Записки научных
семинаров ПОМИ
Том 552, 2026 г.

Y. I. Dementiev, A. A. Ignatiev, V. A. Sidelnik, A. V. Smal, M. S. Ushakov

ADVANCES IN THE HALF-DUPLEX COMMUNICATION COMPLEXITY

Abstract. In this work, we continue the line of research initiated in [9], which introduced the model of half-duplex communication complexity. In contrast to the classical model of communication complexity due to [18], the half-duplex model restricts the parties so that at any moment each of them can either speak or listen, but not both simultaneously, as in communication via a walkie-talkie. This model is motivated by questions arising in the study of the KRW conjecture.

Addressing the open problems posed in [9], we establish lower bounds for the disjointness function in all considered variants of the half-duplex model, and prove upper bound in the half-duplex model with zero. We further show lower and upper bounds on the half-duplex complexity of the Karchmer–Wigderson games for counting functions and for the recursive majority function, adapting and extending techniques from classical communication complexity. In addition, we introduce the notion of non-deterministic half-duplex communication complexity and prove bounds relating it to non-deterministic communication complexity in the classical model.

A preliminary version of this work appeared in [6], and some of the results were subsequently refined in [5]. The present journal version significantly extends the conference version and includes a new section on Karchmer–Wigderson games for threshold and majority function.

§1. Introduction

1.1. Background. Communication complexity is a powerful tool with applications in algorithms, circuit complexity, proof complexity, and many

Key words and phrases: communication complexity and half-duplex communication and Karchmer-Wigderson games.

This work was supported by the Ministry of Science and Higher Education of the Russian Federation (agreement 075-15-2025-344 dated 29/04/2025 for Saint Petersburg Leonhard Euler International Mathematical Institute at PDMI RAS).

other areas of theoretical computer science. In the classical model of communication complexity introduced by Yao [18], two players, Alice and Bob, try to compute $f(x, y)$ for some function f, where Alice only knows x and Bob only knows y. The players can communicate by sending bits to each other, one bit per round, and at the end of the communication, both players must know the result $f(x, y)$. The essential property of this classical model is that in each round of communication, one player sends a bit, and the other receives it.

There are many extensions of this basic two-party communication model, such as randomized communication complexity, non-deterministic communication complexity, various types of multiparty communication models, etc. In [9], the authors suggested considering a communication model where the players speak over *a half-duplex channel*. A well-known example of half-duplex communication is talking using a walkie-talkie: one has to hold a "push-to-talk" button to speak to another person, and the other has to keep it released to listen. If two persons try to speak simultaneously, then they do not hear each other. Formally speaking, every round, each player chooses one of three actions: send 0, send 1, or receive. There are three different types of rounds: *a classical round*, when one player sends some bit while the other one receives, *a wasted*[1] *round*, when both players send bits (these bits get lost), and *a silent round*, when both players receive. In [9], the authors defined three variations of the half-duplex model based on what happens in silent rounds: half-duplex models with silence, with zero, and with adversary (see Section 2 for more information).

It turned out that the communication complexity in the half-duplex models not only differs from the classical case, but also behaves differently. For example, in the classical case, the equality function, the disjointness function, and the inner product function have complexity $n + 1$, meaning that all three are the hardest functions. In the half-duplex models with silence and with zero, the equality is less complex than the other two.

The original motivation for the half-duplex communication models comes from the study of the Karchmer–Wigderson games [11] for the multiplexer relation [7]. In [15], results from the half-duplex communication complexity were used to prove a lower bound on a composition of the universal relation with the Karchmer–Wigderson game for some function.

[1] In the original paper, this type of rounds is called *spent*. We believe that *wasted* is a better term for it.

	EQ_n	IP_n	DISJ_n		$\mathrm{KW}_{\mathrm{MOD2}_n}$	
D_s^{hd}	$\geqslant n/\log 5$	$\geqslant n/2$	$\geqslant n/\log 5$	$\star$	$\geqslant \log n$	$\star$
	$\leqslant n/\log 5 + o(n)$		$\leqslant n/2 + O(1)$		$\leqslant 2\log_3 n$	$\star$
D_0^{hd}	$\geqslant n/\log 3$	$\geqslant n/\log(2/(3-\sqrt{5}))$	$\geqslant n/\log 3$	$\star$	$\geqslant 1.439\log n$	$\star$
	$\leqslant n/\log 3 + o(n)$		$\leqslant 3n/4 + o(n)$	$\star$	$\leqslant 3\log_3 n$	$\star$
D_a^{hd}	$\geqslant n/\log 2.5$	$\geqslant n/\log(7/3)$	$\geqslant n/\log 2.5$	$\star$	$= 2\log n$	

Table 1. Lower and upper bounds for the communication problems considered in [8].

The authors suggest that a better understanding of the half-duplex communication complexity might help to achieve new bound in the study of the KRW conjecture [10] and even prove a supercubic lower bound on the De Morgan formula size of an implicit function.

We continue the research started [9], and close some open questions from it regarding the complexity of the disjointness function. We also study the complexity of the Karchmer–Wigderson games for the counting functions and for the recursive majority function. In addition, we define the non-deterministic half-duplex communication complexity and prove bounds connecting it to the classical non-deterministic communication complexity.

1.2. Overview of results. For a communication problem P, let $\mathrm{D}_s^{hd}(P)$, $\mathrm{D}_0^{hd}(P)$, and $\mathrm{D}_a^{hd}(P)$, denote the (deterministic) half-duplex communication complexity of P with silence, with zero, and with adversary, respectively (see Section 2 for formal definitions). Table 1.2 contains a summary of lower and upper bounds for the communication problems considered in [9], the bounds proved in this paper are marked with $\star$.

In addition to the bounds in Table 1.2, we prove the following upper bounds for special cases of the counting function,

$$\mathrm{D}_s^{hd}(\mathrm{KW}_{\mathrm{MOD3}_n}) \leqslant 3\log_3 n, \quad \mathrm{D}_s^{hd}(\mathrm{KW}_{\mathrm{MOD5}_n}) \leqslant 2.47\log n,$$

$$\mathrm{D}_s^{hd}(\mathrm{KW}_{\mathrm{MOD11}_n}) \leqslant 3.48\log n,$$

and for the recursive majority function,

$$\mathrm{D}_s^{hd}(\mathrm{KW}_{\mathrm{RecMaj}_n}) \leqslant \mathrm{D}_0^{hd}(\mathrm{KW}_{\mathrm{RecMaj}_n}) \leqslant 2\log_3 n.$$

For arbitrary $p \geqslant 7$, we prove that $\mathrm{D}_s^{hd}(\mathrm{KW}_{\mathrm{MOD}p_n}) \leqslant 1.16 \left\lceil 1 + \log_3 \frac{p}{2} \right\rceil \cdot \log n$. For threshold function we show upper bound,

$$\mathrm{D}_s^{hd}(\mathrm{KW}_{\mathrm{THR}_{k,n}}) \leqslant \log_3(n) + (k-2) \cdot \log_3(\log_k(n)) + (k-1)^2 \cdot \log_3(k-1) + k^2 - 2.$$

We also show that the lower bounds for $\mathrm{KW}_{\mathrm{MOD}2_n}$ in Table 1.2 apply for $\mathrm{KW}_{\mathrm{MOD}p_n}$ for arbitrary p in all three models.

In Section 5, we introduce non-deterministic half-duplex communication complexity. Let $\mathrm{N}_s^{hd}(f)$, $\mathrm{N}_0^{hd}(f)$, and $\mathrm{N}_a^{hd}(f)$ denote the non-deterministic half-duplex communication complexity of f with silence, with zero, and with adversary, respectively. For any function $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, we show that

$$\mathrm{N}_s^{hd}(f) = \mathrm{N}(f)/\log 5 + \Theta(\log \mathrm{N}(f)),$$
$$\mathrm{N}_0^{hd}(f) = \mathrm{N}(f)/\log 3 + \Theta(\log \mathrm{N}(f)),$$
$$\mathrm{N}_a^{hd}(f) \geqslant \mathrm{N}(f)/\log 3,$$

where $\mathrm{N}(f)$ denotes the classical non-deterministic communication complexity.

1.3. Organization of this paper. In Section 2, we review the half-duplex communication models. In Section 3, we prove lower and upper bounds for the disjointness function in various half-duplex models. Then, in Section 4.1, we prove lower and upper bounds on the Karchmer–Wigderson games for the counting functions. In Section 4.2, we prove upper bounds on the Karchmer–Wigderson games for recursive majority, majority and threshold function. In Section 5, we define the non-deterministic half-duplex communication complexity and prove bounds connecting it to the classical non-deterministic complexity. Section 6 contains open problems.

§2. Half-duplex communication complexity

We expect that the reader is familiar with the standard definitions of communication complexity that can be found in [13]. It will be necessary to understand that a communication protocol can be described by a binary tree, where every node has an associated combinatorial rectangle of all input pairs (x, y) such that if the players are given x and y, then the communication goes through this node. Finally, we expect the reader to understand why the rectangles associated with the leaves of the protocol tree are monochromatic.

Let's assume that the players have some synchronizing mechanism, e.g., synchronized clocks, that allows them to understand when each round begins. In the half-duplex communication, every round, each player chooses one of three *actions*: `send(0)`, `send(1)`, or `receive`. So, there are three different types of rounds.

- *A classical round:* one player sends some bit and the other one receives it.
- *A wasted round:* both players send bits, and these bits get lost.
- *A silent round:* both players receive.

In [9], the authors considered the following three variations of this model.

- *The half-duplex communication model with silence.* In a silent round, both players receive a special symbol `silence`, so it is possible for both players to distinguish a silent round from a classical one.
- *The half-duplex communication model with zero.* In a silent round, both players receive 0, so the players can not distinguish a silent round from a classical round where the other player sends 0.
- *The half-duplex communication model with adversary.* In a silent round, each player receives arbitrary bit, not necessarily the same as the other player.

In the half-duplex model with zero, there is no need to send zeros – a player can choose to receive instead and the other player will not notice the difference.

In the classical case a communication protocol is described by a binary tree with labels. A communication of players on inputs (x, y) corresponds to a root-to-leaf path in this tree. After every round each player knows what the other player's action was, and that allows them to determine the next node of the path. Unlike the classical case, in the half-duplex communication models a player does not always know what the other player's action – the information about it can be "lost", i.e., in a wasted round a player do not know what the other player's action was. If we extend the definition of a classical protocol for the half-duplex models then in some rounds a player might not know how to choose the next node of the path. That is why in the half-duplex case, a protocol is described by two trees – one for each player. The protocol trees have arity 5 in the half-duplex model with silence, arity 3 in the model with zero, and arity 4 in the model with adversary. The arity of the tree correspond

to a number of different *events* a player can observe: `sent(0)`, `sent(1)`, `received(0)`, `received(1)`, and `silence`. In the half-duplex model with silence all five events are possible, in the half-duplex model with zero there is no silence and the players never send 0, in the half-duplex model with adversary a silence is not possible. For the formal definition of the half-duplex communication protocols see [9]. It should be noted that despite the differences, every node of a half-duplex protocol tree has an associated rectangle of inputs, and every leaf rectangle is monochomatic.

