

# A plurality problem with three colors and query size three

Dániel Gerbner<sup>a,\*</sup>, Dániel Lenger<sup>b</sup>, Máté Vizer<sup>a</sup>

<sup>a</sup> Alfréd Rényi Institute of Mathematics, Budapest, Hungary

<sup>b</sup> Department of Computer Science, ELTE, Budapest, Hungary

## ARTICLE INFO

### Article history:

Received 28 June 2021

Received in revised form 5 August 2022

Accepted 20 August 2022

Available online 6 September 2022

### Keywords:

Plurality problem

Partition problem

Worst case analysis

## ABSTRACT

The Plurality problem - introduced by Aigner - has many variants. In this article we deal with the following version: suppose we are given  $n$  balls, each of them colored by one of three colors. A *plurality ball* is one such that its color class is strictly larger than any other color class. Questioner asks a triplet (or a  $k$ -set in general), and Adversary as an answer gives the partition of the triplet (or the  $k$ -set) into color classes. Questioner wants to find a plurality ball as soon as possible or show that there is no such ball, while Adversary wants to postpone this.

We denote by  $A_p(n, k)$  the largest number of queries needed to ask in the worst case if both play optimally. We provide an almost precise result in the case of even  $n$  by proving that for  $n \geq 4$  even we have

$$\frac{3}{4}n - 2 \leq A_p(n, 3) \leq \frac{3}{4}n - \frac{1}{2},$$

and for  $n \geq 3$  odd we have

$$\frac{3}{4}n - O(\log n) \leq A_p(n, 3) \leq \frac{3}{4}n - \frac{1}{2}.$$

We also prove some bounds on the number of queries needed to ask in the case  $k \geq 3$ .

© 2022 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

In combinatorial search theory the so-called **Plurality problem** is the following problem: we are given  $n$  indexed balls, say  $\{1, 2, \dots, n\}$  ( $=: [n]$ ), each colored with one of  $c \geq 2$  colors given at the beginning. A **plurality ball** is one such that its color class is strictly larger than any other color class. The aim in the Plurality problems is to decide whether there exists a plurality ball and even to show one (if there exists one).

One can reformulate this problem as a game played between two players, Questioner and Adversary. In each **round** Questioner asks a **query** of size  $k \geq 2$  (i.e. a subset of  $[n]$  that contains  $k$  elements) and Adversary answers the query in some way that we will specify later. The game ends, if Questioner can decide whether there is a plurality ball (and even show one if there exists one). Note that if Questioner asks all the  $k$ -subsets of  $[n]$ , then we can see for any two balls whether they have the same color, hence we can completely identify the color classes, thus we can either find a plurality ball or show that there is none. Therefore, Questioner's aim is to minimize the number of asked queries in this game, while

\* Corresponding author.

E-mail addresses: [gerbner@renyi.hu](mailto:gerbner@renyi.hu) (D. Gerbner), [leda1648@gmail.com](mailto:leda1648@gmail.com) (D. Lenger), [vizermate@gmail.com](mailto:vizermate@gmail.com) (M. Vizer).

Adversary wants to maximize it. In this paper we will be interested in the **maximal length** of the game, i.e. the maximum number of asked queries, when both players play optimally. In other words, we study the **worst case complexity** of the Plurality problem.

A **model** of the Plurality problem is defined by the number  $n$  of balls, the number  $c$  of colors, the size  $k$  of the queries asked by Questioner, and the possible answers of Adversary.

If  $c = 2$ , this problem is called the **Majority problem** (where Questioner should find a ball of the color that occurs more than half of the time) that was raised by J. Moore [15] in connection with finding the majority vote among  $n$  processors, using the minimum number of paired comparisons. The **Pairing model** of the Majority problem is where the query is a pair of balls and the answer is whether their colors coincide or not. It was investigated by Saks and Werman [16], who determined the maximum length precisely (later a simpler proof was found by Wiener [18]), while generalizations for larger query size were investigated in [6,9,10,12,14].

In [1] Aigner introduced the Plurality problem as one of the possible generalizations of the Majority problem. Since then the Plurality problem has attracted a lot of attention, see e.g. [2–5,8,11,13,17]. In [8] it is written, that “[The Plurality problem] seems to be a more difficult variant [than other variants]”, and this is supported by the fact that the Plurality problem in the pairing model is still unsolved. A lower bound of  $3n/2 + O(1)$  and an upper bound of  $5n/3 + O(1)$  were proved in [3], and this upper bound was conjectured to be sharp in [2].

In this article we consider models of the Plurality problem where the query size is larger than two. Such research for the Majority problem were carried out in [6,12]. Note that unlike in the case of queries of size two, there are several different ways Adversary can answer, which creates several different models. In this article we will be interested in the model where Adversary partitions the query into color classes. Another model was studied in [7].

**Structure of the paper.** In Section 2 we introduce our model and state our results, in Section 3 we prove the upper bounds of both Theorem 1 and Theorem 3. In Section 4, we prove the lower bound of Theorem 1. In Section 5, we prove the lower bound of our main result Theorem 3. In Subsection 5.1 we present the proof for  $n$  even in full details (lower bound of Theorem 3 (i)) and in Subsection 5.2 we sketch the lower bound of our main result for  $n$  odd (lower bound of Theorem 3 (ii)). We finish our article with some open questions and remarks in Section 6.

## 2. Our results

First let us specify our model for the Plurality problem, and then state our result. We are given  $n$  indexed balls, each colored with one of **three** given colors. Questioner asks queries of **size**  $k$  (i.e. subsets of  $[n]$  of size  $k$ ) and Adversary's answer is a **partition of the query set into color classes**. We emphasize that Adversary does not give any information of the color of the balls, only a partition of the query set to three parts. We denote the maximum number of queries if both players play optimally by  $A_p(n, k)$ .

Let us remark that even asking all the possible queries does not give any information on the actual colors, thus the most information one can obtain is a partition of  $[n]$  into three color classes. If this is Questioner's goal, we call this problem the **Partition problem** and we denote by  $A_{par}(n, k)$  the maximum length of this game. Note that  $A_p(n, k) \leq A_{par}(n, k)$ .

### 2.1. General result - query size of $k$

First we prove some general bounds on  $A_p(n, k)$  and  $A_{par}(n, k)$ :

**Theorem 1.** For  $n$  even with  $n \geq k \geq 2$  we have

$$\frac{n-2}{k-1} + \frac{n}{2(k-1)^2} \leq A_p(n, k) \leq A_{par}(n, k) \leq \left\lceil \frac{n-1}{k-1} \right\rceil + \left\lceil \frac{n-1}{(k-1)^2} \right\rceil.$$

For  $n$  odd with  $n \geq k \geq 2$  we have

$$\frac{n-5}{k-1} + \frac{n-k}{2(k-1)^2} \leq A_p(n, k) \leq A_{par}(n, k) \leq \left\lceil \frac{n-1}{k-1} \right\rceil + \left\lceil \frac{n-1}{(k-1)^2} \right\rceil.$$

**Remark 2.** We remark that for  $k = 2$ , the lower bound is up to additive constant the same as the one in [3].

### 2.2. Query size of three

In the case of  $k = 2$  there is a linear (in the number of balls) gap between the best known lower and upper bound on  $A_p(n, 2)$  (see [3]), as we mentioned in the introduction. In Theorem 1 we also have linear gaps between the lower and upper bound on  $A_p(n, k)$  for larger  $k$ .

Surprisingly, if the query size is three, then we can prove an (almost) precise result for the Plurality and the Partition problem in this model in the cases  $n = 4m$  or  $n = 4m + 2$ . That means that in the former case we can exactly determine  $A_p(n, 3)$ , while we give two possible values in the latter case. If  $n$  is odd we can give the asymptotics of  $A_p(n, 3)$ .

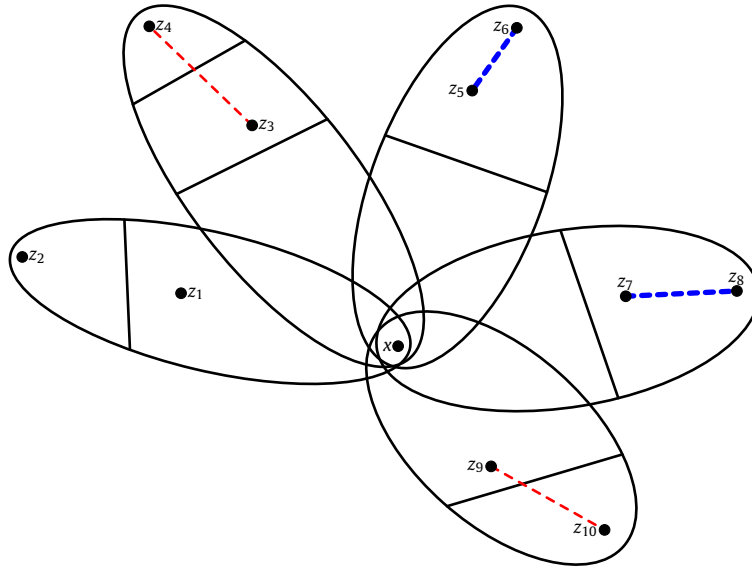


Fig. 1. Phase 1. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

**Theorem 3.** (i) For  $n \geq 4$  even we have

$$\frac{3}{4}n - 2 \leq A_p(n, 3) \leq A_{par}(n, 3) \leq \frac{3}{4}n - \frac{1}{2}.$$

(ii) For  $n \geq 3$  odd we have

$$\frac{3}{4}n - \frac{1}{2} \log_2(n) - 5 \leq A_p(n, 3) \leq A_{par}(n, 3) \leq \frac{3}{4}n - \frac{1}{2}.$$

### 3. Proof of the upper bounds of Theorem 1 and Theorem 3

We describe a strategy for Questioner that will prove the upper bound  $A_{par}(n, k) \leq \left\lceil \frac{n-1}{k-1} \right\rceil + \left\lceil \frac{n-1}{(k-1)^2} \right\rceil$ . It consists of 2 phases.

