An application of communication complexity, Kolmogorov complexity and extremal combinatorics to parity games

Alexander Kozachinskiy*[1] and Mikhail Vyalyi†[1,2,3]

[1]National Research University Higher School of Economics, Russian Federation
[2]Moscow Institute of Physics and Technology, Russian Federation
[3]Dorodnicyn Computing Centre, FRC CSC RAS, Russian Federation

Abstract

So-called separation automata are in the core of several recently invented quasi-polynomial time algorithms for parity games. An explicit q-state separation automaton implies an algorithm for parity games with running time polynomial in q. It is open whether a polynomial-state separation automaton exists. A positive answer will lead to a polynomial-time algorithm for parity games, while a negative answer will at least demonstrate impossibility to construct such an algorithm using separation approach.

In this work we prove exponential lower bound for a restricted class of separation automata. Our technique combines communication complexity and Kolmogorov complexity. One of our technical contributions belongs to extremal combinatorics. Namely, we prove a new upper bound on the product of sizes of two families of sets with small pairwise intersection.

1 Introduction

For a game with two competitive players one can consider a problem of deciding which player has a winning strategy. Parity games are classical example when this problem lies in NP∩coNP yet for which no polynomial-time algorithm is known. To specify an instance of a parity game one needs to specify:

- n-node directed graph in which any node has at least one outgoing-edge;

*akozachinskiy@hse.ru
†vyalyi@gmail.com

- indicated initial node;
- labeling of edges by integers from $\{1, 2, \ldots, d\}$ (*priorities*);
- partition of nodes into two parts, V_0 and V_1.

There are two players named *Player 0* (Even) and *Player 1* (Odd). A position of a game is specified by a node of a graph. It is possible to move from node u to node v if and only if (u, v) is an edge of a graph. For each node u it is predetermined which player makes a move in u. Namely, Player 0 makes a move in V_0 and Player 1 makes a move in V_1.

Since all nodes have out-going edges, a play can always last for infinite number of moves. In this way we obtain an infinite sequence of nodes $\{v_k\}_{k=1}^{\infty}$ visited by players. We can also look at the sequence of corresponding priorities. Namely, let l_k be a priority of an edge (v_k, v_{k+1}). Winning conditions in parity game are the following: Player i wins if and only if

$$\limsup_{k\to\infty} l_k \equiv i \pmod 2.$$

Such a winning condition is Borel, which means due to Martin's theorem ([18]) that either Player 0 or Player 1 has a winning strategy. Moreover, it turns out that a player having a winning strategy in a parity game has also a *memoryless* winning strategy, i.e. one in which every move depends only on a current node ([7, 20]). This fact means a lot for the complexity of ParityGames, a problem of determining the winner of a parity game. Namely, due to this fact ParityGame is in NP∩coNP (a short certificate for a player is his memoryless winning strategy). More involved argument shows that actually ParityGames is in UP∩coUP ([14]).

All this leaves a hope that ParityGames is solvable in polynomial time. Yet this is still an open problem. A lot of work was done to improve an obvious n^n-time algorithm checking all memoryless strategies (see, e.g., [21, 16, 19, 22]). This finally led in 2017 to a *quasi-polynomial* time algorithm for ParityGames:

Theorem 1 (Calude et. al., [4]). ParityGames *with n nodes and d priorities can be solved in $n^{O(\log d)}$ time.*

Although we made no assumptions on d but it is clear that we can always reduce a given instance of parity game to one in which d is linear in the number of edges. Thus in a worst case the algorithm of Calude et. al. takes $n^{O(\log n)}$ time.

1.1 Separation approach to parity games

Since the paper of Calude et. al., several other quasi-polynomial time algorithms were invented for ParityGames ([15, 9, 17]). The paper of Czerwiński et. al. ([6]) argues that all these works follow so-called *separation approach*. Let us briefly summarize this approach.

The main idea is to reduce ParityGames to *reachability games*. To specify a reachability game one needs to specify a graph and mark some of its nodes as winning. The goal of one player is to visit one of winning nodes at least once. Correspondingly, the goal of the other player is to avoid winning nodes.

A standard analysis of complete information games works also for reachability games, which leads to a polynomial-time algorithm for the these games. In separation approach, a parity game on a graph G with n nodes is reduced to a reachability game on a *product* of G and the transition graph of some specific deterministic finite automaton $\mathcal{A}$.

The input alphabet of $\mathcal{A}$ is a set $\{1, \ldots, n\} \times \{1, \ldots, d\}$ (pairs of the form $\langle \text{a node of } G, \text{a priority}\rangle$). We use this alphabet to encode infinite paths in n-node labeled graphs with d priorities. Namely, assume that an infinite path starts with node v_1, then goes to v_2, then to v_3 and so on. Moreover, assume that the priority of the first edge in a path is p_1, the priority of the second edge in a path is p_2 etc. Then this path corresponds to the infinite sequence $(v_1, p_1)(v_2, p_2)(v_3, p_3)\ldots$ over the alphabet $\{1, \ldots, n\} \times \{1, \ldots, d\}$. In what follows by saying that $\mathcal{A}$ does something on an infinite path we mean that $\mathcal{A}$ does something on the input sequence corresponding to this path.

To make a reduction correct, we impose the following requirement on $\mathcal{A}$. There should be a state q_{accept} of $\mathcal{A}$ with the following properties:

- $\mathcal{A}$ reaches q_{accept} on all paths produced by memoryless strategies of Player 0 which are winning for some n-node graph with d priorities.

- $\mathcal{A}$ never reaches q_{accept} on any path produced by a memoryless strategy of Player 1 which is winning for some n-node graph with d priorities.

Automata satisfying the above requirements are called *separation automata*. It follows immediately from definition that a memoryless winning strategy in a parity game on G yields a winning strategy in a reachability game on a product of G and $\mathcal{A}$, where $\mathcal{A}$ is a separation automaton and winning nodes correspond to a state q_{accept}. Thus indeed to solve a parity game on G it is enough to solve such a reachability game, and this takes time polynomial in the number of states of $\mathcal{A}$.

It is possible to simplify a little bit a definition of separation automata. Take any winning positional strategy of Player i in a parity game on G. Notice that if we remove all edges contradicting this strategy from G, then we obtain a graph in which, on any cycle, the maximum of priorities has the same parity as i. A graph is called *even* (*odd*) if this maximum of priorities is even (odd) for all cycles.

In [6], Czerwiński et. al. define the following two languages consisting of infinite words over $\{1, \ldots, n\} \times \{1, \ldots, d\}$. Denote by $\text{EvenCycles}_{n,d} \subseteq (\{1, \ldots, n\} \times \{1, \ldots, d\})^{\mathbb{N}}$ the set of all inputs sequences to $\mathcal{A}$ which correspond to some infinite path in an even graph with at most n nodes and d priorities. Define $\text{OddCycles}_{n,d}$ similarly. Now from the observation above it follows that we can define separation automata equivalently as follows: $\mathcal{A}$ should reach q_{accept}

on sequences from $\mathrm{EvenCycles}_{n,d}$ and should avoid q_{accept} on sequences from $\mathrm{OddCycles}_{n,d}$.

As far as we know, before [6] a formalization of separation approach appears in a textbook of Bojańczyk and Czerwiński [2]. However, instead of $\mathrm{EvenCycles}_{n,d}$ and $\mathrm{OddCycles}_{n,d}$ they used another two languages, $\mathrm{EvenLoops}_{n,d}$ and $\mathrm{OddLoops}_{n,d}$[1]. Namely, $\mathrm{EvenLoops}_{n,d}$ ($\mathrm{OddLoops}_{n,d}$) consists of all infinite paths in which the maximum of priorities between any two visits of a same node is always even (odd). It is clear that $\mathrm{EvenCycles}_{n,d} \subsetneq \mathrm{EvenLoops}_{n,d}$ and $\mathrm{OddCycles}_{n,d} \subsetneq \mathrm{OddLoops}_{n,d}$. Thus it is easier to construct separation automata in a sense of [6] than in a sense of [2]. Correspondingly, it is easier to obtain lower bounds against the latter than against the former. We stress that in this paper we follow the approach of [6], i.e., we use $\mathrm{EvenCycles}_{n,d}$ and $\mathrm{OddCycles}_{n,d}$.

To describe the main lower bound of [6] we shall introduce two more sets of infinite sequences from $(\{1,\ldots,n\}\times\{1,\ldots,d\})^{\mathbb{N}}$. Namely, let $\mathrm{LimSupEven}_{n,d}$ be the set of all sequences $(v_1,p_1)(v_2,p_2)(v_3,p_3) \in (\{1,\ldots,n\}\times\{1,\ldots,d\})^{\mathbb{N}}$ satisfying $\limsup\limits_{i\to\infty} p_i \equiv 0 \pmod 2$. Define $\mathrm{LimSupEven}_{n,d}$ similarly. Again, it is clear that

$$\mathrm{EvenCycles}_{n,d} \subsetneq \mathrm{LimSupEven}_{n,d}, \qquad \mathrm{OddCycles}_{n,d} \subsetneq \mathrm{LimSupOdd}_{n,d}.$$

First Czerwiński et. al. demonstrate that actually all quasi-polynomial time algorithms for parity games listed above provide a quasi-polynomial-state automaton separating $\mathrm{LimSupEven}_{n,d}$ from $\mathrm{OddCycles}_{n,d}$ (in the same sense of separation as above — an automaton should reach q_{accept} on sequences from the first set an avoid q_{accept} from sequences of the second set). It is more than required in separation approach — however, no quasi-polynomial state automaton doing "no more than required" is known.

On the other hand Czerwiński et. al. show that any automaton separating $\mathrm{LimSupEven}_{n,d}$ from $\mathrm{OddCycles}_{n,d}$ has $n^{\Omega(\log d)}$ number of states. This exactly matches known constructions. To obtain such a lower bound they introduce a combinatorial object called "universal trees" and show that automata separating $\mathrm{LimSupEven}_{n,d}$ from $\mathrm{OddCycles}_{n,d}$ should contain a universal tree within the set of its states. Then they prove a quasi-polynomial lower bound on universal trees.

It is not clear how to generalize this technique to separation $\mathrm{EvenCycles}_{n,d}$ from $\mathrm{OddCycles}_{n,d}$ (for which no better lower bound that just n is known). One of the obstacles is that the lower bound based on universal trees works also for non-deterministic automata. At the same time separation of $\mathrm{EvenCycles}_{n,d}$ and $\mathrm{OddCycles}_{n,d}$ is very easy with non-determinism allowed — just guess a node appearing more then once and compute the maximum between two occurrences of this node.

[1] Actually, [2] contains no name for these two languages and we use a terminology of [6]

1.2 Our contribution

We attack the question of obtaining lower bound on automata separating $\mathrm{EvenCycles}_{n,d}$ and $\mathrm{OddCycles}_{n,d}$. To do so we first relax a notion of separation automata by introducing an additional parameter t. Namely, recall that for any $w \in \mathrm{EvenCycles}_{n,d}$ a separation automaton should reach an accepting state on some finite prefix on w. The length of such prefix is not anyhow bounded. We suggest to simplify the problem and study it for automata in which such prefix is of length at most t.

More specifically, we say that a deterministic finite automaton separates $\mathrm{EvenCycles}_{n,d}$ from $\mathrm{OddCycles}_{n,d}$ in **time t** if for all $w \in (\{1, \ldots, n\} \times \{1, \ldots, d\})^{\mathbb{N}}$:

- if $w \in \mathrm{EvenCycles}_{n,d}$, then an automaton reaches q_{accept} while reading $w_1 w_2 \ldots w_t$ and always stays in q_{accept} after that;
- if $w \in \mathrm{OddCycles}_{n,d}$, then an automaton never reaches q_{accept} on w.

A requirement that an automaton stays in q_{accept} forever after reading $w_1 w_2 \ldots w_t$ is not essential as we can always make q_{accept} an absorbing state.

It is easy to see that a deterministic automaton separating $\mathrm{EvenCycles}_{n,d}$ from $\mathrm{OddCycles}_{n,d}$ necessarily does it in qn-time (for the sake of completeness we include the proof in Appendix A). Thus to show a lower bound $l(n, d)$ on size of separating automata it is enough to do so for those automata, which separate in $(n \cdot l(n, d))$-time.

According to this approach in order to obtain at least super-linear lower bound for unbounded time we need a lower bound for some time t such that t/n^2 is unbounded. Unfortunately, we prove lower bounds only for automata separating $\mathrm{EvenCycles}_{n,d}$ from $\mathrm{OddCycles}_{n,d}$ in at most $O(n^{5/4})$-time.