The minimal number of rounds that is enough to solve a communication problem R on all inputs defines the communication complexity of R. For the classical communication model, we denote it by $\mathrm{D}(R)$, for the half-duplex models with silence, with zero, and with adversary, we denote it by $\mathrm{D}_s^{hd}(R)$, $\mathrm{D}_0^{hd}(R)$, and $\mathrm{D}_a^{hd}(R)$, respectively.

§3. Bounds for the disjointness function

The disjointness function $\mathrm{DISJ}_n : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ is defined by

$$\mathrm{DISJ}_n(x,y) = 1 \iff \forall i \in [n] : x_i \neq 1 \vee y_i \neq 1.$$

This is one of the hardest functions in the classical communication model – it has communication complexity exactly $n+1$. In other words, the trivial protocol where Alice sends all her bits to Bob, Bob computes the result and then sends it back to Alice, is optimal. This is not the case in the half-duplex models. In [9], the authors prove an upper bound $n/2 + O(1)$ on DISJ_n in the half-duplex model with silence (Theorem 16 in [9]). In this section we prove lower bounds for DISJ_n in all three half-duplex models, and an upper bound in the model with zero. We start with the lower bounds.

Theorem 1. *For all* $n \in \mathbb{N}$,

$$\mathrm{D}_s^{hd}(\mathrm{DISJ}_n) \geqslant n/\log 5, \quad \mathrm{D}_0^{hd}(\mathrm{DISJ}_n) \geqslant n/\log 3,$$
$$\mathrm{D}_a^{hd}(\mathrm{DISJ}_n) \geqslant n/\log 2.5.$$

To prove this theorem, we will need the following folklore property of communication protocols for DISJ_n. For a Boolean vector $x \in \{0,1\}^n$, let $\bar{x}$ denotes its complement, i.e. $\bar{x}_i = 1 - x_i$ for all $i \in [n]$.

Lemma 1 ([13]). *Let* $x, y \in \{0,1\}^n$ *and* $x \neq y$. *The pair of inputs* $(x, \bar{x})$ *and* $(y, \bar{y})$ *do not belong to the same monochromatic rectangle of* DISJ_n.

Proof. Note that $\mathrm{DISJ}_n(x, \bar{x}) = \mathrm{DISJ}_n(y, \bar{y}) = 1$. Consider a rectangle containing $(x, \bar{x})$ and $(y, \bar{y})$. Due to combinatorial rectangle properties it necessarily contains also $(x, \bar{y})$ and $(y, \bar{x})$. If $y \subset x$ then $\mathrm{DISJ}_n(x, \bar{y}) = 0$; otherwise, $\mathrm{DISJ}_n(\bar{x}, y) = 0$. Hence this rectangle is not monochromatic. □

This property of DISJ_n allows us to define a sub-additive measure $\mu(R)$ that is equal to the number of pairs $(x, \bar{x})$ in rectangle R. In [9], a special framework was developed specifically for such measures, *the rectangle elimination technique*. The following lemma allows us to get a lower bound by showing that for every protocol the measure of the root rectangle is large while the measures of all leaf rectangles are small. We say that some rectangle of inputs R is *good* for a protocol Π if restricting the problem to R allows the players to omit the first round of Π.

Lemma 2 (Lemma 10 in [9])**.** *Let μ be some sub-additive measure on rectangles such that $\mu(X \times Y) \geqslant \mu_r$ and for any leaf rectangle R_l, $\mu(R_l) \leqslant \mu_\ell$. If for any rectangle R there is always a good subrectangle for function $f \restriction R$ of measure at least $\alpha \cdot \mu(R)$ then the depth of the protocol is at least $\log_{1/\alpha} \frac{\mu_r}{\mu_\ell}$.*

Now we are ready to prove Theorem 1.

Proof of Theorem 1. For the first two lower bounds it is enough to use *the fooling set method* [13]. There are 2^n pairs $(x, \bar{x})$, and hence there are at least 2^n 1-monochromatic rectangles, that gives a lower bound on the number of leaves in the protocol. It remains to note that the protocol trees in the half-duplex models with silence and with zero have arities 5 and 3, respectively.

The third lower bound requires a little bit more effort. The same argument would prove only $n/2$ lower bound which is trivial. Instead, we use the rectangle elimination technique. For that we will use the fact that for any protocol solving DISJ_n on some rectangle R there is a set of five good rectangles covering each element of R twice. Hence, one of these rectangles has measure at least $2/5 \cdot \mu(R)$ and we can reduce the problem to it (for more details see [8, Theorem 13]). Application of Lemma 2 for μ and $\alpha = 2/5$ concludes the proof. □

Now we proceed to proving an upper bound for DISJ_n in the half-duplex model with zero. In order to show a protocol with $\frac{3}{4}n + o(n)$ rounds, we start with a less efficient protocol and then improve it. Let us remind that

Case 1

Block	Alice	Bob
00	`receive`	`send(1)`
01	`send(1)`	`receive`
10	`receive`	`receive`
11	`receive`	`receive`

Case 2

Block	Alice	Bob
00	`receive`	`receive`
01	`send(1)`	`send(1)`
10	`receive`	`receive`
11	`receive`	`receive`

Case 3

Block	Alice	Bob
00	`send(1)`	`receive`
01	`receive`	`receive`
10	`receive`	`send(1)`
11	`receive`	`receive`

Table 2. The first stage of the half-duplex protocol for DISJ_n with zero.

in the half-duplex model with zero there is no need to send zeros, so the players never do it.

Theorem 2. *For all* $n \in \mathbb{N}$, $\mathrm{D}_0^{hd}(\mathrm{DISJ}_n) \leqslant 5n/6 + O(\log n)$.

Proof. W.l.o.g., we assume that n is even. Consider the following protocol. Alice and Bob split their input strings into blocks of size 2, so each player has $n/2$ such blocks. Let $\sharp(ab)$ denotes the number of blocks ab among the blocks of Bob. Note that one of the following cases holds:

(1) $\sharp(00) \geqslant n/6$, then $\sharp(01) + \sharp(10) + \sharp(11) \leqslant n/2 - n/6 = n/3$,
(2) $\sharp(01) \geqslant n/6$, then $\sharp(00) + \sharp(10) + \sharp(11) \leqslant n/2 - n/6 = n/3$,
(3) $\sharp(00) + \sharp(01) < n/3$.

At the beginning Bob must check which of the cases applies and tell Alice using two bits of communication. Further communication depends on it. We will show that in all the cases the players can solve the problem using at most $5n/6 + O(\log n)$ rounds. The communication will be divided into two stages. In the first stage, Alice and Bob process their input considering one pair of corresponding blocks per round. Each round they act as it is described in Table 3. Alice and Bob need to be able to determine the situations when the corresponding blocks intersect, that is, to distinguish block pairs $(01, 01)$, $(01, 11)$, $(11, 01)$, $(11, 10)$, $(11, 11)$, $(10, 10)$, $(10, 11)$ from others.

Case 1. After $n/2$ rounds, Bob tells Alice whether he ever has received 1 while processing a block 11 or 01. If he has, then the corresponding block of Alice was 01, and hence their inputs intersect. This corresponds to identifying $(01, 01)$ and $(01, 11)$. Further, Alice tells Bob whether she ever has a silent round while processing 11. If she has, then the corresponding block of Bob was one of 01, 10 or 11, and hence their inputs intersect, so the players identified block pairs $(11, 01)$, $(11, 10)$, and $(11, 11)$. If any intersecting blocks have been found, the players stop the communication and output 0. Otherwise, they proceed to the second stage.

In the second stage, to identify the remaining two block pairs $(10, 10)$ and $(10, 11)$, Alice needs to distinguish zeros she received when Bob was processing 10 or 11, and zeros she received when Bob was processing 01. Bob sequentially (starting from his first silent round) goes through all his blocks 01, 10, 11 corresponding to silent rounds. When he processes 10 and 11, he sends 1, and when he processes 01, he sends 0. Alice knows how many bits Bob will send her, since he sends one bit for every silent round. Alice simultaneously processes her blocks corresponding to the silent rounds. Now for every such block she knows whether Bob has 10 or 11. If she receives 1 having 10 then they identified $(10, 10)$ and $(10, 11)$. At the end of the protocol, Alice needs one more round to tell Bob whether she has found an intersection.

The complexity of the first stage is $n/2 + O(1)$, the complexity of the second stage is $n/3 + O(1)$, so the total complexity is $5n/6 + O(1)$.

Case 2. After $n/2$ rounds, Bob tells Alice whether he has ever received 1 while processing 11. If he has, then Alice had 01 in the corresponding block, and hence their inputs intersect, so the players identified $(01, 11)$. After that, Alice tells Bob whether she has ever received 1 while processing 11. If she has, then they identified $(11, 01)$. Next, Bob tells Alice the number of rounds where he sent 1. Alice compares it with the number of rounds where she received 1. If these two numbers are equal, then there were no wasted rounds in the first stage, otherwise the players identified a block pair $(01, 01)$. If any intersecting blocks have been found, the players stop the communication and output 0. Otherwise, they proceed to the second stage.

In the second stage, to identify the remaining four block pairs $(10, 10)$, $(10, 11)$, $(11, 10)$ and $(11, 11)$, Alice needs to distinguish zeros received when Bob was processing 10 or 11 from zeros received when Bob was

processing 00. Bob sequentially goes through all his blocks 00, 10, 11 corresponding to silent rounds. When he processes 10 and 11, he sends 1, and when he processes 00, he sends 0. Similarly to the previous case, if she receives 1 having 10 or 11 then they identified one of the desired pairs of blocks. At the end of the protocol Alice needs one more round to tell Bob whether she has found an intersection.

The complexity of the first stage is $n/2 + O(1)$, the complexity of the second stage is $n/3 + O(\log n)$, so the total complexity is $5n/6 + O(\log n)$.
Case 3. After $n/2$ rounds, Bob tells Alice if there was a silent round corresponding to his block 11. If there was such a round, the Alice had 01, 10 or 11 in the corresponding block, so the players identified block pairs $(01, 11)$, $(10, 11)$, and $(11, 11)$. Next, Alice tells Bob if she has ever received 1 while processing 10 or 11. If she has, then Bob had 10, and hence they identified a block $(10, 10)$ or $(11, 10)$. If any intersecting blocks have been found, the players stop the communication and output 0. Otherwise, they proceed to the second stage.