**Phase 1:** Questioner chooses an arbitrary ball  $x \in [n]$ , puts all the other balls into  $\left\lceil \frac{n-1}{k-1} \right\rceil$  arbitrary (not necessarily disjoint) sets of size  $k-1$  and asks as queries each of these sets with  $\{x\}$  added.

After these queries he makes an auxiliary edge colored graph  $G$  (that he will always change during Phase 2) the following way:

- the vertex set of  $G$  is  $[n]$  minus those balls that turned out to have the same color as  $x$  (including  $x$ ) during Phase 1.
- the edge set is the following: if two balls  $z$  and  $z'$  was asked in the same query, and the answer was that  $x$  has different color than  $z$  and  $z'$ , then he adds the edge  $(z, z')$  to  $G$ . He colors the edge  $(z, z')$  **red** if  $z$  and  $z'$  have different colors, and **blue** if  $z$  and  $z'$  have the same color (Fig. 1). Let  $C_i \subset [n]$  be the set of balls with color  $i$  (for  $i = 1, 2, 3$ ). We can assume that  $x$  has color 1. So we have that  $V(G) = C_2 \cup C_3 = [n] \setminus C_1$ .

Let us remark that inside a connected component of  $G$ , we know the two color classes. Note that as  $G$  contains all the edges between the balls in  $C_2 \cup C_3$  in the queries,  $G$  has at most  $\left\lceil \frac{n-1}{k-1} \right\rceil$  components. If  $k = 3$ , then it is easy to see (using the last query in the case  $n-1$  is odd) that the number of components is exactly  $\left\lfloor \frac{n-1}{2} \right\rfloor$ .

Let us describe the strategy of the next phase informally. Questioner's next goal is to make  $G$  connected, to be able to separate the remaining two color classes. We again connect two balls if they appear in the same query, i.e. if we know whether they have the same color or not.

**Phase 2:** Questioner asks queries that contain balls from as many different components of  $G$  as possible. If he asks  $\{x_1, x_2, \dots, x_k\}$ , then he adds the edges  $(x_i, x_j)$  to  $G$ . He again colors the edge  $(x_i, x_j)$  **red** if  $x_i$  and  $x_j$  have different colors, and **blue** if they have the same color. He continues to ask queries of this type as long as there are at least 2 components in the graph. After each query, the (at most)  $k$  components containing the balls of the query unite in a connected component of the resulting graph (Fig. 2).

Note that at the beginning of Phase 2 there are at most  $\left\lceil \frac{n-1}{k-1} \right\rceil$  components in  $G$ , and their number decreases by  $k-1$  in each round except for the last one, thus there were at most

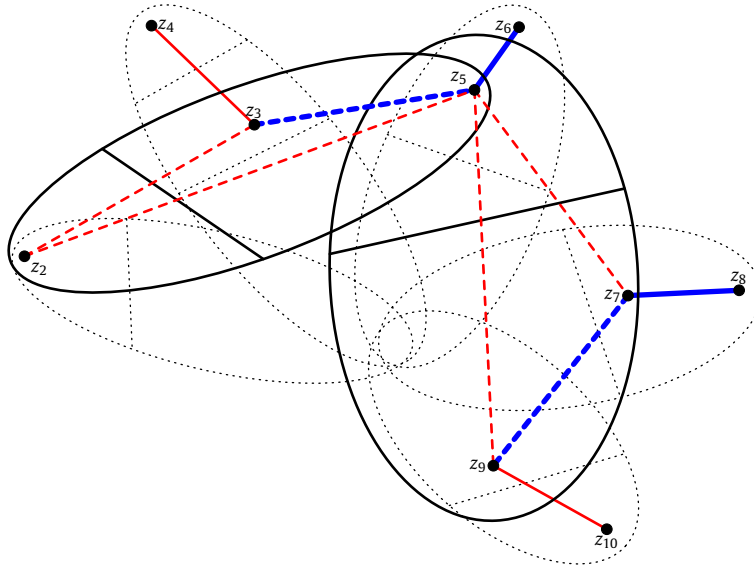


Fig. 2. Phase 2: connecting components from Phase 1.

$$\left\lceil \frac{\left\lfloor \frac{n-1}{k-1} \right\rfloor - 1}{k-1} \right\rceil \leq \left\lceil \frac{n-1}{(k-1)^2} \right\rceil$$

queries during Phase 2. For  $k = 3$ , this value is

$$\left\lceil \frac{\left\lfloor \frac{n-1}{2} \right\rfloor - 1}{2} \right\rceil.$$

Let  $G'$  denote the resulting auxiliary graph, then  $G'$  is connected. In this case Questioner can easily figure out the color classes: let  $y$  be a vertex of  $G'$ , and let its color be e.g. 2. For each other vertex  $z$ , there is a path from  $y$  to  $z$ . Then the parity of the number of the red edges in this (and each other) path between  $y$  and  $z$  shows the color of  $z$ . This means that the sets  $C_2, C_3$  are determined.

For  $k > 3$  altogether at most

$$\left\lceil \frac{n-1}{k-1} \right\rceil + \left\lceil \frac{n-1}{(k-1)^2} \right\rceil$$

queries were asked, so we are done with the proof of the upper bound of Theorem 1.

For  $k = 3$  altogether at most

$$\left\lceil \frac{n-1}{2} \right\rceil + \left\lceil \frac{\left\lfloor \frac{n-1}{2} \right\rfloor - 1}{2} \right\rceil$$

queries were asked. One can easily calculate that if  $n \equiv 0, 1, 2, 3 \pmod{4}$ , then it is equal to

$$\frac{3}{4}n - 1, \frac{3}{4}(n-1), \frac{3}{4}(n-2) + 1, \frac{3}{4}(n-3) + 1,$$

respectively. As

$$\frac{3}{4}(n-2) + 1 = \frac{3}{4}n - \frac{1}{2}$$

is the largest of these numbers, we are done with the proof of the upper bound of Theorem 3.  $\square$

#### 4. Proof of the lower bound of Theorem 1

Here we prove  $\frac{n-2}{k-1} + \frac{n}{2(k-1)^2} \leq A_p(n, k)$ . We generalize the proof of the case  $k = 2$  by Aigner, De Marco and Montanero ([3], Theorem 3.2).

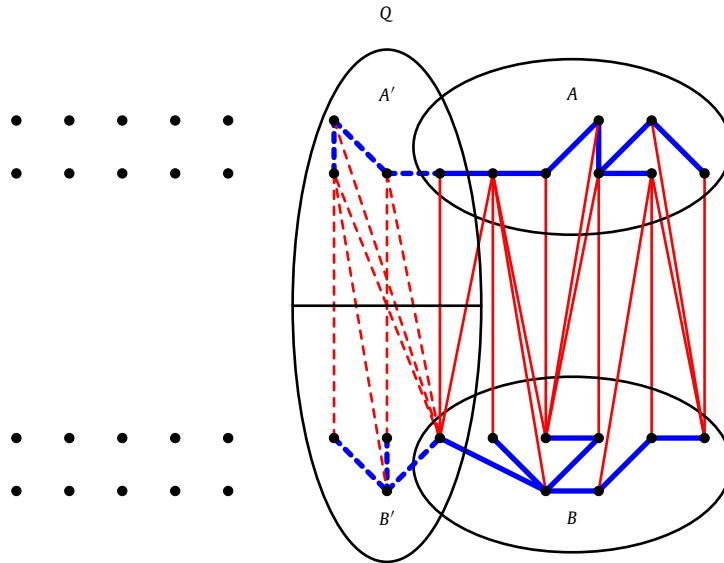


Fig. 3. Asking a query  $Q$ . The newly added edges are dashed.

We say that a coloring of  $n$  balls is a *balanced two-coloring* if there are two colors with  $n/2$  balls and there is no ball of the third color. We say that a coloring is an *almost balanced two-coloring* if there is a color with  $(n-1)/2$  balls, another color with  $(n+1)/2$  balls and there is no ball of the third color.

First we consider the case  $n$  is even. We give a strategy for Adversary. First we define two conditions.

**Condition 1:** we say that an answer of Adversary **fulfills** Condition 1, if there is a balanced 2-coloring of all the balls consistent with all the answers (previous and the one just given).

**Condition 2:** we say that an answer of Adversary **fulfills** Condition 2, if the  $k$  balls of the query belong to at least two color classes.

We will specify the answers of Adversary to the queries only later, but the following will always be true. If there is a possibility to choose an answer that fulfills both Condition 1 and Condition 2, then he chooses such an answer. If there is no such answer, then he answers in such a way that Condition 1 is fulfilled, if possible. If even that is impossible, then he **stops**.

Note that as long as Adversary can fulfill Condition 1, it is a possible solution to the Plurality problem that there is no plurality ball. In other words, the final answer of Questioner can only be that there is no plurality ball.

Adversary consecutively builds an auxiliary graph  $G$  during his algorithm, that is very similar to the graph that was used in the proof of the upper bound. As the graph changes during the algorithm, it would be more precise to denote the auxiliary graph after the  $i$ th query by  $G_i$ , but we omit  $i$  as it is always clear from the context. The vertex set of  $G$  is a subset of the balls. At the beginning  $G$  does not contain any vertex or edge. At the end  $G$  has two (consecutively built) subsets  $A$  and  $B$  of the vertex set that both contain  $n/2$  vertices, and it is consistent with the answers that  $A$  consists of balls of color 1 and  $B$  consists of balls of color 2. Similarly to the proof of the upper bound, blue edges will represent the knowledge that the endpoints have the same color and red edges will represent the knowledge that the endpoints have different color.