Theorem 2. *Any deterministic finite automata $\mathcal{A}$ separating* $\mathrm{EvenCycles}_{n,2}$ *from* $\mathrm{OddCycles}_{n,2}$ *in time t has* $\exp\left(\Omega(n^5/t^4)\right)$ *number of states.*

Notice that this theorem is true even for the case of 2 priorities, $d = 2$. This is in some sense bad news as there exists a simple $O(n)$-state deterministic automaton, separating $\mathrm{EvenCycles}_{n,2}$ from $\mathrm{OddCycles}_{n,2}$ in $O(n^2)$-time (namely, accept if and only if you have seen at least $n + 1$ priorities which are equal to 2). This means that to obtain a good lower bound for $\omega(n^2)$-time we need an argument for which it is crucial to have more than 2 priorities.

2 Preliminaries

We denote the set $\{1, 2, \ldots, n\}$ by $[n]$ and the set $\{a, a + 1, \ldots, b\}$ by $[a, b]$. By $2^{[n]}$ we mean the set of all subsets of $[n]$ and by $\binom{[n]}{k}$ we mean the set of all k-element subsets of $[n]$. Notation $X \triangle Y$ is used for the symmetric difference of two sets X, Y.

2.1 Separation automata

Let Σ be a finite alphabet. For $w \in \Sigma^* \cup \Sigma^{\mathbb{N}}$ by $|w|$ we denote the length of w. We assume that subscripts enumerating letters of w start with 1, i.e., we write $w = w_1 w_2 w_3 \ldots$

A deterministic finite automaton $\mathcal{A}$ over Σ is specified by a finite set Q of its states, an indicated initial state $q_{start} \in Q$ and a transition function $\delta_{\mathcal{A}} \colon Q \times \Sigma \to Q$. As usual, we extend $\delta_{\mathcal{A}}$ to be a function of the form $\delta \colon Q \times \Sigma^* \to Q$ by setting $\delta_{\mathcal{A}}(q, w_1 \ldots w_p)$ to be a state reached by the automaton from $q \in Q$ after reading $w_1 \ldots w_p \in \Sigma^*$.

For $A, B \subseteq \Sigma^{\mathbb{N}}$, $A \cap B = \varnothing$, we say that a deterministic finite automaton $\mathcal{A}$ *separates* A from B if there exists a state $q_{accept} \in Q$ such that for all $w = w_1 w_2 \ldots \in \Sigma^{\mathbb{N}}$ the following holds:

- if $w \in A$, then there exists $i_0 \in \mathbb{N}$ such that $\delta_{\mathcal{A}}(q_{start}, w_1 \ldots w_i) = q_{accept}$ for all $i \geqslant i_0$;
- if $w \in B$, then for all $i \in \mathbb{N}$ it holds that $\delta_{\mathcal{A}}(q_{start}, w_1 \ldots w_i) \neq q_{accept}$.

We say that an automaton separates A from B in time t if instead of the first condition the stronger one holds: if $w \in A$, then $\delta(q_0, w_1 \ldots w_i) = q_{accept}$ for all $i \geqslant t$.

A *game graph* with n nodes and d priorities is a pair $G = \langle E, \pi \rangle$, where

- E is a subset of $\{1, \ldots, n\}^2$ satisfying the following condition: for all $u \in \{1, \ldots, n\}$ there is $v \in \{1, \ldots, n\}$ such that $(u, v) \in E$;
- π is a function of the form $\pi \colon E \to \{1, \ldots, d\}$.

I.e., we consider G as a directed graph in which nodes are elements of $\{1, \ldots, n\}$ and edges are elements of E. Moreover, edge e has a label $\pi(e) \in \{1, \ldots, d\}$ on it. Edge labels are called priorities. A game graph should satisfy the following requirement: for each node, there exists at least one out-going edge. We stress that we allow loops but do not allow parallel edges[2].

A game graph $G = \langle E, \pi \rangle$ is called even (odd), if the maximum of π on every cycle of G is even (odd). More formally, G is called even (odd) if for all $k \geqslant 1$ and $v_1, \ldots, v_k \in \{1, \ldots, n\}$ satisfying:

$$(v_1, v_2), (v_2, v_3), \ldots, (v_{k-1}, v_k), (v_k, v_1) \in E,$$

it holds that:

$$\max\{\pi((v_1, v_2)), \pi((v_2, v_3)), \ldots, \pi((v_{k-1}, v_k)), \pi((v_k, v_1))\} \text{ is even (odd).}$$

Now let us define two sets (languages) consisting of infinite words over an alphabet $\Sigma = \{1, \ldots, n\} \times \{1, \ldots, d\}$. These two languages will be called $\mathrm{EvenCycles}_{n,d}$ and $\mathrm{OddCycles}_{n,d}$. Namely, an infinite sequence

[2] Our main lower bound holds for graphs without loops as well and the proof is easily adaptable. To simplify an exposition, we present a weaker result.

$(v_1, l_1)(v_2, l_2)(v_3, l_3)\ldots \in (\{1, \ldots, n\} \times \{1, \ldots, d\})^{\mathbb{N}}$ belongs to $\mathrm{EvenCycles}_{n,d}$ if there exists an even game graph $G = \langle E, \pi\rangle$ with at most n nodes and d priorities such that for all $i \geqslant 1$ it holds that $(v_i, v_{i+1}) \in E$ and $\pi((v_i, v_{i+1})) = l_i$. I.e., we put $(v_1, l_1)(v_2, l_2)(v_3, l_3)\ldots$ into $\mathrm{EvenCycles}_{n,d}$ if and only if this sequence can be realized as an infinite path in some even game graph with at n nodes and d priorities.

If instead of G being even we require that G is odd, we obtain a definition of $\mathrm{OddCycles}_{n,d}$.

2.2 Communication complexity

For our main lower bound we use non-deterministic communication complexity in the *number-in-hand* model, but let us start with the deterministic case. In the number-in-hand model there are k parties and their goal is to compute some (fixed in advance, possibly partial) function $f\colon \mathcal{X}_1 \times \ldots \times \mathcal{X}_k \to \{0, 1\}$, where sets $\mathcal{X}_1, \ldots, \mathcal{X}_k$ are finite. The i^{th} party receives an element X_i of $\mathcal{X}_i$ on input. Parties have a shared blackboard on which they can write binary messages. Blackboard is seen by all parties. A deterministic protocol specifies at each moment of time:

- whose turn is to write on a blackboard (depending on what is already written there);
- a message of the corresponding party (which depends not only on what is written on a blackboard but also on the player's input).

In the end of the communication, parties output a single bit which is assumed to be the value of f on $(X_1, \ldots, X_k)$. This bit is a function of the history of communication, i.e. it can be computed by an external observer who can see only blackboard but does not see inputs of players. The communication complexity of a deterministic protocol π (denoted below by $CC(\pi)$) is the maximal possible (over all inputs) number of bits written on a blackboard in π.

Now let us switch to non-deterministic protocols. The most convenient definition for us is the following one. A non-deterministic protocol is a set $\mathcal{P}$ of deterministic protocols. A run of a non-deterministic protocol has two phases. At first parties guess $\pi \in \mathcal{P}$. The guess is public so all the parties have the same π. Then the parties run π on $(X_1, \ldots, X_k)$. By a communication complexity of $\mathcal{P}$ we mean the following expression:

$$CC(\mathcal{P}) = \lceil \log_2 (|\mathcal{P}|\rceil) + \max_{\pi \in \mathcal{P}} CC(\pi).$$

In particular, besides communication in π the number of bits needed to specify π also counts. For brevity, we use a term "c-bit protocol" for a protocol with communication complexity at most c.

We say that $\mathcal{P}$ computes f if for all $(X_1, \ldots, X_k) \in \mathcal{X}_1 \times \ldots \times \mathcal{X}_k$ it holds that:

- if $f(X_1, \ldots, X_k) = 1$, then there is $\pi \in \mathcal{P}$ such that π outputs 1 on $(X_1, \ldots, X_k)$;
- if $f(X_1, \ldots, X_k) = 0$, then for all $\pi \in \mathcal{P}$ it holds that π outputs 0 on $(X_1, \ldots, X_k)$.

Finally, by the non-deterministic communication complexity of f we mean the minimal $c \in \mathbb{N}$ such that there exists a c-bit non-deterministic communication protocol, computing f.

More formal introduction to the number-in-hand model can be found, for instance, in [13, Chapter 5]. For our lower bound we use only very basic technique of *monochromatic boxes*. This technique is a generalization of a standard two-party monochromatic rectangle technique. A box is a set of the form $\mathcal{F}_1 \times \ldots \times \mathcal{F}_k$ for some $\mathcal{F}_1 \subseteq \mathcal{X}_1, \ldots, \mathcal{F}_k \subseteq \mathcal{X}_k$. We exploit the following feature of protocols: a c-bit non-deterministic protocol computing f induces a *cover* of $\{(X_1, \ldots, X_k) \in \mathcal{X}_1 \times \ldots \times \mathcal{X}_k : f(X_1, \ldots, X_k) = 1\}$ by at most 2^c boxes such that each box in the cover does not contain a tuple on which f is defined and takes value 0.

2.3 Kolmogorov complexity

Consider any two binary strings x and y. Informally speaking, the conditional Kolmogorov complexity of x given y is the minimal length of a program producing x from y (length is measured in bits). To define it formally, consider any partial computable function $D : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^*$. Let $C_D(x|y)$ denote

$$\min\{|p| : p \in \{0,1\}^* \text{ and } D(p, y) = x\}.$$

(here, as above, $|p|$ stands for the length of p). So $C_D(x|y)$ can be viewed as a compressed size of x given y with respect to "decompressor" D. Kolmogorov – Solomonoff theorem states that there exists an "optimal" decompressor; more precisely, there is a partial computable function $D_0 : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^*$ such that for any partial computable function $D : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^*$ there exists $A > 0$ such that for all $x, y \in \{0,1\}^*$ we have $C_{D_0}(x|y) \leqslant C_D(x|y) + A$. We fix any such D_0 and let $C(x|y) = C_{D_0}(x|y)$ be Kolmogorov complexity of x given y. We also define the unconditional Kolmogorov complexity of x as Kolmogorov complexity of x given the empty word.

Let us list some standard properties of Kolmogorov complexity which will be used in this paper. Proofs of them can be found, for instance, in [23].

Proposition 3. *For any $z \in \{0,1\}^*$ the number of $x \in \{0,1\}^*$ satisfying $C(x|z) \leqslant a$ is less than 2^{a+1}.*

Proposition 4. *For any computable function $f(\cdot,\cdot)$ and for all $x, y \in \{0,1\}^*$ the following holds:*

$$C(f(x,y)|y) \leqslant C(x|y) + O(1)$$

(constant hidden in $O(\cdot)$ depends only on f but not on x and y).

Proposition 5. *For all $m \in \mathbb{N}$ and for all $x_1, \ldots, x_m, y \in \{0,1\}^*$ the following holds:*

$$C(x_1, x_2, \ldots, x_m | y) \leqslant O(1) + \sum_{i=1}^{m} (2C(x_i | x_1, \ldots, x_{i-1}, y) + 2)$$

(constant hidden in $O(\cdot)$ is absolute)[3].

Kolmogorov Complexity can be defined not only for binary strings but for other "finite objects", like tuples of strings, finite sets, graphs etc. To do so we have to fix some encoding of these objects by binary strings. Different encodings lead to the same complexity up to $O(1)$ additive term.

3 Proof of Theorem 2

For our lower bound we define the following communication problem which is a variation of Disjointness problem. Fix $n, k \in \mathbb{N}$ and $\gamma > 0$. There are k parties. The i^{th} party receives a set $X_i \subseteq \{1, 2, \ldots, n\}$ of size $\lfloor n/k \rfloor$. It is promised that either $X_1, X_2, \ldots, X_k$ are disjoint or $\forall i, i' \in \{1, \ldots, k\}$ $|X_i \triangle X_{i'}| \leqslant \gamma \cdot \lfloor n/k \rfloor$. The goal of parties is to output 1 in the first case and 0 otherwise. We denote this problem by $\mathrm{DISJ}'_{k,\gamma}(n)$.

We show the following lower bound on $\mathrm{DISJ}'_{k,\gamma}(n)$:

Theorem 6. *For all large enough n and for all $k \in \{2, \ldots, n-1\}$ and $\gamma \in (0,1)$ satisfying $\frac{k}{\gamma} \leqslant \frac{\sqrt{n}}{100}$ the non-deterministic communication complexity of $\mathrm{DISJ}'_{k,\gamma}(n)$ is at least $\frac{\gamma^2 n}{10^4 \cdot k} - 2\log_2(n)$.*

A similar problem (without restrictions on sizes of input sets) in the two-party setting was considered in [11]. We postpone proof of Theorem 6 to Section 5.