In the second stage, to identify the remaining two block pairs $(01, 01)$ and $(11, 01)$, Alice needs to distinguish zeros received while Bob was processing a block 00 and zeros received while Bob was processing a block 01. Bob sequentially goes through all his blocks 00 and 01 corresponding to silent rounds. When he processes a block 01, he sends 1, and when he processes 00, he sends 0. Similarly to the previous cases, if Alice receives 1 while processing 01 or 11, she identifies one of the desired block pairs.

The complexity of the protocol in this case is $5n/6 + O(1)$. □

The protocol proposed in Theorem 2 can be improved if we notice that reiterating over a part of the blocks that happens in the second stage can be reduced to solving disjointness problem on smaller inputs.

Theorem 3. *For all* $n \in \mathbb{N}$, $\mathrm{D}_0^{hd}(\mathrm{DISJ}_n) \leqslant 3n/4 + O(\log^2 n)$.

Proof. We are going to modify the protocol from the proof of Theorem 2. In the modified protocol, the players consider the same three cases, and they have the same first stages in all cases. The second stage is different. Instead of reiterating all blocks corresponding to silent rounds, Alice and Bob reduce this problem to solving disjointness on strings of size at most $n/3$, and then run the same protocol for disjointness recursively. Thus, we

get the following bound

$$\mathrm{D}_0^{hd}(\mathrm{DISJ}_n) \leqslant \sum_{i=0}^{\lceil \log_3(n) \rceil} \frac{n}{2 \cdot 3^i} + O(\log^2 n)$$
$$\leqslant \sum_{i=0}^{\infty} \frac{n}{2 \cdot 3^i} + O(\log^2 n) = \frac{3n}{4} + O(\log^2 n).$$

It remains to understand how the second stage works in each of the cases.
Case 1. To identify two block pairs $(10, 10)$ and $(10, 11)$, Bob writes down a string x' that has one bit for every silent round: 1 for a block 10 or 11, and 0 for 01. Similarly, Alice writes down a string y' that has one bit for every silent round: 1 for a block 10, and 0 for other blocks. It is not hard to see, that $\mathrm{DISJ}(x', y') = 0$ if and only if there was a silent round corresponding to $(10, 10)$ or $(10, 11)$.
Case 2. To identify four block pairs $(10, 10)$, $(10, 11)$, $(11, 10)$ and $(11, 11)$, Bob writes down a string x' that has one bit for every silent round: 1 for a block 10 or 11, and 0 for 00. Similarly, Alice writes down a string y' using the same rules. It is each to verify, that $\mathrm{DISJ}(x', y') = 0$ if and only if there was a silent round corresponding to one of the desired block pairs.
Case 3. To identify the remaining two block pairs $(01, 01)$ and $(11, 01)$, Bob writes down a string x' that has one bit for every silent round: 1 for a block 01, and 0 for others (at this point, he already knows that there were no silent rounds corresponding to 11). Similarly, Alice writes down a string y' that has one bit for every silent round: 1 for a block 01 or 11, and 0 for 10. And again, it is not hard to see, that $\mathrm{DISJ}(x', y') = 0$ if and only if there was a silent round corresponding $(01, 01)$ and $(11, 01)$. □

§4. Bounds on the Karchmer–Wigderson games

The seminal work of Karchmer and Wigderson [11] established a correspondence between De Morgan formulas for non-constant Boolean function f and communication protocols for the Karchmer–Wigderson game for f. *De Morgan formula* is a Boolean formula over the De Morgan basis $\{\wedge, \vee, \neg\}$, where $\neg$ operation is only applied to input variables. *The Karchmer–Wigderson game* for Boolean function $f : \{0,1\}^n \to \{0,1\}$ is the following communication problem: Alice gets an input $x \in \{0,1\}^n$ such that $f(x) = 0$, and Bob gets an input $y \in \{0,1\}^n$ such that $f(y) = 1$. Their goal is to find a coordinate $i \in [n]$ such that $x_i \neq y_i$. The Karchmer–Wigderson game for f can be considered as a communication problem for

the Karchmer–Wigderson relation for f:

$$\mathrm{KW}_f = \{(x, y, i) \mid x, y \in \{0,1\}^n, i \in [n], f(x) = 0, f(y) = 1, x_i \neq y_i\}.$$

Karchmer and Wigderson showed that the communication complexity of KW_f is exactly equal to the formula depth complexity of f. This observation allows us to use communication complexity methods for proving formula depth lower bounds.

In this section, we prove bounds on the half-duplex communication complexity of the Karchmer–Wigderson games for *the counting function* $\mathrm{MOD}p_n : \{0,1\}^n \to \{0,1\}$, defined by

$$\mathrm{MOD}p_n(x) = 0 \iff x_1 + \ldots + x_n = 0 \mod p,$$

and for *the recursive majority function* $\mathrm{RecMaj}_n : \{0,1\}^n \to \{0,1\}$, defined by

$$\begin{aligned}\mathrm{RecMaj}_n(x_1, \ldots, x_n) = \mathrm{Maj}_3\big(&\mathrm{RecMaj}_{\frac{n}{3}}(x_1, \ldots, x_{\frac{n}{3}}),\\ &\mathrm{RecMaj}_{\frac{n}{3}}(x_{\frac{n}{3}+1}, \ldots, x_{\frac{2n}{3}}),\\ &\mathrm{RecMaj}_{\frac{n}{3}}(x_{\frac{2n}{3}+1}, \ldots, x_n)\big),\end{aligned}$$

where $\mathrm{Maj}_3 : \{0,1\}^3 \to \{0,1\}$ is majority of three bits, and n is a power of three. And for *threshold function* $\mathrm{THR}_{k,n} : \{0,1\}^n \to \{0,1\}$ defined by

$$\mathrm{THR}_{k,n}(x) = 1 \iff \sum_{i=1}^{n} x_i \geqslant k.$$

A special case of the threshold function is the majority function defined by

$$\mathrm{Maj}_n = \mathrm{THR}_{\frac{n}{2},n}.$$

This does not immediately imply any bounds for De Morgan formulas – the correspondence between formulas and protocols works only for the classical model. On the other hand, better understanding of the half-duplex model might help to prove bounds in the classical case as it was done in [15].

4.1. Counting function. We start with lower bounds for $\mathrm{MOD}p_n$ functions. In the classical case, a lower bound on the communication complexity of the Karchmer – Wigderson game for some function f corresponds to a lower bound on De Morgan formula depth complexity of f. For $\mathrm{MOD}2_n$, the parity function, we have the tight bound $2 \log n$: the lower bound is due to the famous work of Khrapchenko [12], and the upper bound is straightforward by implementing binary search. The method of Khrapchenko can also be used to prove $2 \log n - O(1)$ lower bounds for $\mathrm{MOD}p_n$ for arbitrary

p. In [9], the authors proved $2\log n$ lower bound for $\mathrm{KW}_{\mathrm{MOD}2_n}$ in the half-duplex model with adversary. We use similar ideas to prove general lower bounds for $\mathrm{MOD}p_n$ in all the half-duplex models.

Theorem 4. *For any $p \geqslant 2$ and $n \in \mathbb{N}$,*

$$\mathrm{D}_s^{hd}(\mathrm{KW}_{\mathrm{MOD}p_n}) > \log n - O(1),$$
$$\mathrm{D}_0^{hd}(\mathrm{KW}_{\mathrm{MOD}p_n}) > 1.439 \log n,$$
$$\mathrm{D}_a^{hd}(\mathrm{KW}_{\mathrm{MOD}p_n}) \geqslant 2\log n - O(1).$$

Remark 1. In earlier versions of this paper and in the conference version [6], this theorem states better lower bounds that were erroneously obtained by using an analysis that does not work for any distribution on the inputs.

We prove this theorem using the information-theoretic approach from [9]. For basic definitions of information theory, we refer to [4]. We show that there is a probability distribution over the inputs of Alice and Bob, such that at the beginning of the communication each player has uncertainty roughly $\log n$ bits about the input of the other player. At the end of the communication (for this specific distribution) each player necessarily knows the input of the other player. This means that during the protocol each player learns roughly $\log n$ bits of information. For the classical communication model, this would be enough to show the $2\log n$ lower bound, because in every round one of the players learns at most one bit of information and the other learns nothing, so there must be at least $2\log n$ rounds. In the half-duplex communication, the situation is more complicated – in silent rounds both players might learn some information. To estimate the amount of information the players learn during half-duplex communication, we prove upper bounds on it similar to upper bounds proved in [9, Theorems 18, 21, and 24].

Proof. Let $(\mathcal{X}, \mathcal{Y})$ be a pair of jointly distributed random variables where $\mathcal{X}$ is uniformly distributed on the set $\{x \in \{0,1\}^n \mid \mathrm{MOD}p_n(x) = 0, n/4 < \|x\|_1 < 3n/4\}$, that corresponds to Alice's inputs of Hamming weight between $n/4$ and $3n/4$, and $\mathcal{Y}$ is obtained from $\mathcal{X}$ by flipping one 0 bit uniformly at random. Thus, $H(\mathcal{Y} \mid \mathcal{X}) \geqslant \log(n/4)$ and $H(\mathcal{X} \mid \mathcal{Y}) \geqslant \log(n/4)$. Before any communication takes place $H(\mathcal{Y} \mid \mathcal{X}) + H(\mathcal{X} \mid \mathcal{Y}) \geqslant 2\log n - O(1)$. Given inputs from the distribution $(\mathcal{X}, \mathcal{Y})$ the players have to find the unique bit of difference.

Let $\mathcal{P}$ be a protocol for MODp_n. W.l.o.g., we assume that all the leaves of $\mathcal{P}$ are on the same depth. For any natural k, let Π_A^k and Π_B^k be the marginal distributions over Alice's and Bob's partial transcripts after running $\mathcal{P}$ for k rounds induced by $(\mathcal{X}, \mathcal{Y})$. If the protocol $\mathcal{P}$ has depth d, then $H(\mathcal{Y} \mid \mathcal{X}, \Pi_A^d) + H(\mathcal{X} \mid \mathcal{Y}, \Pi_B^d) = 0$. That the players have to learn at least $2\log n - O(1)$ bits in d rounds. If they can learn at most α bits of information in every round, then $d \geqslant \frac{2}{\alpha}\log n - O(1)$.

