Majority in the Three-Way Comparison Model

by

Azin Nazari

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Computer Science

Waterloo, Ontario, Canada, 2019

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

In this thesis, we study comparison based problems in a new comparison model called three-way, where a comparison can result in $\{>, =, <\}$. We consider a set of n balls with fixed ordered coloring. Particularly, we are interested in finding a ball of the majority color, the color that occurs more than half, when there are 2 colors, partition problem, where the goal is to determine groups of balls with the same color when there are 2 and 3 colors, respectively. We study these problems using both deterministic and randomized approaches.

Acknowledgements

First of all, I would like to thank my supervisors Ian Munro and Semih Salihoglu. Specially, Ian who introduced me to a family of exciting, classical algorithm problems which later shaped the idea of this thesis. I also want to thank Semih for making several insightful remarks on my thesis. I would like to thank Sebastian Wild for his invaluable inputs and ideas. I would like to extend my gratitude toward my readers, Naomi Nishimura and Grant Weddell. Last but not least, I want to thank my parents and my brother for their unconditional support during my studies and throughout my life in general.

Dedication

This is for you, Majin.

Table of Contents

List of Algorithms

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Comparison-Based Problems

The computational complexity of comparison-based problems deals with finding the minimum number of comparisons necessary to solve certain problems. Interestingly, there are some fundamental problems in this area which remain unsolved. For example, theoretically, sorting requires at least $\lg(n!)$ comparisons even in the expected case and at least $\lceil \lg(n!) \rceil$ in the worst case. This bound is not achievable for certain values of n including 12. The Ford-Johnson merge-insertion sort algorithm comes with approximately $0.028n$ more comparisons than the information-theoretic lower bound [14]. Similarly, by the works of Dor and Zwick, it is known that $2n$ comparisons are necessary and $3n$ are sufficient to find the median of n values [9, 10], while the constant is conjectured by Paterson to be $\log_{4/3} 2$ [18]. Beside sorting and median-finding, several other fundamental problems fit into the category of comparison-based problems and there is still gap between their lower and upper bounds [4, 13, 8]. In this work, we focus on the problem of determining whether any value is in the majority of n values (and if so finding this value). We are also interested in determining the most frequently occurring value (plurality) and also partitioning the elements based on their value. Our interest is in both the equality-test comparison model, i.e. the set of outcomes of a comparison is $\{=, \neq\}$, and the three-way comparison model, i.e. values are ordered and the set of outcomes is $\{<, =, >\}$. Formally,

Definition 1.1.1. The three-way comparison model is a type of comparison model in which, regarding an ordering on the values of the elements, any two elements can be compared with respect to their values and the outcome of any comparison is either $<, =$, or $>$.

Consistent with the literature, the scenario in which we express the problem is a set of n balls with a fixed coloring from which we can take any two and, without knowing the color of each ball, compare their colors. Our work focuses on cases in which the number of distinct possible colors is known and restricted to two or three, and considers both deterministic and randomized Las Vegas methods along with proof of correctness for each proposed algorithm. One of the classical results in this area is for the problem of finding the majority in a set of n colored balls with unrestricted number of colors in linear time. In 1980 (though not published until 1991), Boyer and Moore [7] found a clever solution by sweeping the sequence of balls from left to right and keeping track of a candidate and a counter: starting from the first number and counter = 0, depending on whether the next number equals to the candidate or not increase or decrease the counter by 1, respectively, and if the counter drops down to zero, change the candidate to the next number. It is easy to see that the last candidate is the only potential candidate, therefore, by comparing this candidate to all other numbers we can confirm if it is, in fact, the majority. This beautiful algorithm uses no storage and $2n$ comparisons to find the majority; a natural question is what is the minimum number of comparisons needed to do so. Fischer and Salzberg [12] answered this question by modifying the mentioned algorithm. The modified version, with the help of a storage of size n though, finds the majority in $\lceil \frac{3n}{2} \rceil - 2$ comparisons. They also proposed an adversary which shows this solution is indeed the best one for all deterministic algorithms. Recently, new results have improved these bounds by introducing randomness. For instance, in the majority problem with only two colors, any deterministic algorithm needs at least $n - o(n)$ comparisons (the exact number is discussed in 1.2.1), while a random pairing of balls with the assumption that colors occur uniformly has a run-time of $\frac{2n}{3}$ [3]. For reference, Table 1.1 summarizes the previous results on these problems [16].

Table 1.1: Previous Results on Equality-test Comparison Model

Problem	Deterministic		Randomized	
	Lower bound	Upper bound	Lower bound	Upper bound
Majority				
2 colors	$n - B(n)$	$n - B(n)$	$\frac{2n}{3} - o(n)$	$\frac{2n}{3} + o(n)$
unrestricted colors	$\lceil \frac{3n}{2} \rceil - 2$	$\lceil \frac{3n}{2} \rceil - 2$	$cn - o(n)$	$\frac{7n}{6} + o(n)$
Plurality				
3 colors	$\frac{3n}{2} - O(1)$	$\frac{5n}{3} + O(1)$	$\frac{3n}{2} - o(n)$	$\frac{3n}{2} + O(1)$
Partition				
2 colors	$n - 1$	$n - 1$	$n - 1$	$n - 1$
3 colors	$2n - 3$	$2n - 3$	$\frac{5n}{3} - \frac{8}{3}$	$\frac{5n}{3} - \frac{8}{3} + o(1)$

n is the size of input and $B(n)$ is number of $1'$s in binary representation of n [16].

However, there are problems for which there is a gap between the known lower and upper bounds for both deterministic and randomized algorithms. Besides, some of them have been studied in the equality-test comparison model, only one of the possible comparison systems. Although previous studies mostly focus on the equality-test model, the three-way comparison is actually being used in practice. Many processors support three-way comparison on primitive types by having relevant instruction sets. For instance, some signed number representations allow machines to differentiate positive, negative, and zero integers. This thesis studies the problems of plurality and partition under a three-way comparison, where colors are ordered and the result of a comparison can be any of $>, <$, or $=$. A three-way comparison system can be simulated by combining two comparisons such as $A = B$ and $A < B$, or $A < B$ and $A > B$. This observation naturally raises the question of how many comparisons can be saved by having direct access to the results of these two comparisons in a three-way model, which is the main theme of this thesis. As the problems studied in this thesis concern fundamental operations being atoms of high-level computations, even a small decrease in the number of comparisons can be tremendously beneficial on a larger scale. In the remainder of this chapter, we overview the contributions of this thesis along with a number of directions for future research.

1.2 Our Contributions

1.2.1 Majority

In the majority problem, we are interested in finding the color (if any) that occurs more than half of the times in a sequence of colored balls. The input consists of n balls tagged with different colors where the number of colors can be restricted or unrestricted. In the equality-test comparison model, we only can check if the two colors are the same or not. In this setting, Fischer and Salzberg [12] proved that for an unrestricted number of colors, in the deterministic case $\lceil \frac{3n}{2} \rceil - 2$ is the number of comparisons needed which is necessary and sufficient in the worst case. For this problem, Yang [24] showed the three-way comparison model cannot improve the bound for equality-test deterministic algorithms, but it can result in better upper bounds for probabilistic strategies. A special case of the problem when there are only two colors was later considered by Saks and Werman [19]. They showed that $n - B(n)$ number of comparisons is a tight bound, where $B(n)$ is the number of 1's in the binary representation of n. In this thesis, we study the majority problem in both the equality-test and the three-way comparison models. We propose a randomized algorithm for the problem of 2-color majority in equality-test comparison model and prove

an upper bound on the randomized 2-color majority in both models. We prove a lower bound on the number of comparisons in the problem of deterministic 2-color majority in the three-way comparison model by giving an adversary argument. We also prove that there exist families of inputs in 2-color majority in which the upper bound for the three-way comparison model is better than the tight bound for the equality-test model. Our contributions to the majority problem can be formalized as below:

Theorem 2.2.1. *There is no deterministic algorithm for 2-color majority with n balls in the three-way comparison model that makes less than cn comparisons for $c < 1$: there is an adversary showing the lower bound is at least $n - 2\sqrt{n}$.*

Theorem 2.3.1. *Compared to equality-test comparison model, three-way can solve the 2-color majority problem with fewer comparisons for the following families of input size n:*

$$F_1 = \{n = 2^k \mid k \in \mathbb{N}, k \geq 3\},$$
$$F_2 = \{n = 2^k + 1 \mid k \in \mathbb{N}, k \geq 3\}.$$

Theorem 3.0.1. *Let $\alpha \in \left[\frac{1}{2}, 1\right]$ be the fraction of the majority color. Then, there is a Las Vegas algorithm which solves 2-color majority problem with at most $g(\alpha)n + o(n)$ comparisons for input size of n with high probability, where $g(\alpha)$ is defined as below and is always between $\frac{1}{2}$ and $\frac{2}{3}$:*

$$g(\alpha) = \frac{2\alpha - 1}{4\alpha} \sum_{k=0}^{\lg n} \frac{1}{2^k} \frac{\alpha^{2^k} + (1-\alpha)^{2^k}}{\alpha^{2^k} - (1-\alpha)^{2^k}}.$$

1.2.2 Plurality and Partition

The plurality problem was first introduced by Aigner [1] in 2004. In this problem, we are given with n balls each colored with one of the possible k colors and the objective is to find a ball with the plurality color: the ball for which the number of balls with this color is more than those of any other color; if no such color exists, we declare that there is a tie. As in the previous problem, we are only allowed to compare two balls at each turn. It is clear that in the case of 2 distinct colors, the plurality and the majority problems are the same. A deterministic algorithm for 3 colors is proposed by [2] which can solve the problem in $\frac{5n}{3} - 2$ comparisons; the authors also proved a lower bound of $\left\lfloor \frac{3n}{2} \right\rfloor - 2$ comparisons for 3-color plurality. A probabilistic approach by [11] revealed that $\frac{3n}{2} + o(n)$ comparisons for n balls and 3 colors is necessary and sufficient in the expected case.

In the same context of the previous problems, the partition problem asks for partitioning the balls with respect to their colors. In 2005, [11] showed that any deterministic algorithm solving the problem with n balls and k colors needs at least $(k-1)n - \binom{k}{2}$ comparisons, and this number is also sufficient. For 3 colors, they proved the necessary and sufficient expected number of comparisons is $\frac{5n}{3} - \frac{8}{3} + o(1)$. For the problem of partition with 3 distinct colors, we show an upper bound with both deterministic and randomized approaches. Because the partition problem yields plurality, these are upper bounds for 3-color plurality as well. Our results are formalized as follows:

Theorem 3.5.1. *There is a Las Vegas algorithm which solves 2-color partition problem in the three-way comparison model making at most $2\alpha g(\alpha)n + o(n)$ comparisons with high probability, where α is the fraction of majority color and $\frac{2}{3} \leq 2\alpha g(\alpha) \leq 1$ is given by the following:*

$$g(\alpha) = \frac{2\alpha - 1}{4\alpha} \sum_{k=0}^{\lg n} \frac{1}{2^k} \frac{\alpha^{2^k} + (1-\alpha)^{2^k}}{\alpha^{2^k} - (1-\alpha)^{2^k}}$$

Theorem 4.0.1. *Given a set of n balls colored with numbers $1 < 2 < 3$, there is a deterministic algorithm which partitions the balls using at most $\frac{3n}{2} - 2$ comparisons.*

Theorem 5.0.1. *Given a set of n balls colored with numbers $1 < 2 < 3$, there is a Las Vegas algorithm which partitions the balls with $h(\beta)n + o(n)$ comparisons in the expected case, where $1 \leq h(\beta) \leq \frac{3}{2}$ is a function of the distribution of colors.*

Table 1.2 summarizes the results studied in this section and the preceding one.

Table 1.2: New Results on the Three-way Comparison Model

Problem	Deterministic		Randomized
	Lower bound	Upper bound	Upper bound
Majority			
2 colors	$n - 2\sqrt{n}$	$n - B(n) - 1$ $\forall n = 2^k$ or $2^k + 1$ and $k \geq 8$	$g(\alpha)n + o(n)$ $\left(\frac{1}{2} \leq g(\alpha) \leq \frac{2}{3}\right)$
Partition			
2 colors			$2\alpha g(\alpha)n + o(n)$ $\left(\frac{2}{3} \leq 2\alpha g(\alpha) \leq 1\right)$
3 colors		$\frac{3n}{2} - 2$	$h(\beta)n + o(n)$ $\left(1 \leq h(\beta) \leq \frac{3}{2}\right)$

$B(n)$ denotes the number of 1's in the binary representation of n. $g(\alpha)$ and $h(\beta)$ are functions based on the fraction of the majority color, $\alpha \in [\frac{1}{2}, 1]$, and the fraction of the plurality color, $\beta \in [0, 1]$.

