The plurality problem with three colors and more / M., Aigner; G., De Marco; Montangero, Manuela. - In: THEORETICAL COMPUTER SCIENCE. - ISSN 0304-3975. - STAMPA. - 337:1-3(2005), pp. 319-330. [10.1016/j.tcs.2004.12.035]

(Article begins on next page)

The Plurality Problem with Three Colors and More [1,2]

Martin Aigner [a] Gianluca De Marco [b,c,*]
Manuela Montangero [b,d]

[a] *Freie Universität Berlin, Institut für Mathematik II, Arnimallee 3, 14195 Berlin, Germany.*

[b] *Istituto di Informatica e Telematica, Consiglio Nazionale delle Ricerche, via Moruzzi 1, 56124 Pisa, Italy.*

[c] *Dipartimento di Informatica ed Applicazioni, Università di Salerno, via S. Allende, 84081 Baronissi (SA), Italy.*

[d] *Facoltà di Scienze della Comunicazione e dell'Economia, Università di Modena e Reggio Emilia, 42100 Reggio Emilia, Italy.*

Abstract

The *plurality problem* is a game between two participants: Paul and Carole. We are given n balls, each of them is colored with one out of c colors. At any step of the game, Paul chooses two balls and asks whether they are of the same color, whereupon Carole answers yes or no. The game ends when Paul either produces a ball a of the plurality color (meaning that the number of balls colored like a exceeds those of the other colors), or when Paul states that there is no plurality. How many questions $L_c(n)$ does Paul have to ask in the worst case?

For $c = 2$, the problem is equivalent to the well known *majority problem* which has already been solved [11]. In this paper we show that $3\lfloor n/2\rfloor - 2 \leq L_3(n) \leq \lfloor 5n/3\rfloor - 2$. Moreover, for any $c \leq n$, we show that surprisingly the naive algorithm for the plurality problem is asymptotically optimal.

Key words: Combinatorial Search, Algorithm Analysis, Majority Problem, Game.

* Corresponding author.

Email addresses: aigner@math.fu-berlin.de (Martin Aigner), gianluca.demarco@iit.cnr.it (Gianluca De Marco), manuela.montangero@iit.cnr.it (Manuela Montangero).

[1] Work supported in part by the European RTN Project under contract HPRN-CT-2002-00278,COMBSTRU.

[2] A preliminary version of this paper appeared in Proc. of STACS 2004 [4].

1 Introduction

The plurality problem can be stated as a game between two players: Paul and Carole. There are n balls, each of them colored with one out of c colors. The *plurality color* is the color that has been used the most, *i.e.*, such that the balls colored with it strictly outnumber the balls of any other color. A *plurality ball* is any ball colored with the plurality color (see Figure 1.a). Notice that a plurality color (and ball) not always exists (see Figure 1.b).

At any step of the game, Paul chooses two balls and asks whether they are of the same color, whereupon Carole answers yes or no. The game ends when Paul either produces a ball of the plurality color, or when Paul states that there is no plurality. How many questions $L_c(n)$ does Paul have to ask in the worst case?

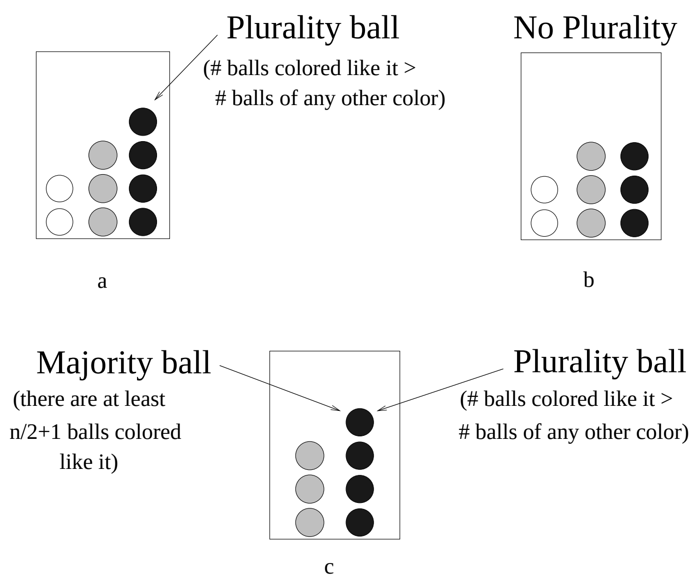

Fig. 1.