Now we need to upper bound the amount of information that can be learned in one round. That is to show that the amount of information the players learn in k rounds is upper bounded by αk for some constant α,

$$I(\mathcal{X} : \Pi_B^k \mid \mathcal{Y}) + I(\mathcal{Y} : \Pi_A^k \mid \mathcal{X}) \leqslant \alpha k.$$

We will induct on k: the number of rounds. For $k = 0$, there is only one possible partial transcript for either player, the empty transcript, and thus the result is immediate. Now suppose that this is true for some k. Let $\mathcal{E}_A^{k+1}$ and $\mathcal{E}_B^{k+1}$ be the marginal distributions over which event each player will observe (`sent(0)`, `sent(1)`, `received(0)`, `received(1)`, and `silence`). Note that

$$\begin{aligned} I(\mathcal{X} : \Pi_B^{k+1} \mid \mathcal{Y}) &= I(\mathcal{X} : \Pi_B^k, \mathcal{E}_B^{k+1} \mid \mathcal{Y}) \\ &= I(\mathcal{X} : \Pi_B^k \mid \mathcal{Y}) + I(\mathcal{X} : \mathcal{E}_B^{k+1} \mid \mathcal{Y}, \Pi_B^k). \end{aligned}$$

Thus, it suffices to show that $I(\mathcal{X} : \mathcal{E}_B^{k+1} \mid \mathcal{Y}, \Pi_B^k) + I(\mathcal{Y} : \mathcal{E}_A^{k+1} \mid \mathcal{X}, \Pi_A^k) \leqslant \alpha$. By definition of mutual information,

$$\begin{aligned} I(\mathcal{X} : \mathcal{E}_B^{k+1} \mid \mathcal{Y}, \Pi_B^k) &= H(\mathcal{E}_B^{k+1} \mid \mathcal{Y}, \Pi_B^k) - H(\mathcal{E}_B^{k+1} \mid \mathcal{X}, \mathcal{Y}, \Pi_B^k) \\ &= H(\mathcal{E}_B^{k+1} \mid \mathcal{Y}, \Pi_B^k). \end{aligned}$$

The second term here is zero because values of $\mathcal{X}$ and $\mathcal{Y}$ unambiguously determine the entire protocol. So it is enough to bound $H(\mathcal{E}_B^{k+1} \mid \mathcal{Y}, \Pi_B^k) = \mathbb{E}_{y,\pi}[H(\mathcal{E}_B^{k+1} \mid \mathcal{Y} = y, \Pi_B^k = \pi)]$.

Let $\mathcal{A}_A^{k+1}$ and $\mathcal{A}_B^{k+1}$ be the marginal distributions over players actions (`send(0)`, `send(1)`, and `receive`) in round $k+1$. Note that $\mathcal{A}_B^{k+1}$ is a function of $\mathcal{Y}$ and Π_B^k. If for some pair (y, π) Bob sends, i.e. $\mathcal{A}_B^{k+1} =$ `send(0)` or $\mathcal{A}_B^{k+1} =$ `send(1)`, then $H(\mathcal{E}_B^{k+1} \mid \mathcal{Y} = y, \Pi_B^k = \pi) = 0$. For the sake of brevity we use r as a shortcut for action `receive`.

$$\begin{aligned} H(\mathcal{E}_B^{k+1} \mid \mathcal{Y}, \Pi_B^k) &= H(\mathcal{E}_B^{k+1} \mid \mathcal{Y}, \Pi_B^k, \mathcal{A}_B^{k+1}) \\ &= \Pr[A_B^{k+1} = r] \cdot H(\mathcal{E}_B^{k+1} \mid \mathcal{Y}, \Pi_B^k, \mathcal{A}_B^{k+1} = r). \end{aligned}$$

Given that Bob receives, his event is determined by the action of Alice, thus

$$H(\mathcal{E}_B^{k+1} \mid \mathcal{Y}, \Pi_B^k, \mathcal{A}_B^{k+1} = r) = H(\mathcal{A}_A^{k+1} \mid \mathcal{Y}, \Pi_B^k, \mathcal{A}_B^{k+1} = r) \leqslant H(\mathcal{A}_A^{k+1} \mid \mathcal{A}_B^{k+1} = r).$$

This gives us the following bound.

$$I(\mathcal{X} : \Pi_B^k \mid \mathcal{Y}) = H(\mathcal{E}_B^{k+1} \mid \mathcal{Y}, \Pi_B^k) \leqslant \Pr[\mathcal{A}_B^{k+1} = r] \cdot H(\mathcal{A}_A^{k+1} \mid \mathcal{A}_B^{k+1} = r).$$

The same argument works for $I(\mathcal{Y} : \Pi_A^k \mid \mathcal{X})$ and hence we get,

$$I(\mathcal{X} : \Pi_B^k \mid \mathcal{Y}) + I(\mathcal{Y} : \Pi_A^k \mid \mathcal{X}) \leqslant \Pr[\mathcal{A}_B^{k+1} = r] \cdot H(\mathcal{A}_A^{k+1} \mid \mathcal{A}_B^{k+1} = r) + \Pr[\mathcal{A}_A^{k+1} = r] \cdot H(\mathcal{A}_B^{k+1} \mid \mathcal{A}_A^{k+1} = r).$$

Now let a_r and b_r denote the probabilities that Alice and Bob receive in round $k+1$, respectively. In addition, let p_{0r}, p_{1r}, p_{r0}, p_{r1}, and p_{rr} denote the probabilities of all the possible situations in round $k+1$ (p_{0r} corresponds to a situation where Alice sends 0 and Bob receives, and so on). The right hand side of the above inequality can be rewritten as follows.

$$b_r \cdot \left(\frac{p_{0r}}{b_r} \cdot \log \frac{b_r}{p_{0r}} + \frac{p_{1r}}{b_r} \cdot \log \frac{b_r}{p_{1r}} + \frac{p_{rr}}{b_r} \cdot \log \frac{b_r}{p_{rr}} \right) + a_r \cdot \left(\frac{p_{r0}}{a_r} \cdot \log \frac{a_r}{p_{r0}} + \frac{p_{r1}}{a_r} \cdot \log \frac{a_r}{p_{r1}} + \frac{p_{rr}}{a_r} \cdot \log \frac{a_r}{p_{rr}} \right).$$

Numerical analysis of this expression with all the necessary restrictions shows that its maximum is at most 2, hence $I(\mathcal{X} : \Pi_B^k \mid \mathcal{Y}) + I(\mathcal{Y} : \Pi_A^k \mid \mathcal{X}) \leqslant 2k$. That gives $(2 \log n - O(1))/2 = \log n - O(1)$ lower bound for the half-duplex model with silence.

For the half-duplex model with zero, we can use the same analysis, but the players never send 0. So at the end, we maximize

$$b_r \cdot \left(\frac{p_{1r}}{b_r} \cdot \log \frac{b_r}{p_{1r}} + \frac{p_{rr}}{b_r} \cdot \log \frac{b_r}{p_{rr}} \right) + a_r \cdot \left(\frac{p_{r1}}{a_r} \cdot \log \frac{a_r}{p_{r1}} + \frac{p_{rr}}{a_r} \cdot \log \frac{a_r}{p_{rr}} \right).$$

Numerical analysis shows that the maximum of this expression is slightly less then 1.389. That gives $2 \log n / 1.389 \geqslant 1.439 \log n$ lower bound for the half-duplex model with zero.

Finally, in the half-duplex model with adversary, for any distribution on the inputs, the players can learn at most one bit per round [8, Theorem 24], that concludes the proof. □

Round 1		
Case	**Alice**	**Bob**
1	`receive`	`receive`
2	`receive`	`receive`
3	`send 0`	`send 1`
4	`send 1`	`send 0`

Round 2		
Case	**Alice**	**Bob**
1	`send 0`	`send 1`
2	`send 1`	`send 0`
3	`receive`	`receive`
4	`receive`	`receive`

Table 3. The half-duplex protocol for $\mathrm{KW}_{\mathrm{MOD2}_n}$ with silence.

Now we proceed to the upper bounds. First, we will consider a few special cases of the counting function, and then we will prove the general upper bound for arbitrary $p \geqslant 7$. The following two theorems establish upper bounds for $\mathrm{MOD2}_n$ in the half-duplex models with silence and with zero. Both protocols use the idea of ternary search but in a slightly different manner.

Theorem 5. *For all* $n \in \mathbb{N}$, $\mathrm{D}_s^{hd}(\mathrm{KW}_{\mathrm{MOD2}_n}) \leqslant 2\log_3 n$.

Proof. Alice and Bob split their input strings into three equal parts and compute $\mathrm{MOD2}_n$ for the resulting substrings. There are four possible cases for each player.

- Parities of Alice's substrings: 1) 000; 2) 011; 3) 101; 4) 110.
- Parities of Bob's substrings: 1) 111; 2) 100; 3) 010; 4) 001.

Using two rounds of communication the players determine which pair of corresponding substrings have different parities, and then repeat the protocol recursively for these substrings of size $n/3$. That gives the desired bound, $\mathrm{D}_s^{hd}(\mathrm{MOD2}_n) \leqslant 2\log_3 n + O(1)$. These two rounds are described in Table 4.1.

If one of two rounds was silent, i.e., the players received the symbol of silence, then they know that the first substrings have different parities. Otherwise, if none of the rounds was silent, then there were no wasted rounds (if the first was wasted, then the second was necessarily silent, and vice versa, if the second was wasted, then the first was silent). In this case, the players know that they have the same parity of the first substrings, and each player have sent one bit that was received by the other player. Moreover, these bits correspond to the parities of the second substrings. So, both Alice and Bob know the parities of the first two substrings of

the other player's input, hence both players know the parities of all the substrings. □

Theorem 6. *For all* $n \in \mathbb{N}$, $\mathrm{D}_0^{hd}(\mathrm{KW}_{\mathrm{MOD2}_n}) \leqslant 3\log_3 n$.

Proof. Similarly to the proof of Theorem 5, Alice and Bob split their input strings into three equal parts and compute $\mathrm{MOD2}_n$ for all the resulting substrings. There are the same four possible cases for each player as in the proof of Theorem 5. The players use three rounds to determine which pair of corresponding substrings have different parities, and then repeat the protocol recursively for these substrings of size $n/3$. That gives the desired bound $\mathrm{D}_0^{hd}(\mathrm{MOD2}_n) \leqslant 3\log_3 n + O(1)$.

In the first round, Alice and Bob send 1 if they both have Cases 1, otherwise they receive.

- If the first round was silent (i.e., both players received 0) then they both know that none of them has Case 1. In the second round, Alice sends 1 if her first substring is even (Case 2), otherwise she receives. Bob does the reverse, he sends 1 if his first substring is odd (Case 2), otherwise he receives. The third round is similar, Alice and Bob send 1 in their Cases 3, otherwise they receive.
- If at least one of them sent 1 in the first round, then again both know about it. Alice sends 1 if her first substring is odd, otherwise she receives. Bob does the reverse, he sends 1 if his first substring is even, otherwise he receives. In the third round, the players do the same for the second substrings.

If the second or the third rounds were silent, then, respectively, the first or the second substrings have different parities. Otherwise, the third substrings have different parities. □

The next theorem considers the $\mathrm{MOD3}_n$ function. In the classical case, the best known upper bound for it is $2.881\log n$ [3]. We show a simple upper bound based on the idea of ternary search.