1.3 Overview

In Chapter 2, we propose a lower bound on the problem of the 2-color majority in the three-way comparison model. We then study the upper bound on the problem and provide two families of inputs in which the upper bound beats the tight bound on the problem in the equality-test model. In Chapter 3, we present a randomized approach to solving the problem of 2-color majority which can be used both in the equality-test and the three-way comparison models. Then, we provide an upper bound based on the same approach for the three-way 2-color partition. In Chapter 4, we move to the problem of the 3-color partition and plurality. We study the problem in the deterministic setting. In Chapter 5, we use a randomized approach to the problem. Lastly, in Chapter 6, we discuss open problems and future work in this area.

Chapter 2

Deterministic 2-color Majority

The problem of finding the majority when there are only two colors was first solved in the equality-test model by Saks and Werman [19] by giving an elegant proof using generating functions. Later, Alonso et al. [3] found a combinatorial proof based on a number-theoretic argument on the number of leaves in the decision tree of the problem. Completely solved in the equality-test model, the upper bound and lower bound are found to be $n - B(n)$, where $B(x)$ denotes the number of $1's$ in the binary representation of natural number x. The methods used to prove the lower bound in equality-test relied on the following property: at each step of the problem, the information obtained so far can be described by a vector $(C_1, C_2, \ldots, C_s)$ where each C_i is a set of balls known to have the same color, i.e, a homogeneous component. Starting from $(\underbrace{1, 1, \ldots, 1}_{\text{n times}})$, if two balls, representatives of C_i, C_j, happen to be of the same color in a comparison, then in the next step, instead of C_i and C_j, we will have a new homogeneous component of size $|C_i| + |C_j|$. On the other hand, if they are not of the same color, we can replace C_i and C_j with a component of size $||C_i| - |C_j||$; no more information can be obtained by an inequality. This is however not true in the three-way model as an inequality can reveal the color of two balls: the greater one should be of color 2 while the other should be of color 1. This seemingly minor difference makes the proofs in equality-test comparison irrelevant to the case of three-way comparison. In fact, by knowing $a > b$ and $c > d$ (obviously, in three-way comparison model) we can deduce $a = c = 2$ and $b = d = 1$. To obtain the same information we would have needed 3 comparisons in the equality-test model, and therefore three-way saves us one comparison. This may suggest that the lower bound and upper bound for 2-color majority problem in the three-way model is different from those in the equality-test model. In this chapter, we propose an adversary which shows the lower bound is at least $n - 2\sqrt{n}$, and

therefore, even if three-way is better, there is no algorithm with at most cn comparisons for $c < 1$. However, we also characterize two families of the input where three-way comparison model outperforms the equality-test model by at least one comparison.

The problem of finding the majority can be formalized as follows:

Definition 2.0.1 (2-color Majority)**.** Given n balls of 2 different ordered colors, 1 and 2 which $1 < 2$, the goal is to find the color that occurred more than half of the times with the minimum number of comparisons.

2.1 Preliminaries

In order to count the number of comparisons comparison, we need to first define the following:

Definition 2.1.1 (Homogeneous Component)**.** A set of balls known to have the same color, which is unknown. For a homogeneous component C, we refer to its size by $|C|$.

Definition 2.1.2 (Known Component)**.** A and B: A is a component known to have balls of color 1 and B is a component known to have balls of color 2.

Definition 2.1.3 (I_{UU})**.** The number of comparisons made until now between two homogeneous components that result in an inequality.

We assume that an algorithm does not perform any comparison between two known components. By having the above definitions, we can now state the following lemma on the number of comparisons made up to now:

Lemma 2.1.1. *If there are m homogeneous components, then the number of comparisons is equal to*

$$c = n - m - I_{UU}.$$

Proof. Using induction on m, we show that each comparison increases c by 1. At first, $m = n$ and $I_{UU} = 0$, therefore $c = 0$ and therefore the identity holds. Suppose the lemma is true for all $m > k$. We show it also true for $m = k$. Indeed, the next comparison is either between two homogeneous components or between one known and one homogeneous, as

there is no benefit in comparing two known components. In the former case, if the outcome is equality, then two components will be unified and m changes to $m-1$ and therefore c increases by 1; otherwise, the outcome is inequality and therefore two components will be split among known components. In this way, m decreases by 2, while I_{UU} increases by 1. Thus c increases by 1. Similarly, in the case when one of the components is known and the other is not, the result of comparison determines to which known component the homogeneous one belongs. Therefore, m decreases by 1 and I_{UU} remains unchanged, so c increases by 1. □

2.2 Lower Bound

In light of 2.1.1, an adversary maximizing the number of comparisons should keep $m + I_{UU}$ as low as possible. We know regardless of the comparison m decreases either by 1 or 2 and this change is common between the three-way and equality-test comparison models. Consequently, in order to achieve a lower bound for the three-way, we need to minimize I_{UU}. To this end, we propose an adversary in which I_{UU} increases only if the total size of the two components being paired is more than $\sqrt{n}$. At the same time, to make the game longer, upon each occurrence of I_{UU}, we split two components such that A and B have almost the same size. We give a formal proof in the following theorem.

Theorem 2.2.1 (Deterministic 2-color Majority Lower Bound)**.** *No deterministic algorithm exists for 2-color majority in three-way comparison model that can solve the problem with less than cn comparisons for $c < 1$: there is an adversary showing the lower bound is at least $n - 2\sqrt{n}$.*

In our adversary, if the opponent asks for the result of comparison $x : y$ then

- If x, y belong to two homogeneous components C_x, C_y, and $|C_x| + |C_y| < \sqrt{n}$ we announce $x = y$; otherwise, we announce the colors such that smaller of C_x and C_y goes to the larger of A and B.
- If x belongs to a known component, we announce the color of y to be the color of the smaller one between A and B.

We show this adversary can force the opponent to make at least $n - 2\sqrt{n}$ comparisons. We start with the following lemma:

Lemma 2.2.2. *For any homogeneous component C, $|C|-1$ comparisons have been made.*

Proof. This is true for all components of size 1. We proceed by induction on $|C|$. If $|C| > 1$, then the component has been made by merging two components C_1 and C_2 where $|C_1| + |C_2| = |C|$, and because $|C_1|, |C_2| < |C|$, the induction hypothesis implies that each component needs $|C_1| - 1$ and $|C_2| - 1$ comparisons to be created. Together with the last comparison, we have $|C_1| - 1 + |C_2| - 1 + 1 = |C| - 1$ comparisons. □

Lemma 2.2.3. *To obtain two known components A and B, exactly $|A| + |B| - I_{UU}$ comparisons have been made.*

Proof. By the definition of I_{UU}, we know that exactly I_{UU} comparisons resulted in inequalities. Each of these inequalities reveal the color of two components, thus adds one component (say C_1) to A and one component (say C_2) to B. In this way, $|A|+|B|$ increases by $|C_1|+|C_2|$. The number of comparisons made to obtain C_1 and C_2 is $|C_1|-1+|C_2|-1 = |C_1| + |C_2| - 2$; together with the last comparison, $|C_1| + |C_2| - 2 + 1 = |C_1| + |C_2| - 1$ comparisons increase $|A| + |B|$ by $|C_1| + |C_2|$, if there was an inequality between C_1 and $|C_2|$. On the other hand, if $|A| + |B|$ increases as a result of comparing one homogeneous component (say C_3) to a known component (A), then $|C_3|-1+1 = |C_3|$ comparisons have been made. With having these in mind, a similar argument to the above lemma yields the identity stated in the lemma. □

Lemma 2.2.4. *I_{UU} is at most $\sqrt{n}$.*

Proof. The adversary only responds inequality when the sizes of the two components X and Y exceeds $\sqrt{n}$, which implies by each inequality $|A| + |B|$ increases by at least $\sqrt{n}$. Hence, we have at most $\frac{n}{\sqrt{n}} = \sqrt{n}$ comparisons between two homogeneous components that result in an inequality. □

Lemma 2.2.5. *In all steps $||A| - |B|| \leq \sqrt{n}$.*

Proof. Clearly, this true at the beginning where $|A| = |B| = 0$. Because of the first condition in the adversary, all the homogeneous components are of the size ranging from 1 to $\sqrt{n}$. At each step, if A and B, with $|A| \geq |B|$ become updated and change to A' and B' then depending on the order of $|A'|$ and $|B'|$ we have:

$$||A'| - |B'|| \leq \max(|A| - |B|, |B| - |A| + ((\sqrt{n} - 1) - 1)) \leq \sqrt{n}.$$

□

Proof of Theorem 2.2.1. A *terminal state* is either a state in which one of the known components is larger than $\frac{n}{2}$, or both A and B are known to be $\frac{n}{2}$, i.e, we have no majority. In the proof below we assumed the first case happens because the proof of the latter case is clear: if $|A|$ becomes larger than $\frac{n}{2}$, by Lemma 2.2.5, $|A|+|B| \geq |A|+|A|-\sqrt{n} \geq n-\sqrt{n}$. Hereafter, by terminal state, we refer to the first case when there is a majority.

We have two scenarios for the terminal states. The first one is when we compare two balls x and y from homogeneous components C_x and C_y, and the second one is when we compare on ball x from homogeneous component C_x with a ball from a known component. In the first scenario, $|C_x|+|C_y| > \sqrt{n}$ so that the adversary has to respond inequality and split the two components between A and B in a way that the difference between their sizes is being kept low. However, as this is a terminal state, in either dividing $A \leftarrow x, B \leftarrow y$ or $A \leftarrow y, B \leftarrow x$ the algorithm terminates. Thus either of these two happens:

$$\begin{aligned}&\left(|A|+|C_x| > \frac{n}{2} \quad \vee \quad |B|+|C_y| > \frac{n}{2}\right) \quad \wedge \\ &\left(|A|+|C_y| > \frac{n}{2} \quad \vee \quad |B|+|C_x| > \frac{n}{2}\right)\end{aligned}$$

Suppose $|A| \geq |B|$, thus by Lemma 2.2.5 we conclude that $|A| \geq \frac{n}{2} - \sqrt{n}$ and $|B| \geq \frac{n}{2} - 2\sqrt{n}$. Using Lemma 2.2.2 and Lemma 2.2.3, we know for C_x and C_y and for A and B, $|C_x|-1+|C_y|-1$ and $|A|+|B|-I_{UU}$ comparisons were made, respectively. The total number of comparisons is then $|A|+|B|+|C_x|+|C_y|-1-I_{UU}$. As a consequence of Lemma 2.2.4, to show this number is at least $n-2\sqrt{n}$, we only need to prove $|A|+|B|+|C_x|+|C_y|-1$ is at least $n-\sqrt{n}$. We prove this by considering each of four possible cases separately:

Case 1. $|A|+|C_x| > \frac{n}{2} \quad \wedge \quad |A|+|C_y| > \frac{n}{2}$

Combining Lemma 2.2.5 with the given condition, in this case, we have

$$\begin{aligned}&|A|+|B|+|C_x|+|C_y| \\ &= (|A|+|C_x|)+(|A|+|C_y|)+|B|-|A| \\ &\geq \frac{n}{2}+\frac{n}{2}-\sqrt{n} \\ &= n-\sqrt{n}.\end{aligned}$$

Case 2. $|A|+|C_x| > \frac{n}{2} \quad \wedge \quad |B|+|C_x| > \frac{n}{2}$

Using the fact that a homogeneous component size is at least 1 and at at most $\sqrt{n}$,

$|C_y| - |C_x|$ is at least $1 - \sqrt{n}$ and we have

$$\begin{aligned}
&|A| + |B| + |C_x| + |C_y| \\
&= (|A| + |C_x|) + (|B| + |C_x|) + |C_y| - |C_x| \\
&\geq \frac{n}{2} + \frac{n}{2} + 1 - \sqrt{n} \\
&= n + 1 - \sqrt{n} \\
&> n - \sqrt{n}
\end{aligned}$$

Case 3. $|B| + |C_y| > \frac{n}{2} \quad \wedge \quad |A| + |C_y| > \frac{n}{2}$

Using the fact that $|C_x| - |C_y|$ is at least $1 - \sqrt{n}$, we have

$$\begin{aligned}
&|A| + |B| + |C_x| + |C_y| \\
&= (|A| + |C_y|) + (|B| + |C_y|) + |C_x| - |C_y| \\
&\geq \frac{n}{2} + \frac{n}{2} + 1 - \sqrt{n} \\
&= n + 1 - \sqrt{n} \\
&> n - \sqrt{n}
\end{aligned}$$

Case 4. $|B| + |C_y| > \frac{n}{2} \quad \wedge \quad |B| + |C_x| > \frac{n}{2}$

Using the assumption that $|A| \geq |B|$, we have

$$\begin{aligned}
&|A| + |B| + |C_x| + |C_y| \\
&= (|B| + |C_y|) + (|B| + |C_x|) + |A| - |B| \\
&\geq \frac{n}{2} + \frac{n}{2} \\
&= n \\
&> n - \sqrt{n}
\end{aligned}$$

In the second scenario for the terminal state, we have a ball x from a component C_x that is compared to a ball from one of the known components; it is a terminal state because announcing x as each of two colors will result in a known component of size more than $\frac{n}{2}$. This condition can be written as

$$|A| + |C_x| > \frac{n}{2} \quad \wedge \quad |B| + |C_x| > \frac{n}{2}.$$

Similar to above cases, we need to show $|A| + |B| + |C_x|$ is at least $n - \sqrt{n}$, which is true because $|C_x|$ is less than $\sqrt{n}$ and

$$\begin{aligned}
&|A| + |B| + |C_x| \\
&= (|A| + |C_x|) + (|A| + |C_x|) - |C_x| \\
&\geq \frac{n}{2} + \frac{n}{2} - \sqrt{n} \\
&= n - \sqrt{n}.
\end{aligned}$$

□

2.3 Upper Bound

The three-way comparison model includes the equality-test model in itself; hence, the upper bound $n - B(n)$ holds for the three-way comparison model as well. In particular, this association can be described by mapping three-way comparisons resulting in $<$ or $>$ to a $\neq$ in the equality-test model. The question is can we solve 2-color majority problem with less than $n - B(n)$ comparisons in the three-way model. Although the general answer is not known to date, in this section, we show that for two families of the input size we can save at least one comparison and find the majority by using $n - B(n) - 1$ comparisons in total.