This problem is a generalization of the well known *majority problem* in which we are given n balls and two colors, *e.g.*, white and black. The aim is to produce a ball of the *majority color* (meaning that the number of balls with that color is strictly greater than that of the other color), or to state that there is no majority (this happens when there is the same number of white and black balls). The *majority problem* asks to determine how many questions Paul needs in the worst case. It is straightforward to observe that the plurality problem with two colors is equivalent to the majority problem (see Figure 1.c).

This kind of problems finds several interesting applications in the field of fault diagnosis of multiprocessor systems introduced in [10].

The majority problem was first solved by Saks and Werman [11], later Alonso, Reingold, Schott [5] gave a different proof. The elegant combinatorial result is that $L_2(n) = n - \nu(n)$ questions are necessary and sufficient in the worst case, where $\nu(n)$ denotes the number of 1's in the binary representation of n. Alonso, Reingold, Schott [6] also gave the solution for the average case.

Aigner [2,3] introduced several variants and generalizations of the majority problem. In particular, in the (n, k)-majority game Paul must exhibit a k-majority ball z (that is, there are at least k balls colored like z), or declare there is no k-majority. De Marco and Pelc [8] considered randomized solutions for the majority problem in the more general case when the balls correspond to the nodes of an undirected graph and the comparisons can only be made between adjacent nodes (of course, the problem reduces to the original majority problem on the complete graph). Fisher and Salzberg [9] studied the majority problem when the number of colors is any integer up to n. In this case the majority color is the color such that there are at least $n/2+1$ balls of that color. Observe that if a majority ball exists, then this is also a plurality ball; while a plurality ball might exist when there is no majority ball. They solved the problem by showing that $\lceil 3n/2 \rceil - 2$ comparisons are sufficient and necessary.

As for the plurality problem, it seems to be surprisingly difficult: while it is mentioned in the 1997 Alonso et al. paper, no results were known, even for the case of 3 colors. Therefore, prior to the present work, nothing substantial was known for $3 \leq c \leq n-1$. It is easy to see that $L_n(n) = \binom{n}{2}$. Indeed, $\binom{n}{2}$ is the total number of distinct comparisons which, of course, allows to determine the number of balls for each color. On the other hand, if a comparison is omitted, and hence two balls a and b are never compared, when Carole always gives "no" answers, Paul is not able to distinguish the case when there are exactly n balls colored with n distinct colors (no plurality) from the case when $n-2$ balls are colored with $n-2$ distinct colors and a and b are colored with the $(n-1)$-th color (a and b are plurality balls). Hence, the current knowledge about the problem can be depicted as follows

$$n - \nu(n) = L_2(n) \leq L_3(n) \leq \ldots \leq L_n(n) = \binom{n}{2},$$

where the inequalities are obvious.

In Section 2 we give some notation and present a naive algorithm that basically makes all the possible comparisons avoiding redundancy. The algorithm uses $O(c \cdot n)$ comparisons in the worst case. In Section 3, we consider the plurality problem with 3 colors. We exhibit an algorithm that solves the problem using $\lfloor 5n/3 \rfloor - 2$ comparisons in the worst case. On the other hand, we show that any algorithm that correctly determines the plurality must use at least $3\lfloor n/2 \rfloor - 2$ comparisons. Note that it was not previously known that $n + O(1)$ comparisons would not suffice. Finally, in Section 4 we show the surprising result that the naive algorithm is asymptotically optimal, by giving an $\Omega(c \cdot n)$ lower bound on the number of comparisons, for any $2 \leq c \leq n$.

2 Preliminaries

Throughout the paper, we use the following notation. A comparison between two balls a and b, is denoted $a : b$. The outcome of a comparison (the answer given by Carole) might be YES or NO. We say that Paul wins when the game ends and he gives the correct solution. A *color class* is a set of balls having the same color.

Let us start with a naive algorithm for the plurality problem for c colors, where $2 \leq c \leq n$. The algorithm uses $O(c \cdot n)$ comparisons. Surprisingly, in Section 4 we will show that this asymptotic bound cannot be improved. The algorithm is very simple: Paul makes all the possible comparisons, avoiding redundancy.

Namely, the algorithm consists of a sequence of at most $c - 1$ steps. At any step i, $1 \leq i < c$, a new ball is handled and the correspondent color class C_i is determined.

Initialization. Set S be the set of balls and $C_0 = \emptyset$.
Step i (for $1 \leq i < c$). Let $R = S \setminus (C_0 \cup \cdots \cup C_{i-1})$. Paul handles any ball $b \in R$, if it exists. In this step, the following comparisons are made: $b : b'$, for all $b' \in R \setminus \{b\}$. Set $C_i = \{a : a \text{ and } b \text{ have the same color}\}$.