To show Theorem 6 we prove the following result which is interesting on its own:

Theorem 7. *For all $n, a, t \in \mathbb{N}$ satisfying $t < a < n$ the following holds. If $\mathcal{F} \subseteq \binom{[n]}{a}$ and $\mathcal{G} \subseteq \binom{[n]}{a}$ are such that $|F \cap G| \leqslant t$ for all $F \in \mathcal{F}$ and $G \in \mathcal{G}$, then*

$$|\mathcal{F}| \cdot |\mathcal{G}| \leqslant 32a(n-a) \cdot e^{-(a-t-1)^2/(20a)} \binom{n}{a}^2.$$

We postpone proof of Theorem 7 to Section 4. For a special case when $t = \Omega(n)$ and $a - t = \Omega(n)$ this bound can be found in a classical work of Frankl and Rödl [10]. Moreover, their result only requires that $|F \cap G| \neq t + 1$ for all $F \in \mathcal{F}, G \in \mathcal{G}$. However, the paper [10] does not contain a complete proof of this bound and it is unclear how to restore details omitted. Also, it is quite

[3] There is a more tight relation between the left and the right hand side known as "chain rule". However, Proposition 5 is enough for our purposes.

hard to turn a proof of Frankl and Rödl into an explicit bound for sublinear m and t.

To simplify the analysis below we need the following lower bound on separating $\mathrm{EvenCycles}_{n,2}$ from $\mathrm{OddCycles}_{n,2}$ without any time restrictions.

Lemma 8. *Any deterministic finite automaton separating* $\mathrm{EvenCycles}_{n,2}$ *from* $\mathrm{OddCycles}_{n,2}$ *has at least* $n+1$ *states.*

Proof. Assume that a deterministic finite automaton $\mathcal{A}$ separates $\mathrm{EvenCycles}_{n,2}$ from $\mathrm{OddCycles}_{n,2}$. For $i = 0, 1, \ldots, n-1$ define

$$q_i = \delta_{\mathcal{A}}(q_{start}, (1,2)(2,2)\ldots(i,2)),$$

where q_{start} is the initial state of $\mathcal{A}$. Note that $(1,2)(2,2)\ldots(n-1,2)$ is a prefix of a word from $\mathrm{OddCycles}_{n,2}$. Indeed, consider a graph which for $i \in [n-1]$ has an edge from i to $i+1$ labeled by 2 and also has a loop labeled by 1 at node n. This means that $q_0 \neq q_{accept}, q_1 \neq q_{accept}, \ldots, q_{n-1} \neq q_{accept}$. Now assume that $\mathcal{A}$ has at most n states. Then there are $i, j \in \{0, 1, \ldots, n-1\}$, $i < j$, such that $q_i = q_j$. Consider a graph G with n nodes which has all possible directed edges (including loops) and all of them are labeled by 2. Obviously, G is an even game graph. Let $C_{i,j}$ be a cycle of G obtained by going from $i+1$ to j and then back to $i+1$ (in particular if $j = i+1$, then $C_{i,j}$ is a loop at j). Consider an infinite path in G which goes from 1 to i and then stays on $C_{i,j}$ forever. By definition $\mathcal{A}$ should reach q_{accept} on this path at some point. On the other hand it is easy to see that the set of states $\mathcal{A}$ reaches on this path is $\{q_0, q_1, \ldots, q_i, \ldots, q_{j-1}\}$. □

In this section we actually prove a more specified version of Theorem 2.

Theorem 9. *For all large enough n the following holds. If $8n \leqslant t \leqslant \frac{n^{5/4}}{10^3}$, then any deterministic finite automaton separating* $\mathrm{EvenCycles}_{n,2}$ *from* $\mathrm{OddCycles}_{n,2}$ *in time t has more than*

$$2^{\frac{n^5}{(10^3 \cdot t)^4}}$$

states.

Theorem 2, however, has no restrictions on t, unlike Theorem 9, so let us explain how Theorem 9 implies Theorem 2. For

$$t > \frac{n^{5/4}}{10^3}$$

the lower bound of Theorem 2 is just constant, and we even have a better bound by Lemma 8. Theorem 2 for $n \leqslant t < 8n$ follows from Theorem 9 for $t = 8n$ (with some constant loss in the exponent). Finally, we observe that for $t < n$ there is no deterministic finite automaton separating $\mathrm{EvenCycles}_{n,2}$ from $\mathrm{OddCycles}_{n,2}$ in time t at all. Indeed, a word $(1,1)(2,1)\ldots(n-1,1)$ is a prefix of a word $\mathrm{EvenCycles}_{n,2}$ and also a prefix of a word from $\mathrm{OddCycles}_{n,2}$.

So now we proceed to a proof of Theorem 9. Let us start with the following observation. There exists an algorithm, which decides, given $n, t \in \mathbb{N}$ and a deterministic finite automaton $\mathcal{A}$, whether $\mathcal{A}$ separates $\mathrm{EvenCycles}_{n,2}$ from $\mathrm{OddCycles}_{n,2}$ in time t. First we go through all odd game graphs with at most n nodes and 2 priorities. For each such graph we check whether there is a path in it which leads $\mathcal{A}$ to q_{accept}. Obviously this reduces to a computable problem of determining whether there is a path in a finite directed graph from one node to another. Then we go through all even game graphs with at most n nodes and 2 priorities. For each such graph we check whether it is possible to reach any state other than q_{accept} after making t steps. This reduces to a computable problem of checking whether there is a path of length *at least* t from one node of a finite directed graph to another.

Consider an algorithm $\mathbf{ALG_1}$, which, given n and t, tries to find by a brute-force search a deterministic finite automaton with at most

$$Q = 2^{\left\lceil \frac{n^5}{(10^3 \cdot t)^4} \right\rceil} \tag{1}$$

states separating $\mathrm{EvenCycles}_{n,2}$ from $\mathrm{OddCycles}_{n,2}$ in time t. Once such automaton is found, $\mathbf{ALG_1}$ outputs it and halts. If there is no such automaton at all, $\mathbf{ALG_1}$ halts with no output.

Assume for contradiction that Theorem 9 is false for some n and $8n \leqslant t \leqslant \frac{n^{5/4}}{10^3}$. Let

$$\mathcal{A} = \mathbf{ALG_1}(n, t). \tag{2}$$

Note that $\mathcal{A}$ separates $\mathrm{EvenCycles}_{n,2}$ from $\mathrm{OddCycles}_{n,2}$ and has at most Q states. Hence by Lemma 8 we get

$$Q \geqslant n + 1 \tag{3}$$

(we only need that Q is super-constant).

Set

$$\begin{aligned} n' &= \lceil n/2 \rceil, \qquad k = 20 \cdot \left\lfloor \frac{t}{n} \right\rfloor, \\ \gamma &= \frac{1}{k}, \qquad a = \lfloor n'/k \rfloor. \end{aligned} \tag{4}$$

From the hypotheses of Theorem 9 it is easy to derive the following bound:

$$k = O(n^{1/4}). \tag{5}$$

We will use the following notation:

$$\begin{aligned} \mathcal{D} &= \left\{ (X_1, \ldots, X_k) \in \binom{[n']}{a}^k : X_1, X_2, \ldots, X_k \text{ are disjoint} \right\}, \\ \mathcal{I} &= \left\{ (Y_1, \ldots, Y_k) \in \binom{[n']}{a}^k : \forall i, i' \in \{1, \ldots, k\} \ |Y_i \triangle Y_{i'}| \leqslant \gamma a \right\}. \end{aligned} \tag{6}$$

Note that $\mathrm{DISJ}'_{k,\gamma}(n')$ is a problem to output 1 on $\mathcal{D}$ and output 0 on $\mathcal{I}$.

Below we construct a non-deterministic communication protocol $\mathcal{P}$, involving $\mathcal{A}$. Protocol $\mathcal{P}$ appears to be too short to solve $\mathrm{DISJ}'_{k,\gamma}(n')$. So there is an input on which $\mathcal{P}$ works incorrectly. Using this input, we construct two special words on which $\mathcal{A}$ comes into the same state. One word is a prefix of a sequence from $\mathrm{EvenCycles}_{n,2}$. Moreover, its length is at least t. The other word is a prefix of a sequence from $\mathrm{OddCycles}_{n,2}$. This gives us a contradiction with the definition of separation in time t.

In the description of $\mathcal{P}$ we use the following notation. For $X \subseteq [n]$ denote:

$$(X, 1) = (x_1, 1)(x_2, 1) \dots (x_m, 1) \in ([n] \times \{1\})^*,$$

where $x_1, x_2, \dots, x_m \in [n]$ are such that $x_1 < x_2 < \dots < x_m$ and $X = \{x_1, x_2, \dots, x_m\}$.

Description of the protocol $\mathcal{P}$. In this protocol there are k parties and the ith party receives a set $X_i \in \binom{[n']}{a}$. At the beginning parties non-deterministically guess a state q_0 of $\mathcal{A}$ and a set $U \subseteq [n']$ satisfying:

$$|U| \leqslant \frac{2n'}{5}, \qquad C(U|n, t) \leqslant k \log_2(Q).$$

Then parties communicate in k stages. Stages are numbered from 1 to k. At the ith stage the ith party writes $\log_2(Q)$ bits specifying a state of $\mathcal{A}$ on the blackboard. Namely,

at the 1st stage the 1st party writes $q_1 = \delta_{\mathcal{A}}(q_0, (X_1 \setminus U, 1))$;

at the 2nd stage the 2nd party writes $q_2 = \delta_{\mathcal{A}}(q_1, (X_2 \setminus U, 1))$;

$$\vdots$$

at the kth stage the kth party writes $q_k = \delta_{\mathcal{A}}(q_{k-1}, (X_k \setminus U, 1))$.

Observe that

$$q_k = \delta_A(q_0, (X_1 \setminus U, 1)(X_2 \setminus U, 1) \dots (X_k \setminus U, 1)).$$

After performing these k stages parties finish communication. It remains to explain how the output of the protocol $\mathcal{P}$ is computed. Parties output 1 if and only if there is no $(Y_1, \dots, Y_k) \in \mathcal{I}$ such that

$$q_k = \delta_A(q_0, (Y_1 \setminus U, 1)(Y_2 \setminus U, 1) \dots (Y_k \setminus U, 1)).$$

In other words, parties output 1 if and only if there is no input from $\mathcal{I}$ on which $\mathcal{P}$ produces the same q_k for a guess (q_0, U). **Description of the protocol is finished.**

Lemma 10. *$CC(\mathcal{P})$ is smaller than the non-deterministic communication complexity of* $\mathrm{DISJ}'_{k,\gamma}(n')$.

Proof. Let us compute the communication complexity of $\mathcal{P}$. By Proposition 3 in $\mathcal{P}$ there are at most $Q \cdot 2^{k\log_2(Q)+1}$ possible non-deterministic guesses. After making a guess parties communicate exactly $k\log_2(Q)$ bits. Therefore:

$$CC(\mathcal{P}) \leqslant \log_2(Q) + k\log_2(Q) + 1 + k\log_2(Q) = (2k+1)\log_2(Q) + 1.$$

The last expression is at most $3k\log_2(Q)$. Indeed, $k = 20\lfloor t/n \rfloor$ by (4) and $t \geqslant 8n$ by requirements of Theorem 9. Hence $k \geqslant 160$ and $\frac{5k}{2} \geqslant 2k+1$. Note also that $\log_2(Q)$ is super-constant by (3). Thus

$$\begin{aligned}(2k+1)\log_2(Q) + 1 &\leqslant \frac{5k}{2} \cdot \log_2(Q) + 1 \\ &\leqslant 3k\log_2(Q).\end{aligned}$$

In this way we conclude

$$CC(\mathcal{P}) \leqslant 3k\log_2(Q). \tag{7}$$

Let us verify that $\frac{k}{\gamma} \leqslant \frac{\sqrt{n'}}{100}$. Indeed, again by (4) and by the hypotheses of Theorem 9 we have:

$$\frac{k}{\gamma} = k^2 \leqslant \frac{400 \cdot t^2}{n^2} \leqslant \frac{400 \cdot \frac{n^{5/2}}{10^6}}{n^2} = \frac{400 \cdot \sqrt{2}}{10^6} \cdot \sqrt{\frac{n}{2}} \leqslant \frac{400 \cdot \sqrt{2}}{10^6} \cdot \sqrt{n'} < \frac{\sqrt{n'}}{100}.$$

Hence by Theorem 6 the non-deterministic communication complexity of $\mathrm{DISJ}'_{k,\gamma}(n')$ is at least

$$\frac{\gamma^2 n'}{10^4 \cdot k} - 2\log_2(n') \geqslant \frac{\gamma^2 n}{2 \cdot 10^4 \cdot k} - 2\log_2(n) \geqslant \frac{\gamma^2 n}{3 \cdot 10^4 \cdot k}.$$

Here in the first inequality we use the definition of n' (see (4)). The second inequality is true because $\gamma^2 n/k = n/k^3 = \Omega(n^{1/4})$ by (5).