Theorem 7. *For all* $n \in \mathbb{N}$, $\mathrm{D}_s^{hd}(\mathrm{KW}_{\mathrm{MOD3}_n}) \leqslant 3\log_3 n + O(1)$.

Proof. Similarly to the protocols for $\mathrm{MOD2}_n$, Alice and Bob split their inputs into three equal parts and for all the resulting substrings compute the number of ones modulo 3. Then they spend three rounds to find a pair of corresponding substrings with different remainders modulo 3, and run the protocol recursively on these substrings. Thus, we get the desired bound. In the first two rounds, Alice sends her remainders for the

first and the second substrings naturally encoded in a ternary alphabet {`0`,`1`,`silence`}. Bob compares the received numbers with the remainders of his first two substrings, decides on which part they should proceed, and sends this number to Alice in the third round using the same encoding. □

Next, we consider the $\mathrm{MOD}5_n$ function. The best known upper bound in the classical case is $3.475 \log n$ [3]. For this and for all the following upper bounds for $\mathrm{MOD}p_n$ functions, we adapt the prefix code technique used in [3].

Theorem 8. *For all* $n \in \mathbb{N}$, $\mathrm{D}_s^{hd}(\mathrm{KW}_{\mathrm{MOD}5_n}) \leqslant 2.47 \log n$.

Proof. Alice and Bob split their inputs into two parts: the first of length εn, and the second of length $(1-\varepsilon)n$, and compute *the remainder* for every resulting substring, that is the number of ones modulo 5. During the protocol the players will narrow the search area from the current string to one of its substrings repeatedly. At the beginning of the communication, Alice sends the remainder of her first substring to Bob. Then the players speak in turns, starting with Bob, sending each other pairs (r, b), where r is a remainder and b is a bit flag. Every turn, the speaking player does the following.

- (except for the very first turn) If $b = 0$ in the previous message then the player narrows the search area to the first substring, otherwise the player narrows the search area to the second substring. The player subdivides the new search area into two parts in proportion $\varepsilon : (1-\varepsilon)$.
- The player choose one of the substrings and narrows the search area to it. If the remainder r received in the previous message is equal to the remainder of the first substring, then the player chooses the second substring, otherwise the player chooses the first one. The player subdivides the new search area into two parts in proportion $\varepsilon : (1-\varepsilon)$.
- The player sends message (r, b), where r is the remainder of the first substring of the current search area, and b is set to 1 if and only if the second substring was chosen in the previous step.

It remains for us to discuss, how exactly the pair (r, b) is encoded. The encoding is the key ingredient of this technique. We are going to use the following prefix-free code in ternary alphabet:

$b\backslash r$	0	1	2	3	4
0	00	01	0s	10	11
1	s00	s01	s0s	s10	s11

Note that "s" stands for "silence". The code is chosen such that every message with $b = 0$ has encoding of length 2, and every message with $b = 1$ has encoding of length 3. To estimate the number of rounds, we need to solve the following system of recurrent relations, where $T(n)$ stands for the number of rounds.

$$\begin{cases} T(n) = 2 + T(\varepsilon n) \\ T(n) = 3 + T((1-\varepsilon)n) \end{cases} \implies \begin{cases} T(n) = 2\log_{\frac{1}{\varepsilon}} n \\ T(n) = 3\log_{\frac{1}{1-\varepsilon}} n \end{cases}$$

From $\frac{2}{3} = \frac{\log \varepsilon}{\log(1-\varepsilon)}$ we get $\varepsilon < 0.57$, and hence $T(n) < 2.466 \log n$. □

Following [3], we use the same methods for the $\mathrm{MOD}11_n$ function with the best known upper bound is $4.93 \log n$ in the classical case.

Theorem 9. *For all* $n \in \mathbb{N}$, $\mathrm{D}_s^{hd}(\mathrm{KW}_{\mathrm{MOD}11_n}) \leqslant 3.48 \log n$.

Proof. The proof is almost identical to proof of Theorem 8. The players use a prefix-free encoding, such that every message with $b = 0$ has encoding of length 3, and every other message has encoding of length 4.

$b\backslash r$	0	1	2	3	4	5	6	7	8	9	10
0	000	001	010	011	00s	0s0	01s	0s1	0ss	100	101
1	s000	s001	s010	s011	s00s	s0s0	s01s	s0s1	s0ss	s100	s101

That leads to the following system of recurrent relations:

$$\begin{cases} T(n) = 3 + T(\varepsilon n) \\ T(n) = 4 + T((1-\varepsilon)n) \end{cases} \implies \begin{cases} T(n) = 3\log_{\frac{1}{\varepsilon}} n \\ T(n) = 4\log_{\frac{1}{1-\varepsilon}} n \end{cases}$$

From $\frac{3}{4} = \frac{\log \varepsilon}{\log(1-\varepsilon)}$ we get $\varepsilon < 0.55$, and hence $T(n) < 3.48 \log n$. □

To generalize the previous results, we prove a general bound for $\mathrm{MOD}p_n$ for arbitrary $p \geqslant 7$.

Theorem 10. *For all* $p \geqslant 7$ *and* $n \in \mathbb{N}$, $\mathrm{D}_s^{hd}(\mathrm{KW}_{\mathrm{MOD}p_n}) \leqslant 1.16 \left\lceil 1 + \log_3 \frac{p}{2} \right\rceil \cdot \log n$.

Note that this upper bound is not useful for some p, e.g., for $p \in \{7, 8, 9\}$ this bound gives $3.48 \log n$, while regular binary search requires only $3 \log n$. Moreover, in the classical case, the corresponding bound [3] is surpassed

by the upper bound on all symmetric functions [2] starting with some p. We expect that the similar happens in the half-duplex model with silence.

Proof. The protocol is similar to the protocol from the proof of Theorem 8. The prefix-free code is chosen such that for $b = 0$ the encoding starts with `0` or `1`, and for $b = 1$ the encoding starts with `s`. The length of the encoding for $b = 0$ is $\lceil \log_3 \frac{3p}{2} \rceil$, for $b = 1$ is $1 + \lceil \log_3 p \rceil$. Thus, we get the following system:

$$\begin{cases} T(n) = 1 + \lceil \log_3 \frac{p}{2} \rceil + T(\varepsilon n) \\ T(n) = 1 + \lceil \log_3 p \rceil + T((1-\varepsilon)n) \end{cases} \Longrightarrow \begin{cases} T(n) = (1 + \lceil \log_3 \frac{p}{2} \rceil) \log_{\frac{1}{\varepsilon}} n \\ T(n) = (1 + \lceil \log_3 p \rceil) \log_{\frac{1}{1-\varepsilon}} n \end{cases}$$

$$\frac{1 + \lceil \log_3 \frac{p}{2} \rceil}{1 + \lceil \log_3 p \rceil} = \frac{\log \varepsilon}{\log(1-\varepsilon)}$$

Note that the right hand side of the last equation decreases with ε, and the left hand side increases with p. So it is easy to see, that $\frac{1+\lceil \log_3 \frac{p}{2} \rceil}{1+\lceil \log_3 p \rceil} \in [\frac{3}{4}, 1]$, and thus $\varepsilon \in [0.5, 0.55)$. That gives us the desired bound $T(n) \leqslant \frac{1+\lceil \log_3 \frac{p}{2} \rceil}{\log \frac{1}{0.55}} \log n \leqslant 1.16 \lceil 1 + \log_3 \frac{p}{2} \rceil \cdot \log n$. □

4.2. Threshold and majority function.

4.2.1. *Recursive majority function.* To conclude this series of results, we consider the recursive majority function RecMaj_n. In the classical case, its communication complexity is known to be bounded between $2 \log_3 n$ and $3 \log_3 n$ [14]. The structure of this function is ideal for implementing a ternary search. So, given the fact that the half-duplex model with silence allows the players to send messages encoded in ternary, the following theorem is straightforward.

Theorem 11. *For $n \in \mathbb{N}$ that is a power of 3, $\mathrm{D}_s^{hd}(\mathrm{KW}_{\mathrm{RecMaj}_n}) \leqslant 2 \log_3 n$.*

Proof. The players split their input strings into three equal parts and implement a ternary search with two rounds per iteration. Every iteration, Alice sends the index of a substring with $\mathrm{RecMaj}_n = 1$ (there is at most one such substring), or 0 if all substrings have $\mathrm{RecMaj}_n = 0$. Bob chooses a substring with $\mathrm{RecMaj}_n = 1$ among the other two substrings and sends its number back to Alice. □

The same upper bound holds in the half-duplex model with zero.

Theorem 12. *For all $n \in \mathbb{N}$ that is a power of 3, $\mathrm{D}_0^{hd}(\mathrm{KW}_{\mathrm{RecMaj}_n}) \leqslant 2\log_3 n$.*

Proof. The players split their input strings into three equal parts and implement a ternary search. Each iteration consists of two rounds. In the first round, Alice is silent if her first substring has $\mathrm{RecMaj}_n = 0$, otherwise she sends 1, while Bob is silent if his first substring has $\mathrm{RecMaj}_n = 1$, otherwise he sends 1. In the second round they repeat it for the second substrings. Note that not both rounds can be wasted, since Alice has at most one substring with $\mathrm{RecMaj}_n = 1$. If one of the rounds was silent then the players narrow the search area to the corresponding substrings. Otherwise, they continue with the third substrings. □

4.2.2. *Threshold function.* We now consider the Karchmer – Wigderson game for the threshold function

$$\mathrm{THR}_{k,n}(x) = 1 \iff \sum_{i=1}^{n} x_i \geqslant k.$$

In the game $\mathrm{KW}_{\mathrm{THR}_{k,n}}$, Alice receives an input x such that $\sum_{i=1}^{n} x_i \geqslant k$, and Bob receives an input y such that $\sum_{i=1}^{n} y_i < k$. Their goal is to find an index i with $x_i \neq y_i$:

$$\mathrm{KW}_{\mathrm{THR}_{k,n}} = \{(x, y, i) | x \in \mathrm{THR}_{k,n}^{-1}(1), y \in \mathrm{THR}_{k,n}^{-1}(0), x_i \neq y_i\}.$$

In the classical communication model, it is known that the communication complexity of the threshold function is bounded above by $\mathcal{O}(\log_2(k^{4,3} n \log_2(n)))$ [1], and bounded below by $\log_2(\lfloor \frac{k}{2} \rfloor n \log_2(\frac{n}{k-1}))$ [17]. In half-duplex models, only special cases are known: upper bounds in the silence model and lower bounds in all three models. In [19], the following lower bounds were proved:

(1) $\mathrm{D}_s^{hd}(\mathrm{KW}_{\mathrm{THR}_{k,n}}) \geqslant \frac{\log_2(n)}{2}$,
(2) $\mathrm{D}_0^{hd}(\mathrm{KW}_{\mathrm{THR}_{k,n}}) \geqslant 0.71 \log_2(n)$,
(3) $\mathrm{D}_a^{hd}(\mathrm{KW}_{\mathrm{THR}_{k,n}}) \geqslant \log_2(n)$

These bounds were obtained via an information-theoretic approach. The same work also presented the following upper bounds:

(1) $\mathrm{D}_s^{hd}(\mathrm{KW}_{\mathrm{THR}_{2,n}}) \leqslant \log_3(n) + \log_3(\log_3(n)) + 6$,
(2) $\mathrm{D}_s^{hd}(\mathrm{KW}_{\mathrm{THR}_{3,n}}) \leqslant \log_3(n) + \log_3(\log_3(n)) + 8$,
(3) $\mathrm{D}_s^{hd}(\mathrm{KW}_{\mathrm{THR}_{4,n}}) \leqslant \log_3(n) + 3\log_3(\log_9(n)) + 12$

The proofs are based on ternary search supplemented with additional information about the structure of the players' inputs, which the players exchange during the protocol. Developing these ideas, we prove a generalized upper bound stated in Theorem 14. Moreover, under additional restrictions on the players' inputs, we obtain improved bounds in all three half-duplex models. To state these results, we first introduce several definitions.