At the end of any step a new color class is determined. Therefore, at the end, Paul knows all the color classes, and hence he can give the correct solution. In order to count the total number of comparisons, it is sufficient to observe that there are at most $c - 1$ steps and that during the i-th step at most $n - i$ comparisons are made. Therefore, the algorithm uses at most $\sum_{i=1}^{c-1}(n - i) = O(c \cdot n)$ comparisons.

3 Three colors

In this section we consider the plurality problem with 3 colors. We show that Paul has a strategy that uses no more than $5n/3 - 2$ comparisons to solve the problem. On the other hand, we prove that any algorithm that correctly determines the plurality must use at least $3\lfloor n/2 \rfloor - 2$ comparisons.

3.1 The upper bound

Theorem 3.1 *We have* $L_3(n) \leq \frac{5}{3}n - 2$, *for* $n \geq 2$.

PROOF. The proof is by induction on n. This is clear for $n \leq 3$, so let us assume $n \geq 4$. Paul arranges the balls $b_1, \ldots, b_n$ and compares them one by one according to Phase I.

Phase I. The phase consists of a sequence of states. Every state S_i (after b_i has been handled) is inductively described by a vector (k_i, ℓ_i, m_i), where $k_i \geq \ell_i \geq m_i$ are the color classes cardinalities. For $i \geq 1$, let $r_i = n - i$ be the number of the remaining balls (those that have not been involved in any comparison yet) and set $t_i = r_i - (k_i - \ell_i - 1)$. The phase ends at state S_i, for $i \geq 1$, when one of the following conditions arises:

(*A*) $k_i = \ell_i = m_i$;
(*B*) $t_i = 0$;
(*C*) $t_i = 1$.

(Notice that (A) and (B) cannot arise together, as well as (B) and (C). Moreover, if (A) and (C) hold, then $i = n$.)

Condition (A) simply says that the three color classes have the same cardinality. The problem can, thus, be reduced to the same problem with smaller size $(n - 3k_i)$ and Paul can use induction.

The special cases when $t_i = 0, 1$ give a precise indication on the plurality and Paul can handle them easily.

Claim 1 *Paul has a strategy such that at every state* S_i *of Phase I, the following conditions hold:*

(i) $k_i \geq \ell_i \geq m_i$*;*
(ii) *a representative ball* K_i, L_i *of the two largest classes* k_i, ℓ_i *is known (if not empty);*

(iii) *the number T_i of comparisons up to (and including) S_i is less than or equal to $2k_i + \ell_i + 2m_i - 2$.*

PROOF. Proof by induction. After the first ball has been handled, $S_1 = (1,0,0)$, $T_1 = 0 \leq 2\cdot 1 + 0\cdot 1 + 0 \cdot 1 - 2$, $K_1 = b_1$ and L_1 is unknown as the class is empty. Let $1 \leq i < n$. Suppose K_i and L_i are the representatives of k_i and ℓ_i respectively and that b_{i+1} is handled. Conditions (i),(ii) are clearly preserved if Paul uses the following strategy.

If $k_i > \ell_i > m_i$:

$$b_{i+1} : L_i \begin{cases} \text{if YES } S_{i+1} = (k_i, \ell_i + 1, m_i) \\ \text{if NO } \quad b_{i+1} : K_i \begin{cases} \text{if YES } S_{i+1} = (k_i + 1, \ell_i, m_i) \\ \text{if NO } \quad S_{i+1} = (k_i, \ell_i, m_i + 1) \end{cases} \end{cases}$$

If $k_i > \ell_i = m_i$:

$$b_{i+1} : K_i \begin{cases} \text{if YES } S_{i+1} = (k_i + 1, \ell_i, \ell_i) \\ \text{if NO } \quad S_{i+1} = (k_i, \ell_i + 1, \ell_i), L_{i+1} = b_{i+1} \end{cases}$$

If $k_i = \ell_i$ then $\ell_i > m_i$ (otherwise finished by (A)):

$$b_{i+1} : K_i \begin{cases} \text{if YES } S_{i+1} = (k_i + 1, k_i, m_i) \\ \text{if NO } \quad b_{i+1} : L_i \begin{cases} \text{if YES } S_{i+1} = (k_i + 1, k_i, m_i) \\ \qquad K_{i+1} = b_{i+1}, L_{i+1} = K_i \\ \text{if NO } \quad S_{i+1} = (k_i, k_i, m_i + 1) \end{cases} \end{cases}$$

Unless differently stated $K_{i+1} = K_i$ and $L_{i+1} = L_i$.