Thus by (7) it remains to show that:

$$\log_2(Q) < \frac{\gamma^2 n}{9 \cdot 10^4 \cdot k^2} = \frac{n}{9 \cdot 10^4 \cdot k^4}.$$

The right hand side by definition of k (see (4)) is at least

$$\frac{n}{9 \cdot 10^4 \cdot \left(\frac{20t}{n}\right)^4} \geqslant \frac{n^5}{10^{11} \cdot t^4}.$$

In turn the left hand side by definition of Q (see (1)) is at most

$$\log_2(Q) = \left\lceil \frac{n^5}{(10^3 \cdot t)^4} \right\rceil < \frac{n^5}{10^{12} \cdot t^4} + 1 \leqslant 2 \cdot \frac{n^5}{10^{12} \cdot t^4} < \frac{n^5}{10^{11} \cdot t^4},$$

where the second inequality is because $t^4 \leqslant \frac{n^5}{10^{12}}$ due to requirements of Theorem 9. □

Lemma 10 means that $\mathcal{P}$ does not compute $\mathrm{DISJ}'_{k,\gamma}(n')$. In particular this gives the following corollary, which allows us to forget about $\mathcal{P}$ for the rest of the proof.

Corollary 11. *There exists $\overline{X} = (X_1, \dots, X_k) \in \mathcal{D}$ such that for every state q_0 of $\mathcal{A}$ and for every $U \subseteq [n']$ satisfying:*

$$|U| \leqslant 2n'/5, \qquad C(U|n,t) \leqslant k \log_2(Q),$$

there exists $(Y_1, \dots, Y_k) \in \mathcal{I}$ such that:

$$\delta_A(q_0, (X_1 \setminus U, 1) \dots (X_k \setminus U, 1)) = \delta_{\mathcal{A}}(q_0, (Y_1 \setminus U, 1) \dots (Y_k \setminus U, k)).$$

Proof. Obviously $\mathcal{P}$ on any input from $\mathcal{I}$ for any possible guess outputs 0. Since $\mathcal{P}$ does not compute $\mathrm{DISJ}'_{k,\gamma}(n')$, this means that there is $\overline{X} = (X_1, \dots, X_k) \in \mathcal{D}$ such that $\mathcal{P}$ on input $\overline{X}$ for any guess outputs 0. It is easy to see that this is is equivalent to the statement of Corollary 11. □

From now on $\overline{X} = (X_1, X_2, \dots, X_k)$ stands for a tuple satisfying Corollary 11.

Let us introduce some notation needed to formulate the next lemma. Namely, let $<_{\overline{X}}$ be the linear order on $X_1 \cup X_2 \cup \dots \cup X_k$ drawn on Figure 1.

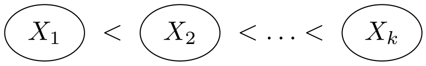

Figure 1: $<_{\overline{X}}$ order

Formally, we say that $p <_{\overline{X}} q$ if and only if at least one of the following two conditions holds:

- $p \in X_i, q \in X_{i'}$ for some $i, i' \in [k], i < i'$;
- $p < q$ and $p, q \in X_i$ for some $i \in [k]$.

Next, let us say that a word $(v_1, l_1) \dots (v_m, l_m) \in ([n] \times \{1,2\})^*$ is $\overline{X}$-increasing if $v_1, v_2, \dots, v_m \in X_1 \cup X_2 \cup \dots \cup X_k$ and $v_1 <_{\overline{X}} v_2 <_{\overline{X}} \dots <_{\overline{X}} v_m$.

For a word $w = (v_1, l_1) \dots (v_m, l_m) \in ([n] \times \{1,2\})^*$ denote $v(w) = \{v_1, \dots, v_m\}$.

Finally, for $r \in \mathbb{N}$ let $\#_r$ denote a pair $(n' + r, 2)$. We will use symbols $\#_r$ only for $r \leqslant k/5$. Recall that $k = O(n^{1/4})$ (see (5)), which means that for any $r \leqslant k/5$ it holds that $\#_r \in [n] \times \{1,2\}$, i.e., $\#_r$ belongs to the input alphabet of $\mathcal{A}$.

Lemma 12. *For every $r = 1, \dots, k/5$ there are words $f^1, \dots, f^r, g^1, \dots, g^r \in ([n'] \times \{1\})^*$ satisfying the following conditions:*

$$v(f^1), \dots, v(f^r) \text{ are disjoint and } |v(f^1)| \leqslant 2n'/k, \dots, |v(f^r)| \leqslant 2n'/k, \tag{8}$$

$$C(f^j | f^1, \dots, f^{j-1}, n, t) \leqslant 2\log_2(Q) \text{ for } j = 1, \dots, r, \tag{9}$$

$$|v(g^1)| = |g^1|, \dots, |v(g^r)| = |g^r| \text{ and } |g^1| \geqslant 4n'/7, \dots, |g^r| \geqslant 4n'/7, \tag{10}$$

$$g^1, \dots, g^r \text{ are } \overline{X}\text{-increasing}, \tag{11}$$

$$\delta_{\mathcal{A}}(q_{start}, f^1 \#_1 f^2 \#_2 \dots f^r \#_r) = \delta_{\mathcal{A}}(q_{start}, g^1 \#_1 g^2 \#_2 \dots g^r \#_r). \tag{12}$$

Here q_{start} is the initial state of $\mathcal{A}$.

Proof. Let $\mathbf{ALG_2}$ be the following algorithm. An input to $\mathbf{ALG_2}$ consists of two parts:

- $n, t \in \mathbb{N}$ and tuple $\alpha = (f^1, \dots, f^j)$, where $f^1, \dots, f^j \in ([n'] \times \{1\})^*$ and $j \geqslant 0$ (when $j = 0$ we assume that α is empty);
- a binary word $q \in \{0, 1\}^{\log_2(Q)}$;

here n' is defined as in (4) and Q is defined as in (1).

At the beginning, $\mathbf{ALG_2}$ applies $\mathbf{ALG_1}$ to n and t to find a deterministic finite automaton $\mathcal{A}$ with at most Q states separating $\text{EvenCycle}_{n,2}$ from $\text{OddCycle}_{n,2}$ in time t. If there is no such $\mathcal{A}$, an algorithm $\mathbf{ALG_2}$ halts and outputs "not found". Otherwise $\mathbf{ALG_2}$ interprets q as a state of $\mathcal{A}$. The algorithm $\mathbf{ALG_2}$ sets

$$U = v(f^1) \cup v(f^2) \cup \dots \cup v(f^j),$$

$$q_0 = \delta_{\mathcal{A}}(q_{start}, f^1 \#_1 f^2 \#_2 \dots f^j \#_j).$$

where q_{start} is the initial state of $\mathcal{A}$. Then $\mathbf{ALG}_2$ tries to find $(Y_1, \dots, Y_k) \in \mathcal{I}$ (where $\mathcal{I}$ is defined as in (6)) such that

$$\delta(q_0, (Y_1 \setminus U, 1) \dots (Y_k \setminus U, 1)) = q.$$

Once such $(Y_1, \dots, Y_k)$ is found, an algorithm $\mathbf{ALG_2}$ outputs a word $f = (Y_1 \setminus U, 1) \dots (Y_k \setminus U, 1)$. If there is no such $(Y_1, \dots, Y_k)$ at all, $\mathbf{ALG_2}$ halts and outputs "not found". **Description of an algorithm $\mathbf{ALG_2}$ is finished**.

We will use $\mathbf{ALG_2}$ to show lemma by induction on r. Induction base and induction step will be proved by the same argument. Namely, assume that $f^1, \dots, f^{r-1}, g^1, \dots, g^{r-1}$ satisfying (8–12) are already constructed for some $r \leqslant k/5$ (case $r = 1$ corresponds to the induction base). Define

$$U = v(f^1) \cup v(f^2) \cup \dots v(f^{r-1}),$$

$$q_0 = \delta_{\mathcal{A}}(q_{start}, f^1 \#_1 f^2 \#_2 \dots f^{r-1} \#_{r-1})$$

(for $r = 1$ we have $U = \varnothing$ and $q_0 = q_{start}$). Note that by (12) we also have

$$q_0 = \delta_{\mathcal{A}}(q_{start}, g^1 \#_1 g^2 \#_2 \dots g^{r-1} \#_{r-1}).$$

It is enough to construct $f^r, g^r \in ([n'] \times \{1\})^*$ satisfying:

$$v(f^r) \cap U = \varnothing \text{ and } |v(f^r)| \leqslant 2n'/k, \tag{13}$$
$$C(f^r|f^1, \ldots, f^{r-1}, n, t) \leqslant 2\log_2(Q), \tag{14}$$
$$|v(g^r)| = |g^r| \text{ and } |g^r| \geqslant 4n'/7, \tag{15}$$
$$g^r \text{ is } \overline{X}\text{-increasing}, \tag{16}$$
$$\delta_{\mathcal{A}}(q_0, f^r) = \delta_{\mathcal{A}}(q_0, g^r). \tag{17}$$

We define g^r as follows:

$$g^r = (X_1 \setminus U, 1)(X_2 \setminus U, 1) \ldots (X_k \setminus U, 1).$$

At first, we derive (15) and (16). The latter is clear from construction. As for (15), the fact that $|v(g^r)| = |g^r|$ follows from the fact that $X_1, X_2, \ldots, X_k$ are disjoint. Moreover, the following holds:

$$|g^r| = |(X_1 \cup X_2 \cup \ldots \cup X_k) \setminus U| \geqslant k \cdot \lfloor n'/k \rfloor - |U|.$$

By (8) and by definition of U its size is at most $(k/5) \cdot (2n'/k) = 2n'/5$. As $k = O(n^{1/4})$ by (5), we obtain $|g^r| \geqslant 4n'/7$.

It remains to derive (13), (14) and (17) (these conditions involve f^r). Set $q = \delta_{\mathcal{A}}(q_0, g^r) = \delta_{\mathcal{A}}(q_0, (X_1 \setminus U, 1)(X_2 \setminus U, 1) \ldots (X_k \setminus U, 1))$ and define

$$f^r = \mathbf{ALG_2}((n, t, (f^1, f^2, \ldots, f^{r-1})), q).$$

Here, however, we first should show that the output of $\mathbf{ALG_2}$ on such input is different from "not found". I.e., we should show that there exists $(Y_1, \ldots, Y_k) \in \mathcal{I}$ satisfying

$$\delta_{\mathcal{A}}(q_0, (Y_1 \setminus U, 1)(Y_2 \setminus U, 1) \ldots (Y_k \setminus U, 1)) = q.$$

Since $(X_1, X_2, \ldots, X_k)$ is a tuple from Corollary 11, to do so it is enough to demonstrate that $|U| \leqslant 2n'/5$ and $C(U|n, t) \leqslant k\log_2(Q)$. The first inequality is already shown above. To show the second one, recall that $U = v(f^1) \cup v(f^2) \cup \ldots \cup v(f^{r-1})$, i.e., U is an image of $(f^1, \ldots, f^{r-1})$ under some computable mapping. By Proposition 4 this means that

$$C(U|n, t) \leqslant C(f^1, f^2, \ldots, f^{r-1}|n, t) + O(1).$$

In turn, the right hand sized of the last inequality by Proposition 5 can be upper bounded by

$$O(1) + \sum_{j=1}^{r-1} (2C(f^j|f^1, \ldots, f^{j-1}, n, t) + 2).$$

The last sum by (9) is at most $(k/5) \cdot (4\log_2(Q) + 2) + O(1) \leqslant k\log_2(Q)$. The last inequality is true because $k \geqslant 160$ (see proof of Lemma 10) and because Q is super-constant by (3).

So $f^r = \mathbf{ALG_2}((n, t, (f^1, f^2, \dots, f^{r-1})), q)$ is different from "not found". Hence

$$f^r = (Y_1 \setminus U, 1) \dots (Y_k \setminus U, 1)$$

for some $(Y_1, \dots, Y_k) \in \mathcal{I}$ satisfying

$$q = \delta_{\mathcal{A}}(q_0, (Y_1 \setminus U, 1) \dots (Y_k \setminus U, 1)).$$

This already implies (17) (recall that by definition $q = \delta_{\mathcal{A}}(q_0, g^r)$) and the first part of (13). To show the second part of (13) observe that $v(f^r) \subseteq Y_1 \cup Y_2 \cup \dots \cup Y_k$. Hence

$$\begin{aligned}|v(f^r)| &\leqslant |Y_1| + |Y_2 \setminus Y_1| + \dots + |Y_k \setminus Y_1| \\ &\leqslant \frac{n'}{k} + (k-1)\frac{\gamma n'}{k} \leqslant \frac{2n'}{k}.\end{aligned}$$

Here in the second inequality we use the fact that $(Y_1, Y_2, \dots, Y_k) \in \mathcal{I}$ and in the third inequality we use definition of γ (see (4)).