Definition 1. A hypergraph $G_h = (V(G_h), E(G_h))$ consists of a vertex set $V(G_h)$ and a set of hyperedges $E(G_h) = \{e_1, \dots, e_m\}$, where each $e_i \subseteq V(G_h)$ and $|e_i| \geqslant 2$ for all $i \in \{1, \dots, m\}$.

Definition 2. A hypergraph G_h is called *k-uniform* if every hyperedge contains exactly k vertices.

Definition 3. The *degree* of a vertex v in a hypergraph G_h is the number of hyperedges containing v and is denoted by $\deg_{G_h}(v)$.

Definition 4. The *codegree* of two distinct vertices u and v is the number of hyperedges that contain both u and v and is denoted by $\operatorname{codeg}_{G_h}(u, v)$.

Definition 5. The *chromatic index* $\chi(G_h)$ of a hypergraph G_h is the minimum number of colors needed to color the hyperedges so that any two intersecting hyperedges receive different colors.

We will use the following bound on the chromatic index of a hypergraph, which can be viewed as an analogue of Vizing's theorem. Define

$$\begin{aligned} D(G_h) &= \max_{v \in V(G_h)} \deg_{G_h}(v), \\ d(G_h) &= \min_{v \in V(G_h)} \deg_{G_h}(v), \\ C(G_h) &= \max_{v,u \in V(G_h), u \neq v} \operatorname{codeg}_{G_h}(u, v). \end{aligned}$$

Theorem 13 (Pippenger and Spencer [16])**.** *For any $k \geqslant 2$ and $\delta > 0$, there exists $\delta' > 0$ and n_0 such that if a k-*uniform hypergraph G_h *satisfies*

(1) $d(G_h) \geqslant (1 - \delta')D(G_h)$,
(2) $C(G_h) \leqslant \delta' D(G_h)$,

then for any such G_h on n vertices with $n \geqslant n_0$ one has

$$\chi(G_h) \leqslant (1 + \delta)D(G_h).$$

Theorem 14. *For $n > 2$,*

$$\mathrm{D}_s^{hd}(\mathrm{KW}_{\mathrm{THR}_{k,n}}) \leqslant \log_3 n + (k-2)\log_3(\log_k n) + (k-1)^2 \log_3(k-1) + k^2 - 2.$$

Proof. Note that Alice can keep only k ones in her string, replacing the rest with 0, and then it is necessary and sufficient for the players to find a position index i such that $x_i = 1$ and $y_i = 0$.

Before describing the algorithm for finding the required position, let us describe a tree structure that will help the players do this. Each player divides their input string into k substrings and for each substring computes $\mathrm{THR}_{1,\frac{n}{k}}$. The players will repeat the previous step for each new substring until its length becomes 1. As a result, the players will have a tree where each vertex contains the value of $\mathrm{THR}_{1,\frac{n}{k^i}}$ for the corresponding substring of length $\frac{n}{k^i}$, where i is the level number in the tree. Note that each player will have their own tree, corresponding to the player's input. Denote these trees as T_A and T_B, respectively. Also note that finding a differing bit is the same as finding a leaf number where Alice's tree has 1 and Bob's tree has 0.

After constructing T_A and T_B, the players run an iterative procedure. In each iteration they move from the current trees to a pair of corresponding subtrees in which Bob has fewer ones in the leaves. Therefore, the number of iterations is at most $k - 1$.

We now describe one iteration.

- Bob finds the first level l_0^B in the current tree at which there are at least two vertices labeled with 1, and sends l_0^B to Alice. He encodes l_0^B in ternary and transmits it using $\log_3(\log_k n)$ rounds as follows: ternary digit 0 corresponds to the action *send* 0, digit 1 corresponds to *send* 1, and digit 2 corresponds to *receive* (i.e., sending the special silence symbol).
- Alice inspects level l_0^B in her tree. Let us distinguish the following cases.
  (1) **Alice has exactly one 1-vertex at level l_0^B.** Alice informs Bob of this in one round by sending 0. Then, for each level up to l_0^B, Alice tells Bob which child to follow in order to reach her unique 1-vertex. After that, the players replace T_A and T_B by the corresponding subtrees rooted at these vertices (denote them by T'_A and T'_B) and repeat the iteration on (T'_A, T'_B). This requires $l_0^B \cdot \log_3(k)$ rounds. Note that the number of 1-leaves in Bob's new tree decreases by at least one.

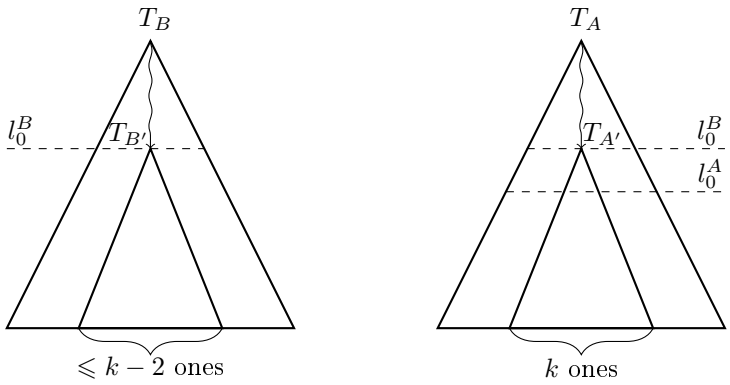

Figure 1. Case 1.

(2) **Alice has at least two 1-vertices at level l_0^B.** Let l_0^A be the first level in Alice's tree at which the number of 1-vertices becomes at least two. Then $l_0^A \leqslant l_0^B$. We consider two subcases.

Subcase $l_0^A < l_0^B$. Alice informs Bob of this in one round by sending 1 (see Figure 2). At level l_0^A Bob has exactly one 1-vertex (by the choice of l_0^B), while Alice has at least two. Therefore, there exists a vertex at level l_0^A where Alice has label 1 and Bob has label 0.
To find such a vertex, for each level up to $l_0^A - 1$ Alice tells Bob which child to follow so that they reach a vertex labeled 1 in T_A. Bob then transmits to Alice the labels of the k children of the corresponding vertex in T_B (as a bit string of length k), which takes k rounds. After receiving this information, Alice chooses a child where she has label 1 and Bob has label 0; once such a child is fixed, Alice can direct Bob to a differing leaf in the corresponding subtree. Communicating the remaining path to a leaf takes $(\log_k n - l_0^A) \cdot \log_3(k)$ rounds.
Hence, the total number of rounds in this subcase is

$$(\log_k n - l_0^A) \cdot \log_3(k) + (l_0^A - 1) \cdot \log_3(k) + k = \log_3(n) - \log_3(k) + k.$$

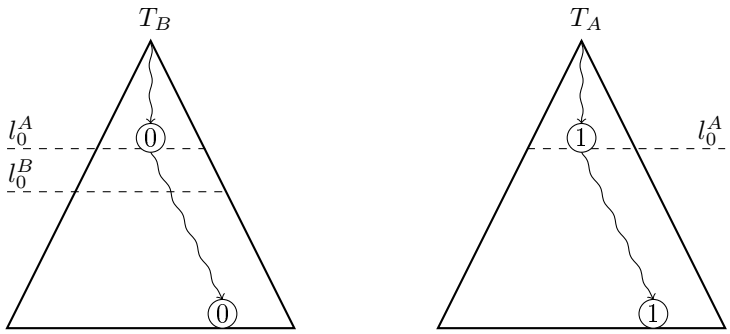

Figure 2. Case 2.

Subcase $l_0^A = l_0^B$. Alice informs Bob of this case in one round by choosing the action *receive* (see Figure 3). Alice leads Bob to the subtree rooted at the (unique) 1-vertex at level $l_0^A - 1$, which takes $(l_0^A - 1) \cdot \log_3(k)$ rounds. Then Bob transmits to Alice the labels of the k children of the corresponding vertex in T_B in k rounds.

If the received bit string contains no ones, then Alice can immediately specify a path to a differing leaf; this requires $(\log_k n - l_0^A) \cdot \log_3(k)$ rounds. Otherwise, Bob additionally transmits to Alice, for each 1 in the string, the number of 1-leaves in the corresponding subtree; this requires at most $(k-1) \cdot \log_3(k-1)$ rounds. Alice then chooses a child at level l_0^A whose subtree contains more 1-leaves in T_A than in T_B (such a child exists since initially Alice has more ones than Bob). Alice communicates the index of this child in $\log_3(k)$ rounds, and the players replace T_A, T_B by the corresponding subtrees T'_A, T'_B and continue. Note that the number of 1-leaves in Bob's new tree decreases by at least one.

Hence, the number of rounds in this subcase is at most

$$(l_0^A - 1) \cdot \log_3(k) + k + (k-1) \cdot \log_3(k-1).$$

- Finally, consider the extreme situation when the players have already reached level $\log_k n$. Then they are considering a block of k leaves of the original trees in which Alice has more ones than Bob. Bob transmits the labels of his k leaves to Alice in k rounds, and

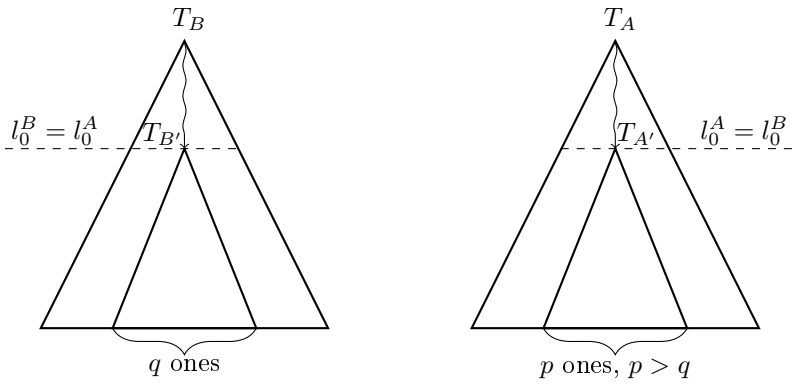

Figure 3. Case 3.

then Alice identifies an index where she has 1 and Bob has 0 and communicates this index to Bob in $\log_3(k)$ rounds. Thus, this case requires $k + \log_3(k)$ rounds.