Theorem 2.3.1. *For the following families of input size n, the three-way model can solve the 2-color majority problem with $n - B(n) - 1$ comparisons:*

$$\begin{aligned}
F_1 &= \{n = 2^k \mid k \in \mathbb{N}, k \geq 3\}, \\
F_2 &= \{n = 2^k + 1 \mid k \in \mathbb{N}, k \geq 3\}.
\end{aligned}$$

2.3.1 Upper bound for $F_1 = \{n = 2^k \mid k \in \mathbb{N}, k \geq 3\}$

When $n = 2^k$ $(k \geq 3)$ we would like to show $n - 2$ $(= n - B(n) - 1)$ comparisons suffices. From Lemma 2.1.1 if $I_{UU} \geq 2$ we are done. Therefore, we must consider the cases in which $I_{UU} = 0$ or 1. Based on these ideas, we divide the input into three groups of roughly the same sizes. In each group, we compare one ball with all other balls, if there is an

inequality inside a group, we know the color of the ball we are comparing in that group, hence the color of all balls in that group. Therefore, in each group, we can have at most one inequality between two unknown balls. We consider the two cases $I_{UU} = 0$ and $I_{UU} = 1$ and solve the problem in these cases.

Algorithm 2.3.1 2-color majority for $n = 2^k$

```
1: Group the balls in 3 groups of equal sizes
2: I_UU ← 0
3: while I_UU ≤ 1 and in each group there is at least one ball left to compare do
4:     In each group, compare one ball with all others
5:     If there is an inequality between two homogeneous components: I_UU ← I_UU + 1
6: end while
7: switch I_UU do
8:     case 0
9:         Compare the two homogeneous components of the same size
10:        if they are equal then Return their color as majority
11:        else Return the color of the third homogeneous component as majority end if
12:    case 1
13:        Compare the two homogeneous components
14:        if they are equal then Return their color as majority
15:        else Count the balls of each color and determine the majority end if
16:    case 2
17:        Compare one known ball with all remaining unknown balls
18:        Count the balls of each color and determine the majority
```

2.3.2 Correctness and Complexity

Depending on the remainder of 2^k when divided by 3, we can either group the balls into three groups of sizes $\lceil \frac{n}{3} \rceil$, $\lceil \frac{n}{3} \rceil$ and $\lfloor \frac{n}{3} \rfloor$ or of sizes $\lfloor \frac{n}{3} \rfloor$, $\lfloor \frac{n}{3} \rfloor$ and $\lceil \frac{n}{3} \rceil$; note in both cases there are exactly two groups with the same size. In each group, we compare one ball with all the other balls in that group.

Lemma 2.3.2. *In each group, we can have at most one comparison between two balls of unknown color that results in an inequality.*

Proof. After the first inequality, the color of the ball that is compared to all other balls is determined. Therefore, further comparisons, if any, are between a known and an unknown

ball and I_{UU} remains unchanged. □

We count the number of these inequalities and, as before, call it I_{UU}. If I_{UU} reaches 2, we stop comparing within groups and start to compare one known ball with all the remaining unknown balls. We prove the correctness in all three possible cases:

1. $I_{UU} = 0$

   If there is no comparison resulting in inequality in any group, the result of all comparisons within each group was equality. Therefore, we have three components in each of which all balls have the same color. We compare two balls from the components of the same size. If they are equal, it means that we have a component of size at least $\frac{2n-2}{3}$ which is more than $\frac{n}{2}$, hence we found the color of majority. By Lemma 2.1.1, the total number of comparisons is

   $$n - m - I_{UU} = n - 2 - 0 = n - 2 = n - B(n) - 1.$$

   Otherwise, if the result is an inequality, it means they have different colors and because they have the same size, we can discard them. Thus, in this case, the remaining component is the majority. $I_{UU} = 1$ and by Lemma 2.1.1, the total number of comparisons is

   $$n - m - I_{UU} = n - 1 - 1 = n - 2 = n - B(n) - 1.$$

2. $I_{UU} = 1$

   In this case, we know two groups are homogeneous and we determined colors of balls in the third group in which the inequality happened. We compare two balls from the two homogeneous groups. If they are equal, their size will be at least $\frac{n-1}{3} + \frac{n-1}{3} > \frac{n}{2}$ and thus we found the majority. In this case, by Lemma 2.1.1, the number of comparisons is

   $$n - m - I_{UU} = n - 1 - 1 = n - 2 = n - B(n) - 1$$

   If they are not equal, we know the colors of all balls so we can count the number of occurrences of each ball and determine the majority color (if it exists). The number of comparisons is still

   $$n - m - I_{UU} = n - 0 - 2 = n - 2 = n - B(n) - 1$$

3. $I_{UU} \geq 2$

   By Lemma 2.3.2, we know that the inequalities happened in two different groups. We stop comparing and compare one ball of known color with all remaining unknown balls. We know the color of each ball after the algorithm and can determine the majority. By Lemma 2.1.1, the total number of comparisons is

$$n - m - I_{UU} = n - 0 - 2 = n - 2 = n - B(n) - 1$$

2.3.3 Upper bound for $F_2 = \{n = 2^k + 1 \mid k \in \mathbb{N}, k \geq 3\}$

With the same approach as Algorithm 2.3.1, we can force the opponent to give us at most two inequalities between two homogeneous components. This time, we group the balls in five groups of sizes $2^{k-2} - 1$, $2^{k-2} - 1$, $2^{k-2} + 1$, $2^{k-2} + 1$ and 1. We compare one ball with all the other balls in a group. If there are three inequalities between homogeneous components, we stop and compare one ball with all the other unknown balls. This way, $n - m - I_{UU} = n - 0 - 3$ comparisons are made, which is one comparison better than the tight bound given by the equality-test model. Three other cases happen: $I_{UU} = 0, 1, 2$. We solve the problem for each case separately, knowing that $I_{UU} = 0$ means five homogeneous components of the sizes of groups, $I_{UU} = 1$ means we know the color of all the balls in one group (this can be of size $2^{k-2} - 1$ or $2^{k-2} + 1$) and all the other four groups are homogeneous components and $I_{UU} = 2$ means we have two homogeneous components (this can be any two from the four groups of sizes more than 1) and two groups of known color balls.

Algorithm 2.3.2 2-color majority for $n = 2^k + 1$

Group the balls in five groups of sizes $2^{k-2} - 1$, $2^{k-2} - 1$, $2^{k-2} + 1$, $2^{k-2} + 1$ and 1.
$I_{UU} \leftarrow 0$
while $I_{UU} \leq 2$ and in each group there is at least one ball left to compare **do**
    In each group, compare one ball with all others
    If there is an inequality between two homogeneous components: $I_{UU} \leftarrow I_{UU} + 1$
end while
switch I_{UU} **do**
    case 0
        Compare homogeneous components $2^{k-2} + 1$ and $2^{k-2} + 1$
        if they are equal **then** Return as majority
        else Compare homogeneous components $2^{k-2} - 1$ and $2^{k-2} - 1$
            if they are equal **then** Return as majority
            else Return the color of the group of size 1 as the majority **end if end if**
    case 1
        if inequality was in component $2^{k-2} - 1$ **then** Compare homogeneous components $2^{k-2} + 1$ and $2^{k-2} + 1$
            if they are equal **then** Return as majority
            else Compare a known ball with a ball in the homogeneous component of size $2^{k-2} - 1$
                Count the balls of each color and determine the majority **end if**
        else Compare homogeneous components $2^{k-2} - 1$ and $2^{k-2} + 1$
            if they are equal **then** Compare remaining homogeneous components
                if they are equal **then** Return as majority
                else Count the balls of each color and determine the majority **end if**
            else find the color of the last component
                Count the balls of each color and determine the majority **end if**
    case 2
        Compare the two homogeneous components
        if they are equal **then** Return as majority
        else Count each color and determine the majority **end if**
    case 3
        Compare one known ball with all remaining unknown balls
        Count the balls of each color and determine the majority

2.3.4 Correctness and Complexity

We divide the balls into five groups of sizes $2^{k-2}-1$, $2^{k-2}-1$, $2^{k-2}+1$, $2^{k-2}+1$ and 1. Similar to the proposed algorithm for F_1, we compare one ball in each group with all other balls and compute I_{UU}. This time, we stop the algorithm when $I_{UU} \geq 3$ and compare one known ball with the unknown remaining balls.

Observation 2.3.3. If there is an equal number of two colors in the first four groups, the last ball is the majority. Otherwise, as the total number of balls is even, the number of balls of the majority color in the first four groups is at least 2 more than the minority. Therefore, the last ball does not need to be checked.

Four cases might happen:

1. $I_{UU} = 0$

   In this case, all groups are homogeneous components. We compare the two components of sizes $2^{k-2}+1$ and $2^{k-2}+1$. If they are equal, they are the majority because the sum is $2^{k-2}+1+2^{k-2}+1 = 2^{k-1}+2 > \frac{n}{2}$. By Lemma 2.1.1, the number of comparisons is

   $$n - m - I_{UU} = n - 3 = n - B(n) - 1$$

   If they are not equal, we can discard them as they have the same size. We then compare components of sizes $2^{k-2}-1$ and $2^{k-2}-1$ together. If these two components are equal, announce them as the majority because they are more than half of the remaining balls. The number of comparisons is

   $$n - m - I_{UU} = n - 2 - 1 = n - 3 = n - B(n) - 1.$$

   If these two components are not equal, we can discard them because they have the same size; consequently, the last component of size 1 is the majority color. In this case, the number of comparisons is

   $$n - m - I_{UU} = n - 1 - 2 = n - 3 = n - B(n) - 1.$$

2. $I_{UU} = 1$

   If the inequality is in one of the groups of size $2^{k-2}-1$, compare two components of size $2^{k-2}+1$. If they are equal, they are the majority color, and the number of comparisons we made is

   $$n - m - I_{UU} = n - 3 - 1 = n - 4 = n - B(n) - 2.$$

If they are not, discard these two components. Then compare one ball from the homogeneous component of size $2^{k-2}-1$ with one ball with known color. Colors of all balls are determined in this way and the number of comparisons is

$$n - m - I_{UU} = n - 1 - 2 = n - 3 = n - B(n) - 1.$$

If the one inequality occurred in a component of size $2^{k-2}+1$, compare a component of size $2^{k-2}-1$ with the homogeneous component of size $2^{k-2}+1$. If they are not equal, compare the remaining $2^{k-2}-1$ component with a known ball and all colors are determined with the following number of comparisons:

$$n - m - I_{UU} = n - 1 - 2 = n - 3 = n - B(n) - 1.$$

If they are equal, compare one ball from them with the component $2^{k-2}-1$. If they are equal, they are the majority and

$$n - m - I_{UU} = n - 2 - 1 = n - 3 = n - B(n) - 1.$$

If they are not, we know all colors except the last one, which does not to be checked by Observation 2.3.3. The number of comparisons is

$$n - m - I_{UU} = n - 1 - 2 = n - 3 = n - B(n) - 1.$$

3. $I_{UU} = 2$

   By Lemma 2.3.2, the two inequalities happen in two different groups and we know the colors of all balls in them. By comparing a ball with known color with two homogeneous groups we can determine the colors of the first four groups and the majority can be found by Observation 2.3.3. Total number of comparisons is

   $$n - m - I_{UU} = n - 1 - 2 = n - 3 = n - B(n) - 1.$$

4. $I_{UU} \geq 3$

   If that happens, we stop comparing within groups and compare one ball with known color with all the remaining unknown balls except the last ball in the fifth group (Observation 2.3.3). By Lemma 2.1.1, we used

   $$n - m - I_{UU} = n - 3 - 0 = n - 3 = n - B(n) - 1$$

   comparisons in total.