Finally, to show (14) recall once again that

$$f^r = \mathbf{ALG_2}((n, t, (f^1, f^2, \dots, f^{r-1})), q).$$

Hence by definition of conditional Kolmogorov complexity we have:

$$C(f^r | f^1, \dots f^{r-1}, n, t) \leqslant |q| + O(1) = \log_2(Q) + O(1) \leqslant 2\log_2(Q),$$

where the last inequality is due to (3). □

To finish the proof we take $f^1, \dots, f^{k/5}, g^1, \dots, g^{k/5} \in ([n'] \times \{1\})^*$ satisfying conditions (8–12) of Lemma 12.

Lemma 13. *The length of a word $g^1 \#_1 g^2 \#_2 \dots g^{k/5} \#_{k/5}$ is at least t.*

Proof. Indeed, the length of $g^1 \#_1 g^2 \#_2 \dots g^{k/5} \#_{k/5}$ by (10) is at least $(k/5) \cdot (4n'/7)$. Furthermore, by (4) we have:

$$|g^1 \#_1 g^2 \#_2 \dots g^{k/5} \#_{k/5}| \geqslant (k/5) \cdot (4n'/7) \geqslant 4\left(\frac{t}{n} - 1\right) \cdot \frac{2n}{7}.$$

Recall that $t \geqslant 8n$ by requirements of Theorem 9. Hence $\frac{t}{n} - 1 \geqslant \frac{7t}{8n}$. □

To obtain a contradiction it is enough to show that

$$f^1 \#_1 f^2 \#_2 \dots f^{k/5} \#_{k/5} \text{ is a prefix of a word from OddCycles}_{n,d}, \tag{18}$$

$$g^1 \#_1 g^2 \#_2 \dots g^{k/5} \#_{k/5} \text{ is a prefix of a word from EvenCycles}_{n,d}. \tag{19}$$

Indeed Lemma 13 together with (19) gives that

$$\delta_{\mathcal{A}}(q_{start}, g^1 \#_1 g^2 \#_2 \dots g^{k/5} \#_{k/5}) = q_{accept}.$$

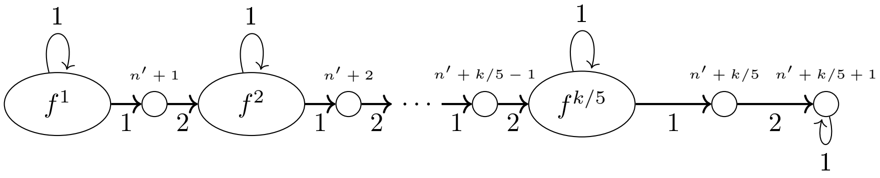

Figure 2: a graph for $f^1\#_1 f^2\#_2 \dots f^{k/5}\#_{k/5}$.

By (12) this also means that

$$\delta_{\mathcal{A}}(q_{start}, f^1\#_1 f^2\#_2 \dots f^{k/5}\#_{k/5}) = q_{accept},$$

but this contradicts (18).

Let us at first show (18). Consider the following graph G_{odd} (see Figure 2).

Nodes of this G_{odd} are elements of

$$v(f^1) \cup v(f^2) \cup \dots \cup v(f^{k/5}) \cup \{n'+1, \dots, n'+k/5+1\}.$$

Recall that $v(f^1), \dots, v(f^{k/5})$ are subsets of $[n']$. Moreover, by (8) we have that $v(f^1), \dots, v(f^{k/5})$ are disjoint. Now let us specify edges of G_{odd}. First of all, for each $j \in [k/5]$ we draw all possible edges between nodes from $v(f^j)$ (including loops), each edge labeled by 1. Next, for all $j \leqslant k/5$ we draw all edges that start at $v(f^j)$ and end at $n'+j$, each again labeled by 1. Further, for all $j < k/5$ we draw all edges that start at $n'+j$ and end at $v(f^{j+1})$, now each edges labeled by 2. Finally, we draw an edge from $n'+k/5$ to $n'+k/5+1$ labeled by 2 and a loop at $n'+k/5+1$ labeled by 1 (we add the last loop to ensure that each node of G_{odd} has at least one out-going edge).

Since $v(f^1), \dots, v(f^{k/5})$ are disjoint, it is easy to see that G_{odd} is an odd game graph with at most n nodes. Moreover, $f^1\#_1 f^2\#_2 \dots f^{k/5}\#_{k/5}$ encodes a path in G_{odd}. Indeed, we move for some time in $v(f^1)$, then through $n'+1$ we go to $v(f^2)$ and so on. Thus (18) is proved.

Now, to show (19) we define another graph, G_{even} (see Figure 3). Its nodes are elements of

$$X_1 \cup X_2 \cup \dots \cup X_k \cup \{n'+1, \dots, n'+k/5\}.$$

First of all, recall a linear order $<_{\overline{X}}$ on a set $X_1 \cup X_2 \cup \dots \cup X_k$. We use this order to define edges of G_{even}. Namely, for all $u, v \in X_1 \cup X_2 \cup \dots \cup X_k$ satisfying $u <_{\overline{X}} v$ we add an edge labeled by 1 from u to v. Moreover, we draw all edges between $X_1 \cup X_2 \cup \dots \cup X_k$ and $\{n'+1, \dots, n'+k/5\}$ (in both directions). In particular, this ensures that each node of G_{even} has at least one out-going edge. Edges, starting at $X_1 \cup X_2 \cup \dots \cup X_k$, are labeled by 1 while opposite edges are labeled by 2.

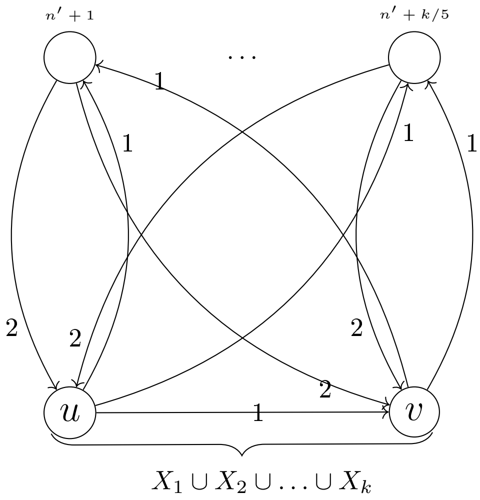

Figure 3: a graph for $g^1\#_1 g^2\#_2 \ldots g^{k/5}\#_{k/5}$.

Note that once we delete all edges labeled by 2 from G_{even}, we obtain an acyclic graph. Hence G_{even} is an even game graph with at most n nodes. On the other hand it is easy to see that $g^1\#_1 g^2\#_2 \ldots g^{k/5}\#_{k/5}$ corresponds to a path of G_{even}. Indeed, by (11) all $g^1, \ldots, g^{k/5}$ are $\overline{X}$-increasing, which means that all edges encoded in $g^1\#_1 g^2\#_2 \ldots g^{k/5}\#_{k/5}$ belong to G_{even}. Thus (19) is proved.

4 Extremal Set Theory Lemma — Proof of Theorem 7

Let us sketch our proof of Theorem 7. First of all, for the sake of brevity we say that two families $\mathcal{F}, \mathcal{G} \subseteq \binom{[n]}{a}$ are *t-far* if $|F \cap G| \le t$ for all $F \in \mathcal{F}, G \in \mathcal{G}$ (so that any member of $\mathcal{F}$ is of Hamming distance at least $2a - 2t$ from any member of $\mathcal{G}$).

Step 1. We use a classical shifting technique of [8] to define so-called *left-compressed* families. We show that it is enough to demonstrate Theorem 7 for the case when $\mathcal{F}$ is left-compressed (Lemma 15).

Step 2. We observe (Proposition 18) that left-compressed families are ideals of a special partial order $\sqsubseteq_a$ (see [1]) on a set $\binom{[n]}{a}$.

Step 3. We give a necessary and sufficient condition for a family $\mathcal{G} \subseteq \binom{[n]}{a}$ to be t-far from an ideal $\mathcal{F}$ of $\sqsubseteq_a$ (Lemma 19).

Step 4. Using this condition we give an upper bound on the probability that $\mathbf{X} \in \mathcal{F}$ and $\mathbf{Y} \in \mathcal{G}$ for two suitably chosen independent random variables

$\mathbf{X}$ and $\mathbf{Y}$ (Lemma 20). From that we easily deduce an upper bound on $|\mathcal{F}| \cdot |\mathcal{G}|$.

4.1 Shifting and compression

For every $i, j \in [n]$ we define so-called *shifting operations* s_{ij} and S_{ij}. Namely, s_{ij} is a unary operation on the set of all subsets of $[n]$. Given $X \subseteq [n]$, the value of $s_{ij}(X)$ is defined as follows:

$$s_{ij}(X) = \begin{cases} (X \setminus \{j\}) \cup \{i\}, & \text{if } j \in X,\ i \notin X, \\ X, & \text{otherwise.} \end{cases}$$

In turn, S_{ij} is a unary operation on the set of all *families* of subsets of $[n]$. Given $\mathcal{X} \subseteq 2^{[n]}$, we define the value of $S_{ij}(\mathcal{X})$ as follows:

$$S_{ij}(\mathcal{X}) = \{s_{ij}(X) : X \in \mathcal{X}, s_{ij}(X) \notin \mathcal{X}\} \cup \{X : X \in \mathcal{X}, s_{ij}(X) \in \mathcal{X}\}.$$

Note that s_{ij} preserves the size of a set, i.e., $|X| = |s_{ij}(X)|$ for all $X \subseteq [n]$. Hence if a family $\mathcal{X}$ consists only of a-element subsets of $[n]$, then the same holds for $S_{ij}(\mathcal{X})$. It is also easy to see that S_{ij} preserves the size of a family, i.e., $|\mathcal{X}| = |S_{ij}(\mathcal{X})|$ for all $\mathcal{X} \subseteq 2^{[n]}$.

Proposition 14 (Lemma 2.1 from [3])**.** *Assume that $1 \le i < j \le n$ and $\mathcal{F}, \mathcal{G} \subseteq \binom{[n]}{m}$ are t-far. Then $S_{ij}(\mathcal{F})$, $S_{ji}(\mathcal{G})$ are also t-far.*

A family $\mathcal{F} \subseteq 2^{[n]}$ is said to be *left-compressed* if $S_{ij}(\mathcal{F}) = \mathcal{F}$ for all $i < j$.

Lemma 15. *If $\mathcal{F}, \mathcal{G} \subseteq \binom{[n]}{m}$ are t-far, then there are $\mathcal{F}', \mathcal{G}' \subseteq \binom{[n]}{a}$ satisfying the following three conditions:*

- *$\mathcal{F}'$ and $\mathcal{G}'$ are t-far;*
- *$|\mathcal{F}'| = |\mathcal{F}|$ and $|\mathcal{G}'| = |\mathcal{G}|$;*
- *$\mathcal{F}'$ is left-compressed.*

It is easy to deduce the last lemma from Proposition 14. Indeed, apply S_{ij} to $\mathcal{F}$ and S_{ji} to $\mathcal{G}$ until $S_{ij}(\mathcal{F}) \neq \mathcal{F}$ for some $i < j$. To show that this can be done only finite number of times observe that

$$\sum_{A \in S_{ij}(\mathcal{F})} \sum_{i \in A} i < \sum_{A \in \mathcal{F}} \sum_{i \in A} i,$$

whenever $S_{ij}(\mathcal{F}) \neq \mathcal{F}$. The proof can also be found in [3] (see the last two paragraphs before Section 3).

4.2 Auxiliary order

For $X \subseteq [n]$ and $1 \leqslant i \leqslant |X|$ define $m(X,i)$ to be the ith smallest element of X. Also define $m(X,0) = 0$.

For any $l \in [n]$ let $\sqsubseteq_l$ be the following partial order on the set $\binom{[n]}{l}$ (see [1]). Namely, for $X, Y \in \binom{[n]}{l}$ we write $X \sqsubseteq_l Y$ if and only if $m(X,i) \leqslant m(Y,i)$ for all $1 \leqslant i \leqslant l$.

Proposition 16. *Let $X = \{x_1, \ldots, x_l\} \in \binom{[n]}{l}$ and $Y \in \binom{[n]}{l}$ be such that $x_i \leq m(Y,i)$ for all $1 \leq i \leq l$. Then $X \sqsubseteq_l Y$.*

Note that x_i in this proposition are not ordered. In other words, a smaller set w.r.t. this order can be produced by decreasing values of some elements of a set.