The first query is partitioned by Adversary into two non-empty parts  $A$  and  $B$ , and Adversary places arbitrary blue spanning trees into both parts, and an arbitrary red spanning tree between the two parts (he would like to fulfill Condition 1 by considering the coloring where the balls in  $A$  are of color 1 and in  $B$  are of color 2). When a query  $Q$  is asked at any point, Adversary considers the balls of  $Q$  that were not asked so far and moves them arbitrarily into  $A$  or  $B$  in such a way that both  $A$  and  $B$  have sizes at most  $n/2$  (including the newly added balls). Let  $A'$  be the set of balls of  $Q$  in  $A$  and  $B'$  be the set of balls of  $Q$  in  $B$ . Then Adversary **answers** in such a way that the balls in  $A'$  are of color 1, the balls in  $B'$  are of color 2, and there are no balls of the third color in  $Q$ . This answer is clearly consistent with the previous answers and Condition 1 will be satisfied always.

He also updates  $G$  and puts an arbitrary blue spanning tree into  $A'$  and an arbitrary blue spanning tree into  $B'$ . Additionally, in the case  $Q$  contains vertices from both  $A$  and  $B$  (after Adversary moves the new vertices there), he also puts a red edge between any ball that was asked first in  $Q$  and a ball on the other side of  $Q$  (if there is a ball on the other side). He does it in such a way that at most  $k-1$  red edges are added (Fig. 3). In all the cases if the edges he would add are already present, he simply does not add them again, so he does not create multiple edges.

This way either at most  $k-1$  blue, or at most  $k-2$  blue and at most  $k-1$  red edges are added to  $G$  at any step. Let the weight of blue edges be  $1/(k-1)$  and the weight of red edges  $1/(k-1)^2$ . So a query adds at most 1 to the total weight. We will show that the total weight of  $G$  is at least

$$\frac{n-2}{k-1} + \frac{n}{2(k-1)^2}$$

at the end of the algorithm.

**Proposition 4.** *When the algorithm is finished (i.e. Questioner can solve the Plurality problem), there are at least  $n - 2$  blue edges.*

**Proof.** We prove by contradiction. If there would be less than  $n - 2$  blue edges, then we can suppose that one side, e.g.  $A$  is disconnected. Then the following coloring is consistent with all the answers: one component of  $A$  is of color 3, the other ones are of color 1, while  $B$  consists of color 2, thus there is a plurality color. However it is also consistent with the answers that  $A$  and  $B$  are both monochromatic, thus there is no plurality ball. It means the algorithm is not finished, a contradiction.  $\square$

**Proposition 5.** *When the algorithm is finished, there are at least  $n/2$  red edges.*

**Proof.** Observe that every ball  $x$  appears in at least one query, otherwise it is consistent with the answers that  $A \setminus \{x\}$ ,  $B \setminus \{x\}$  and  $x$  are the color classes, in which case there is a plurality ball, a contradiction. Thus every ball  $x$  is asked in a query  $Q$ , hence  $x$  is incident to a red edge, unless the answer was that  $Q$  is monochromatic, and its vertices are all in the same part, say  $A$ . If there is no such answer to any query, we are done, as the  $n$  vertices are incident to at least  $n/2$  red edges. Moreover, let us consider only the queries containing at least one new ball (i.e., balls that were not asked earlier). If none of these queries was monochromatic with vertices all in the same part, then again all the  $n$  vertices are incident to at least one red edge and we are done.

Consider the first query  $Q$  that is monochromatic, contains a new ball  $x$  and its vertices are all in  $A$ . If  $x$  was put into  $B$  instead of  $A$ , then Condition 2 would be fulfilled. There is only one possible reason for putting  $x$  into  $A$  instead: putting  $x$  into  $B$  would violate Condition 1. In that case  $B$  already contains  $n/2$  vertices, and whenever a new vertex was added to  $B$ , it was incident to a red edge. Since red edges do not go between vertices of  $B$ , this means there are at least  $n/2$  red edges.  $\square$

The total weight of the at least  $n - 2$  blue edges and at least  $n/2$  red edges is at least

$$(n-2)/(k-1) + n/2(k-1)^2,$$

as needed and we are done with the lower bound in the case  $n$  is even.

Let us consider now the case  $n$  is odd. Adversary's strategy is similar to the previous one, however in this case we have to solve some technical details that emerge because of the fact that there is no exactly balanced two-coloring of odd many balls. We modify the first condition and recall the second.

**Condition 1:** we say that an answer of Adversary **fulfills** Condition 1, if there is a 3-coloring of the balls consistent with all the answers (previous and the one just given), such that there are  $(n-1)/2$  balls of color 1,  $(n-1)/2$  balls of color 2 and one ball of color 3.

**Condition 2:** we say that an answer of Adversary **fulfills** Condition 2, if not all the  $k$  balls of the query are of the same color.

Again, the following will always be true. If there is a possibility to choose an answer that fulfills both Condition 1 and Condition 2, then he chooses such an answer. If there is no such answer, then he answers in such a way that Condition 1 is fulfilled, if possible. If even that is impossible, then he **stops**.

As long as there is a ball that has not appeared in any queries, Adversary answers the same way as in the even case (the size of  $A$  and  $B$  is  $\frac{n-1}{2}$  instead of  $\frac{n}{2}$ ), and blue and red edges are added similarly to the auxiliary graph  $G$  (and we count them with the same weights at the end as in the even case).

The last time a query  $Q$  contains a new ball  $x$  (if there are more balls in  $Q$  that have not appeared in any query earlier, we just pick an arbitrary one of them), Adversary partitions the new balls in  $Q$  in such a way that there are  $(n-1)/2$  balls in  $A$ ,  $(n-1)/2$  balls in  $B$  (that we call  $B$ ) and the third part is  $\{x\}$ . From this point, he answers in such a way that these three sets are going to be the three color classes. Note that this implies that Condition 1 will be satisfied.

Now we describe how the definition of the auxiliary graph  $G$  changes after  $Q$  (including also  $Q$ ). If a query  $R$  contains  $x$ , Adversary puts a spanning tree of blue edges into  $A \cap R$  and  $B \cap R$  like in the even case, and adds a green edge from a vertex of  $A \cap R$  (if such ball exists) to  $x$  and another green edge from a vertex of  $B \cap R$  (if such ball exists) to  $x$ . For queries not containing  $x$  he does the same as in the even case.

**Lemma 6.** *When the algorithm is finished, there are at least  $n - 5$  blue and green edges (altogether).*

**Proof.** Since  $A$ ,  $B$  and  $\{x\}$  is a coloring consistent with Adversary's answers, Questioner must answer that there are no plurality ball.

Let  $G'$  be the subgraph of  $G$  on the vertex set  $A \cup \{x\}$  with the blue and green edges. If  $G'$  is connected, then there are at least  $\frac{n+1}{2} - 1$  edges in it. Assume that  $G'$  is not connected. Let  $A = A_1 \cup A_2$ ,  $A_1 \cap A_2 = \emptyset$ ,  $A_2 \neq \emptyset$ , such that there are no edges between  $A_1 \cup \{x\}$  and  $A_2$ . In this case,  $A_1, B, A_2 \cup \{x\}$  is also a coloring consistent with Adversary's answers.

As there is no plurality ball, we have  $\max(|A_1|, |A_2 \cup \{x\}|) = |B| = \frac{n-1}{2}$ . If  $|A_1| = |B|$ , then  $A_2 = \emptyset$ , which is a contradiction, thus  $|A_2 \cup \{x\}| = |B|$ , so  $|A_1| = 1$ .

Assume that  $A_2$  is not connected in  $G'$ . Let  $A_2 = A_3 \cup A_4$ ,  $A_3 \cap A_4 = \emptyset$ ,  $A_3 \neq \emptyset$ ,  $A_4 \neq \emptyset$ , such that there are no edges between  $A_3$  and  $A_4$ . Then  $A_1 \cup A_3, B, A_4 \cup \{x\}$  is a 3-coloring with plurality balls (in  $B$ ) that is consistent with the answers. This is a contradiction, so  $A_2$  must be connected. Thus, there are at least  $\frac{n-3}{2} - 1$  edges in  $A_2$ .

Hence there are at least  $\frac{n-5}{2}$  edges in  $A \cup \{x\}$ . Similarly, there are at least  $\frac{n-5}{2}$  edges in  $B \cup \{x\}$ . As each edge is counted only once, we are done with the proof.  $\square$

We also have that

**Proposition 7.** *When the algorithm is finished, there are at least  $(n - k)/2$  red edges.*

**Proof.** In fact we prove that there are at least  $(n - k)/2$  red edges before the last time a query  $Q$  contains a new ball  $x$ . Let us assume first that before choosing  $x$ , there was a query  $R$  where the answer was that  $R$  is monochromatic. In that case there are at least  $n/2$  red edges in  $G$  at that point by the same argument that we used in the even case (in the proof of Proposition 5). Let us consider now the case that there was no monochromatic answer before choosing  $x$ . Similarly to the even case, each ball  $y$  is queried at least once. Indeed, without loss of generality  $y$  is not in  $A$  at the end of the algorithm, but then it is consistent with the answers that  $A \cup \{y\}$  is a color class, and it contains a plurality ball, a contradiction. Therefore, at least  $n - k$  balls appeared in the queries before  $Q$ , and each of them is incident to a red edge. Indeed, such an edge was added when that ball was first queried.  $\square$

Let the weight of the red edges be  $1/(k - 1)^2$  and the weight of the blue edges be  $1/(k - 1)$ , just like in the even case. Let the weight of a green edge be  $1/(k - 1)$ . The weight of any query will be at most 1. Indeed, when no green edge is added, then the argument is the same as in the even case, and if green edges are added, then the blue and green edges form a spanning tree in  $Q$  and no red edges are added. The total weight in  $G$  at the end of the algorithm is at least

$$\frac{n - 5}{k - 1} + \frac{n - k}{2(k - 1)^2},$$

that proves the statement in the case of odd  $n$ .  $\square$

## 5. Proof of the lower bounds of Theorem 3

### 5.1. Proof of the lower bound of Theorem 3 (i)

We give a strategy for Adversary which forces any algorithm that can solve the Plurality problem with queries of size 3 to ask at least

$$\frac{3}{4}n - 2$$

questions for every even  $n$ . We remark that this strategy is based on the one given in the proof of the lower bound in Theorem 1, thus familiarity with Section 4 helps follow this proof.