Chapter 3

Randomized 2-color Majority

In the previous chapter, we achieved a lower bound for deterministic algorithms for 2-color majority problem, and we showed three-way cannot decrease the number of comparisons more than $o(n)$. It is then reasonable to approach the same problem with randomized methods to see whether three-way can make a difference. Previously, Alonso et al. [3] used martingales to show that the expected number of comparisons that any randomized algorithm makes for solving 2-color majority in the equality-test model is $\frac{2n}{3} + o(n)$. In this chapter, we propose a Las Vegas algorithm for this problem which can be applied in the equality-test comparison model as well and has expected number of comparisons $g(\alpha)n + o(n)$, where α is the fraction of the majority color in the input and g is a function defined momentarily that takes values $\in \left[\frac{1}{2}, \frac{2}{3}\right]$ for $\alpha \in \left[\frac{1}{2}, 1\right]$. Meeting the lower bound, this algorithm also provides a parameterized cost for this problem. In contrast with previous results, the proof given here is completely elementary: as with the method of Boyer and Moore [7], we use the fact that if we find a group of $2r$ balls, with r of one color and r of the other we can discard those elements. Then, starting from the initial input, we estimate the number of comparisons in each step using an estimated number of comparisons in the previous step, until we reach the terminal state. We then use Chebyshev's inequality to show the propagated error in estimating these numbers is small with high probability. We first formally state the theorem and algorithm. Next, we explain each step of it in detail, and finally prove its correctness and compute the complexity.

Theorem 3.0.1. *Let $\alpha \in \left[\frac{1}{2}, 1\right]$ be the fraction of the majority color. Then, there is a Las Vegas algorithm which solves 2-color majority problem with at most $g(\alpha)n + o(n)$ comparisons for input size of n with high probability, where $g(\alpha)$ is defined as below and is*

always between $\frac{1}{2}$ *and* $\frac{2}{3}$.

$$g(\alpha) = \frac{2\alpha - 1}{4\alpha} \sum_{k=0}^{m} \frac{1}{2^k} \frac{\alpha^{2^k} + (1-\alpha)^{2^k}}{\alpha^{2^k} - (1-\alpha)^{2^k}}$$

Corollary 3.0.1.1. *There is a Las Vegas algorithm which solves* 2*-color majority problem making at most* $\frac{2n}{3} + o(n)$ *comparisons with high probability.*

The graph of function g is depicted in Figure 3.

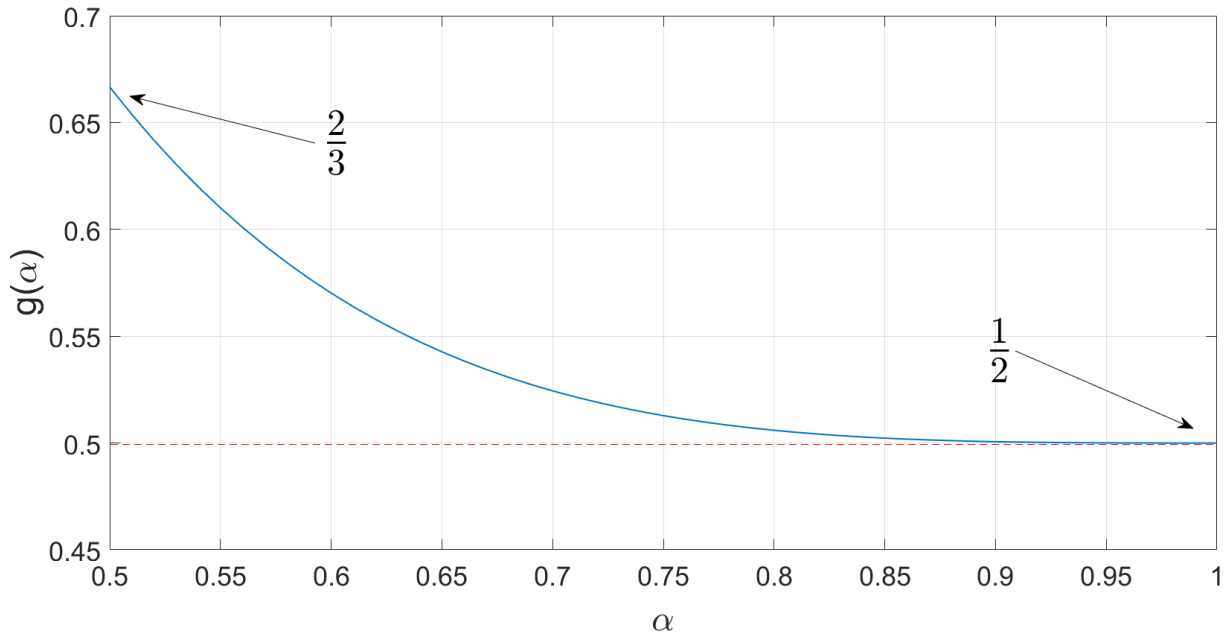

Figure 3.1: Coefficient of n in the expected number of comparisons

3.1 Proposed Algorithm

For a given coloring of n balls consisting of b black and w white balls, let $\alpha = \frac{\max(b,w)}{n} \geq \frac{1}{2}$ be the fraction of the most frequent color. An accurate estimation of α, $\hat{\alpha}$ can be obtained by sampling a small set n_s of size $o(n)$ from the original set. Colors in this set can be

determined by comparing one ball with all others (naïve algorithm). The key idea of the algorithm is the expected number of balls of the majority color in a sample of size $n_0 = \frac{n}{2\alpha}$ balls from the original set is $\frac{n}{2}$. Roughly speaking, by assuming a symmetric distribution, with a probability of 0.5, there are at least $\frac{n}{2}$ balls of the majority color in this sample, which is enough balls to conclude this color is certainly the majority in the input. To increase the probability of finding the majority in a sample, one can increase the size of the sample n_0 by a factor of $1 + \lambda$. With the same argument, this time the expected number of balls of the majority color in our sample is $\alpha \times \frac{n}{2\alpha} \times (1 + \lambda) = (1 + \lambda)\frac{n}{2}$; therefore, any sample in which the number of balls of majority color is in interval $\left[\frac{n}{2}, (1 + \lambda)\frac{n}{2}\right]$ is desirable. As a result of concentration inequalities such as Chebyshev [23], the probability of having such a sample is very high. We are interested in decreasing the total number of comparisons and therefore smaller λ is desired. However, in the analysis, we show by decreasing λ we need to decrease the error in our estimation of the fraction of the majority color, which, in turn, requires a higher number of balls to be sampled. This trade-off is addressed by suggesting certain values for these variables in terms of n so that $(1 + \lambda)\frac{n}{2\alpha} = \frac{n}{2\alpha} + n^{\nu}$ for some $\nu < 1$ - asymptotically compensating for the cost induced by adding $(1 + \lambda)$.

Pairing in each step happens only between groups of equal size; we pair two balls from two different components, compare the two balls in each pair, merge the two components if the balls are of the same color and discard them if they are not. Consequently, the size of the components is always a power of 2. During each round, if there is an odd number of components, we put one of the components aside and continue with the others. In the end, sizes of the remaining components are distinct powers of 2; otherwise, the pairing could have been done. Consequently, the number of remaining components is at most $\lg(n)$, and determining the majority in them takes at most $\lg(n) \in o(n)$ comparisons, which does not asymptotically change the total number of comparisons which is in $\Theta(n)$ and depends on the number of pairings during previous steps.

Algorithm 3.1.1 Randomized 2-color majority

Randomly sample $n_s \in o(n)$ balls.
Count the number of balls of each color b and w
Define $\hat{\alpha} = \frac{\max(b,w)}{n_s}$.
Randomly sample $n_0 = (1+\lambda)\frac{n}{2\hat{\alpha}}$ balls for a $\lambda > 0$
FOLDED-PAIRING(n_0)
if majority found **then**
    Return the majority color
else
    Switch to the naïve algorithm
end if

Algorithm 3.1.2 Folded pairing

procedure FOLDED-PAIRING(n_0)
    while more than one ball remains **do**
        Pair the balls in n_0 randomly
        Compare the two balls in each pair
        if they are equal **then**
            discard one of the balls
            take the other ball to the next round as a representative of the two balls
        else
            discard both of them
            keep the number of balls of each color which is 2^i balls for round i
        end if
        if $|n_0|$ is odd **then**
            Keep the number of the unpaired ball which is a representative of 2^i balls
        end if
        Update n_0 with the balls we take to the next round
    end while
    Compare the possible remaining ball and all unpaired balls with each other
    Determine if there is a majority
end procedure

3.2 Correctness

In the i^{th} round of the Algorithm 3.1.2, each ball is a representative of 2^{i-1} balls of the same color. When we discard a heterogeneous pair, we discard exactly 2^{i-1} balls of each color. Keeping track of the number of balls of each color, the algorithm ends when there are at least $\frac{n}{2}$ balls of the same color in the sample of $n_0 = (1+\lambda)\frac{n}{2\hat{\alpha}}$ balls, which is clearly the majority color in the whole input set as well. If the algorithm does not find the $\frac{n}{2}$ balls of the same color among these n_0 balls, it switches to the naïve algorithm and compares a ball with known color with all other balls, until it finds a majority or there are no other balls. Note that there is always at least one inequality in this case, as otherwise all the $\frac{n}{2\alpha} \geq \frac{n}{2}$ were of the same color and algorithm would not switch to the naïve algorithm.

3.3 Comparison Complexity

In this section, a formal, step-by-step analysis of the complexity of the algorithm proposed in the first part of the chapter is given. To compute the expected number of comparisons, we need to calculate p_s, the probability of an early-success - finding the majority in the small sample of $\frac{n}{2\hat{\alpha}}(1+\lambda)$ balls. Then, we can compute the average cost of our Las Vegas algorithm as

$$p_s \times \mathbb{E}[\text{cost given early-success}] + (1-p_s) \times \text{cost given naïve algorithm}$$

3.3.1 Estimating α

As a consequence of the *law of large numbers* [20], the fraction of the majority color in the input, α can be estimated by sampling large enough number of balls. Formally, we have the following theorem from [22]

Theorem 3.3.1 (Estimating α)**.** *By sampling $n_s \geq \frac{2+\epsilon}{\epsilon^2}\ln(\frac{2}{\delta})$ balls, estimated $\hat{\alpha}$ is within $[\alpha-\epsilon, \alpha+\epsilon]$ with probability $1-\delta$.*

The idea here is, by having an accurate estimation of α, namely $\hat{\alpha}$, the number of balls of majority color in a sample of n_0 balls should be approximately $n_0\hat{\alpha}$. We need to have at least $\frac{n}{2}$ balls from the majority color, thus we should take n_0 to be at least $\frac{n}{2\hat{\alpha}}$. Roughly speaking, in half of the cases, the number of majority balls in such sample is less than the expected, which is $\frac{n}{2}\frac{\alpha}{\hat{\alpha}}$, or approximately $\frac{n}{2}$. Therefore, by choosing $n_0 = \frac{n}{2\hat{\alpha}}$ we cannot

decide in half of the cases when the sample average falls below the expected. It motivates us to let n_0 be $(1+\lambda)(\frac{n}{2\hat{\alpha}})$, where the value of λ can regulate the trade-off between solving the problem in the sample of n_0 balls and switching to naïve algorithm.