Proof of Proposition 16. Take any $i \in [l]$. Let j be the largest element of $\{0, 1, \ldots, l\}$ satisfying $m(X,j) \leqslant m(Y,i)$. Note that j is equal to the size of $X \cap [1, m(Y,i)]$. On the other hand we have that $x_1 \leqslant m(Y,1), \ldots, x_i \leqslant m(Y,i)$. Hence $x_1, \ldots, x_i \in X \cap [1, m(Y,i)]$, which means that $j = |X \cap [1, m(Y,i)]| \geqslant i$. Therefore $m(X,i) \leqslant m(X,j) \leqslant m(Y,i)$. □

Proposition 17. *Let $X \in \binom{[n]}{l}$ and $Y = \{y_1, \ldots, y_l\} \in \binom{[n]}{l}$ be such that $m(X,i) \leq y_i$ for all $1 \leq i \leq l$. Then $X \sqsubseteq_l Y$.*

Proof. Apply Proposition 16 to $X' = \{n - y_l + 1, n - y_{l-1} + 1, \ldots, n - y_1 + 1\}$ and $Y' = \{n - j + 1 : j \in X\}$. □

Recall that an *ideal* $\mathcal{A}$ of a partially ordered set $\mathcal{P}$ is a downward-closed subset of $\mathcal{P}$: if $x \leq_P y$ and $y \in \mathcal{A}$, then $x \in \mathcal{A}$.

Proposition 18 (Proposition 3 in [1])**.** *A left-compressed family $\mathcal{F} \subseteq \binom{[n]}{a}$ is an ideal of the order $\sqsubseteq_a$.*

For reader's convenience we also give here a proof sketch of Proposition 18. If $\mathcal{F}$ is not an ideal of $\sqsubseteq_a$, then for some $B \in \mathcal{F}$ there is $A \in \binom{[n]}{a} \setminus \mathcal{F}$ immediately preceding B with respect to $\sqsubseteq_a$. It is not hard to see that A can be obtained from B after decreasing some element of B (say, i) by one. Then $s_{i-1,i}(B) = A$ and hence $\mathcal{F}$ is not left-compressed.

So, it suffice to prove Theorem 7 for a pair $(\mathcal{F}, \mathcal{G})$ in which $\mathcal{F}$ is an ideal of the order $\sqsubseteq_a$.

4.3 Characterizing families which are t-far from ideals

Define the *j-left border* $L_j(X)$ and the *j-right border* $R_j(X)$ of a set $X \subseteq \binom{[n]}{a}$ as

$$L_j(X) = \{m(X,i) : 1 \leq i \leq j\}; \quad R_j(X) = \{m(X,i) : a - j + 1 \leq i \leq a\}.$$

In other words, $L_j(X)$ consists of j smallest elements of X and $R_j(X)$ consists of j largest elements of X.

Lemma 19. *Let $\mathcal{F} \subseteq \binom{[n]}{a}$ be an ideal of $\sqsubseteq_a$. Then for any $\mathcal{G} \subseteq \binom{[n]}{a}$ the following two conditions are equivalent:*

(a) *$\mathcal{F}$ and $\mathcal{G}$ are t-far;*

(b) *$L_{t+1}(G) \not\sqsubseteq_{t+1} R_{t+1}(F)$ for all $F \in \mathcal{F}$ and $G \in \mathcal{G}$.*

Proof. **(b)** $\implies$ **(a)**. Assume for contradiction that $\mathcal{F}$ and $\mathcal{G}$ are *not* t-far. Hence there are $F \in \mathcal{F}$ and $G \in \mathcal{G}$ such that $|F \cap G| \geqslant t+1$. Let X be any $(t+1)$-element subset of $F \cap G$. Then obviously we have that $L_{t+1}(G) \sqsubseteq_{t+1} X \sqsubseteq_{t+1} R_{t+1}(F)$, contradiction.

(a) $\implies$ **(b)**. Assume for contradiction that there are $F \in \mathcal{F}$ and $G \in \mathcal{G}$ such that $L_{t+1}(G) \sqsubseteq_{t+1} R_{t+1}(F)$. Define

$$\mathcal{F}' = \{F' \in \mathcal{F} : L_{t+1}(G) \sqsubseteq_{t+1} R_{t+1}(F')\}.$$

By definition $F \in \mathcal{F}'$, i.e., $\mathcal{F}'$ is non-empty. Let F_0 be any minimal element of $\mathcal{F}'$ with respect to $\sqsubseteq_a$, i.e., assume there is no $F' \in \mathcal{F}'$, $F' \neq F_0$ such that $F' \sqsubseteq_a F_0$. To obtain a contradiction it is enough to show that $|F_0 \cap G| \geqslant t+1$ (this would mean that $\mathcal{F}$ and $\mathcal{G}$ are not t-far).

Assume that $|F_0 \cap G| < t+1$. Hence there is an element of $L_{t+1}(G)$ which is not in F_0. Namely, there is $i \in \{1, 2, \ldots, t+1\}$ such that $m(G, i) \notin F_0$. Define

$$F_1 = (F_0 \setminus \{m(F_0, a-t-1+i)\}) \cup \{m(G, i)\}.$$

First of all, observe that $|F_1| = |F_0| = a$ (this is because $m(F_0, a-t-1+i) \in F_0$ and $m(G, i) \notin F_0$). Let us check that the following three claims hold:

$$F_1 \in \mathcal{F}' \tag{20}$$

$$F_1 \neq F_0 \tag{21}$$

$$F_1 \sqsubseteq_a F_0. \tag{22}$$

These three claims give a contradiction with minimality of F_0.

The simplest one is (21) — observe that F_1 contains $m(G, i)$ and F_0 does not.

Now, let us show (22). Recall that $F_0 \in \mathcal{F}'$, i.e., $L_{t+1}(G) \sqsubseteq_{t+1} R_{t+1}(F_0)$. Hence $m(G, i) = m(L_{t+1}(G), i) \leqslant m(R_{t+1}(F_0), i) = m(F_0, a-t-1+i)$, i.e., F_1 is obtained from F_0 by removing a bigger element and adding a smaller element (which originally was not in F_0). Hence by Proposition 16 we have that $F_1 \sqsubseteq_a F_0$.

To show (20) let us at first show that $F_1 \in \mathcal{F}$. Indeed, $\mathcal{F}$ is an ideal of $\sqsubseteq_a$ and $F_0 \in \mathcal{F}' \subseteq \mathcal{F}$. Hence by (22) we have that $F_1 \in \mathcal{F}$. To show that actually $F_1 \in \mathcal{F}'$ we have to prove that $L_{t+1}(G) \sqsubseteq_{t+1} R_{t+1}(F_1)$. Define

$$X = (R_{t+1}(F_0) \setminus \{m(F_0, a-t-1+i)\}) \cup \{m(G, i)\}.$$

Observe that X is a $(t+1)$-element subset of F_1. Note that $m(G, i) = m(L_{t+1}(G), i) \notin R_{t+1}(F_0)$ and $m(F_0, a-t-1+i) = m(R_{t+1}(F_0), i)$ and recall once again that $L_{t+1}(G) \sqsubseteq_{t+1} R_{t+1}(F_0)$. Thus X is obtained from $R_{t+1}(F_0)$ by removing the ith element of $R_{t+1}(F_0)$ and adding the ith element of $L_{t+1}(G)$. Hence by Proposition 17 we have that $L_{t+1}(G) \sqsubseteq_{t+1} X$. On the other hand obviously $X \sqsubseteq_{t+1} R_{t+1}(F_1)$, which means that (20) is proved. □

4.4 Probabilistic lemma

To upperbound $|\mathcal{F}| \cdot |\mathcal{G}|$, where $\mathcal{F}, \mathcal{G} \subseteq \binom{[n]}{a}$ are t-far and $\mathcal{F}$ is an ideal of the order $\sqsubseteq_a$, we use an approach suggested in [10]. We introduce a probabilistic measure μ_p on the set $2^{[n]}$ such that the probability of a subset $X \in 2^{[n]}$ is equal to $p^{|X|}(1-p)^{n-|X|}$. It is easy to see that this measure is a product of Bernoulli measures: each point x belongs to a random set X with probability p and points are included in the set independently.

Lemma 20. *Let $\mathcal{F}, \mathcal{G} \subseteq \binom{[n]}{a}$ be such that $L_{t+1}(G) \not\sqsubseteq_{t+1} R_{t+1}(F)$ for all $F \in \mathcal{F}, G \in \mathcal{G}$. Define $\mathbf{X}$ and $\mathbf{Y}$ to be two independent random variables, both distributed according to $\mu_{a/n}$. Then*

$$\mathbf{Pr}[\mathbf{X} \in \mathcal{F}, \mathbf{Y} \in \mathcal{G}] \leqslant 4n \cdot \exp\left(-(a-t-1)^2/(20a)\right).$$

We will use the following form of the Chernoff bound:

Proposition 21 ([12], Theorem 1)**.** *Let $Z_1, \ldots, Z_l$ be l independent Bernoulli random variables. Assume that each Z_i takes value 1 with probability p. Then for all $\varepsilon \geqslant 0$:*

$$\mathbf{Pr}\left[\sum_{i=1}^{l} Z_i \geqslant (p+\varepsilon)l\right] \leqslant \exp\left(-D(p+\varepsilon||p) \cdot l\right)$$
$$\mathbf{Pr}\left[\sum_{i=1}^{l} Z_i \leqslant (p-\varepsilon)l\right] \leqslant \exp\left(-D(p-\varepsilon||p) \cdot l\right),$$

where $D(x||y)$ is the Kullback – Leibler divergence:

$$D(x||y) = x \ln\left(\frac{x}{y}\right) + (1-x)\ln\left(\frac{1-x}{1-y}\right).$$

We also need the following lower bound on the Kulback – Leibler divergence:

Proposition 22 ([24])**.** *$D(x||y) \geqslant \frac{(x-y)^2}{2(x+y)}$.*

From Propositions 21 and 22 we obtain:

Corollary 23. *Let $Z_1, \ldots, Z_l$ be l independent Bernoulli random variables. Assume that each Z_i takes value 1 with probability p. Then for all $\varepsilon \geqslant 0$:*

$$\mathbf{Pr}\left[\sum_{i=1}^{l} Z_i \notin [(p-\varepsilon)l, (p+\varepsilon)l]\right] \leqslant 2\exp\left(-\frac{\varepsilon^2 \cdot l}{4p+2\varepsilon}\right).$$

Proof of Lemma 20. Denote $s = (a-t-1)$. Let E be the event that for all $r \in \{1, \ldots, n\}$ it holds that

$$|\mathbf{X} \cap [1,r]| \in \left[\frac{a}{n} \cdot r - s/2, \frac{a}{n} \cdot r + s/2\right]$$
$$\text{and } |\mathbf{Y} \cap [1,r]| \in \left[\frac{a}{n} \cdot r - s/2, \frac{a}{n} \cdot r + s/2\right].$$

Let us show that $\mathbf{X} \in \mathcal{F}, \mathbf{Y} \in \mathcal{G} \implies \neg E$. Indeed, assume for contradiction that there are $X \in \mathcal{F}$ and $Y \in \mathcal{G}$ such that event E holds when $\mathbf{X} = X, \mathbf{Y} = Y$. Note that $L_{t+1}(Y) \not\sqsubseteq_{t+1} R_{t+1}(X)$. Hence for some $j \in \{1, \ldots, t+1\}$ it holds that $m(Y, j) > m(X, a-t-1+j) = m(X, s+j)$. Consider $r = m(X, s+j)$. By definition there are exactly $s+j$ elements of X in $[1, r]$. Since event E holds when $\mathbf{X} = X, \mathbf{Y} = Y$, we get:

$$s + j \leqslant \frac{a}{n} \cdot r + s/2. \tag{23}$$

On the other hand there are at most $j-1$ elements of Y in $[1, m(X, s+j)] = [1, r]$ (this is because $m(Y, j) > m(X, s+j)$). Hence

$$\frac{a}{n} \cdot r - s/2 \leqslant j - 1 \tag{24}$$

(we use once again the fact that E holds for (X, Y)). By adding (24) and (23) we get $0 \leqslant -1$. Thus an implication $\mathbf{X} \in \mathcal{F}, \mathbf{Y} \in \mathcal{G} \implies \neg E$ is proved.

In particular, we get:

$$\mathbf{Pr}[\mathbf{X} \in \mathcal{F}, \mathbf{Y} \in \mathcal{G}] \leqslant \mathbf{Pr}[\neg E].$$

Hence it is enough to upper bound the probability of $\neg E$. If $\neg E$ holds, then for some $r \in \{1, \ldots, n\}$ we have:

$$\begin{aligned} |\mathbf{X} \cap [1, r]| &\notin \left[\frac{a}{n} \cdot r - s/2, \frac{a}{n} \cdot r + s/2\right] = \left[\left(\frac{a}{n} - \frac{s}{2r}\right) r, \left(\frac{a}{n} + \frac{s}{2r}\right) r\right] \\ \text{or } |\mathbf{Y} \cap [1, r]| &\notin \left[\frac{a}{n} \cdot r - s/2, \frac{a}{n} \cdot r + s/2\right] = \left[\left(\frac{a}{n} - \frac{s}{2r}\right) r, \left(\frac{a}{n} + \frac{s}{2r}\right) r\right]. \end{aligned}$$

By Corollary 23 both of these events have probability at most

$$\begin{aligned} 2\exp\left(-\frac{\left(\frac{s}{2r}\right)^2 \cdot r}{4 \cdot \frac{a}{n} + 2 \cdot \frac{s}{2r}}\right) &= 2\exp\left(-\frac{s^2}{16 \cdot \frac{ar}{n} + 4s}\right) \\ &\leqslant 2\exp\left(-\frac{s^2}{16 \cdot \frac{an}{n} + 4s}\right) \\ &= 2\exp\left(-\frac{s^2}{16a + 4s}\right) \leqslant 2\exp\left(-\frac{s^2}{20a}\right) \end{aligned}$$

Hence the probability of the union of these two events is at most twice as large as the last expression. Then by summing over all $a \in \{1, \ldots, n\}$ we get the required bound.