- The players repeat the above iteration (with updated trees) until they find a leaf corresponding to a differing bit.

We now bound the total number of rounds.

(1) Let $l_0^*, l_1^*, \ldots, l_j^*$ be the sequence of levels at which the players stop. Since the players ultimately traverse a root-to-leaf path, we have $\sum_{i=1}^{j} l_i^* = \log_k n$. Communicating a child index at each level costs $\log_3(k)$ rounds, hence specifying the full path costs at most $\log_k n \cdot \log_3(k) = \log_3(n)$ rounds.

(2) During the protocol, Bob sends to Alice at most $k-2$ values of the form l_i^B. Each such value is at most $\log_k n$ and can be transmitted in at most $\log_3(\log_k n)$ rounds. Thus, this contributes at most $(k-2)\cdot\log_3(\log_k n)$ rounds. In addition, Alice uses at most $k-2$ rounds to indicate which case applies. Altogether, this part costs at most $(k-2)\cdot\log_3(\log_k n) + k - 2$ rounds.

(3) In Cases 2 and 3, Bob transmits to Alice the labels of k children (cost k rounds), and in Case 3 he may also transmit the numbers of 1-leaves (cost at most $(k-1)\cdot\log_3(k-1)$ rounds). These transmissions occur at most $k-1$ times, hence their total cost is at

most

$$\sum_{i=1}^{k-1}\Big((k-i)\cdot\log_3(k-i)+k\Big) < (k-1)^2\cdot\log_3(k-1)+k\cdot(k-1).$$

(4) Combining the bounds above, the total number of rounds is at most

$$\log_3(n)+(k-2)\cdot\log_3(\log_k n)+(k-1)^2\cdot\log_3(k-1)+k^2-2,$$

which proves the theorem. □

This bound can be improved under additional restrictions on the players' inputs. Namely, assume that Alice receives a string $x \in \mathrm{THR}_{k,n}(1)^{-1}$, Bob receives a string $y \in \mathrm{THR}_{k,n}(0)^{-1}$, and the following two conditions hold:

(1) if $y_i = 1$, then $x_i = 1$,
(2) $\sum_{i=1}^{n} x_i = k$ and $\sum_{i=1}^{n} y_i = k-1$.

Theorem 15. *If the following conditions hold:*

(1) *if* $y_i = 1$, *then* $x_i = 1$,
(2) $\sum_{i=1}^{n} x_i = k$ *and* $\sum_{i=1}^{n} y_i = k-1$,

then the following bounds are achieved:

(1) $\mathrm{D}_s^{hd}(\mathrm{KW}_{\mathrm{THR}_{k,n}}) \leqslant \log_3\big(k(n-k+1)\big)$,
(2) $\mathrm{D}_0^{hd}(\mathrm{KW}_{\mathrm{THR}_{k,n}}) \leqslant \log_2\big(k(n-k+1)\big)$,
(3) $\mathrm{D}_a^{hd}(\mathrm{KW}_{\mathrm{THR}_{k,n}}) \leqslant \log_2\big(k(n-k+1)\big)$.

Proof. We describe a general construction that allows the players to find a differing bit within the claimed number of rounds. Before the communication starts, the players construct a graph G that captures the possible inputs and the location of the differing bit.

Given their inputs, the players form the sets of indices of ones: $X \subseteq \{1,2,\ldots,n\}$ for Alice and $Y \subseteq \{1,2,\ldots,n\}$ for Bob. By the assumptions, $|X| = k$, $|Y| = k-1$, and $Y \subseteq X$. Therefore, there exists a unique index $i \in X \setminus Y$ such that $x_i = 1$ and $y_i = 0$, and the players' goal is to identify this index.

Define the graph G as follows. The vertex set consists of all k-subsets of $[n]$:

$$V(G) = \{S \subseteq [n] : |S| = k\}.$$

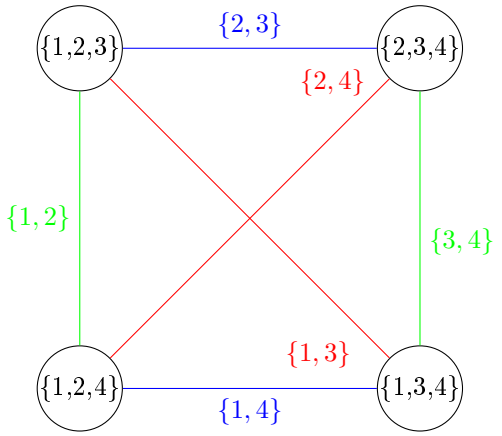

Figure 4. Example construction for $n = 4$ and $k = 3$. Here each clique has size $n-k+1 = 2$. Vertices represent possible sets of indices of ones in Alice's input, and edges represent possible sets of indices of ones in Bob's input. Each edge is labeled by the intersection of the endpoint sets.

Two vertices $U, W \in V(G)$ are connected by an edge if and only if $|U \cap W| = k-1$, that is,

$$E(G) = \{(U, W) : U, W \in V(G),\ |U \cap W| = k-1\}.$$

Vertices of G correspond to possible inputs of Alice. An edge (U, W) corresponds to a possible input of Bob, namely the $(k-1)$-set $U \cap W$. Fix such a $(k-1)$-set E. Then the family of all k-sets that contain E has size $n-k+1$, and any two of them are adjacent; hence it forms a clique of size $n-k+1$. Moreover, each vertex U belongs to exactly k such cliques, since U contains exactly k distinct $(k-1)$-subsets.

After constructing G, the players color the cliques. For this, we view the family of cliques as a hypergraph G_h on the same vertex set $V(G)$, where each hyperedge corresponds to a clique of G. To apply Theorem 13, we use the following lemma.

Lemma 3. *The hypergraph G_h satisfies the conditions of Theorem 13, namely:*

(1) *G_h is m-uniform,*

(2) $d(G_h) \geqslant (1-\delta')D(G_h)$,
(3) $C(G_h) \leqslant \delta' D(G_h)$.

Proof. We verify the conditions one by one.

(1) By construction, each hyperedge corresponds to a clique induced by a fixed $(k-1)$-set, hence it contains exactly $m = n-k+1$ vertices. Therefore, G_h is m-uniform.
(2) Each vertex $U \in V(G_h)$ belongs to exactly k hyperedges (one for each $(k-1)$-subset of U). Thus $d(G_h) = D(G_h) = k$, and the inequality holds for any $\delta' > 0$.
(3) We claim that $C(G_h) = 1$. Suppose, towards a contradiction, that there exist two distinct vertices U, W contained together in two distinct hyperedges E_1 and E_2. Let e_1 and e_2 be the corresponding $(k-1)$-subsets of $[n]$ defining E_1 and E_2. Since $U \in E_1 \cap E_2$, we have $e_1 \subseteq U$ and $e_2 \subseteq U$, hence $|e_1 \cap e_2| = k-2$. Let $e = e_1 \cap e_2$ and write $e_1 = e \cup \{a_1\}$ and $e_2 = e \cup \{a_2\}$ with $a_1 \neq a_2$. Since U contains both e_1 and e_2 and has size k, it must be of the form $U = e \cup \{a_1, a_2\}$. The same argument applies to W, giving $W = e \cup \{a_1, a_2\} = U$, a contradiction. Hence any two vertices are contained together in at most one hyperedge, and so $C(G_h) = 1$. Taking $\delta' = \frac{1}{k}$ yields $C(G_h) \leqslant \delta' D(G_h)$. □

By Theorem 13, for sufficiently large n the hyperedges of G_h can be properly colored with k colors. Equivalently, the cliques of G can be colored with k colors so that any two cliques sharing a vertex receive different colors.

After the coloring is fixed, Bob sends to Alice the color of the clique corresponding to his input y. Then Alice sends Bob the index (within this clique) of her vertex. This identifies the unique index $i \in X \setminus Y$, and hence the differing bit.

Therefore:

(1) In the half-duplex silence model, the players transmit numbers in ternary, requiring at most $\log_3(k) + \log_3(n-k+1)$ rounds, hence $\mathrm{D}_s^{hd}(\mathrm{KW}_{\mathrm{THR}_{k,n}}) \leqslant \log_3\big(k(n-k+1)\big)$.
(2) In the half-duplex zero model, the same strategy uses binary encodings and requires at most $\log_2(k)+\log_2(n-k+1)$ rounds, hence $\mathrm{D}_0^{hd}(\mathrm{KW}_{\mathrm{THR}_{k,n}}) \leqslant \log_2\big(k(n-k+1)\big)$.
(3) In the half-duplex adversary model, the players also use binary encodings, giving $\mathrm{D}_a^{hd}(\mathrm{KW}_{\mathrm{THR}_{k,n}}) \leqslant \log_2\big(k(n-k+1)\big)$. □

Note that in the adversary model, the obtained upper bound under the above restrictions matches the corresponding lower bound.

Corollary 1. $\mathrm{D}_a^{hd}(\mathrm{KW}_{\mathrm{THR}_{k,n}}) = \log_2\bigl(k(n-k+1)\bigr)$.

Similar bounds can be obtained for the special case of the threshold function when $k = \frac{n}{2}$.

Corollary 2. *If the following conditions hold:*

(1) *if* $y_i = 1$, *then* $x_i = 1$,
(2) $\sum_{i=1}^n x_i = \frac{n}{2}$ *and* $\sum_{i=1}^n y_i = \frac{n}{2} - 1$,

then the following bounds hold:

(1) $\mathrm{D}_s^{hd}(\mathrm{KW}_{\mathrm{Maj}_n}) \leqslant 2\log_3(n)$,
(2) $\mathrm{D}_0^{hd}(\mathrm{KW}_{\mathrm{Maj}_n}) \leqslant 2\log_2(n)$,
(3) $\mathrm{D}_a^{hd}(\mathrm{KW}_{\mathrm{Maj}_n}) \leqslant 2\log_2(n)$.

Moreover, in the half-duplex adversary model, the upper bound coincides with the lower bound.

Corollary 3. $\mathrm{D}_a^{hd}(\mathrm{KW}_{\mathrm{Maj}_n}) = 2\log_2(n)$.

In the half-duplex zero model, the obtained bound can be improved.