As for condition (iii), observe that T_{i+1} is equal to T_i plus one or two, according to the number of comparisons Paul did. The proof follows by induction.

Let, for example, $k_i > \ell_i > m_i$ and assume b_{i+1} has the same color of L_i, so that $S_{i+1} = (k_i, \ell_i + 1, m_i)$. Then $T_{i+1} = T_i + 1 \leq 2k_i + \ell_i + 2m_i - 1 = 2k_i + (\ell_i + 1) + 2m_i - 2 = 2k_{i+1} + \ell_{i+1} + 2m_{i+1} - 2$. All the other cases can be proven analogously.

$\Diamond$

Claim 2 *One of $(A), (B), (C)$ eventually occurs.*

PROOF. At state S_1, we have $t_1 = n-1 \geq 3$ as $k_1 = 1$, $\ell_1 = 0$ and $n \geq 4$. Every time a ball is handled t_i changes by $0, -1$ or -2. In fact $t_{i+1} - t_i = -1 - (k_{i+1} - k_i) + (\ell_{i+1} - \ell_i)$ and only the cardinality of exactly one of the three color classes is increased by one. If (A) does not occur, when $i = n$ then $t_n = \ell_n - k_n + 1 \leq 1$ and hence (B) or (C) must occur.

$\diamondsuit$

Let (k, ℓ, m) be the state at the end of Phase I, with K and L representatives of the two largest color classes (if not empty), r remaining balls, $t = r-(k-\ell-1)$ and

$$T \leq 2k + \ell + 2m - 2 \quad (1)$$
$$n = k + \ell + m + r \ . \quad (2)$$

Phase II. Paul acts differently depending on how Phase I ended.

Case 1: (A) occurred first.
This means that $k = \ell = m$ and that the total number of comparisons done in Phase I is $T \leq 5k - 2$, by (1).

If $r = 0$, then there are no remaining balls and Paul learned that the three color classes have the same cardinality. Paul wins the game stating there is no plurality. Hence, as $k = n/3$ concerning the total number of comparisons we have

$$L_3(n) \leq T \leq 5k - 2 = \frac{5}{3}n - 2 \ .$$

If $r = 1$, then Paul wins the game showing the remaining ball as the plurality ball. In this case, $k = (n-1)/3$ and therefore

$$L_3(n) \leq T \leq 5k - 2 = \frac{5}{3}n - \frac{11}{3} \ .$$

If $r \geq 2$ the plurality among the n balls is the plurality among the $r = n - 3k$ remaining balls. As $2 \leq r < n$, by induction, Paul wins the game using $5r/3-2$ extra comparisons. Hence

$$L_3(n) \le T + \frac{5r}{3} - 2 \le 5k - 2 + \frac{5(n-3k)}{3} - 2 = \frac{5n}{3} - 4 \ .$$

Case 2: (B) occurred first.
Paul wins the game claiming that K is of the plurality color. In fact, $t = r - (k - \ell - 1) = 0$ means $k = \ell + r + 1$ and even if all remaining balls have the same color as L, there still is one more ball colored as K. Hence K is the plurality color.

To count the number of comparisons used by Paul observe that by (2),

$$k = \ell + r + 1 = \ell + n - k - \ell - m + 1 = n - k - m + 1 \ ,$$

and

$$3k = k + (\ell + r + 1) + m + (\ell - m) + r + 1 = n + r + (\ell - m) + 2 \ .$$

Suppose $r = 0$, then $\ell > m$. Because if $\ell = m$, then the terminal state is $(k, k-1, k-1)$ and thus the previous state was either $(k-1, k-1, k-1)$ and the game would have finished by (A), or $(k, k-1, k-2)$ and the game would have finished by (C). Hence $\max\{r, \ell - m\} \ge 1$, and so $3k \ge n + 3$ implying $k \ge n/3 + 1$.

It follows that

$$\begin{aligned}
L_3(n) \le T &\le 2k + \ell + 2m - 2 && \text{by (1)} \\
&= 2n - \ell - 2r - 2 && \text{by (2)} \\
&= 2n - (\ell + r + 1) - r - 1 && \\
&= 2n - k - r - 1 && \text{because } t = 0 \\
&\le 5n/3 - r - 2 \ . && \text{because } k \ge n/3 + 1
\end{aligned}$$

Case 3: (C) occurred first.
We have that $t = r - (k - \ell - 1) = 1$ if and only if $k = \ell + r$ and, hence, K is of the plurality color unless all the r remaining balls have the color of L (or M if $\ell = m$) or unless there are no remaining balls.

