Tight Bounds on Plurality

Nikhil Srivastava [1] and Alan D. Taylor [2,*]

Department of Mathematics, Union College, Schenectady, NY 12308, USA

Abstract

We show that $\binom{n-1}{2}$ pairwise equality comparisons are necessary and sufficient (in the worst case) to find a plurality in n colored balls.

Key words: Analysis of algorithms; Computational complexity.

1 Introduction

Given a finite set $\{x_1, \ldots, x_n\}$ of colored balls, a *strict plurality color* is one that occurs more frequently than any other color, and a *plurality color* is one that occurs at least as frequently as any other color. In this paper, we consider the problem of picking a ball of a plurality color using only pairwise equality tests. That is, the only kind of operation we are allowed is to pick two balls and ask if they are the same color; in particular, we cannot actually 'look' at the color of a ball. For strict plurality, it is easy to see that all $\binom{n}{2}$ comparisons are needed in the worst case.

The corresponding question for majority has been studied in depth. Fischer and Salzberg [3] show that $\lceil \frac{3n}{2} \rceil - 2$ comparisons are necessary and sufficient (in the worst case) to pick a majority representative or declare that a majority does not exist; Saks and Werman [4] establish that $n - \nu(n)$ comparisons ($\nu(n)$ being the number of 1s in the binary representation of n) will do if it is known beforehand that the input contains a majority color. Alonso et al. [2] derive a tight bound of $\frac{2n}{3} - \sqrt{\frac{8n}{9\pi}} + O(\log n)$ comparisons in the average case.

* Corresponding author.
[1] nikhil.srivastava@yale.edu
[2] taylora@union.edu

Determining the bounds for plurality was stated as an open problem in [1] and [5]. In this paper, we show that $\binom{n-1}{2}$ comparisons are necessary and sufficient in the worst case. The difference between dealing with strict plurality and plurality is already clear in the case of three balls, where only one comparison is needed for the latter. That is, if we know that balls one and two are the same color, then either of them will definitely represent a plurality (of size two or of size three, although we don't know which). On the other hand, if we know that balls one and two are different colors, then ball three will definitely represent a plurality (either alone or in conjunction with ball one or in conjunction with ball two).

2 The Lower Bound

Theorem 1 *Any correct algorithm for plurality must use at least $\binom{n-1}{2}$ comparisons in the worst case.*

PROOF. We will construct an adversary that forces $\binom{n-1}{2}$ *unequal* comparisons; the adversary will simply answer 'no' to all queries. Suppose an algorithm $\mathcal{A}$ stops after $c < \binom{n-1}{2}$ comparisons and declares that a ball x_p has a plurality. Notice that there are $\binom{n-1}{2}$ pairs among the remaining $(n-1)$ balls $x_1, \ldots, x_{p-1}, x_{p+1}, \ldots, x_n$, so two of these balls, say x_v and x_w, must not have been compared. We simply assert that x_v and x_w are colored white and the rest of the balls are distinct colors. This is consistent with the adversary's responses, but not with $\mathcal{A}$'s output. Thus $\mathcal{A}$ does not work, and any *correct* algorithm must use at least $\binom{n-1}{2}$ comparisons to pick a plurality ball. □

3 The Upper Bound

Theorem 2 *The following algorithm correctly picks a plurality representative from n balls in at most $\binom{n-1}{2}$ comparisons.*

(1) Compare each pair of balls in $\{x_3, \ldots, x_n\}$ and place them in bins so that two balls are in a bin iff they are the same color.

(2) Pick a bin and compare one ball from it with x_2. If the comparison is equal, place x_2 in the bin and move on to the next step. Otherwise, if there are bins left, try the next bin. If all comparisons are unequal and x_2 does not get placed in any of the bins, create a new singleton bin for x_2 and move to the next step.

(3) Set aside all bins of maximal size – let these be $B_1, \ldots, B_d$, each containing s balls (say). If $s = 1$ return x_1. Otherwise, compare x_1 to a ball from each of $B_1, \ldots, B_{d-1}$. If any comparison is equal, return x_1. If they are all unequal, return any ball from B_d.

PROOF. *Correctness.* At the end of step 2, the balls $x_2, \ldots, x_n$ have been distributed into bins according to color; let these be $C_1, \ldots, C_m$. If all bins are singletons, then $s = 1$ and no two of $x_2, \ldots, x_n$ are the same color. There are two possibilities as far as x_1 is concerned: either all the balls are of different colors, or x_1 is the same color as one of the other balls. In both cases, x_1 represents a plurality, so it is safe to return this as the answer without further comparisons.

Otherwise, assume $s > 1$, and consider what happens in step 3. If x_1 is the same color as one of the maximal bins $B_1, \ldots, B_{d-1}$, say B_m, then there are $s + 1$ balls of this color and at most s balls of any other color. Thus x_1 (as well as any ball in B_m) represents a plurality. On the other hand, if x_1 is not the same color as any of $B_1, \ldots, B_{d-1}$, then one of the following occurs:

- x_1 is the same color as B_d. It is clear that B_d represents a plurality.
- x_1 is a different color from every other ball. Again, there are no more than s balls of any color, so B_d represents a plurality.
- x_1 is the same color as some non-maximal bin C_l with $|C_l| < s$. There are at most $|C_l| + 1 \leq s$ balls of color x_1, so B_d still represents a a plurality.

In all cases, B_d represents a plurality, so it is safe to return a ball from B_d as the answer without further comparisons.

Complexity. Step 1 uses exactly $\binom{n-2}{2}$ comparisons. We will count the number of comparisons used in steps 2 and 3:

- If $s = 1$, then all bins are singletons at the end of step 1. Consequently, step 2 takes $n - 2$ comparisons, and step 3 takes no comparisons, for a total of $\binom{n-2}{2} + (n-2) = \binom{n-1}{2}$.
- If $s \geq 2$, then at the end of step 1 there are at least $d - 1$ bins of size s, since at most one ball is added to at most one bin in step 2 and we know there are d bins at the end of step 2. Note that in step 2, x_2 is compared to only one ball from each such bin; in particular, it is *not compared* to one other ball from each bin (because each bin has $s \geq 2$ balls). So the number of comparisons in step 2 is at most

$$\text{(number of balls in bins after step 1)} - (d-1)$$

$$\leq (n-2) - (d-1)$$

It is clear that the number of comparisons in step 3 is at most $(d-1)$, and adding this to the previous comparisons gives a total of at most

$$\binom{n-2}{2} + (n-2) - (d-1) + (d-1) = \binom{n-1}{2}$$

as desired.

□

4 References

(1) L. Alonso, E.M. Reingold, R. Schott, Determining the Majority, *Information Processing Letters* **47** (1993) 253-255.

(2) L. Alonso, E.M. Reingold, R. Schott, Average-case Complexity of Determining the Majority, *SIAM J. Comput.* **26** (1997).

(3) M. J. Fischer and S. L. Salzberg, Solution to Problem 81-5, *J. Algorithms* **3** (1982) 376-379.

(4) M.E. Saks and M. Werman, On Computing Majority by Comparisons, *Combinatorica* **11** (1991) 383-387.

(5) G. Wiener, Search for a Majority Element, *Journal of Statistical Planning and Inference* **100** (2002) 313-318.