3.3.2 Creating Sample of Size n_0

Theorem 3.3.2. *For $\lambda = 2.01n^{-\frac{1}{3}}$ and $\epsilon = n^{-\frac{1}{3}}$, the probability of having more than $\frac{n}{2}$ balls of majority color in a sample of $(1+\lambda)\frac{n}{2\hat{\alpha}}$ balls goes to 1 as n goes to infinity.*

Proof. We are going to compute the probability of having at least $\frac{n}{2}$ balls of majority color in a sample of size $n_0 = (1+\lambda)(\frac{n}{2\hat{\alpha}})$. For $1 \leq i \leq n_0$, let T_i denote the indicator random variable of the i^{th} sample, which is 1 if the i^{th} ball is of the majority color, and otherwise is 0. As $n_0 \in o(n)$, the sampling process can be estimated by sampling with replacement, therefore T_i follows a Bernoulli distribution with parameter α. As a result of linearity of expectation, the random variable $T = T_1 + \cdots + T_{n_0}$ indicating the number of balls of majority color has a mean of $n_0\alpha$ and a variance of $n_0\alpha(1-\alpha)$. The last statement follows from the fact that

$$Var(T_i) = \alpha - \alpha^2$$

As a well-known result about approximation of Bernoulli distributions [6], T can be transformed into a standard normal distribution ($\mu = 0, \sigma^2 = 1$):

$$Z = \frac{T - n_0\alpha}{\sqrt{n_0\alpha(1-\alpha)}},$$

which shows

$$\Pr[T \geq \frac{n}{2}] = \Pr[Z \geq \frac{\frac{n}{2} - n_0\alpha}{\sqrt{n_0\alpha(1-\alpha)}}].$$

What we need to show is $\frac{\frac{n}{2} - n_0\alpha}{\sqrt{n_0\alpha(1-\alpha)}}$ goes to negative infinity as n grows. After substituting $n_0 = (1+\lambda)(\frac{n}{2\hat{\alpha}})$, the right hand side becomes

$$\frac{\frac{n}{2} - n_0\alpha}{\sqrt{n_0\alpha(1-\alpha)}} = \sqrt{\frac{n}{2}}\left(\frac{1 - \frac{\alpha}{\hat{\alpha}}(1+\lambda)}{\sqrt{(1+\lambda)\frac{\alpha}{\hat{\alpha}}(1-\alpha)}}\right).$$

We are going show the coefficient of $\sqrt{\frac{n}{2}}$ in the above expression is a negative number tending to zero asymptotically slower than $\frac{1}{\sqrt{n}}$, which implies the expression tends to

negative infinity as desired. We show this by giving a lower bound on the absolute value of nominator and an upper bound on the denominator of $\frac{1-\frac{\alpha}{\hat{\alpha}}(1+\lambda)}{\sqrt{(1+\lambda)\frac{\alpha}{\hat{\alpha}}(1-\alpha)}}$. From the results of the previous section on the accuracy of our estimation of α, we know $\frac{\alpha}{\alpha+\epsilon} \leq \frac{\alpha}{\hat{\alpha}} \leq \frac{\alpha}{\alpha-\epsilon}$ with probability of $1-\delta$ and thus

$$\frac{1-\frac{\alpha}{\hat{\alpha}}(1+\lambda)}{\sqrt{(1+\lambda)\frac{\alpha}{\hat{\alpha}}(1-\alpha)}} \leq \frac{1-\frac{\alpha}{\alpha+\epsilon}(1+\lambda)}{\sqrt{(1+\lambda)\frac{\alpha}{\hat{\alpha}}(1-\alpha)}}.$$

By the definition in the theorem we have $\epsilon < \frac{\lambda}{2}$ which is at most $\alpha\lambda$, therefore

$$1-\frac{\alpha}{\alpha+\epsilon}(1+\lambda) < 1-\frac{\alpha}{\alpha+\frac{\lambda}{2}}(1+\lambda) \leq 1-\frac{\alpha}{\alpha+\alpha\lambda}(1+\lambda) = 0$$

which is true because $\alpha \geq \frac{1}{2}$; therefore, the coefficient is negative. As long as this number goes to zero slower than $\frac{1}{\sqrt{\frac{n}{2}}}$ we can be sure that $\Pr[T \geq \frac{n}{2}]$ is close to 1. To show this, note that

$$\left|1-\frac{\alpha}{\hat{\alpha}}(1+\lambda)\right| \geq \left|1-\frac{\alpha}{\alpha+\epsilon}(1+\lambda)\right| \geq \left|1-\frac{\frac{1}{2}}{\frac{1}{2}+\epsilon}(1+\lambda)\right| = \frac{\lambda-2\epsilon}{1+2\epsilon}$$

For the direction of inequality, we used the fact that numbers in the absolute function are negative. We also know the maximum value of $\alpha(1-\alpha)$ is $\frac{1}{4}$ and $\hat{\alpha}$ is at least $\frac{1}{2}-\epsilon$ with probability $1-\delta$, therefore

$$\sqrt{(1+\lambda)\frac{\alpha}{\hat{\alpha}}(1-\alpha)} \leq \sqrt{\frac{1+\lambda}{4\hat{\alpha}}} \leq \sqrt{\frac{1+\lambda}{4(\frac{1}{2}-\epsilon)}}$$

Combining these results together, we get

$$\left|\frac{1-\frac{\alpha}{\hat{\alpha}}(1+\lambda)}{\sqrt{(1+\lambda)\frac{\alpha}{\hat{\alpha}}(1-\alpha)}}\right| \geq \frac{\frac{\lambda-2\epsilon}{1+2\epsilon}}{\sqrt{\frac{1+\lambda}{4(\frac{1}{2}-\epsilon)}}}$$

Substituting $\lambda = 2.01\epsilon$ first to remove λ and then $\epsilon = n^{-\frac{1}{3}}$, the right hand side of the above expression is equal to

$$\begin{aligned}
&= \frac{\frac{0.01\epsilon}{1+2\epsilon}}{\sqrt{\frac{1+2.01\epsilon}{(2-4\epsilon)}}} \\
&= \frac{\sqrt{2}}{100}\sqrt{\frac{2-\epsilon}{1+2.01\epsilon}}\epsilon \in \Theta(\epsilon) \subseteq \Theta(n^{-\frac{1}{3}}),
\end{aligned}$$

The latter asymptotic bound is true as $2-\epsilon$ and $1+2.01\epsilon$ tend to 2 increasingly and 1 decreasingly, respectively, as ϵ goes to zero in the sequence $\epsilon = \epsilon_n = n^{-\frac{1}{3}}$. Thus the function $f(\epsilon) = \sqrt{\frac{2-\epsilon}{1+2.01\epsilon}}$ is a monotonous function bounded between $f(1) = \sqrt{\frac{1}{3.01}}$ and $f(0) = \sqrt{2}$.

We showed the coefficient of $\sqrt{\frac{n}{2}}$ is a negative number whose absolute value tends to zero as fast as $n^{-\frac{1}{3}}$ does, which is slower than $\frac{1}{\sqrt{n}}$ and we are done as we proved

$$\frac{\frac{n}{2} - n_0\alpha}{\sqrt{n_0\alpha(1-\alpha)}}$$

is a negative number with absolute value in $\Theta(\sqrt{n}n^{-\frac{1}{3}}) = \Theta(n^{\frac{1}{6}})$ hence

$$\Pr[T \geq \frac{n}{2}] = \Pr[Z \geq \frac{\frac{n}{2} - n_0\alpha}{\sqrt{n_0\alpha(1-\alpha)}}]$$

tends to 1. □

Corollary 3.3.2.1. *λ and ϵ defined in Theorem 3.3.2 results in a sample of size $n_0 = (1+\lambda)(\frac{n}{2\hat{\alpha}}) = \frac{n}{2\hat{\alpha}} + o(n)$, and $n_s \in \Theta(n^{-\frac{1}{3}})$.*

3.3.3 Expected Number of Heterogeneous Pairs

Each round of the algorithm has an instance of the *pairing problem*, which is defined and analyzed below for the case when the number of balls in a round is even. The same probability for odd n can be computed in similar manner, but as the below formula is only intended to demonstrate the nature of the problem and not being used in the analyses, we do not mention it for the sake of simplicity.

Lemma 3.3.3. *(Pairing Problem) A set of b black balls and w white balls are paired up randomly, where $n = b + w$ is even. If $b \leq w$ then for a fixed k, $0 \leq k \leq b$, the probability of having k heterogeneous pairs is*

$$P_{b,w}(k) = \frac{\binom{b}{k}\binom{w}{k}k!\frac{(b-k)!(w-k)!}{(\frac{b-k}{2})!(\frac{w-k}{2})!2^{\frac{b-k}{2}}2^{\frac{w-k}{2}}}}{\frac{n!}{(\frac{n}{2})!2^{\frac{n}{2}}}}.$$

Proof. There are $\binom{b}{k}$ and $\binom{w}{k}$ ways to choose the k black balls and k white balls which are paired with a ball of opposite color. After choosing these $2k$ balls, there are $k!$ ways to pair them up. Other balls need to be paired within their color; for the remaining $b-k$ black balls, there are $\binom{b-k}{2}$ ways for choosing the first pair, $\binom{b-k-2}{2}$ ways for choosing the second pair, $\binom{b-k-4}{2}$ ways for choosing the third pair, and so on. These $b-k$ pairs can be chosen in any order, so each pairing is achieved by $(\frac{b-k}{2})!$ different combinations. Therefore, the number of ways to pair black balls is

$$\begin{aligned}
&\binom{b-k}{2}\binom{b-k-2}{2}\binom{b-k-4}{2}\cdots\binom{2}{2}\frac{1}{(\frac{b-k}{2})!} \\
&= \frac{(b-k)(b-k-1)}{2}\frac{(b-k-2)(b-k-3)}{2}\cdots\frac{(2)(1)}{2}\frac{1}{(\frac{b-k}{2})!} \\
&= \frac{(b-k)!}{(\frac{b-k}{2})!2^{\frac{b-k}{2}}}.
\end{aligned}$$

With the same argument, the number of ways for pairing white balls is $\frac{(w-k)!}{(\frac{w-k}{2})!2^{\frac{w-k}{2}}}$. In a similar way, the number of ways of pairing all balls is

$$\binom{n}{2}\binom{n-2}{2}\binom{n-4}{2}\cdots\binom{2}{2}\frac{1}{(\frac{n}{2})!} = \frac{n!}{(\frac{n}{2})!2^{\frac{n}{2}}},$$

which proves the lemma. □

Let $M_{b,w}$ be the expected number of heterogeneous pairs calculated by $\sum_{k=0}^{b} kP_{b,w}(k)$. Although calculating expectation via the above formula seems to need an enormous effort, we easily derive the explicit form of the expected value and variance in the next two lemmas with the aid of auxiliary random variables.

Lemma 3.3.4. *Given the condition in Lemma 3.3.3,* $M_{b,w} = \frac{bw}{b+w-1} = \alpha(1-\alpha)\frac{n_0^2}{n_0-1}$

Proof. If X_i denotes the indicator random variable which shows the i^{th} black ball is paired with a white one, then the number of heterogeneous pairs is $H = X_1 + X_2 + \cdots + X_b$. From the linearity of expectation, we have

$$\begin{aligned}
M_{b,w} &= \mathbb{E}[H] \\
&= \mathbb{E}[X_1 + X_2 + \cdots + X_b] \\
&= \mathbb{E}[X_1] + \mathbb{E}[X_2] + \cdots + \mathbb{E}[X_b]
\end{aligned}$$

For the i^{th} black ball X_i,

$$\mathbb{E}\left[X_i\right] = \mathbb{P}\left[X_i = 1\right] = \frac{w}{b+w-1}$$

because there are w white balls among the total $b+w-1$ potential balls to be paired with this ball. To complete the proof, note that by the definition of α we have $b+w = n_0$ and $bw = \alpha n_0(1-\alpha)n_0$, thus the identity $\frac{bw}{b+w-1} = \alpha(1-\alpha)\frac{n_0^2}{n_0-1}$ holds. □

Lemma 3.3.5. *The variance of the number of heterogeneous pairs is* $Var(H) = 2\frac{bw(b-1)(w-1)}{(n_0-1)^2(n_0-3)} \approx 2\alpha^2(1-\alpha)^2 n_0$

Proof. With the same notation as in Lemma 3.3.4, to compute expectation and variance of $H = X_1 + \cdots + X_b$ we use the below identity

$$\begin{aligned} Var(H) &= Var(X_1 + \cdots + X_b) \\ &= \sum_{i=1}^{b} Var\left(X_i\right) + \sum_{i \neq j} Cov\left(X_i X_j\right). \end{aligned}$$