Let  $\mathcal{S}$  denote the set of the balanced two-colorings of the  $n$  balls. The most important point in Adversary's strategy is that he makes sure that there is always a coloring in  $\mathcal{S}$  that is consistent with the previous answers. He will never violate this.

He makes an auxiliary graph  $G$  to code the information that one can gain from his answers in the following way:

- The vertices of  $G$  are the balls.
- At the beginning, there are no edges in the graph. If he gets the answer to a query  $\{i, j, \ell\}$  that balls  $i$  and  $j$  have the same color, and  $\ell$  has a different color, then he adds the edges  $i\ell$  and  $j\ell$  colored red, and the edge  $ij$  colored blue. If he gets the answer that balls  $i, j$  and  $\ell$  all have the same color, then he adds two of the three edges  $ij, i\ell$  and  $j\ell$  chosen arbitrarily, colored blue. Note that it is possible that we add edges multiple times, so we create a (colored) multigraph (i.e. multiple edges can occur). However for the sake of simplicity we use the word graph instead of multigraph.
- At each point  $G$  contains all the edges that come from the answers given so far. (We note that at any point the graph depends on the queries asked and answers given earlier, but we do not introduce a new notation, since it will not be misleading at any point of the proof.)

We add weights to these edges. Red edges have weight  $\frac{1}{4}$ , blue edges have weight  $\frac{1}{2}$ , so the total weight of every answer is (at most) 1, thus the number of queries asked is at least the sum of the weights. We denote by  $G_R$  the graph spanned by the red edges of  $G$ , and by  $G_B$  the graph spanned by the blue edges of  $G$ . By the strategy of Adversary and the definition

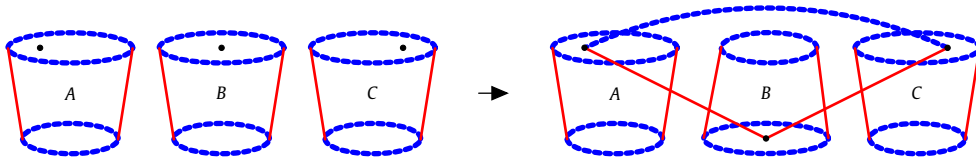


Fig. 4. A possible answer in the case 3.(a), where each  $x_i$  is in the larger class of its component.

of the edges,  $G_R$  is a bipartite graph, and  $G_B$  has at least two components. Note also that each ball in a component of  $G_B$  has the same color.

Consider a component  $D$  of  $G_R$ . It is a bipartite graph, call the two classes of vertices  $D_X$  and  $D_Y$ . The most important property of a component is how the difference between the cardinality of its two classes, thus we introduce a notation for it. Let

$$d(D) := |D_X| - |D_Y|.$$

Then we say  $d(D)$  is the *imbalance* of component  $D$ .

### Strategy of Adversary:

Now we will describe how Adversary should answer queries. First we informally give 3 conditions (we mentioned the first one earlier) that will be the main lines of the strategy of Adversary. Then we specify the strategy more.

**Condition 1:** as we have mentioned, there must be at least one balanced two-coloring consistent with the answers. Adversary will never violate this condition.

**Condition 2:** Adversary answers that two of the three balls have the same color, and the third one has a different color. (He answers according to Condition 2, if he does not violate Condition 1 with the answer.)

**Condition 3:** In the case the three balls are from three different components of  $G$ , Adversary answers in such a way that the resulting component has “small imbalance”. We will give the precise condition below. (He answers this way if he can without violating Condition 1.)

The strategy of Adversary separates two phases. During Phase 1, Adversary does not violate Conditions 1, 2 and 3. Phase 1 ends when he has to violate one of the conditions (which is Condition 2 or 3). During Phase 2, Adversary does not violate Condition 1.

Now we give Adversary's strategy in more details (in the description of the strategy  $G$  means the auxiliary graph before a specific query). Recall that a component of  $G$  is determined by edges without considering their colors.

1. If the query contains three balls from the same component of  $G$ , then the answer is determined by the two-coloring of that component of  $G$ .
2. If the query  $\{x_1, x_2, x_3\}$  contains three balls from exactly two components of  $G$ :  $A$  and  $B$ , then two of the three balls are in the same component and we can assume that  $x_1, x_2 \in A$  and  $x_3 \in B$ . There are two cases:
  - (a) First assume that  $x_1$  and  $x_2$  are in the same color class of  $A$ . In this case, Adversary answers that  $x_1$  and  $x_2$  has the same color, and  $x_3$  has a different color (if that answer does not violate Condition 1. If it does, he has to answer that the three balls have the same color).
  - (b) Now assume that  $x_1$  and  $x_2$  are in different color classes of  $A$ . Then the answer will be that  $x_1$  and  $x_2$  have different colors, and  $x_3$  has a color such that this answer does not violate Condition 1. Note that before this query, there is at least one balanced two-coloring, and  $x_1$  and  $x_2$  are in different color classes, thus there is at least one possible answer that does not violate Condition 1, nor Condition 2.

Note that if  $d(A) = a$ , and  $d(B) = b$ , then the imbalance of the new component in any of the above cases will be either  $a + b$  or  $a - b$  or  $b - a$ .

3. If the query contains three balls from three different components:  $x_1 \in A$ ,  $x_2 \in B$ ,  $x_3 \in C$ , let  $d(A) = a$ ,  $d(B) = b$ ,  $d(C) = c$ . Adversary may have multiple allowed answers, in that case he chooses an arbitrary one of them. More precisely, there is one possible answer that is forbidden by Condition 3. This is the case where Condition 3 appears, thus we give the exact description here.
  - (a) Assume that  $a \geq b \geq c > 0$ . Adversary's goal is to answer in such a way that the imbalance of the new component is not  $a + b + c$  (i.e., the imbalance is at most  $a + b - c$ ), see Fig. 4. To this end, he should not answer in such a way that the balls belonging to the larger class in their component have the same color, and the balls belonging to the smaller class in their component have the other color. Every other answer is allowed by Condition 3, if there is such an answer that does not violate Conditions 1 and 2, then Adversary picks such an answer. Otherwise, Adversary picks an arbitrary answer that does not violate Condition 1.
  - (b) Assume that one of  $a, b, c$  is 0. Then Condition 3 does not give any restriction (note that this is part of the definition of Condition 3, not an implication of any earlier statement). Therefore, Adversary picks an arbitrary answer that does not violate Conditions 1 and 2 if he can, and otherwise he picks an arbitrary answer that does not violate Condition 1.

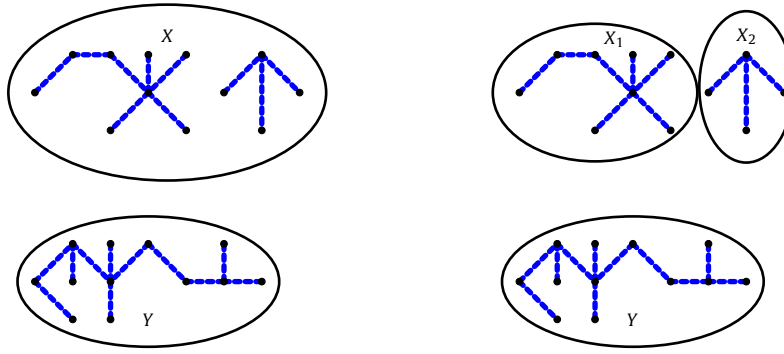


Fig. 5. Two possible coloring for the same blue edges.

**Remark 8.** We note that Adversary could easily answer without violating Condition 2 and 3, if he was allowed to violate Condition 1.

**Proposition 9.** If Questioner can solve the Plurality problem at one point, then  $G$  has at least  $n - 2$  blue edges at that point.

**Proof.** By the strategy of Adversary there is a coloring from  $\mathcal{S}$  that is consistent with the answers, so the only solution to the Plurality problem can be that there is no plurality ball. Let  $X$  and  $Y$  be the two color classes of that coloring. Assume that  $G_B$  has more than two components. Note that each component of  $G_B$  is a subset of either  $X$  or  $Y$ . Assume first without loss of generality that  $G_B$  has at least two components in  $X$ , i.e. there are no blue edges between  $X_1$  and  $X_2$ , where  $X = X_1 \cup X_2$ ,  $X_1 \cap X_2 = \emptyset$ ,  $X_1 \neq \emptyset$ ,  $X_2 \neq \emptyset$ . In this case  $X_1$ ,  $X_2$  and  $Y$  can be the three color class, and any ball from  $Y$  is a plurality ball, a contradiction (Fig. 5).

This means that if Questioner can solve the Plurality problem against Adversary's strategy, then  $G_B$  has only two components at the end, thus there are at least  $n - 2$  blue edges.  $\square$

Our next goal is to show that there are at least  $n - 4$  red edges at the end. This will imply that the sum of the weights of the edges is at least

$$\frac{1}{2}(n - 2) + \frac{1}{4}(n - 4) = \frac{3}{4}n - 2,$$

and that will prove the lower bound in Theorem 3 (i).

**Proposition 10.** Assume that Adversary has not violated Conditions 1, 2, and 3 till a certain point. Then the components of  $G$  are the same as the components of  $G_R$ .

**Proof.** The edges of  $G_R$  are edges in  $G$ , so we only have to show that there are no blue edges between two components of  $G_R$ . Assume there is at least one. Consider the first answer from which we get that kind of edge.

If this answer was that two of the balls (say  $x$  and  $y$ ) have the same color, the other (say  $z$ ) has a different color, then  $xy$  is the blue edge, but  $x$  and  $y$  are also in the same red component, because  $z$  is a common neighbor of them in  $G_R$ .