□

4.5 Tying up loose ends — proof of Theorem 7

Assume that $\mathcal{F} \subseteq \binom{[n]}{a}$ and $\mathcal{G} \subseteq \binom{[n]}{a}$ are t-far. By Lemma 15 there are $\mathcal{F}', \mathcal{G}' \subseteq \binom{[n]}{m}$ satisfying the following three conditions:

- $\mathcal{F}'$ and $\mathcal{G}'$ are t-far;
- $|\mathcal{F}'| = |\mathcal{F}|$ and $|\mathcal{G}'| = |G|$;
- $\mathcal{F}'$ is left-compressed.

By Proposition 18 we have that $\mathcal{F}'$ is an ideal of $\sqsubseteq_a$. Then by Lemma 19 we get that $L_{t+1}(G) \not\sqsubseteq_{t+1} R_{t+1}(F)$ for all $F \in \mathcal{F}'$ and $G \in \mathcal{G}'$. Hence by Lemma 20 we have that

$$\mathbf{Pr}[\mathbf{X} \in \mathcal{F}', \mathbf{Y} \in \mathcal{G}'] \leqslant 4n \exp\left(-(a-t-1)^2/(20a)\right), \tag{25}$$

where $\mathbf{X}$ and $\mathbf{Y}$ are two independent random variables distributed according to $\mu_{a/n}$. The left hand side of (25) equals

$$|\mathcal{F}'| \cdot |\mathcal{G}'| \cdot \left[\left(\frac{a}{n}\right)^a \cdot \left(1 - \frac{a}{n}\right)^{n-a}\right]^2.$$

Finally, from the following lower bound on $\binom{n}{a}$ (see [5, Lemma 2.4.2])

$$\binom{n}{a} \geqslant \sqrt{\frac{1}{8n \cdot \frac{a}{n} \cdot \frac{n-a}{n}}} \cdot \left(\frac{n}{a}\right)^a \cdot \left(\frac{n}{n-a}\right)^{n-a},$$

we get:

$$\begin{aligned}
|\mathcal{F}| \cdot |\mathcal{G}| &= |\mathcal{F}'| \cdot |\mathcal{G}'| \\
&\leqslant \left[\left(\frac{n}{a}\right)^a \cdot \left(\frac{n}{n-a}\right)^{n-a}\right]^2 \cdot 4n \exp\left(-(a-t-1)^2/(20a)\right) \\
&\leqslant \left[8n \cdot \frac{a}{n} \cdot \frac{n-a}{n} \cdot \binom{n}{a}^2\right] \cdot 4n \exp\left(-(a-t-1)^2/(20a)\right) \\
&= 32a(n-a) \cdot \exp\left(-(a-t-1)^2/(20a)\right) \cdot \binom{n}{a}^2.
\end{aligned}$$

5 Communication lower bound — Proof of Theorem 6

Our proof relies on Theorem 7. Since we are dealing with k-party setting, we need the following k-dimensional generalization of Theorem 7. Fortunately, this generalization can be obtained via a very simple induction argument.

Lemma 24. *For all $n, a, t, k \in \mathbb{N}$ satisfying $t < a < n$ the following holds. Assume that $\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_k \subseteq \binom{[n]}{a}$ are such that*

$$|\mathcal{F}_i| \geqslant 2^{k-2} \cdot \sqrt{32a(n-a)} \cdot \exp\left(-\frac{(a-t-1)^2}{40a}\right) \cdot \binom{n}{a} + 2^{k-2}$$

for all $i \in \{1, 2, \ldots, k\}$. Then there are $F_1 \in \mathcal{F}_1, F_2 \in \mathcal{F}_2, \ldots, F_k \in \mathcal{F}_k$ such that $|F_1 \cap F_i| \geqslant t+1$ for all $i \in \{2, \ldots, k\}$.

Proof. For $t < a < n$ let $A_{a,t}^{k,n}$ be the minimal positive integer N such that for all $\mathcal{F}_1, \ldots, \mathcal{F}_k \subseteq \binom{[n]}{a}$ the following holds. If $|\mathcal{F}_i| \geqslant N$ for all $i \in \{1, \ldots, k\}$, then there are $F_1 \in \mathcal{F}_1, F_2 \in \mathcal{F}_2, \ldots, F_k \in \mathcal{F}_k$ such that $|F_1 \cap F_i| \geqslant t+1$ for all $i \in \{2, \ldots, k\}$.

Let us verify that $A_{a,t}^{k,n}$ is non-decreasing in k, i.e.:

$$A_{a,t}^{k,n} \leqslant A_{a,t}^{k+1,n} \tag{26}$$

for all $k \geqslant 2$ and $t < a < n$. Indeed, assume that $\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_k \subseteq \binom{[n]}{a}$ are such that $|\mathcal{F}_i| \geqslant A_{a,t}^{k+1,n}$ for all $i \in \{1, \ldots, k\}$. Apply definition of $A_{a,t}^{k+1,n}$ to $\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_{k+1}$, where $\mathcal{F}_{k+1} = \binom{[n]}{a}$. To do so we should additionally check that $|\mathcal{F}_{k+1}| \geqslant A_{a,t}^{k+1,n}$, but in our case this is obvious because $A_{a,t}^{k+1,n} \leqslant \binom{n}{a}$. In this way we obtain $F_1 \in \mathcal{F}_1, F_2 \in \mathcal{F}_2, \ldots, F_{k+1} \in \mathcal{F}_{k+1}$ satisfying $|F_1 \cap F_i| \geqslant t+1$ for all $i \in \{2, \ldots, k+1\}$. Thus (26) is proved (we only need $F_1, F_2, \ldots, F_k$).

Theorem 7 implies that:

$$A_{a,t}^{2,n} \leqslant \left\lfloor \sqrt{32a(n-a)} \cdot \exp\left(-\frac{(a-t-1)^2}{40a}\right) \cdot \binom{n}{a} \right\rfloor + 1.$$

Indeed, assume that $\mathcal{F}_1, \mathcal{F}_2 \subseteq \binom{[n]}{a}$ are such that:

$$|\mathcal{F}_1|, |\mathcal{F}_2| \geqslant \left\lfloor \sqrt{32a(n-a)} \cdot \exp\left(-\frac{(a-t-1)^2}{40a}\right) \cdot \binom{n}{a} \right\rfloor + 1.$$

Then $|\mathcal{F}_1| \cdot |\mathcal{F}_2|$ is strictly larger than $32a(n-a) \cdot \exp\left(-\frac{(a-t-1)^2}{20a}\right) \cdot \binom{n}{a}^2$. By Theorem 7 this means that there are $F_1 \in \mathcal{F}_1, F_2 \in \mathcal{F}_2$ such that $|F_1 \cap F_2| \geqslant t+1$.

To show the lemma it is enough to demonstrate that

$$A_{a,t}^{k+1,n} \leqslant 2 \cdot A_{a,t}^{k,n},$$

for all $k \geqslant 2$ and $t < a < n$. To do so, fix $k+1$ families $\mathcal{F}_1, \ldots, \mathcal{F}_{k+1} \subseteq \binom{[n]}{a}$. Assume that $|\mathcal{F}_i| \geqslant 2 \cdot A_{a,t}^{k,n}$ for all $i \in \{1, \ldots, k+1\}$. Our goal is to show that there are $F_1 \in \mathcal{F}_1, \ldots, F_{k+1} \in \mathcal{F}_{k+1}$ satisfying

$$|F_1 \cap F_i| \geqslant t+1 \text{ for all } i \in \{2, \ldots, k+1\}.$$

Denote $N = A_{a,t}^{k,n}$. We claim that there are N distinct $G_1, \ldots, G_N \in \mathcal{F}_1$ such that for every $j \in \{1, 2, \ldots, N\}$ there are $F_2^j \in \mathcal{F}_2, \ldots, \mathcal{F}_k^j \in \mathcal{F}_k$ satisfying $|G_j \cap F_i^j| \geqslant t+1$ for all $i \in \{2, \ldots, k\}$.

We construct such $G_1, \ldots, G_N$ one by one. Assume that $G_1, \ldots, G_j$ for some $j \in \{0, \ldots, N-1\}$ are already constructed. Notice that:

$$|\mathcal{F}_1 \setminus \{G_1, \ldots, G_j\}| \geqslant 2 \cdot A_{a,t}^{k,n} - j \geqslant 2 \cdot A_{a,t}^{k,n} - N = A_{a,t}^{k,n},$$

$$|\mathcal{F}_i| \geqslant 2 \cdot A_{a,t}^{k,n} \geqslant A_{a,t}^{k,n}, \qquad i = 2, \ldots, k.$$

This means by definition of $A^{k,n}_{a,t}$ that there are $G \in \mathcal{F}_1 \setminus \{G_1, \ldots, G_j\}$, $H_2 \in \mathcal{F}_2, \ldots, H_k \in \mathcal{F}_k$ satisfying:

$$|G \cap H_i| \geqslant t + 1 \text{ for all } i \in \{2, \ldots, k\}.$$

Then we set $G_{j+1} = G, F_2^{j+1} = H_2, \ldots, F_k^{j+1} = H_k$. Note that G_{j+1} is distinct from $G_1, \ldots, G_j$ because $G \notin \{G_1, \ldots, G_j\}$.

Finally, consider two families $\{G_1, \ldots, G_N\}$ and $\mathcal{F}_{k+1}$. These two families are both of size at least $N = A^{k,n}_{a,t} \geqslant A^{2,n}_{a,t}$ (the last inequality here is by (26)). Hence there are $j \in \{1, \ldots, N\}$ and $H_{k+1} \in \mathcal{F}_{k+1}$ such that $|G_j \cap H_{k+1}| \geqslant t+1$. To finish the proof set $F_1 = G_j, F_2 = F_2^j, \ldots, F_k = F_k^j$ and $F_{k+1} = H_{k+1}$. □

We are now ready to prove Theorem 6.

Proof of theorem 6. Set $a = \lfloor n/k \rfloor$ and $t = \lfloor (1 - \gamma/4)a \rfloor$. Note that

$$a - t \geqslant \frac{\gamma a}{4} \geqslant \frac{\gamma(n/k - 1)}{4} = \frac{n}{4 \cdot \frac{k}{\gamma}} - \frac{\gamma}{4} \geqslant 25\sqrt{n} - \frac{1}{4} \tag{27}$$

Here we use the assumption that $\frac{k}{\gamma} \leqslant \frac{\sqrt{n}}{100}$. In particular (27) means that $t < a$ for all large enough n.

Define

$$\mathcal{D} = \left\{ (X_1, \ldots, X_k) \in \binom{[n]}{a}^k : X_1, X_2, \ldots, X_k \text{ are disjoint} \right\},$$

$$\mathcal{I} = \left\{ (F_1, \ldots, F_k) \in \binom{[n]}{a}^k : |F_i \triangle F_{i'}| \leqslant \gamma a \text{ for all } i, i' \in \{1, \ldots, k\} \right\}.$$

Observe that:

$$|\mathcal{D}| = \binom{n}{a} \cdot \binom{n-a}{a} \cdot \ldots \cdot \binom{n - (k-1) \cdot a}{a} > 0. \tag{28}$$

Assume that there is a c-bit non-deterministic communication protocol for $\mathrm{DISJ}'_{k,\gamma}(n)$. Hence there is a cover of $\mathcal{D}$ by at most 2^c boxes which are disjoint with $\mathcal{I}$. Among these boxes there is one which contains at least $|\mathcal{D}|/2^c$ elements of $\mathcal{D}$. Let this box be $\mathcal{F}_1 \times \ldots \times \mathcal{F}_k$ for some $\mathcal{F}_1, \ldots, \mathcal{F}_k \subseteq \binom{[n]}{a}$.