For each i, we showed in Lemma 3.3.4 that $\mathbb{E}[X_i] = \frac{w}{b+w-1}$. Similar reasoning shows $\mathbb{E}[X_i X_j] = \frac{w}{b+w-1} \cdot \frac{w-1}{b-1+w-1-1}$ because after pairing the first black ball with one of the white balls, these two balls are eliminated from the set of balls and there are $b-1$ black and $w-1$ white balls. The rest can be done by calculating $Var(H)$ using the above formula for $\mathbb{E}[X_i X_j]$ and the following for $Var\left(X_i\right)$

$$\begin{aligned} &Var(H) = Var(X_1 + \cdots + X_b) \\ &= \sum_{i=1}^{b} Var\left(X_i\right) + \sum_{i \neq j} Cov\left(X_i X_j\right) \\ &= \sum_{i=1}^{b} \mathbb{E}[X_i^2] - \mathbb{E}[X_i]^2 + \sum_{i \neq j} \mathbb{E}[X_i X_j] - \mathbb{E}[X_i]\mathbb{E}[X_j] \\ &= \sum_{i=1}^{b} \frac{w}{b+w-1} - \left(\frac{w}{b+w-1}\right)^2 + \sum_{i \neq j} \frac{w}{b+w-1} \cdot \frac{w-1}{b-1+w-1-1} - \sum_{i \neq j} \left(\frac{w}{b+w-1}\right)^2 \\ &= b\left(\frac{w}{b+w-1} - \left(\frac{w}{b+w-1}\right)^2\right) + 2\binom{b}{2}\left(\frac{w}{b+w-1} \cdot \frac{w-1}{b+w-3} - \left(\frac{w}{b+w-1}\right)^2\right) \end{aligned}$$

$$
\begin{aligned}
&= \frac{bw}{b+w-1} - (b+b^2-b)\left(\frac{w}{b+w-1}\right)^2 + \frac{bw(b-1)(w-1)}{(b+w-1)(b+w-3)} \\
&= \frac{bw}{b+w-1}\left(1 - \frac{bw}{b+w-1} + \frac{(b-1)(w-1)}{b+w-3}\right) \\
&= \frac{bw}{b+w-1}\left(\frac{-(b-1)(w-1)}{b+w-1} + \frac{(b-1)(w-1)}{b+w-3}\right) \\
&= \frac{2bw(b-1)(w-1)}{(b+w-1)^2(b+w-3)} \\
&= \frac{2b^2w^2(1-\frac{1}{b})(1-\frac{1}{w})}{(b+w)^3(1-\frac{1}{b+w})^2(1-\frac{3}{b+w})} \\
&= \frac{2b^2w^2}{(b+w)^3}\frac{(1-\frac{1}{b})(1-\frac{1}{w})}{(1-\frac{1}{b+w})^2(1-\frac{3}{b+w})}
\end{aligned}
$$

Substituting b and w by αn_0 and $(1-\alpha)n_0$, respectively, to obtain the following

$$
2\alpha^2(1-\alpha)^2 n_0 \frac{(1-\frac{1}{\alpha n_0})(1-\frac{1}{(1-\alpha)n_0})}{(1-\frac{1}{n_0})^2(1-\frac{3}{n_0})}
$$

When occurrences of both colors are large enough, the second fraction is approximately 1 and therefore the whole expression is equal to $2\alpha^2(1-\alpha)^2 n_0$. □

The following lemma shows how by having the number of heterogeneous pairs at each round other parameters can be derived and do not need be analyzed separately.

Lemma 3.3.6. *Suppose we start with $n_0 = w_0 + b_0$ balls with w_0 white and b_0 black balls. If n_i, w_i, b_i, and H_i denote the total number of balls, white balls, black balls, and number of heterogeneous pairs on the i^{th} call of Algorithm 3.1.2 then the following recurrence relationships hold:*

$$
\begin{aligned}
n_k &= \frac{n_0 - 2\sum_{j=0}^{k-1} 2^j H_j}{2^k}, \\
w_k &= \frac{w_0 - \sum_{j=0}^{k-1} 2^j H_j}{2^k}, \\
b_k &= \frac{b_0 - \sum_{j=0}^{k-1} 2^j H_j}{2^k}
\end{aligned}
$$

Proof. In each round, the number of balls is halved because we only keep a representative from each component. At the same time, each heterogeneous pair deducts a number of balls from both colors. For instance, n_1 is equal to the number of pairs at the first level ($\frac{n_0}{2}$) minus the number of heterogeneous pairs (H_0). With the same reasoning, $n_k = \frac{n_{k-1}}{2} - H_{k-1}$, and the claim follows from an induction. □

Lemma 3.3.7. *m, the total number of rounds of pairing balls before at least one set becomes empty, is not more than* $\log n_0$.

Proof. It follows from Lemma 3.3.6 that $n_k \leq \frac{n_0}{2^k}$, therefore k is at most $\log n_0$. □

Lemma 3.3.8. *The expected number of heterogeneous pairs H_i, the size of n_{i+1}, and α_{i+1} are:*

$$\mathbb{E}[H_i] = \alpha_i(1-\alpha_i)n_i$$
$$\mathbb{E}[n_{i+1}] = n_i\left(\frac{1}{2} - \alpha_i(1-\alpha_i)\right)$$
$$\mathbb{E}[\alpha_{i+1}] = \frac{\alpha_i^2}{2\alpha_i^2 - 2\alpha_i + 1}.$$

Proof. To have a heterogeneous pair, we need to have a ball from the majority color and a ball from the other color. The probability of having a ball of the majority color is α_i and the probability of having a ball of the other color is $(1-\alpha_i)$. Since we are sampling with replacement, the probability of having a heterogeneous pair is $2\alpha_i(1-\alpha_i)$. The 2 in the term comes from the fact that in a pair either the first element or the second one is the majority color. We have $\frac{n_i}{2}$ pairs in total, so the expected number of heterogeneous pairs is

$$\mathbb{E}[H_i] = 2\alpha_i(1-\alpha_i)\frac{n_i}{2} = \alpha_i(1-\alpha_i)n_i$$

After each round, we discard a ball from each pair. We also discard the other ball from the heterogeneous pairs, so n_{i+1} becomes $\frac{n_i}{2} - H_i$. Thus

$$\mathbb{E}[n_{i+1}] = \frac{n_i}{2} - H_i = \frac{n_i}{2} - \alpha_i(1-\alpha_i)n_i = n_i\left(\frac{1}{2} - \alpha_i(1-\alpha_i)\right).$$

The expected number of majority color balls in the sample of size n_i is $\alpha_i n_i$. In the next round for the sample size n_{i+1}, the expected number of majority color balls becomes

$\frac{\alpha_i n_i - H_i}{2}$, and α_{i+1} is the fraction of majority balls in the new set, hence

$$\begin{aligned}
&\mathbb{E}[\alpha_{i+1}] \\
&= \frac{\alpha_i n_i - H_i}{n_i - 2H_i} \\
&= \alpha_i + \frac{(2\alpha_i - 1)H_i}{n_i - 2H_i} \\
&= \alpha_i + \frac{(2\alpha_i - 1)\alpha_i(1-\alpha_i)n_i}{n_i - 2\alpha_i(1-\alpha_i)n_i} \\
&= \frac{\alpha_i^2}{2\alpha_i^2 - 2\alpha_i + 1}
\end{aligned}$$

□

Lemma 3.3.9. *If for $1 \geq \alpha_0 > \frac{1}{2}$ we define the next term $\alpha_{i+1} := \frac{\alpha_i^2}{2\alpha_i^2 - 2\alpha_i + 1}$ then $\{\alpha_k\}_{k\geq 0}$ is a strictly-increasing sequence, $\lim_{i\to\infty}\{\alpha_i\} = 1$, and $\alpha_k = \frac{1}{1+\left(\frac{1}{\alpha_0} - 2\alpha_i^2 - 2\alpha_i + 11\right)^{2^k}}$.*

Proof. Let $\beta_i = \frac{1}{\alpha_i}$ and re-write the relation as

$$\frac{1}{\beta_{i+1}} = \frac{\frac{1}{\beta_i^2}}{\frac{2}{\beta_i^2} - \frac{2}{\beta_i} + 1}$$

Simplifying the expression gives us

$$\beta_{i+1} = \beta_i^2 - 2\beta_i + 2,$$

therefore, to prove $\beta_{i+1} < \beta_i$, we need to show $\beta_i^2 - 2\beta_i + 2 < \beta_i$, which is equivalent to

$$(2 - \beta_i)(1 - \beta_i) < 0$$

which is true by assumption for $\beta_0 = \frac{1}{\alpha_0}$. Thus the sequence is strictly-decreasing.
To show the sequence is bounded, we use induction to prove $1 \leq \beta_i \leq 2$. This is true for $i = 0$ because $\alpha_0 \geq \frac{1}{2}$. If the hypothesis holds for i, we have $\beta_i - 2 \leq 0$, therefore

$$\begin{aligned}
\beta_{i+1} &= (\beta_i - 1)^2 + 1 \geq 1, \\
\beta_{i+1} &= \beta_i(\beta_i - 2) + 2 \leq 2.
\end{aligned}$$

Hence $\{\beta_i\}$ is strictly-decreasing, which means $\{\alpha_i\}$ is strictly-increasing. Now we know $\{\alpha_i\}$ is an increasing bounded sequence and thus has a limit L that must satisfy

$$L = \frac{L^2}{2L^2 - 2L + 1}$$

Solving the equation gives us

$$L = 1 \text{ or } \frac{1}{2}$$

Because we assumed α_0 is not $\frac{1}{2}$, the only possible solution is $L = 1$. It proves the second claim in the lemma. In fact the sequence $\{\alpha_i\}$ is a constant sequence if $\alpha_0 = \frac{1}{2}$.

By defining an auxiliary sequence $\lambda_i = \beta_i - 1$ we can re-write $\beta_{i+1} = \beta_i^2 - 2\beta_i + 2$ as

$$\lambda_{i+1} = \lambda_i^2.$$

For instance, $\lambda_3 = \lambda_2^2 = \lambda_1^4 = \lambda_0^8$. An easy induction shows the general solution to this recurrence relation is

$$\lambda_k = \lambda_0^{2^k} = (\beta_0 - 1)^{2^k}.$$

Substituting back β and then α, we have

$$\alpha_k = \frac{1}{1 + \left(\frac{1}{\alpha_0} - 1\right)^{2^k}}$$

□

We need an algebraic lemma about the length of the interval in which $\mathbb{E}[H_k]$ lies with high probability. Intuitively, the sum of lengths of these intervals bound the total error in estimating the number of heterogeneous pairs. Clearly, we want this value to be as low as possible, however, decreasing the length of each interval is equivalent to decreasing the chance of $\mathbb{E}[H_k]$ lies in that interval. The lemma addresses this trade-off.

Lemma 3.3.10. *For $0 \leq k \leq m$, where m is the maximum number of rounds in Algorithm 3.1.2 bounded by Lemma 3.3.7, there is a choice of $a_k \in o(n_0)$ such that the sum $a_0 + \cdots + a_m$ is also in $o(n_0)$.*

Proof. For a $1 > \phi > 0$, let $a_k = \sqrt{\frac{1}{8}}(\frac{n_0}{2^k})^{\frac{1+\phi}{2}}$, which is clearly in $o(n_0)$. Each a_k can be bounded by the inequality below:

$$a_k = \sqrt{\frac{1}{8}}(\frac{n_0}{2^k})^{\frac{1+\phi}{2}} < \sqrt{\frac{1}{8}} n_0^{\frac{1+\phi}{2}} \frac{1}{\sqrt{2}^k}.$$

As $\sum_{k\geq 0} \frac{1}{\sqrt{2}^k}$ is a finite constant by geometric series test, the sum is bounded by a constant times $n_0^{\frac{1+\phi}{2}}$, which is in $o(n_0)$. □

Theorem 3.3.11. *The probability of having an error more than a_k in estimating H_k by computing $\mathbb{E}[H_k]$ via Lemma 3.3.4 is at most $\left(\frac{n_0}{2^k}\right)^{-\phi}$.*

Proof. By Chebyshev's inequality [5], due to Lemma 3.3.5 the probability of making a *huge* mistake in estimation, as defined in Lemma 3.3.10, is bounded:

$$\Pr\left[|H_k - \mathbb{E}[H_k]| > a_k\right] < \frac{Var(H)}{a_k^2} = \frac{2\alpha^2(1-\alpha)^2 n_0}{\frac{1}{8}(\frac{n_0}{2^k})^{1+\phi}} \leq \left(\frac{n_0}{2^k}\right)^{-\phi}$$

□

The following theorem assures us that the actual number of the heterogeneous pairs at the i^{th} round is close to the one computed by estimating parameters of the j^{th} round from the $(j-1)^{th}$ round by Lemma 3.3.4 for $1 \leq j \leq i-1$.

Theorem 3.3.12. *(Folded-Expectation) Suppose we estimate $\mathbb{E}[H_k]$ by initial conditions i.e. b_0 and w_0 by iteratively computing $\mathbb{E}[H_j]$ for $1 \leq j \leq k-1$ using Lemma 3.3.4. If $\mathbb{E}[H_k|\mathbb{E}[H_{j<k}]]$ denotes this number, then with probability of at least $1 - n_0^{-\phi}\frac{1}{1-2^{-\phi}}$, as defined in Lemma 3.3.10, the actual value of H_k lies in*

$$\left[\mathbb{E}[H_k|\mathbb{E}[H_{j<k}]] - a_k - a_{k-1} - \cdots - a_1,\ \ \mathbb{E}[H_k|\mathbb{E}[H_{j<k}]] + a_k + a_{k-1} + \cdots + a_1\right],$$

where the a_i's are defined in Lemma 3.3.10.