If $r = 0$ then $k = \ell > m$ and the game ends with Paul claiming that there is no plurality. To bound the total number of comparisons, observe that $n = k + \ell + m = 2k + m < 3k$ and hence $k > n/3$. We have

$$
\begin{aligned}
L_3(n) \le T &\le 3k + 2m - 2 \\
&= 2n - k - 2 \qquad\qquad \text{by (2)} \\
&< 5n/3 - 2\ .
\end{aligned}
$$

If $r \geq 1$, Paul takes a ball R from the remaining balls and compares it to the other $r-1$ balls. As soon as Carole answers NO, Paul wins the game claiming K is of plurality color. If Carole always answers YES then Paul wins using one last comparison.

If $\ell = m$:

$$
R : K \begin{cases} \text{if YES} & K \text{ is of plurality color} \\ \text{if NO} & \text{there is no plurality} \end{cases}
$$

If $\ell > m$:

$$
R : L \begin{cases} \text{if YES} & \text{there is no plurality} \\ \text{if NO} & K \text{ is of plurality color} \end{cases}
$$

Altogether, the total number of comparisons is $L_3(n) \leq T + r$. As $n = k + \ell + m + r = 2k + m < 3k$ we have $k > n/3$ and so

$$
\begin{aligned}
L_3(n) &\le T + r \\
&\le 2k + \ell + 2m - 2 + r \qquad\qquad \text{by (1)} \\
&= 2n - \ell - 2r - 2 + r \qquad\qquad \text{by (2)} \\
&= 2n - k - 2 < 5n/3 - 2\ .
\end{aligned}
$$

$\diamondsuit$

3.2 The lower bound

For the sake of presentation, we will first assume that n is even and then explain how to derive the same bound also in the case n is odd. Let the three colors be *red*, *blue* and *green*.

Any algorithm used by Paul can be seen as a sequence of steps in which Paul selects a pair of balls x, y and receives from Carole the answer YES or NO respectively meaning that x and y are colored with the same color or not.

During the game, Carole builds a graph $H = (V, E)$ (*Carole's graph*), where each node in $V \subseteq [n] = \{1, \ldots, n\}$ represents a ball that Paul involved in at least one comparison, and $(x, y) \in E$ if and only if Paul asked to compare x

and y, where the edges are labeled with YES or NO according to the answers Carole gave. The edges of H will be called YES-edges or NO-edges if they are labeled with YES or NO, respectively. Moreover, by H_Y and H_N we denote respectively the graph induced by the set E_Y of YES-edges and the set E_N of NO-edges of H. Assume n is even, unless differently specified.

Definition 3.1 *A graph H is said to be* nice*, if it satisfies the following properties:*

- *$H_N = (S_1 \cup S_2, E_N)$ is a bipartite graph, $V = S_1 \cup S_2$, $S_1 \cap S_2 = \emptyset$;*
- *$|S_1| \leq n/2$ and $|S_2| \leq n/2$;*
- *H_Y has no edge connecting a node $x \in S_1$ with a node $y \in S_2$.*

Let us show by induction that Carole has a strategy such that, at each step of *any* algorithm chosen by Paul, Carole's graph H is nice.

At the beginning of the game, Carole's graph is empty and thus trivially nice. Therefore, assume that Carole has a nice graph $H = (S_1 \cup S_2, E)$.

Let x, y be the pair of balls selected by Paul at the new step. Carole has to deal with one of the following cases.

Case 1: $x \in V$ and $y \in [n] \setminus V$.

Suppose w.l.o.g. that $x \in S_1$. If $|S_2| < n/2$, then Carole adds y to S_2 and answers NO. If $|S_2| = n/2$, then it must be $|S_1| < n/2$. In this case Carole adds y to S_1 and answers YES.

In both cases the new graph $H = (V \cup \{y\}, E \cup \{(x,y)\})$ is nice according to the new partition given by sets $S_1, S_2 \cup \{y\}$ in the former case, and by $S_1 \cup \{y\}, S_2$ in the latter.

Case 2: $x, y \in [n] \setminus V$.

If $|S_1| < n/2$ and $|S_2| < n/2$, Carole adds x to S_1 and y to S_2 and answers NO. Otherwise, suppose w.l.o.g., that $|S_1| = n/2$. Then it must be $|S_2| \leq n/2 - 2$ and Carole adds x and y to S_2 answering YES.

In both cases the new graph $H = (V \cup \{x, y\}, E \cup \{(x,y)\})$ is nice according to the new partition given by sets $S_1 \cup \{x\}, S_2 \cup \{y\}$ in the former case and by $S_1, S_2 \cup \{x, y\}$ in the latter.