If the answer was that the three balls have the same color, but the three balls did not come from the same component of  $G_R$ , then this answer violated Condition 2.  $\square$

Recall that the two color classes of a component  $D$  of  $G_R$  are  $D_X$  and  $D_Y$ . Let us denote by  $e_R(D)$  the number of red edges in  $D$ .

**Proposition 11.** Assume that Adversary has not violated Conditions 1, 2, and 3 till a certain point. Then for every component  $D$  of  $G_R$  with  $d(D) = 0$  we have  $e_R(D) \geq |V(D)|$ .

Note that in the above case we have  $V(D) = |D_X| + |D_Y| = 2|D_X|$ .

**Proof.**  $D$  is connected, so it has at least  $|V(D)| - 1 = 2|D_X| - 1$  red edges. It is easy to see, that each component has even number of red edges, because every answer gives 0 or 2. Thus  $e_R(D)$  is even and  $e_R(D) \geq 2|D_X| - 1$ , so  $e_R(D) \geq 2|D_X| = |V(D)|$ .  $\square$

**Proposition 12.** Assume that Adversary has not violated Conditions 1, 2, and 3 till a certain point. Then for every component  $D$  of  $G_R$  with  $d(D) > 0$  we have  $e_R(D) \geq |V(D)| + d(D) - 2$ .

Note that we have  $|V(D)| + d(D) - 2 = 2 \max(|D_X|, |D_Y|) - 2$ .

**Proof.** We prove the statement by induction on  $|V(D)|$ .

**Case 1:** Assume that  $|V(D)| = 1$ . Then  $D$  is an isolated vertex, so

$$0 = e_R(D) \geq 2 \max(|D_X|, |D_Y|) - 2 = 2 \cdot 1 - 2 = 0$$

is obviously true.

Consider the last query  $Q$  that contains some vertices in  $V(D)$ .

**Case 2:** Assume that  $Q$  is inside  $D$ . Then nothing changes and we are done.

**Case 3:** Assume that  $Q$  contains three balls from two components, call them  $A$  and  $B$ . By induction we have

$$e_R(A) \geq |V(A)| + d(A) - 2,$$

and

$$e_R(B) \geq |V(B)| + d(B) - 2.$$

The answer of Adversary does not violate Condition 2 (by the assumption of the proposition), so

$$e_R(A \cup B) = e_R(A) + e_R(B) + 2.$$

Obviously we have  $d(A \cup B) \leq d(A) + d(B)$ . By all of these, we get

$$\begin{aligned} e_R(A \cup B) &= e_R(A) + e_R(B) + 2 \geq \\ &\geq |V(A)| + d(A) - 2 + |V(B)| + d(B) - 2 + 2 \geq \\ &\geq |V(A \cup B)| + d(A \cup B) - 2, \end{aligned}$$

and we are done in this case.

**Case 4:** Now assume that the query contains three balls from three components, call these components  $A$ ,  $B$  and  $C$ , and let  $d(A) = a$ ,  $d(B) = b$ ,  $d(C) = c$  with  $a \geq b \geq c$ .

**Case 4.1:**  $c \geq 1$ .

This implies  $c - 2 \geq -c$ . By induction, we have

$$\begin{aligned} e_R(A) &\geq |V(A)| + d(A) - 2, \\ e_R(B) &\geq |V(B)| + d(B) - 2, \end{aligned}$$

and

$$e_R(C) \geq |V(C)| + d(C) - 2.$$

The answer does not violate Condition 2, so

$$e_R(A \cup B \cup C) = e_R(A) + e_R(B) + e_R(C) + 2,$$

and Condition 3 provides that

$$d(A \cup B \cup C) \leq a + b - c.$$

By all of these, we get

$$\begin{aligned} e_R(A \cup B \cup C) &= e_R(A) + e_R(B) + e_R(C) + 2 \geq \\ &\geq |V(A)| + d(A) - 2 + |V(B)| + d(B) - 2 + |V(C)| + d(C) - 2 + 2 = \\ &= |V(A \cup B \cup C)| + a + b + c - 4 \geq \\ &\geq |V(A \cup B \cup C)| + (a + b - c) - 2 \geq \\ &\geq |V(A \cup B \cup C)| + d(A \cup B \cup C) - 2, \end{aligned}$$

and we are done in this case.

**Case 4.2:**  $c = 0$ .

In this case  $e_R(B) \geq |V(C)|$  also holds by a). So we have

$$\begin{aligned} e_R(A \cup B \cup C) &= e_R(A) + e_R(B) + e_R(C) + 2 \geq \\ &\geq |V(A)| + d(A) - 2 + |V(B)| + d(B) - 2 + |V(C)| + 2 = \\ &= |V(A \cup B \cup C)| + a + b - 2 = \\ &= |V(A \cup B \cup C)| + (a + b - c) - 2 \geq \\ &\geq |V(A \cup B \cup C)| + d(A \cup B \cup C) - 2, \end{aligned}$$

and we are done with this case and with the proof of Proposition 12.  $\square$

Propositions 11 and 12 immediately give the following:

**Corollary 13.** Assume that Adversary has not violated Conditions 1, 2, and 3 till a certain point. If  $d(D) \neq 1$ , then

$$e_R(D) \geq |V(D)|.$$

Now we go back to the proof of our main goal: to prove that there are at least  $n - 4$  red edges.

**Case 1:** Adversary never violates Conditions 1, 2, and 3 during the algorithm. By the proof of Proposition 9, it is easy to see that there are at most two components of  $G_B$ . If they are not connected by a red edge, then it is consistent with the answers that each ball has the same color thus there is a plurality ball, a contradiction. Therefore, there is only one component of  $G$ . Then Proposition 10 implies that there is only one component of  $G_R$ . By the assumption that  $n$  is even, we have that the imbalance of this component is even, thus we can apply Corollary 13. Therefore, this component has at least  $|V(G_R)| = n > n - 4$  red edges.

**Case 2:** Adversary has to violate Condition 2 or 3 during the algorithm. Consider the first query  $Q$  when he violates one of the conditions. From now on, we will use the notations  $G$ ,  $G_R$  and  $G_B$  for the graphs before this query. We will show that  $G_R$  already has at least  $n - 4$  edges.

**Case 2.1:** Assume that  $Q$  contains three balls from three components (note that before the query  $Q$ , the components of  $G$  and  $G_R$  are the same by Proposition 10). We call these components  $A$ ,  $B$  and  $C$ , with  $d(A) = a$ ,  $d(B) = b$ , and  $d(C) = c$ . We can assume that  $a \geq b \geq c$ . There is a balanced two-coloring  $S \in \mathcal{S}$  consistent with this answer and the previous ones. Fix it, and denote the two color classes with  $X$  and  $Y$ . For a component  $D$ , we will use the notation  $D_X = V(D) \cap X$  and  $D_Y = V(D) \cap Y$ .

The fact that Adversary has to violate Condition 2 or 3 implies the following.

Property ( $\star$ ): There are no balanced two-colorings  $S', S'' \in \mathcal{S}$  such that in  $S'$  the balls in  $A$  and  $C$  have the **same** colors as in  $S$ , and the balls in  $B$  have the **other** colors as in  $S$ , while in  $S''$  the balls in  $A$  and  $B$  have the **same** colors as in  $S$ , and the balls in  $C$  have the **other** colors as in  $S$ .

Indeed, both Conditions 2 and 3 forbid at most one coloring of  $A, B, C$ , thus three different colorings in  $\mathcal{S}$  would mean the existence of a coloring in  $\mathcal{S}$  that satisfies both Conditions 2 and 3.

We can assume that  $|B_X| \leq |B_Y|$ . Now we introduce another imbalance parameter: we give a sign for the imbalance. For a component  $D$ , the new definition of its imbalance is

$$\bar{d}(D) = |D_X| - |D_Y|.$$

That means  $\bar{d}(B) \leq 0$ . Let  $\mathcal{H}$  be the set of components of  $G$  other than  $A, B$  or  $C$  and for  $i \in \mathbb{Z}$  let

$$\mathcal{H}_i := \{D \in \mathcal{H} \mid \bar{d}(D) = i\}.$$

Several times in the rest of the proof, to obtain a contradiction, we will **reverse** the color classes of  $B$ , and some other components not including  $A$  and  $C$ . By this we mean that we exchange  $B_X$  and  $B_Y$ , and for the other components  $D$  involved, we exchange  $D_X$  and  $D_Y$ . This way we obtain another balanced two-coloring  $S'$  (see Fig. 6). Similarly we will reverse the color classes of some components including  $C$  but not  $A$  and  $B$ , in order to obtain another balanced two-coloring  $S''$ . If both of these are consistent with the answers, that violates property ( $\star$ ).

**Proposition 14.** We have  $|\mathcal{H}_1| < b$ .

**Proof.** If  $|\mathcal{H}_1| \geq b$ , then we could reverse the two color classes of  $B$  and also of  $b$  components from  $\mathcal{H}_1$ . Similarly we could reverse the two color classes of  $C$  and also of  $c$  components from  $\mathcal{H}_1$  (Fig. 7). The resulting coloring contradicts property ( $\star$ ).  $\square$

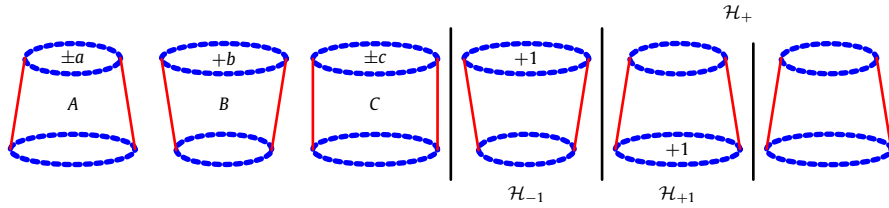


Fig. 6. The most important components,  $A$ ,  $B$ ,  $C$ , the components with imbalance 1 and  $\mathcal{H}_+$ .