Proof. By union bound, the probability of having at least one i for which the estimation at round i has an error more than a_i is at most the sum of all these errors for $0 \leq i \leq m$,

where the total number of round $m \leq \log(n_0)$ by Lemma 3.3.7. This probability is bounded by Theorem 3.3.11 as follows

$$\begin{aligned}
&\sum_{k=0}^{m} (\frac{n_0}{2^k})^{-\phi} \\
&= n_0^{-\phi} (\sum_{0}^{m} \frac{1}{2^{k\phi}}) \\
&< n_0^{-\phi} \frac{1}{1-2^{-\phi}}.
\end{aligned}$$

Hence with probability of at least $1 - n_0^{-\phi} \frac{1}{1-2^{-\phi}}$, estimation at level i is correct up to an error of $\pm a_i$. Finally, note that adding/removing t balls to/from a configuration of balls can increase/decrease the number of heterogeneous pairs by at most $t/2$. Consequently, carry-over errors defined in Theorem 3.3.11, which are deviations from the correct expected number of comparisons at each step, add up linearly. Therefore,

$$\mathbb{E}[H_k|\mathbb{E}[H_{j<k}]] - a_1 - \cdots - a_k \leq H_k \leq \mathbb{E}[H_k|\mathbb{E}[H_{j<k}]] + a_1 + \cdots + a_k.$$

□

3.3.4 Expected Number of Comparisons in Early-Success

Theorem 3.3.13. *If c denotes the number of comparisons, then the difference between the estimated expected cost of the algorithm $\mathbb{E}[c]$ and the actual number is at most $o(n_0)$ with probability of at most $p_s = 1 - n_0^{-\phi} \frac{1}{1-2^{-\phi}}$ as defined in Theorem 3.3.12 and Lemma 3.3.10.*

Proof. To complete the proof, note that Lemma 3.3.10 assures us that $a_1 + \cdots + a_k \in o(n_0)$, and therefore by using Theorem 3.3.12 about our estimation of H_k's and their contribution to n_m described in Lemma 3.3.6 we get

$$\mathbb{E}[c_k] - o(n_0) \leq c_k \leq \mathbb{E}[c_k] + o(n_0)$$

with probability of at least $1 - n_0^{-\phi} \frac{1}{1-2^{-\phi}}$. □

Theorem 3.3.14. *The expected cost of the Folded Pairing algorithm is $g(\alpha)n$, where $\alpha = \alpha_0$ is the proportion of the majority color in the initial set of balls, and $\frac{1}{2} \leq g(\alpha) \leq \frac{2}{3}$ is a function given by the following expression:*

$$g(\alpha) = \frac{2\alpha - 1}{4\alpha} \sum_{k=0}^{m} \frac{1}{2^k} \frac{\alpha^{2^k} + (1-\alpha)^{2^k}}{\alpha^{2^k} - (1-\alpha)^{2^k}}.$$

Proof. Relations in Lemma 3.3.8 and Lemma 3.3.9 provide us with the expected value of n_k in terms of n_0 and α_0 as follows

$$\begin{aligned} n_k &= n_0 \frac{\alpha_0}{2^k} \frac{(2-\frac{1}{\alpha_0})\left(1+(\frac{1}{\alpha_0}-1)^{2^k}\right)}{1-\left(\frac{1}{\alpha_0}-1\right)^{2^k}} \\ &= n_0 \frac{1}{2^k} \frac{2\alpha_0-1}{\alpha_0} \frac{{\alpha_0}^{2^k}+(1-\alpha_0)^{2^k}}{{\alpha_0}^{2^k}-(1-\alpha_0)^{2^k}}. \end{aligned}$$

The result follows from the fact that the number of comparisons in Algorithm 3.1.2 is $\sum_{i=0}^{m} \frac{n_i}{2}$, where n_i denotes the size of remaining components in the i^{th} round of the algorithm and $n_0 = (1+\lambda)(\frac{n}{2\hat{\alpha}})$ is as defined in Theorem 3.3.2. □

3.3.5 Total Number of Comparisons

Putting all these results together, we can finally compute the expected number of comparisons for the proposed algorithm by Theorem 3.3.13 as below:

$$\begin{aligned} & p_s \times \mathbb{E}[\text{cost given early-success}] + (1-p_s) \times \text{cost given naïve algorithm} \\ &= (1-n_0^{-\phi}\frac{1}{1-2^{-\phi}}) \times g(\alpha)n + (n_0^{-\phi}\frac{1}{1-2^{-\phi}})n \\ &= g(\alpha)n + (1-g(\alpha))n_0^{-\phi}\frac{1}{1-2^{-\phi}}n \\ &= g(\alpha)n + o(n), \end{aligned}$$

where the latter equation follows from the fact that $n_0 = \frac{n}{2\alpha} \in \Theta(n)$.

3.4 Notes on the Pairing Problem

3.4.1 Change in α_i

Before finding the proof for the convergence of Algorithm 3.1.2, we studied the behavior of the algorithm by simulating it for different values of α_0 and $n = 10^8$ number of balls. As can be seen in Figure 3.4.1, regardless of α_0, the subsequent α_i's tend to 1 very quickly; a behavior which we proved later (see Lemma 3.3.9).

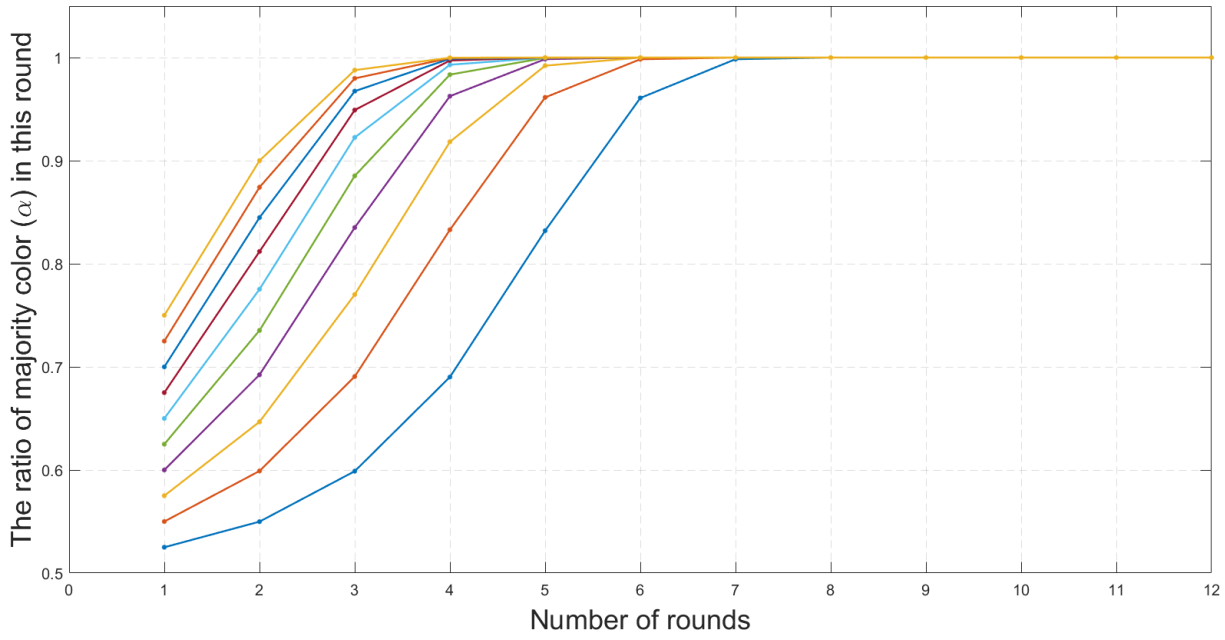

Figure 3.2: Increase of α in each round

3.4.2 Calculating the Exact Expected Value of H

Applying the recurrence relationships in Lemma 3.3.6, one can compute the expected value of the few first terms of each sequence as follows

$$\mathbb{E}[n_1] = \mathbb{E}[\frac{n_0}{2} - H_0] = \frac{n_0}{2} - \mathbb{E}[H_0] = \frac{n_0}{2} - M_{b_0,w_0}$$

$$\mathbb{E}[n_2] = \mathbb{E}[\frac{n_1}{2} - H_1] = \sum_{t=0}^{b_0} \mathbb{E}_{H_1|H_0=t}[\frac{n_1}{2} - H_1|H_0 = t]P_{b_0,w_0}(t)$$

$$= \sum \mathbb{E}[\frac{\frac{n_0}{2} - t}{2} - H_1]P_{b_0,w_0}(t) = \frac{n_0}{4} - \frac{1}{2}M_{b_0,w_0} - \sum M_{\frac{b_0-t}{2},\frac{w_0-t}{2}}P_{b_0,w_0}(t)$$

More generally, the expected number of heterogeneous pairs in each round can be tracked by the previous ones from the equation below:

$$\mathbb{E}\left[H_k|H_{k-1} = t_{k-1}, \ldots, H_1 = t_1, H_0 = t_0\right] = M_{b_j,w_j}(\vec{t})$$

where $M_{b_j,w_j}(\vec{t}) = M_{b_j,w_j}$ for the following values

$$w_k = \frac{w_0 - \sum_{j=0}^{k-1} 2^j t_j}{2^k}, b_k = \frac{b_0 - \sum_{j=0}^{k-1} 2^j t_j}{2^k}.$$

Using this notation, the complete formula is

$$\mathbb{E}[H_k] = \sum_{t_j,\ldots,t_0} \mathbb{E}[H_k | H_{k-1} = t_{k-1}, \ldots, H_1 = t_1, H_0 = t_0] \, \mathbb{P}[H_{k-1} = t_{k-1}, \ldots, H_1 = t_1, H_0 = t_0]$$

$$= \sum_{t_j,\ldots,t_0} M_{b_k,w_k}(\vec{t}) \prod_{j=0}^{k-1} \mathbb{P}[H_j = t_j | H_{j-1} = t_{j-1}, \ldots, H_1 = t_1, H_0 = t_0]$$

$$= \sum_{t_j,\ldots,t_0} M_{b_k,w_k}(\vec{t}) \prod_{j=0}^{k-1} P_{w_j,b_j}(t_j)$$

Although the recurrence relation gives us the exact value of the expected number of comparisons, it is relatively hard to solve it. To find a way around it, we suggested using the expected number of heterogeneous pairs at the i^{th} round to compute the same quantity at the $(i+1)^{th}$ round.

3.5 Randomized 2-Color Partition

Surprisingly, through Algorithm 3.1.2, we actually partitioned our sample of size n_0: every inequality in pairs reveals colors of both components; therefore, in each round of the algorithm, we know the color of the balls we have discarded so far. The only balls with unknown colors are the surviving component in the last round (if any) and the unpaired components in rounds in which the size of n_i is odd. After the last round, we compare the possible surviving component with a ball of known color and we also compare it with all the remaining components of unknown color. In this way, the color of all balls can be determined. As the cost of this partitioning for $n_0 \approx \frac{n}{2\alpha}$ is $g(\alpha)n$, the number of comparisons needed to partition n numbers in Algorithm 3.1.2 is $2\alpha g(\alpha)n$, where $\frac{2}{3} \leq 2\alpha g(\alpha) \leq 1$ for $\frac{1}{2} \leq \alpha \leq 1$. The following theorem formally states this result.

Theorem 3.5.1. *There is a Las Vegas algorithm which solves the 2-color partition problem in the three-way comparison model making at most $2\alpha g(\alpha)n + o(n)$ comparisons with high probability, where α is the fraction of the majority color and $\frac{2}{3} \leq 2\alpha g(\alpha) \leq 1$ is given by the following:*

$$g(\alpha) = \frac{2\alpha - 1}{4\alpha} \sum_{k=0}^{m} \frac{1}{2^k} \frac{\alpha^{2^k} + (1-\alpha)^{2^k}}{\alpha^{2^k} - (1-\alpha)^{2^k}}$$

Chapter 4

Deterministic 3-color Partition

The partition problem is completely solved for deterministic algorithms in the equality-test comparison mode. It is shown that for n balls the lower bound and upper bound are $2n - 3$. For the 3-color plurality problem in the equality-test comparison model, on the other hand, the lower bound of $\frac{3n}{2} - O(1)$ and the upper bound of $\frac{5n}{3} + O(1)$ have been found. In this chapter, we propose an algorithm for the 3-color partition problem in the three-way comparison model with $\frac{3n}{2} - 1$ comparisons. Clearly, any upper bound for the partition problem yields the plurality problem; we, therefore, give an upper bound for the plurality problem as well. In this chapter we prove the following theorem:

Theorem 4.0.1. *Given a set of n balls colored with numbers 1, 2 and 3 with $1 < 2 < 3$, Algorithm 4.1.1 partitions the balls into the sets of their color in $\frac{3n}{2} - 2$ comparisons.*

4.1 Proposed Algorithm

Observation 4.1.1. If we have a ball known to have color 2, we can partition a set of balls with one comparison with each ball: We take a ball of color 2 and compare it with all other balls. The balls of lesser value have color 1, the equal ones are 2 and the greater ones are 3.