Case 3: $x, y \in V$.

If $x \in S_1$ and $y \in S_2$, then Carole answers NO, otherwise she answers YES.

Therefore, in any case the new graph $H = (V, E \cup \{(x,y)\})$ is nice according to the partition sets S_1 and S_2.

Since we have shown that Carole has a strategy that allows her to maintain a graph that is nice, in the following we will always assume that Carole's graph is nice. Observe that Carole is always guaranteed that

$$|E_N| \geq \max\{|S_1|, |S_2|\} \ . \tag{3}$$

In fact, any new node inserted in H is inserted with a new NO-edge incident on it, unless $\max\{|S_1|, |S_2|\}$ is already $n/2$.

In the following we will say that a nice graph admits a coloring if the coloring is consistent with the labelling of YES and NO edges.

Lemma 3.1 *Let $H = (S_1 \cup S_2, E)$ be Carole's graph at the end of the game. Paul wins the game only if S_1 and S_2 are* YES*-components of cardinality $n/2$ each.*

PROOF. In order to prove the lemma, we will show that if S_1 and S_2 are *not* YES-components of cardinality $n/2$ each, then whenever Paul claims that there is no plurality, Carole is able to show that H admits a coloring having a plurality color. On the other hand, whenever Paul indicates that u is of plurality color, Carole is able to show that H admits another coloring in which u is not of the plurality color. In the following, given a color $col \in \{red, blue, green\}$, for any $u \in V$, $f(u) = col$ means that u is colored with col and for any set $S \subseteq V$, $f(S) = col$ means that all the balls in S are colored with col.

Assume first that $\min\{|S_1|, |S_2|\} = |S_1| < n/2$. Let $V_1, V_2 \subseteq V$ be two disjoint sets of nodes such that $V_1 \cup V_2 = V \setminus (S_1 \cup S_2)$ and $|V_j| + |S_j| = n/2$, for $j \in \{1, 2\}$. Of course, $|V_1| > 0$ and $|V_2| \geq 0$.

If Paul claims that there is no plurality or if he claims that $u \in S_1$ is of the plurality color, Carole shows the coloring f such that $f(S_1) = red$, $f(V_1) = blue$ and $f(S_2 \cup V_2) = green$. Graph H admits f, but f has a plurality color different from $f(u)$.

If Paul claims that $u \in S_2$ is of the plurality color, Carole shows the coloring f such that $f(S_1 \cup V_1) = red$ and $f(S_2 \cup V_2) = green$. It is easy to see that H admits f, but f has no plurality color.

In any case Paul is wrong.

Therefore, we can assume $|S_1| = |S_2| = n/2$. To prove that S_1 and S_2 have to be YES-components, we can proceed analogously, assuming there is a third YES-component that plays the role of V_1.

$\Diamond$

Theorem 3.2 *To solve the plurality problem with 3 colors, Paul needs at least $3n/2 - 2$ comparisons in the worst case.*

PROOF. Let $H = (S_1 \cup S_2, E)$ be Carole's graph at the end of a game Paul won. Then by Lemma 3.1 S_1 and S_2 are YES-components of cardinality $n/2$ each. Thus, the number of YES-edges in each YES-component is at least $n/2-1$. From (3) it follows that the number of NO-edges in H is at least $n/2$.

The number of comparisons used by Paul is the number of edges in H, that is, the number of edges in H_Y plus the number of edges in H_N, *i.e.*, $3n/2 - 2$.

$\Diamond$

Let us now see how to derive the same lower bound in the case n is odd. When n is odd, Carole cannot generalize the strategy she used for the case n even by just building a nice graph in which S_1 has cardinality $\lfloor n/2 \rfloor$ and S_2 has cardinality $\lceil n/2 \rceil$ (or vice-versa). In fact, once Paul has a YES-component of cardinality $\lceil n/2 \rceil$, he wins the game by claiming that the color of the nodes in that YES-component is the plurality color. The point is that Paul can build a YES-component of $\lceil n/2 \rceil$ nodes using only $2\lceil n/2 \rceil = n + 1$ comparisons.

Hence Carole's strategy has to be slightly modified. As in the case n even, she builds a nice graph H where the cardinality of sets S_1 and S_2 is bounded by $\lfloor n/2 \rfloor$. When Paul involves the last node, say l, in a comparison for the first time, Carole puts l in a third set S_3 and answers that the two nodes have different colors. In the sequel, whenever l will be involved in a comparison, Carole will say that the two nodes have different colors and will label all edges incident on l with NN. Such edges are called NN-edges and the set of all NN-edges is denoted by E_{NN}.