Let us show that for some $i \in \{1, \ldots, k\}$ it holds that

$$|\mathcal{F}_i| < 2^{k-2} \cdot \sqrt{32a(n-a)} \cdot \exp\left(-\frac{(a-t-1)^2}{40a} \right) \cdot \binom{n}{a} + 2^{k-2}. \tag{29}$$

Indeed, assume that it is not true. Then, since $t < a < n$, we can apply Lemma 24 to find $F_1 \in \mathcal{F}_1, F_2 \in \mathcal{F}_2, \ldots, F_k \in \mathcal{F}_k$ such that $|F_1 \cap F_i| \geqslant t + 1$ for all

$i \in \{2, \dots, k\}$. Note also that $|F_1 \cap F_1| = |F_1| = a \geqslant t+1$. From that for every $i, i' \in \{1, \dots, k\}$ we obtain:

$$\begin{aligned}|F_i \triangle F_{i'}| &\leqslant |F_i \triangle F_1| + |F_{i'} \triangle F_1| \\ &= |F_i| + |F_1| - 2|F_i \cap F_1| + |F_{i'}| + |F_1| - 2|F_{i'} \cap F_1| \\ &\leqslant 4 \cdot a - 4 \cdot (t+1) \leqslant \gamma a.\end{aligned}$$

This means that $\mathcal{F}_1 \times \mathcal{F}_2 \times \ldots \times \mathcal{F}_k$ intersects $\mathcal{I}$, contradiction.

Take any $i \in \{1, 2, \dots, k\}$ satisfying (29). Recall that by definition there are at least $|\mathcal{D}|/2^c$ elements of $\mathcal{D}$ in $\mathcal{F}_1 \times \mathcal{F}_2 \times \ldots \times \mathcal{F}_k$. On the other hand, notice that for any fixed $X \in \binom{[n]}{a}$ there are exactly

$$\binom{n-a}{a} \cdot \ldots \cdot \binom{n-(k-1)\cdot a}{a}.$$

elements of $\mathcal{D}$ with the ith coordinate equals to X. Hence there are at most

$$|\mathcal{F}_i| \cdot \binom{n-a}{a} \cdot \ldots \cdot \binom{n-(k-1)\cdot a}{a}$$

elements of $\mathcal{D}$ in $\mathcal{F}_1 \times \mathcal{F}_2 \times \ldots \times \mathcal{F}_k$. By combining these two bounds we obtain:

$$|\mathcal{D}|/2^c \leqslant |\mathcal{F}_i| \cdot \binom{n-a}{a} \cdot \ldots \cdot \binom{n-(k-1)\cdot a}{a}.$$

By (28) this transforms to

$$2^c \geqslant \frac{\binom{n}{a}}{|\mathcal{F}_i|}.$$

Recall that the size of $\mathcal{F}_i$ satisfies (29). This gives us the following:

$$\begin{aligned}2^c &\geqslant \frac{\binom{n}{a}}{2^{k-2} \cdot \sqrt{32a(n-a)} \cdot \exp\left(-\frac{(a-t-1)^2}{40a}\right) \cdot \binom{n}{a} + 2^{k-2}} \\ &\geqslant \frac{1}{2} \min\left\{\frac{\exp\left(\frac{(a-t-1)^2}{40a}\right)}{2^{k-2}\sqrt{32a(n-a)}}, \frac{\binom{n}{a}}{2^{k-2}}\right\} \\ &\geqslant \frac{1}{2} \min\left\{\frac{\exp\left(\frac{(a-t-1)^2}{40a}\right)}{2^{k-2}\sqrt{32} \cdot n}, \frac{\binom{n}{a}}{2^{k-2}}\right\}.\end{aligned}$$

After taking $\log_2$ of the last inequality (bearing in mind that $\log_2(e) > 1$) we obtain that c for all large enough n satisfies the following:

$$c \geqslant \min\left\{\frac{(a-t-1)^2}{40a} - k - 1.5\log_2(n), \log_2\left(\binom{n}{a}\right) - k\right\}$$

(here we subtract $0.5\log_2(n)$ from the first argument of min to compensate negative constant terms). It remains to demonstrate that both expressions in the minimum above are at least $\frac{\gamma^2 n}{10^4 \cdot k} - 2\log_2(n)$ for all large enough n:

$$\frac{(a-t-1)^2}{40a} - k - 1.5\log_2(n) \geqslant \frac{\gamma^2 n}{10^4 \cdot k} - 2\log_2(n), \tag{30}$$

$$\log_2\left(\binom{n}{a}\right) - k \geqslant \frac{\gamma^2 n}{10^4 \cdot k} - 2\log_2(n). \tag{31}$$

Let us start with (30). At first, note that

$$a - t - 1 \geqslant \frac{\gamma a}{4} - 1 \geqslant \frac{\gamma a}{8},$$

where the last inequality is because γa is large enough:

$$\gamma a \geqslant \gamma(n/k - 1) \geqslant 100\sqrt{n} - 1.$$

(in the second inequality of the last line we use the assumption that $\frac{k}{\gamma} \leqslant \frac{\sqrt{n}}{100}$). In particular, $a - t - 1$ is positive. Hence

$$\begin{aligned}\frac{(a-t-1)^2}{40a} - k - 1.5\log_2(n) &\geqslant \frac{\gamma^2 a}{2560} - k - 1.5\log_2(n)\\ &\geqslant \frac{\gamma^2 n}{2560 \cdot k} - k - 2\log_2(n)\end{aligned}$$

(here once again we subtract $0.5\log_2(n)$ to compensate a negative constant term which is due to rounding of $a = \lfloor n/k \rfloor$). To prove (30) it remains to notice that $k \leqslant \frac{\gamma^2 n}{10^4 \cdot k}$ because $\frac{k}{\gamma} \leqslant \frac{\sqrt{n}}{100}$.

To show (31) we will actually show that the left hand side of (31) is at least the left hand side of (30). Indeed,

$$\log_2\left(\binom{n}{a}\right) \geqslant a \cdot \log_2\left(\frac{n}{a}\right) \geqslant a.$$

The last inequality is because $k \geqslant 2$ and hence $a = \lfloor n/k \rfloor \leqslant n/2$. But recall that $a - t - 1$ is positive, which implies that a is at least $\frac{(a-t-1)^2}{40a}$.

□

Acknowledgment. The article was prepared within the framework of the HSE University Basic Research Program and funded by the Russian Academic Excellence Project '5-100'. Mikhail Vyalyi is partially supported by RFBR grant 17–01-00300 and by the state assignment topic no. 0063-2016-0003.

Authors are sincerely grateful to anonymous reviewers for valuable comments.

References

[1] Bashov, M. On minimisation of the double-sided shadow in the unit cube. *Discrete Mathematics and Applications 21* (2011), 517–535.

[2] Bojańczyk, M., and Czerwiński, W. An automata toolbox. A book of lecture notes, available at `https://www.mimuw.edu.pl/ bojan/upload/reduced-may-25.pdf`, 2018.

[3] Borg, P. The maximum product of sizes of cross-t-intersecting uniform families. *Australasian J. Combinatorics 60* (2014), 69–78.

[4] Calude, C. S., Jain, S., Khoussainov, B., Li, W., and Stephan, F. Deciding parity games in quasipolynomial time. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing* (2017), ACM, pp. 252–263.

[5] Cohen, G., Honkala, I., Litsyn, S., and Lobstein, A. *Covering codes*, vol. 54. Elsevier, 1997.

[6] Czerwiński, W., Daviaud, L., Fijalkow, N., Jurdziński, M., Lazić, R., and Parys, P. Universal trees grow inside separating automata: Quasi-polynomial lower bounds for parity games. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms* (2019), SIAM, pp. 2333–2349.

[7] Emerson, E. A., and Jutla, C. S. Tree automata, mu-calculus and determinacy. In *Foundations of Computer Science, 1991. Proceedings., 32nd Annual Symposium on* (1991), IEEE, pp. 368–377.

[8] Erdos, P., Ko, C., and Rado, R. Intersection theorems for systems of finite sets. *The Quarterly Journal of Mathematics 12* (1961), 313–320.

[9] Fearnley, J., Jain, S., Schewe, S., Stephan, F., and Wojtczak, D. An ordered approach to solving parity games in quasi polynomial time and quasi linear space. In *Proceedings of the 24th ACM SIGSOFT International SPIN Symposium on Model Checking of Software* (2017), ACM, pp. 112–121.

[10] Frankl, P., and Rödl, V. Forbidden intersections. *Transactions of the American Mathematical Society 300*, 1 (1987), 259–286.

[11] Gruska, J., Qiu, D., and Zheng, S. Communication complexity of promise problems and their applications to finite automata. *arXiv preprint arXiv:1309.7739* (2013).

[12] HOEFFDING, W. Probability inequalities for sums of bounded random variables. *J. Am. Stat. Associ. 58*, 301 (1963), 13–30.

[13] Jukna, S. *Boolean function complexity: advances and frontiers*, vol. 27. Springer Science & Business Media, 2012.

[14] Jurdziński, M. Deciding the winner in parity games is in up∩ co-up. *Information Processing Letters 68*, 3 (1998), 119–124.

[15] Jurdzinski, M., and Lazic, R. Succinct progress measures for solving parity games. In *2017 32nd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)* (2017), IEEE.

[16] Jurdziński, M., Paterson, M., and Zwick, U. A deterministic subexponential algorithm for solving parity games. *SIAM Journal on Computing 38*, 4 (2008), 1519–1532.

[17] Lehtinen, K. A modal μ perspective on solving parity games in quasipolynomial time. In *2018 33nd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)* (2018), IEEE.

[18] Martin, D. A. A purely inductive proof of borel determinacy. In *Recursion Theory, Proceedings of Symposia in Pure Mathematics* (1985), vol. 42, American Mathematical Society, pp. 303–308.

[19] McNaughton, R. Infinite games played on finite graphs. *Annals of Pure and Applied Logic 65*, 2 (1993), 149–184.

[20] Mostowski, A. W. Games with forbidden positions. Tech. Rep. 78, Uniwersytet Gdánski, Instytut Matematyki, 1991.

[21] Petersson, V., and Vorobyov, S. G. A randomized subexponential algorithm for parity games. *Nordic Journal of Computing 8*, 3 (2001), 324–345.

[22] Schewe, S. Solving parity games in big steps. In *International Conference on Foundations of Software Technology and Theoretical Computer Science* (2007), Springer, pp. 449–460.

[23] Shen, A., Uspensky, V. A., and Vereshchagin, N. *Kolmogorov complexity and algorithmic randomness*, vol. 220. American Mathematical Soc., 2017.

[24] Topsoe, F. Some inequalities for information divergence and related measures of discrimination. *IEEE Transactions on information theory 46*, 4 (2000), 1602–1609.

A Reduction to finite time

Proposition 25. *Assume that a deterministic finite automaton $\mathcal{A}$ with q states separates* $\text{EvenCycles}_{n,d}$ *from* $\text{OddCycles}_{n,d}$*. Then $\mathcal{A}$ separates* $\text{EvenCycles}_{n,d}$ *from* $\text{OddCycles}_{n,d}$ *in time qn.*

Proof. Let Q be the set of states of $\mathcal{A}$ and let q_{start} be the initial state of $\mathcal{A}$. Without loss of generality we may assume that q_{accept} is an absorbing state of $\mathcal{A}$, i.e.,

$$\delta_{\mathcal{A}}(q_{accept}, a) = q_{accept}$$

for all $a \in [n] \times \{1, 2, \dots, d\}$. Thus it is enough to show that for every $w \in \mathrm{EvenCycles}_{n,d}$ there exists $i \in \{1, 2, \dots, qn\}$ such that

$$\delta_{\mathcal{A}}(q_{start}, w_1 \dots w_i) = q_{accept}.$$

Assume that for some $w = (v_1, l_1)(v_2, l_2)(v_3, l_3) \dots \in \mathrm{EvenCycles}_{n,d}$ this is false. Let G be an even game graph with at most n nodes which has an infinite path corresponding to w. Define a mapping $\phi\colon [qn + 1] \to [n] \times Q$ as follows:

$$\phi(i) = (v_i, \delta_{\mathcal{A}}(q_{start}, w_1 \dots w_{i-1})).$$

By the pigeonhole principle there are $i, j \in [qn + 1]$, $i < j$ such that

$$\phi(i) = \phi(j).$$

I.e., $v_i = v_j$ and $\delta_A(q_{start}, w_1 \dots w_{i-1}) = \delta_A(q_{start}, w_1 \dots w_{j-1})$. Consider the following infinite path w' of G. This path starts at v_1 and goes to v_i by edges encoded in $w_1 \dots w_{i-1}$. Then it stays forever on a cycle starting at $v_i = v_j$ and formed by edges encoded in $w_i \dots w_{j-1}$. It is easy to see that the only states $\mathcal{A}$ reaches on w' are:

$$q_{start}, \delta_{\mathcal{A}}(q_{start}, w_1), \dots, \delta_{\mathcal{A}}(q_{start}, w_1 w_2 \dots w_{j-1}).$$

By our assumption $\delta_{\mathcal{A}}(q_{start}, w_1), \dots, \delta_{\mathcal{A}}(q_{start}, w_1 w_2 \dots w_{j-1})$ are all different from q_{accept} (and q_{start} is obviously too, because otherwise $\mathcal{A}$ reaches q_{accept} on every word). On the other hand w' is an infinite path of an even game graph with at most n nodes, contradiction. □