Theorem 16. *If the following conditions hold:*

(1) *if* $y_i = 1$, *then* $x_i = 1$,
(2) $\sum_{i=1}^n x_i = \frac{n}{2}$ *and* $\sum_{i=1}^n y_i = \frac{n}{2} - 1$,

then

$$\mathrm{D}_0^{hd}(\mathrm{KW}_{\mathrm{Maj}_n}) \leqslant 3\log_3(n).$$

Proof. Since the players' input strings differ by exactly one 1, they have different parity. Therefore, we can apply the protocol from Theorem 6 in [6] and obtain the desired bound. □

§5. Non-deterministic half-duplex complexity

The standard definition of the non-deterministic communication complexity does not involve any communication at all, so it is applicable to the half-duplex model without any changes. Let X and Y be non-empty finite sets.

Definition 6. We say that a function $f : X \times Y \to \{0,1\}$ has *non-deterministic communication protocol* of complexity d if there are two functions $A : X \times \{0,1\}^d \to \{0,1\}$ and $B : Y \times \{0,1\}^d \to \{0,1\}$, such that

- $\forall (x,y) \in f^{-1}(1)\ \exists w \in \{0,1\}^d : A(x,w) = B(y,w) = 1$,
- $\forall (x,y) \in f^{-1}(0)\ \forall w \in \{0,1\}^d : A(x,w) \neq 1 \vee B(y,w) \neq 1$.

The non-deterministic communication complexity of f, denoted $\mathrm{N}(f)$, is the minimal complexity of a non-deterministic communication protocol for f.

There is also an alternative definition of the non-deterministic communication complexity that uses communication between players [15].

Definition 7. We say that a function $f : X \times Y \to \{0,1\}$ has *privately non-deterministic communication protocol* of complexity d if there is a function $\hat{f} : (X \times \{0,1\}^*) \times (Y \times \{0,1\}^*) \to \{0,1\}$ of (deterministic) communication complexity at most d such that

- $\forall (x,y) \in f^{-1}(1)\ \exists w_x, w_y \in \{0,1\}^* : \hat{f}((x,w_x),(y,w_y)) = 1$,
- $\forall (x,y) \in f^{-1}(0)\ \forall w_x, w_y \in \{0,1\}^* : \hat{f}((x,w_x),(y,w_y)) = 0$.

The privately non-deterministic communication complexity of f is the minimal depth of a privately non-deterministic communication protocol for f.

This alternative definition of non-deterministic communication uses private witnesses instead of a public one, and hence the players need to communicate. In the classical case, Definition 6 and Definition 7 are equivalent (see [15] for more details). We think that this way of defining it is the right way to define *the non-deterministic half-duplex communication complexity*. So, we define it by replacing the communication model in Definition 7 with the half-duplex models. Let $\mathrm{N}^{hd}_s(f)$, $\mathrm{N}^{hd}_0(f)$, and $\mathrm{N}^{hd}_a(f)$ denotes *the non-deterministic half-duplex communication complexity of f with silence, with zero, and with adversary*, respectively. We are going to prove bounds that connect the classical non-deterministic communication complexity with the non-deterministic half-duplex communication complexity.

Let's start with the lower bounds.

Theorem 17. *For any function $f : X \times Y \to \{0,1\}$,*

$$\mathrm{N}^{hd}_s(f) \geqslant \mathrm{N}(f)/\log 5, \quad \mathrm{N}^{hd}_0(f) \geqslant \mathrm{N}(f)/\log 3, \quad \mathrm{N}^{hd}_a(f) \geqslant \mathrm{N}(f)/\log 3.$$

Proof. In the half-duplex model with silence, the protocol is a pair of trees of arity 5. The following (classical) non-deterministic protocol can simulate any non-deterministic half-duplex protocol Π with silence. Alice and Bob publicly guess a root-to-leaf path π_A in the tree of Π corresponding to Alice. Alice checks that this transcript is a valid transcript for her input. Bob checks that there exists a root-to-leaf path π_B in his tree that is a valid

transcript for his input, and at the same time π_B matches π_A (in all rounds where Alice receives in π_A, Bob in π_B does the corresponding action, and in all rounds where Alice sends, Bob receives the corresponding bit or sends some bit). If Π has complexity d, then the length of the description of π_A is $\lceil d \cdot \log 5 \rceil$. This gives us the first lower bound.

Similarly, in the half-duplex model with zero, the protocol is a pair of trees of arity 3. The same reasoning shows that for any non-deterministic half-duplex protocol Π with zero of complexity d, there exists a (classical) non-deterministic protocol of complexity $\lceil d \cdot \log 3 \rceil$.

In the half-duplex model with adversary, we can consider only such transcripts where the players always receive zeros in silent rounds. Thus, the lower bound for the half-duplex model with zero applies. □

The upper bounds are based on the upper bounds for the equality function. We do not know any non-trivial upper bounds for the equality in the half-duplex model with adversary, so we last upper bound is trivial.

Theorem 18. *For any function* $f : X \times Y \to \{0,1\}$,

$$\begin{aligned}
\mathrm{N}_s^{hd}(f) &\leqslant \mathrm{N}(f)/\log 5 + O(\log \mathrm{N}(f)),\\
\mathrm{N}_0^{hd}(f) &\leqslant \mathrm{N}(f)/\log 3 + O(\log \mathrm{N}(f)),\\
\mathrm{N}_a^{hd}(f) &\leqslant \mathrm{N}(f).
\end{aligned}$$

Proof. For any (classical) non-deterministic protocol Π, Alice and Bob can privately guess a public witness w and then check that they both guessed the same witness. This requires solving the equality problem on strings of length $N(f)$. Together with the upper bounds on the equality [8, Theorems 15 and 19], this gives us the desired bounds. □

§6. Open problems

In addition to the open questions in [9], we state the following open problems.

(1) Prove new lower bound for disjointness using information-theoretic methods.
(2) Prove an upper bound for the KW game for $\mathrm{MOD}p_n$ in the model with zero.
(3) Prove better upper bound for RecMaj_n in the model with silence.

References

1. R. B. Boppana, *Amplification of Probabilistic Boolean Formulas*, in: 2013 IEEE 54th Annual Symposium on Foundations of Computer Science, IEEE Computer Society, 1985, pp. 20—29.
2. G. S. Brodal and T. Husfeldt, *A Communication Complexity Proof that Symmetric Functions Have Logarithmic Depth*, BRICS, 1996.
3. A. Chin, *On the Depth Complexity of the Counting Functions.* — Information Processing Letters **35** (1990), no. 6, 325–328.
4. T. M. Cover and J. A. Thomas, *Elements of Information Theory*, Wiley-Interscience, 2006.
5. Y. Dementiev, A. Ignatiev, V. Sidelnik, A. Smal, and M. Ushakov, *New Bounds on the Half-Duplex Communication Complexity.* – Electronic Colloquium on Computational Complexity **27** (2020), 117. Available: https://eccc.weizmann.ac.il/report/2020/117.
6. Y. Dementiev, A. Ignatiev, V. Sidelnik, A. Smal, and M. Ushakov, *New Bounds on the Half-Duplex Communication Complexity*, in: SOFSEM 2021: Theory and Practice of Computer Science, Lecture Notes in Computer Science **12607**, Springer, 2021, pp. 233–248. Available: https://doi.org/10.1007/978-3-030-67731-2_17.
7. J. Edmonds, R. Impagliazzo, S. Rudich, and J. Sgall, *Communication Complexity Towards Lower Bounds on Circuit Depth.* — Computational Complexity **10** (2001), no. 3, 210–246.
8. K. Hoover, R. Impagliazzo, I. Mihajlin, and A. Smal, *Half-Duplex Communication Complexity.* — Electronic Colloquium on Computational Complexity **25** (2018), 89. Available: https://eccc.weizmann.ac.il/report/2018/089.
9. K. Hoover, R. Impagliazzo, I. Mihajlin, and A. V. Smal, *Half-Duplex Communication Complexity*, in: 29th International Symposium on Algorithms and Computation (ISAAC 2018), LIPIcs **123**, 2018, pp. 10:1–10:12. Available: https://doi.org/10.4230/LIPIcs.ISAAC.2018.10.
10. M. Karchmer, R. Raz, and A. Wigderson, *Super-Logarithmic Depth Lower Bounds via the Direct Sum in Communication Complexity.* — Computational Complexity **5** (1995), no. 3/4, 191–204.
11. M. Karchmer and A. Wigderson, *Monotone Circuits for Connectivity Require Super-Logarithmic Depth*, in: Proceedings of the 20th Annual ACM Symposium on Theory of Computing, 1988, pp. 539–550.
12. V. M. Khrapchenko, *Complexity of the Realization of a Linear Function in the Class of* Π*-Circuits.* — Mathematical Notes of the Academy of Sciences of the USSR **9** (1971), no. 1, 21–23.
13. E. Kushilevitz and N. Nisan, *Communication Complexity*, Cambridge University Press, 1997.
14. S. Laplante, T. Lee, and M. Szegedy, *The Quantum Adversary Method and Classical Formula Size Lower Bounds*, in: 20th Annual IEEE Conference on Computational Complexity (CCC 2005), 2005, pp. 76–90. Available: https://doi.org/10.1109/CCC.2005.29.

15. I. Mihajlin and A. Smal, *Toward Better Depth Lower Bounds: The XOR-KRW Conjecture.* — Electronic Colloquium on Computational Complexity **27** (2020), 116. Available: https://eccc.weizmann.ac.il/report/2020/116.
16. N. Pippenger and J. Spencer, *Asymptotic Behavior of the Chromatic Index for Hypergraphs.* — Journal of Combinatorial Theory, Series A **51** (1989), no. 1, 24–42.
17. J. Radhakrishnan, *Better Lower Bounds for Monotone Threshold Formulas.* — Journal of Computer and System Sciences **54** (1997), 221–226.
18. A. Yao, *Some Complexity Questions Related to Distributive Computing (Preliminary Report)*, in: Proceedings of the 11th Annual ACM Symposium on Theory of Computing, 1979, pp. 209–213.
19. В. А. Сидельник, *Новые оценки на полудуплексную коммуникационную сложность игр Карчмера–Вигдерсона для булевых функций*, 2022. Available: http://hdl.handle.net/11701/40394.

Saint Petersburg State University,
St. Petersburg Department of Steklov
Mathematical Institute of Russian Academy of Sciences

E-mail:, Y. I. Dementiev `yuru.dementiev@gmail.com`

E-mail:, A. A. Ignatiev `artur.ignatev@csspace.io`

E-mail:, V. A. Sidelnik `slava.sidelnik@gmail.com`

E-mail:, A. V. Smal `avsmal@gmail.com`

E-mail:, M. S. Ushakov `mikushakov@gmail.com`

Поступило 24 февраля 2026 г.