Let $H = (S_1 \cup S_2 \cup \{l\}, E)$ be Carole's graph at the end of the game and assume that S_i contains k_i YES-components, for $i = 1, 2$.

It is clear that "no plurality" is always possible by coloring S_1 red, S_2 blue and l green. Hence since Paul wins he must be able to exclude the possibility that there is a plurality. From this we conclude:

1. Node l must be connected to S_1 and S_2. Otherwise, if *e.g.*, l is not connected to S_1, $f(S_1 \cup \{l\}) = red$, $f(S_2)$ would be a plurality coloring.
2. If $k_i \geq 3$ ($i = 1, 2$), then l must be connected to every YES-component of S_i. Otherwise, if $C \subseteq S_1$ is a component not connected to l then $f(S_1 \setminus C) = red$, $f(S_2) = blue$ $f(C \cup \{l\}) = green$ would give a blue plurality.

It follows that l is connected by at least $k_i - 1$ edges to S_i. With $|E_N| \geq \lfloor n/2 \rfloor$ (as in the case when n is even) we have that

$$L_3(n) \geq |E_N| + |E_Y| + |E_{NN}| \geq \lfloor n/2 \rfloor + 2\lfloor n/2 \rfloor - k_1 - k_2 + (k_1 + k_2 - 2) = 3\lfloor n/2 \rfloor - 2 \ .$$

This concludes the proof of Theorem 3.2 both for n even and odd.

4 More colors

In this section, we prove that in order to solve the plurality problem with c colors, Paul needs $\Omega(c \cdot n)$ questions in the worst case. In view of the naive algorithm of section 2 this bound is asymptotically optimal.

We are given c colors. For the sake of presentation, we assume that n is a multiple of c. As in the lower bound for $c = 3$, during the game, Carole keeps a graph $H = (V, E)$ according to Paul's questions, where nodes correspond to balls and there is a YES-edge (respectively a NO-edge) between two nodes if and only if Carole's answer on these two nodes was yes (respectively no). As in Section 3.2, a YES-component is a component of H connected only by YES-edges. Nodes that are not in any YES-component are called singletons.

At the beginning, Carole arranges all n nodes in c disjoint sets, $S_1, \ldots, S_c$, with n/c nodes each. A singleton $v \in S_i$ is j-movable, for some $j \neq i$, if it has no NO-edge towards set S_j. Sets can be in two different states: marked or unmarked. At the beginning all sets are unmarked.

Carole uses the following strategy:

Whenever Paul asks for two nodes from different sets, Carole answers NO.

Whenever Paul asks for two nodes x, y from the same set S_i:

1. If the set is marked, Carole answers YES.
2. If the set is unmarked:
   a. If both nodes belong to YES-components, then Carole answers YES.
   b. If each of the two nodes has at least $c/2$ incident NO-edges, then Carole answers YES.
   c. If there are at least $n/5$ incident NO-edges in S_i, Carole answers YES

and mark S_i.

d. Let x be the node not in the YES-component:

   i. If x is j-movable (for some $j \neq i$) and in S_j there is an i-movable node, Carole exchanges the nodes and answers NO.

   ii. Otherwise Carole answers YES.

The following fact is straightforward.

Fact 3 *Before Carole goes through* 2.d.ii*, then in any unmarked set, every node that belongs to a* YES*-component has at least* $c/2$ *incident* NO*-edges.*

Lemma 4.1 *When the game is over, there is no movable node.*

PROOF. Let $x \in S_j$ be an i-movable node. Then the graph admits the following two colorings:

- For $i = 1, 2, \ldots, c$, color set S_i with color c_i; this coloring has no plurality.
- Color $S_i \cup \{x\}$ with color c_i; color $S_j \setminus \{x\}$ with color c_j; color any other set S_h, $h \neq i, j$, with color c_h; this coloring has plurality in S_i.

$\diamondsuit$

Lemma 4.2 *If there are* $c/4$ *marked sets, there are at least* $c \cdot n/40$ NO*-edges.*

PROOF. By simple calculation

$$\frac{1}{2} \cdot \#\text{ marked sets } \cdot \#\text{ NO-edges per set } \geq \frac{1}{2} \cdot \frac{c}{4} \cdot \frac{n}{5} = \frac{c}{40} n.$$

$\diamondsuit$

Theorem 4.1 *To solve the plurality problem with* c *colors, Paul needs* $\Omega(c \cdot n)$ *questions in the worst case.*

PROOF. When the game ends, two cases are possible: either (a) Carole never went through 2.d.ii, or (b) Carole did. To prove the theorem, we show that in both cases the number of NO-edges is $\Omega(c \cdot n)$.