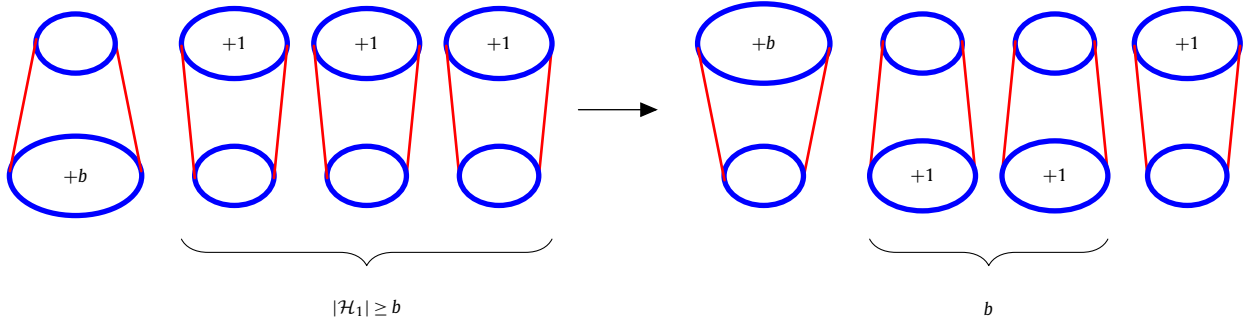


Fig. 7. Proof of Proposition 14.

Using that  $|\mathcal{H}_1|$  is small, we can easily deal with the case  $|\mathcal{H}_{-1}|$  is also small.

**Case 2.1.1:**  $|\mathcal{H}_{-1}| \leq a + c - 1$ .

By Propositions 11 and 12 and by Corollary 13, there are at least

$$n - |\mathcal{H}_1| - |\mathcal{H}_{-1}| + (a - 2) + (b - 2) + (c - 2)$$

red edges. Using  $|\mathcal{H}_1| + 1 \leq b$  (Proposition 14) and  $|\mathcal{H}_{-1}| + 1 \leq a + c$ , we get that there are at least

$$n + (b - |\mathcal{H}_1|) + (a + c - |\mathcal{H}_{-1}|) - 6 \geq n + 1 + 1 - 6 = n - 4$$

red edges.

**Case 2.1.2:**  $|\mathcal{H}_{-1}| \geq a + c$

In this case we want to reverse color classes with positive but not too large imbalance. Then we can use the (many) components of  $|\mathcal{H}_{-1}|$  to obtain the balance. This can be easily done if the total imbalance of such components is large. Let

$$\mathcal{H}_+ := \{D \in \mathcal{H} \mid 0 < \bar{d}(D) \leq |\mathcal{H}_{-1}| + 1\}.$$

**Proposition 15.** *We have*

$$\sum_{D \in \mathcal{H}_+} \bar{d}(D) < b.$$

**Proof.** Assume by contradiction that  $\sum_{D \in \mathcal{H}_+} \bar{d}(D) \geq b$ . Then we first reverse the two color classes of  $B$ . This makes the sum of the imbalances  $-2b$ . To compensate this, we can greedily reverse the color classes of some components  $\mathcal{K} \subseteq \mathcal{H}_+$  such that

$$b \leq \sum_{D \in \mathcal{K}} \bar{d}(D) \leq b + |\mathcal{H}_{-1}|.$$

After this, the total imbalance is  $2k$ , for some  $k$  with  $-2|\mathcal{H}_{-1}| \leq 2k \leq 0$ . And now we can again compensate this by changing the color classes of  $k$  components from  $\mathcal{H}_{-1}$ . We got a balanced two-coloring  $S' \in \mathcal{S}$ . Similarly, as  $\sum_{D \in \mathcal{H}_+} \bar{d}(D) \geq c$  also holds, we could reverse  $C$  and some components of  $\mathcal{H}_+$  to obtain another balanced two-coloring  $S''$ , giving the contradiction with property  $(\star)$ .  $\square$

**Case 2.1.2.1:** there is a component  $D \in \mathcal{H}$  with

$$\bar{d}(D) > |\mathcal{H}_{-1}| + 1.$$

The above assumption gives an upper bound on  $|\mathcal{H}_{-1}|$ , which we can use as earlier. More precisely, by Propositions 11 and 12 and by Corollary 13, there are at least

$$n - |\mathcal{H}_1| - |\mathcal{H}_{-1}| + (a - 2) + (b - 2) + (c - 2) + (\bar{d}(D) - 2)$$

red edges. Using that  $|\mathcal{H}_1| + 1 \leq b$  and  $|\mathcal{H}_{-1}| + 2 \leq \bar{d}(D)$ , we have at least

$$n + (b - |\mathcal{H}_1|) + (\bar{d}(D) - |\mathcal{H}_{-1}|) - 8 + a + c \geq n - 5 + a + c \geq n - 4$$

red edges.

**Case 2.1.2.2:** there is no component  $D \in \mathcal{H}$  with

$$\bar{d}(D) > |\mathcal{H}_{-1}| + 1.$$

We remark that this means that components with positive imbalance are in  $\mathcal{H}_+$  besides  $A$  and  $C$ . This means their total imbalance is small. We can use this property the following way.  $S$  is a balanced two-coloring, so we have

$$\sum_{D: \bar{d}(D) > 0} \bar{d}(D) + \sum_{D: \bar{d}(D) < 0} \bar{d}(D) = 0.$$

By the assumption we have  $\{D \mid \bar{d}(D) > 0\} \subseteq \mathcal{H}_+ \cup \{A, C\}$ , so

$$\sum_{D: \bar{d}(D) > 0} \bar{d}(D) \leq \sum_{D \in \mathcal{H}_+} \bar{d}(D) + a + c \leq b - 1 + a + c.$$

On the other hand we have  $\mathcal{H}_{-1} \cup \{B\} \subseteq \{D \mid \bar{d}(D) < 0\}$ , so

$$\sum_{D: \bar{d}(D) < 0} \bar{d}(D) \leq -|\mathcal{H}_{-1}| - b \leq -a - c - b.$$

That gives us

$$0 = \sum_{D: \bar{d}(D) > 0} \bar{d}(D) + \sum_{D: \bar{d}(D) < 0} \bar{d}(D) \leq b - 1 + a + c - a - c - b = -1,$$

a contradiction.

**Case 2.2:** Assume that  $Q$  (the first query violating Condition 2 or 3) contains three balls from two components. We call these components  $A$  and  $B$  and assume that  $d(A) \geq d(B)$ .

The proof in this case is analogous to the case with three components. The underlying ideas are the same in each case, but there are some small differences in the details. We fix the balanced two-coloring  $S \in \mathcal{S}$  with color classes  $X$  and  $Y$ . Condition 3 does not give any restriction in this case, thus we can strengthen property  $(\star)$  the following way.

Property  $(\star\star)$ : There is no balanced two-coloring  $S' \in \mathcal{S}$  such that in  $S'$  the balls in  $A$  and  $C$  have the **same** colors as in  $S$ , and the balls in  $B$  have the **other** colors as in  $S$ .

Indeed, either  $S$  or  $S'$  does not violate Condition 2 and neither of them violates Conditions 1 and 3, thus Adversary would not need to end Phase 1.

We define the signed imbalance parameter the same way as in Case 2.1, that means  $\bar{d}(B) \leq 0$ . Let now  $\mathcal{H}$  be the set of components of  $G$  other than  $A$  and  $B$  and for  $i \in \mathbb{Z}$  let

$$\mathcal{H}_i := \{D \in \mathcal{H} \mid \bar{d}(D) = i\}.$$

**Proposition 16.** *We have*

$$|\mathcal{H}_1| < b.$$

**Proof.** Same as the proof of Proposition 14, if  $|\mathcal{H}_1| \geq b$ , then we could reverse the two color classes of  $B$  and also of  $b$  components from  $\mathcal{H}_1$ . The resulting coloring contradicts property  $(\star\star)$ .  $\square$

**Case 2.2.1:**  $|\mathcal{H}_{-1}| \leq a - 1$

By Propositions 11 and 12 and by Corollary 13, there are at least

$$n - |\mathcal{H}_1| - |\mathcal{H}_{-1}| + (a - 2) + (b - 2)$$

red edges. Using  $|\mathcal{H}_1| + 1 \leq b$  (Proposition 16) and  $|\mathcal{H}_{-1}| + 1 \leq a$ , we get that there are at least

$$n + (b - |\mathcal{H}_1|) + (a + c - |\mathcal{H}_{-1}|) - 4 \geq n + 1 + 1 - 4 = n - 2$$

red edges.

**Case 2.2.2:**  $|\mathcal{H}_{-1}| \geq a$ .

Now let

$$\mathcal{H}_+ := \{D \in \mathcal{H} \mid 0 < \bar{d}(D) \leq |\mathcal{H}_{-1}| + 1\}.$$

**Proposition 17.** *We have*

$$\sum_{D \in \mathcal{H}_+} \bar{d}(D) < b.$$

**Proof.** The proof is the same as the proof of Proposition 15.  $\square$

**Case 2.2.2.1:** there is a component  $D \in \mathcal{H}$  with  $\bar{d}(D) > |\mathcal{H}_{-1}| + 1$ .

By Propositions 11 and 12 and by Corollary 13, there are at least

$$n - |\mathcal{H}_1| - |\mathcal{H}_{-1}| + (a - 2) + (b - 2) + (\bar{d}(D) - 2)$$

red edges. Using  $|\mathcal{H}_1| + 1 \leq b$  and  $|\mathcal{H}_{-1}| + 2 \leq \bar{d}(D)$ , we get that there are at least

$$n + (b - |\mathcal{H}_1|) + (\bar{d}(D) - |\mathcal{H}_{-1}|) - 6 + a \geq n - 3 + a \geq n - 2$$

red edges.