Case (a). Let m and u be respectively the number of marked and unmarked sets at the end of the game. We have $0 \leq m \leq c$ and $u = c - m$

By Fact 3, every node that belongs to a YES-component has at least $c/2$ incident NO-edges. By Lemma 4.1, singletons cannot be movable, hence, each of them must have at least $(c-1)$ incident NO-edges. Therefore, in any unmarked set S, if Y denotes the number of nodes involved in YES-components and N denotes the number of singletons (where $Y + N = n/c$), we have that the number of NO-edges incident on S is at least

$$\frac{c}{2}Y + (c-1)N \geq \frac{c}{2} \cdot \frac{n}{c} = \frac{n}{2}.$$

Moreover, any marked set has at least $n/5$ incident NO-edges. In total, there are at least

$$\frac{1}{2}\left(m\frac{n}{5} + u\frac{n}{2}\right) \geq \frac{c}{10}n$$

NO-edges.

Case (b). Consider the first time when Carole goes through 2.d.ii. Let C be the family of sets S_i such that x is i-movable. Since Carole went through 2.d.ii, then the number of NO-edges incident on x is less than $c/2$. This implies that $|C| > c/2$.

By Lemma 4.2, we can assume that $m < c/4$. Let C' be the family of unmarked sets in C, we have $|C'| \geq |C| - m > c/4$.

Let s be the number of singletons in C'. As there is no i-movable node in C, then each singleton in C' must have a NO-edge towards set S_i. Therefore, we have $s < n/5$, otherwise Carole would have passed through 2.c.

The number of nodes in C' that belong to YES-components is at least

$$\#\text{ nodes } - \#\text{ singletons} \geq |C'| \cdot \frac{n}{c} - s > \frac{c}{4} \cdot \frac{n}{c} - \frac{n}{5} = \frac{n}{20}.$$

Hence, recalling Fact 3, the number of NO-edges incident on these sets is at least

$$\#\text{ nodes in YES-comp} \cdot \#\text{ NO-edges per node} \geq \frac{n}{20} \cdot \frac{c}{2} = \frac{c}{40}n.$$

◊

5 Conclusion and open problems

In this paper we studied the plurality problem which is a generalization of the well known majority problem. We gave the first algorithm for this problem with three colors and an almost matching lower bound.

The first natural question left open by this paper is to close the gap between upper and lower bound for three colors. Needless to say, it would be very nice to find a general optimal strategy for any number $c \leq n$ of colors.

Asymptotically, we have showed that $\Theta(c \cdot n)$ questions are necessary and sufficient. As our $\Omega(c \cdot n)$ lower bound shows that the naive deterministic algorithm is asymptotically the best possible, it would be interesting to see if randomization might help.

References

[1] M. Aigner, Combinatorial Search. Wiley 1988.

[2] M. Aigner, Two colors and more, preprint.

[3] M. Aigner, Variants of the majority problem, Discrete Applied Mathematics 137 (1) 2004, 3-26.

[4] M. Aigner, G. De Marco and M. Montangero, The plurality problem with three colors, Proc. 21st Ann. Symp. on Theoretical Aspects of Computer Science, STACS 2004, Montpellier, France, LNCS 2996, 513-521.

[5] L. Alonso, E. M. Reingold, and R. Schott, Determining the majority, Information Processing Letters 47 (1993), 253-255.

[6] L. Alonso, E. M. Reingold, and R. Schott, The average-case complexity of determining the majority, SIAM Journal on Computing 26 (1997), 1-14.

[7] L. Alonso, P. Chassaing, E. M. Reingold, and R. Schott, The chip problem, preprint. (Available at `http://emr.cs.uiuc.edu/~reingold/chips.ps`.)

[8] G. De Marco and A. Pelc, Randomized algorithms for determining the majority on graphs, Proc. 28th Int. Symp. on Mathematical Foundations of Computer Science, MFCS 2003, Bratislava, Slovak Republic, LNCS 2747, 368-377.

[9] M. Fisher and S. Salzberg, Finding a majority among n votes, Journal of Algorithms 3 (1982), 375-379.

[10] F.P. Preparata, G. Metze and R.T. Chien, On the connection assignment problem of diagnosable systems, IEEE Trans. on Electr. Computers 16 (1967), 848-854.

[11] M. E. Saks and M. Werman, On computing majority by comparisons, Combinatorica 11 (1991), 383-387.

[12] G. Wiener, Search for a majority element, Journal of Statistical Planning and Inference, 100 (2002), 313-318.