**Case 2.2.2.2:** there is no component  $D \in \mathcal{H}$  with  $\bar{d}(D) > |\mathcal{H}_{-1}| + 1$ .

We have a balanced two-coloring, so

$$\sum_{D: \bar{d}(D) > 0} \bar{d}(D) + \sum_{D: \bar{d}(D) < 0} \bar{d}(D) = 0.$$

By the assumption  $\{D \mid \bar{d}(D) > 0\} \subseteq \mathcal{H}_+ \cup \{A\}$  we have

$$\sum_{D: \bar{d}(D) > 0} \bar{d}(D) \leq \sum_{D \in \mathcal{H}_+} \bar{d}(D) + a \leq b - 1 + a.$$

On the other hand,  $\mathcal{H}_{-1} \cup \{B\} \subseteq \{D \mid \bar{d}(D) < 0\}$ , therefore

$$\sum_{D: \bar{d}(D) < 0} \bar{d}(D) \leq -|\mathcal{H}_{-1}| - b \leq -a - b.$$

That gives us

$$0 = \sum_{D: \bar{d}(D) > 0} \bar{d}(D) + \sum_{D: \bar{d}(D) < 0} \bar{d}(D) \leq b - 1 + a - a - b = -1,$$

a contradiction.

We are done with the proof of the lower bound of Theorem 3 (i).  $\square$

### 5.2. Proof of the lower bound of Theorem 3 (ii)

Now we sketch the proof of Theorem 3 (ii) that is similar to the one of (i). These proofs are connected a similar way to the connection of the proof of the even and odd case in Theorem 1. The problem again lies in the fact that there is no exactly balanced two partition of odd many balls. In the following sketch of proof, we assume familiarity with the proof of Theorem 3 (i).

Now we divide Adversary's strategy into two phases. In Phase 1 he can't violate Conditions 2-4 (see later) and wants to produce at least  $n - O(\log_2 n)$  red edges in the auxiliary graph (defined as in the even case).

**Phase 1:** Before defining the strategy of Adversary we make some initial comments.

During Phase 1 the auxiliary graph will be the same as in the case of even  $n$ . We will call *components* the components of  $G_R$ , but we will see that an analogue of Proposition 10 is true, so the components of  $G_R$  are the same as the components of  $G$ . The imbalance function  $d$  will also be the same.

We will call a component  $D$  a *deficient component*, if it has exactly  $|V(D)| - 1$  red edges. Note that if a component is not deficient, then it has more than  $|V(D)| - 1$  red edges.

We will call a ball  $x$  of *potential third color* (or a *p3c ball* for short) if the blue degree of  $x$  is zero and  $x$  is contained in a deficient component. The name will be explained shortly. Observe that an isolated vertex is a deficient component and it is p3c.

During Phase 1 Adversary will have four conditions on his strategy:

**Condition 1:** there is a 3-coloring of the balls with color class sizes  $\frac{n-1}{2}$ ,  $\frac{n-1}{2}$ , 1, consistent with the answers.

**Condition 2**, and **Condition 3** will be the same as in the case of even  $n$ .

**Condition 4:** there is an almost balanced two-coloring of the balls that is consistent with the answers and the largest color class contains a ball of potential third color.

Note that Condition 4 implies Condition 1. Indeed, if there is an almost balanced two-coloring with color classes  $X'$  and  $Y'$  ( $|X'| > |Y'|$ ), which is consistent with the previous answers, and there is a p3c vertex  $z$  in the larger color class, then the three-coloring  $X = X' \setminus \{z\}$ ,  $Y = Y'$ ,  $Z = \{z\}$  is consistent with Condition 1 and the previous answers. This explains why  $z$  is called a ball of potential third color. Note that in this case, i.e., as long as Condition 4 is not violated, the algorithm has not been finished, Questioner cannot present a solution.

Now we define the answers of Adversary during Phase 1.

The answers will be the same as in the case of even  $n$ , except that there is a little restriction in the case when the query contains three p3c balls from three different deficient components, which we will describe below.

But first note that it is easy to see that a deficient component  $D$  with more than one vertex must have imbalance  $d(D) = 1$ . Moreover,  $D$  was made from three different deficient components, i.e. the last red edges were added to  $D$  after a query containing balls from three different deficient components and  $D$  is the union of those three components. Indeed, any other way we get a component with more red edges.

Now we define the answer of Adversary to a query in Phase 1 containing three p3c balls from three different deficient components. Recall that each component has two sides, with red edges between them.

- If all the three p3c balls were on the larger side of their components, then Adversary answers that the one with the smallest red-degree has different color, and the other two have the same color.
- If all the three p3c balls were on the smaller side of their components, then Adversary answers that the one with the smallest red-degree has different color, and the other two have the same color.
- If exactly two of the three p3c balls were on the larger side of their components, then Adversary answers that the one (of those two) with the smaller red-degree has different color, and the other has the same color as the third ball.
- If exactly two of the three p3c balls were on the smaller side of their components, then Adversary answers that the one (of those two) with the smaller red-degree has different color, and the other has the same color as the third ball.

It is not hard to see that if there is an answer that does not violate Conditions 2, 3 and 4, then the above defined answer also does not violate those conditions. Recall that the three components each have imbalance 1, thus the only essential difference between them in a 2-coloring is whether the balls asked in the query are on the larger or smaller side of the component. More precisely, let  $a$ ,  $b$  and  $c$  be the three balls asked. If  $a$  is on the larger side and  $b$  is on the smaller side of their components, then reversing both their components gives a coloring with the same size of color classes.

By Condition 2, the answer is that two balls have the same color and the third has a different color. In the first two cases, we can pick the one with different color to be the one with the smallest red-degree, without changing the imbalance of the new component. Therefore, the resulting coloring satisfies Conditions 3 and 4 (thus also Condition 1). The other two cases are only slightly more complicated. If  $a$  and  $b$  are both on the larger side,  $a$  has smaller red-degree than  $b$  and  $c$  is on the smaller side, it is possible that Conditions 2, 3 and 4 are satisfied by the answer that  $a$  and  $b$  have the same color and  $c$  has a different color. This is different from the answer we just defined. However, we can reverse now the components of  $b$  and  $c$  to obtain another coloring that satisfies Condition 3 and 4 (thus also Condition 1) with an answer that satisfies Condition 2. The last case is analogous to the above one.

The following propositions are similar to Proposition 10, 11, 12 and Corollary 13, and the proofs are the same.

**Proposition 18.** Assume that Adversary has not violated Conditions 1, 2, 3, and 4 till a certain point. Then the components of  $G$  are the same as the components of  $G_R$ .

**Proof.** Similar to the proof of Proposition 10.  $\square$

**Proposition 19.** Assume that Adversary has not violated Conditions 1, 2, 3 and 4 till a certain point. Then for every component  $D$  of  $G_R$  with  $d(D) = 0$  we have  $e_R(D) \geq |V(D)|$ .

Note that in the above case we have  $V(D) = |D_X| + |D_Y| = 2|D_X|$ .

**Proof.** Similar to the proof of Proposition 11.  $\square$

**Proposition 20.** Assume that Adversary has not violated Conditions 1, 2, 3 and 4 till a certain point. Then for every component  $D$  of  $G_R$  with  $d(D) > 0$  we have  $e_R(D) \geq |V(D)| + d(D) - 2$ .

**Proof.** Similar to the proof of Proposition 12.  $\square$

Propositions 19 and 20 immediately give the following:

**Corollary 21.** Assume that Adversary has not violated Conditions 1, 2, 3, and 4 till a certain point. If  $d(D) \neq 1$ , then

$$e_R(D) \geq |V(D)|.$$

**Lemma 22.** Assume that Adversary has not violated Conditions 1, 2, 3, and 4 till a certain point. Then the following two statements are true for every deficient component  $D$ .

- a) There exists a p3c vertex in  $D$ .
- b) There exists a p3c vertex in  $D$  with red-degree at most  $2 \log_2(|V(D)|)$ .

**Proof.** a) There are  $|V(D)|$  vertices in  $D$ , so there are  $|V(D)| - 1$  red edges. Thus there were  $\frac{|V(D)|-1}{2}$  queries such that red edges were added to  $D$  with the answer to those queries. Hence Condition 2 provides that there are  $\frac{|V(D)|-1}{2}$  blue edges in  $D$ . That means there must be at least one vertex that has blue-degree 0, so that vertex is a p3c vertex.

b) We will prove the statement by induction on  $|D|$ . For an isolated vertex it is obviously true. Let us assume the statement holds for every component of size less than  $|D|$  and prove it for  $D$ . Consider the query  $Q$  such that the red edges added after  $Q$  are the last ones added to  $D$ . As we have already observed, the three balls  $x_1, x_2, x_3$  in  $Q$  must be from three different components,  $x_i \in D_i$  for  $i \leq 3$ . If any of these balls, say  $x_1$  was not p3c before querying  $Q$ , then there was a p3c ball  $x'$  in  $D_1$ , with red degree at most  $2 \log_2(|V(D_1)|) \leq 2 \log_2(|V(D)|)$ . Its red degree does not change, finishing the proof in this case.

If  $x_1, x_2, x_3$  are all p3c, then Adversary chooses the ball with the different color (the one that will get two red edges) from at least two balls, say  $x_1$  and  $x_2$ . By induction, their red-degree is at most  $2 \log_2(|V(D_1)|)$  and  $2 \log_2(|V(D_2)|)$ . Adversary chooses the one with the smaller degree, so it will be a p3c ball, and its red degree is at most

$$\begin{aligned} & \min(2 \log_2(|V(D_1)|), 2 \log_2(|V(D_2)|)) + 2 = 2(\log_2(\min(|V(D_1)|, |V(D_2)|)) + 1) = \\ & = 2 \log_2(2 \min(|V(D_1)|, |V(D_2)|)) \leq 2 \log_2(|V(D_1)| + |V(D_2)|) \leq \\ & \leq 2 \log_2(|V(D_1)| + |V(D_2)| + |V(D_3)|) = 2 \log_2(|V(D)|). \quad \square \end{aligned}$$

In a coloring provided by Condition 1, there is no plurality ball. In a coloring provided by Condition 4, there are plurality balls. So if the Questioner can solve the Plurality problem, then Adversary must violate some Condition at some point, however it cannot be Condition 1. Now we will consider the first answer, where the Adversary had to violate Condition 2, 3, or 4.

**Lemma 23.** Assume that Adversary has not violated Conditions 1, 2, 3, and 4 till a certain point, but he has to violate Condition 2 or 3 or 4 with his next answer. Then we already have at least  $n - 10$  red edges in  $G$ .

**Sketch of the proof.** The proof is very similar to the proofs of **Case 2.1** and **Case 2.2** in the proof of the even case.

We just highlight some differences that create only some modification of the constants:

- in the analogues of Proposition 14 and Proposition 16, we need  $|\mathcal{H}_1| < b + 1$ , because we need one p3c vertex in the larger class after the color-changing. Similarly  $b$  is replaced by  $b + 1$  in the analogues of Propositions 15 and 17.
- property  $(\star)$  is modified, we forbid only those balanced two-colorings  $S'$  and  $S''$  which satisfy Condition 4.

- We change the definition of  $\mathcal{H}_+$ , it is now  $\{D \in \mathcal{H} | 0 < \bar{d}(D) \leq |\mathcal{H}_{-1}|\}$ ,
- at the end, instead of

$$0 = \sum_{D: \bar{d}(D) > 0} \bar{d}(D) + \sum_{D: \bar{d}(D) < 0} \bar{d}(D),$$

we need to use

$$-1 \leq \sum_{D: \bar{d}(D) > 0} \bar{d}(D) + \sum_{D: \bar{d}(D) < 0} \bar{d}(D). \quad \square$$

Let us recall that  $Q$  is the first query, for which the answer violates Condition 2, 3, or 4. This is the end of Phase 1. Before  $Q$ , there was at least one almost balanced two-coloring with at least one p3c ball in the larger color class. Fix such a two-coloring ( $[n] = X' \cup Y'$ ,  $|X'| = \frac{n+1}{2}$ ,  $|Y'| = \frac{n-1}{2}$ ), and choose a p3c ball  $z \in X'$  with red-degree at most  $2 \log_2(n)$ . Such a ball exists according to Lemma 22 b) and because of  $|V(D)| \leq n$ .

As we have noted, the three-coloring  $X = X' \setminus \{z\}$ ,  $Y = Y'$ ,  $Z = \{z\}$  is consistent with the previous answers and Condition 1. Now we fix this three-coloring, and Adversary will answer according to it, starting with the answer to  $Q$ .

We also make some changes in the auxiliary graph. Note that the red edges containing  $z$  are in pairs because they come from the queries involving  $z$ . Let us denote them by  $e_{1,1}, e_{1,2}, e_{2,1}, e_{2,2}, \dots, e_{l,1}, e_{l,2}$ , where  $2l$  is the red-degree of  $z$ . Note that we have  $l \leq \log_2(n)$ .

Now we define a new color of the edges. Green edge simply means that it comes from  $z$ . Basically, they are special red edges.

We change the colors of  $e_{1,1}, e_{2,1}, \dots, e_{l,1}$  from red to green, and we delete the edges  $e_{1,2}, e_{2,2}, \dots, e_{l,2}$  from the graph. By Lemma 23, there were at least  $n - 10$  red edges before  $Q$ , so now we have at least  $n - 10 - 2l \geq n - 10 - 2 \log_2(n)$  red edges.

**Phase 2:** After  $Q$ , Adversary answers according to the previously fixed coloring  $X, Y, Z$ .

We will have some new rules in the auxiliary graph in Phase 2. If the Questioner asks the query  $\{u, v, z\}$ , where  $u$  and  $v$  have the same color, then Adversary adds the edge  $uv$  and colors it blue, and adds one of  $uz$  and  $vz$ , colored green. (That is also how we changed the auxiliary graph at the end of Phase 1.) If Questioner asks the query  $\{u, v, z\}$ , where  $u$  and  $v$  have different colors, he adds  $uz$  and  $vz$  and colors both edges green, but does not add the edge  $uv$ . If Questioner asks a query disjoint from  $z$ , the rules do not change. We give the green edges weight  $\frac{1}{2}$ , so every answer has a total weight of 1.

We proved that at the end of Phase 1, there were at least  $n - 2 \log_2 n - 10$  red edges, which did not decrease in Phase 2. Observe that we can apply Lemma 6 also in this setting, thus at the end of Phase 2 there are altogether at least  $n - 5$  blue and green edges. That means the number of queries is at least  $\frac{1}{4}(n - 2 \log_2(n) - 10) + \frac{1}{2}(n - 5) = \frac{3}{4}n - \frac{1}{2} \log_2(n) - 5$ , which proves Theorem 3 (ii).  $\square$

## 6. Conclusions and further directions

Our Theorem 3 (i) gives an exact result for  $n = 4k + 2$ , and an almost exact result (it can be one of two values) for  $n = 4k$ , while Theorem 3 (ii) gives an asymptotic result for  $n$  odd. Of course, it would be very interesting to prove an exact result for  $n = 4k, 4k + 1, 4k + 3$ . Theorem 1 provides bounds for  $k \geq 4$ . It would be nice to prove some asymptotics or even sharper results for  $A_p(n, k)$  for  $k \geq 4$ .

Our model is just one of the possible models with three colors. One can also imagine other answers, or the model can be non-adaptive, when all the queries are asked in advance, rather than sequentially, after the answer to the previous query.

In this article we investigated a model with three colors, however all Plurality problems can be asked with more colors. We note that the more colors one has, the more models can be imagined, as there can be more possible answers.

If we have two colors, then - as we mentioned in the introduction - we get the so called Majority problem. It would be interesting to apply to those problems (some variant of) the techniques we introduced here.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Data availability

No data was used for the research described in the article.

## Acknowledgement

We would like to thank all the participants of the Combinatorial Search Seminar at the Alfréd Rényi Institute of Mathematics for fruitful discussions. We would like to thank the reviewer for the insightful comments and for pointing out a mistake in the previous version of Proposition 11.

Research of Gerbner was supported by the János Bolyai Research Fellowship of the Hungarian Academy of Sciences and by the National Research, Development and Innovation Office – NKFIH, grants K 116769, KH 130371, SNN 129364, FK 132060, and KKP-133819.

Research of Vizer was supported by the Hungarian National Research, Development and Innovation Office – NKFIH under the grant SNN 129364, KH 130371 and FK 132060, by the János Bolyai Research Fellowship of the Hungarian Academy of Sciences and by the New National Excellence Program under the grant number ÚNKP-20-5-BME-45.

Research of Gerbner and Vizer was supported by the Taiwanese-Hungarian Mobility Program of the Hungarian Academy of Sciences, by Ministry of Science and Technology Project-based Personnel Exchange Program 107 -2911-I-005 -505.

## References

- [1] M. Aigner, Variants of the majority problem, *Discrete Appl. Math.* 137 (1) (2004) 3–25.
- [2] M. Aigner, Two colors and more, in: *Entropy, Search, Complexity*, Springer, Berlin, Heidelberg, 2007, pp. 9–26.
- [3] M. Aigner, G. De Marco, M. Montanero, The plurality problem with three colors and more, *Theor. Comput. Sci.* 337 (1–3) (2005) 319–330.
- [4] L. Alonso, E.M. Reingold, Determining plurality, *ACM Trans. Algorithms* 4 (3) (2008) 26.
- [5] L. Alonso, E.M. Reingold, Average-case analysis of some plurality algorithms, *ACM Trans. Algorithms* 5 (2) (2009) 17.
- [6] A.M. Borzyszkowski, Computing majority via multiple queries, *Theor. Comput. Sci.* 539 (2014) 106–111.
- [7] H. Chang, D. Gerbner, B. Patkós, Finding non-minority balls with majority and plurality queries, *Discrete Appl. Math.* 284 (30) (2020) 631–639.
- [8] F. Chung, R. Graham, J. Mao, A. Yao, Oblivious and adaptive strategies for the majority and plurality problems, *Algorithmica* 48 (2) (2007) 147–157.
- [9] G. De Marco, E. Kranakis, Searching for majority with  $k$ -tuple queries, *Discrete Math. Algorithms Appl.* 7 (02) (2015) 1550009.
- [10] G. De Marco, E. Kranakis, G. Wiener, Computing majority with triple queries, *Theor. Comput. Sci.* 461 (2012) 17–26.
- [11] Z. Dvořák, V. Jelínek, D. Král, J. Kynčl, M. Saks, Probabilistic strategies for the partition and plurality problems, *Random Struct. Algorithms* 30 (1–2) (2007) 63–77.
- [12] D. Gerbner, M. Vizer, Majority problems of large query size, *Discrete Appl. Math.* 254 (2019) 124–134.
- [13] D. Gerbner, G.O.H. Katona, D. Pálvölgyi, B. Patkós, Majority and plurality problems, *Discrete Appl. Math.* 161 (6) (2013) 813–818.
- [14] D. Gerbner, B. Keszegh, D. Pálvölgyi, B. Patkós, M. Vizer, G. Wiener, Finding a non-minority ball with majority answers, *Discrete Appl. Math.* 219 (2017) 18–31.
- [15] J. Moore, Proposed problem 81-5, *J. Algorithms* 2 (1981) 208–210.
- [16] M.E. Saks, M. Werman, On computing majority by comparisons, *Combinatorica* 11 (4) (1991) 383–387.
- [17] J. Sgall, T. Tichý, Randomized strategies for the plurality problem, *Discrete Appl. Math.* 156 (17) (2008) 3305–3311.
- [18] G. Wiener, Search for a majority element, *J. Stat. Plan. Inference* 100 (2) (2002) 313–318.