We start by pairing the balls and then study the problem in three cases. These cases depend on whether we find a ball of color 2 (Observation 4.1.1).

Algorithm 4.1.1 3-color partition

```
1: procedure PARTITION(S)
2:     Pair all balls
3:     Compare the two balls in each pair
4:     if they are equal then put them in E
5:     else Put the larger in G and the smaller in S  end if
6:     if all pairs were equal then
7:         partition the set containing one ball from each pair of E
8:         Put the other ball from each pair in the same set as the other ball in the pair
9:     else
10:        Compare one ball from G with all other balls in the set G
11:        Compare one ball from S with all other balls in the set S
12:        if there is an inequality in G or S then
13:            Partition using ball of color 2
14:        else
15:            Compare a ball from G with one ball from each pair of E and find all balls
    with the same color → label them as the first group
16:            Compare a ball from S with one ball from each pair of E and find all balls
    with the same color → label them as the second group
17:            Label the remaining balls (i.e. balls not equal to the colors of balls of G and
    S) as the third group
18:        end if
19:    end if
20: end procedure
```

4.2 Correctness

We pair the balls and compare the two balls in each pair. We might have three outcomes for each comparison. We label each ball and divide the balls into these three sets according to the result of the comparison:

- G: The balls of the larger value in an inequality.
- S: The balls of the smaller value in an inequality.
- E: The balls from an equality comparison.

Observation 4.2.1. The balls in the set G can only be 2 and 3 for the reason that 1 is never greater than other balls. By the same argument, balls in the set S can only be 1 and 2. The balls in the set E can have any label.

Observation 4.2.2. E consists of pairs of equal balls. Therefore, in order to partition the set E, we only need to consider one ball from each pair and put the other ball of the pair in the same set. Thus, we can only consider half of the balls from the set.

By the nature of comparison, we know that we have the same number of G's as we have S's. Suppose we have a balls of G's and thus a balls of S's. We consider different cases for a:

1. $a = 0$

   This means that all the balls fall into E. By Observation 4.2.2, we have the same problem with half the size of the original problem. We can solve the problem by induction on the size of the input n.

2. $a \neq 0$

   For each of sets of G and S we do the following: we compare one of the balls from the set with all other balls in that set. Depending on finding an inequality in the sets two cases might happen:

   (a) **We have at least one inequality in one of the sets of G or S**

   If we have an inequality, then by Observation 4.2.1 we can find the label of the balls in that set; therefore we found a ball with label 2. If there is an inequality in only one of the sets of G and S, it means that the other set is a homogeneous component. Because the homogeneous is always larger (smaller) than the other set, we know that its color cannot be 2, so we know its color. Then we compare the ball labeled 2 with one of each pair in the set E to find their colors.

   (b) **We do not have any inequality in the sets of G and S**

   We know that both G and S are homogeneous components. We compare ball g from G with one ball from each pair of E. We find all balls with the same color as the color of set G. Then we compare s from S with one ball from each pair of E and find all the balls with the same color as set S. The remaining balls will be the other set.

4.3 Complexity

We show that the total number of comparisons is less than $\frac{3n}{2} - 2$. We used $\frac{n}{2}$ comparisons to pair the balls and compare balls in each pair. We compute the number of comparisons for each case separately:

1. $a = 0$ The induction hypothesis tells us the number of comparisons for the halved problem is at most $\frac{3n}{4} - 2$, which after adding $\frac{n}{2}$ comparisons that we made, in the beginning, is

$$\frac{n}{2} + \frac{3n}{4} - 2 = \frac{5n}{4} - 2 < \frac{3n}{2} - 2$$

2. $a \neq 0$

   We used $2(a-1)$ comparisons for comparing one ball of each of sets G and S with other balls in that set.

   (a) **There is inequality in G or S**

   In this case we compare the ball labeled 2 with half of the balls in the set E. There are $n - 2a$ balls in set E; thus, because $a \leq \frac{n}{2}$ we use at most

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

   comparisons.

   (b) **There is no inequality in G or S**

   We compare one ball from G with one ball from each pair of E and one ball from S with one ball from each pair of E. Therefore, we use at most

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

   comparisons in total.

Therefore, in all cases the number of comparisons is bounded by $\frac{3n}{2} - 2$.

Chapter 5

Randomized 3-Color Partition

Suppose we have a set of n balls colored with three different colors: 1, 2 and 3 with $1 < 2 < 3$. We want to know the expected number of comparisons needed to find the color of each ball and partition the balls according to their color. In randomized algorithms, this problem is solved in the equality-test comparison model with $\frac{5n}{3} - \frac{8}{3} + o(1)$ comparisons for the partition problem and with $\frac{3n}{2} + O(1)$ comparisons for the plurality problem. We give an upper bound on 3-color partition in the three-way comparison model. The bound is also an upper bound for the plurality problem because knowing the color of each ball implies the frequency of each color, which obviously yields the plurality. In this chapter we show the following result:

Theorem 5.0.1. *Given a set of n balls colored with numbers $1 < 2 < 3$, the expected number of comparisons of Algorithm 5.1.1 for partitioning the balls is between $n + o(n)$ and $\frac{3n}{2} + o(n)$ with high probability.*

5.1 Proposed Algorithm

Based on Observation 4.1.1 for partitioning balls of three colors, after finding a ball of color 2 we can partition the balls by comparing them with that ball. We sample k balls from the input and then partition the balls in the sample using the naïve algorithm as defined below to see whether there is a ball of color 2:

Lemma 5.1.1 (Naïve Algorithm)**.** *Compare one ball with all other balls, determine balls of the same color and put these balls aside. Then compare one ball from the remaining set with other balls to partition the remaining set.*

The correctness of the algorithm is trivial and the complexity is $2n - 3$ for a set of n balls. If we find a ball of color 2, we compare that ball with all the balls of the input. If not, we know that the sample consists of only two different colors. Between the two colors, we choose the color of the majority and follow the naïve approach defined in Lemma 5.1.1: compare that ball with all balls outside the sample. Then, compare a ball of the other color in the sample, if any, with all balls that do not have the same color as the majority one, in case there are three different colors in the input but not in the sample. The algorithm is as follows:

Algorithm 5.1.1 Randomized 3-color partition

```
1: Sample k balls
2: Partition the balls in the sample using naïve algorithm
3: if found a ball with color 2 then
4:     Compare the ball with all remaining balls and partition balls
5: else
6:     Choose the majority color in the sample
7:     Compare the majority color with n − k balls outside the sample
8:     Compare the other color with balls not equal to the majority color
9: end if
```

5.2 Correctness

We partition the sample by first comparing one ball with all other balls in the sample and find all the balls with the same color as this ball. We then, for the remaining balls, which can have two different colors, compare one ball with all the balls to find the balls with the same color as that ball. If some balls are left, they belong to the third group. After partitioning the sample, two cases might happen:

1. The sample has three different colors

   Because of the nature of the three-way comparison model, if we have three different colors in the sample we know the color of each group as well. By Observation 4.1.1 we know that we can partition the input using a ball colored 2.

2. The sample has less than three different colors

   There still might be a ball of color 2 in the sample but because there are not three different colors in the sample, we cannot distinguish it. We take the ball of the

majority color in the sample and compare it with all the balls in the input. Thus, we found all the balls with the same color as the majority color. If there are two colors in the sample, we compare the other ball from the sample and compare it with the remaining balls in the input and find all the balls of the same color. The other balls belong to the third group. If there is only one color in the sample, we choose the second ball from the balls in the input that are not the same color as the majority color. We can find the balls of the same color as that ball, and all the other balls are the third color. Therefore, we partitioned the input.

5.3 Complexity

Restating Theorem 3.3.1, we know given a set of balls with two colors, by choosing $k = \frac{2+\epsilon}{\epsilon^2}\ln(\frac{1}{\delta}) \in O(\frac{1}{\epsilon^2}\ln(\frac{1}{\delta}))$ the difference between the fraction of majority color and the estimated fraction of majority in a sample of size k is at most ϵ with probability $1-\delta$ [22]. Because in this problem we have three colors, the sampling is from a multinomial distribution [21]. To utilize Theorem 3.3.1 here, we will use this bound three times where each time a color is compared against two other colors. In this way, we can bound the error in estimating the proportion of each color separately and then finish the problem by the union bound. To this end, we sample $n_s = \frac{2+\epsilon}{\epsilon^2}\ln(\frac{3}{\delta})$; then it follows from Theorem 3.3.1 that the probability of having an error of at least ϵ in estimating the proportion of a color against the other two is at most $\frac{\delta}{3}$. The union bound assures us that the probability of having at least one estimation (among these three estimations) with such an error is at most

$$\frac{\delta}{3} + \frac{\delta}{3} + \frac{\delta}{3} = \delta.$$

Therefore, the probability of estimating all proportions with error less than ϵ is at least $1-\delta$

We set $\epsilon = n^{-\frac{1}{3}}$ and $\delta = n^{-2}$ in our analysis. To compute the expected number of comparisons, we need to calculate $p_s = 1-\delta$, the probability of success in finding a ball of color 2 in the small sample of n_s balls. Then, we can compute the average cost of our Las Vegas algorithm as

$$p_s \times \mathbb{E}[\text{cost given success}] + (1-p_s) \times \text{cost given naïve algorithm}$$

In the sampling, we used at most $n_s - 1 + n_s - 2 = 2n_s - 3$ comparisons in order to partition the sample. For the cases that can happen, we compute the complexity separately:

1. The sample has three different colors

   If the sample has three different colors, it means that we found a ball of color 2 and we can compare this number with all the balls outside the sample. Thus, in total, we used $2n_s - 3 + n - n_s = n + n_s - 3$ comparisons.

2. The sample has less than three different colors

   Call the fraction of the majority color in the sample α. In this case, we compare a ball of this color with all other balls in the input. We used $n - n_s$ comparisons. With high probability, an α fraction of the input is the majority color and $(1-\alpha)$ of the input is one of the other two colors. Therefore, $\alpha(n - n_s)$ of them are the same color as the majority color. We compare one ball of another color (from the sample if the sample consists of two different colors or from the input if there is only one color in the sample) with all the remaining balls. We use an expected $(1-\alpha)(n-n_s)$ number of comparisons to find the balls of the same color as the non-majority ball. The remaining balls belong to the third group. Thus, in total, we used

$$\begin{aligned} &2n_s - 3 + n - n_s + (1-\alpha)(n-n_s) \\ &= (2-\alpha)n + n_s(2 - 1 - 1 + \alpha) - 3 \\ &= (2-\alpha)n + n_s\alpha - 3 \end{aligned}$$

   Comparison. Because $\alpha \geq (1-\alpha)$, we know that $\alpha \geq \frac{n}{2}$. Therefore, the number of comparisons in total in this case is $\leq \frac{3n}{2} n + o(n)$.

Thus, the average cost of our Las Vegas algorithm is

$$\begin{aligned} &p_s \times \mathbb{E}[\text{cost given success}] + (1-p_s) \times \text{cost given naïve algorithm} \\ &= p_s(n + n_s - 3) + (1-p_s)\left((2-\alpha)n + n_s\alpha - 3\right) \\ &= n(\alpha p_s - p_s - \alpha + 2) + n_s(-\alpha p_s + p_s + \alpha) - 3 \\ &= n(1 + \delta(1-\alpha)) + n_s(-\alpha p_s + p_s + \alpha) - 3 \\ &= n\left(1 + (1-\alpha)(1-p_s)\right) + O(1) \end{aligned}$$

The term n_s is approximated by $O(1)$ in the last line as a result of Theorem 3.3.1. The coefficient of n in the above expression depends on p_s, the probability of finding a ball of color 2 in a sample of size n_s; although it depends on the distribution of colors which is not known to us, we can say

$$1 \leq 1 + (1-\alpha)(1-p_s) \leq 1 + 1 - \alpha = 2 - \alpha \leq \frac{3}{2}$$

As the estimation of α, the proportion of the majority, is correct up to an error of ϵ with probability of $1-\delta$, the cost is within n and $\frac{3n}{2}+O(n\epsilon)$ with probability $1-\delta$. Now by plugging in $\epsilon = n^{\frac{-1}{3}}$ and $\delta = n^{-2}$, we have $n\epsilon \in o(n)$ and $n_s \in O(\frac{1}{\epsilon^2}) \subset O(n^{\frac{2}{3}}) \subset O(n)$.

Chapter 6

Conclusion and Future Work

In this thesis we studied under a three-way comparison model $(<, =, >)$ problems that had previously been considered on an equality-test comparison model $(=, \neq)$. We also proposed a new approach for computing the complexity of a family of randomized algorithms for which the outcome of the i^{th} step of the underlying stochastic process can be approximated by the expected value of the process at the $(i-1)^{th}$ step. We studied comparison-based problems for different numbers of colors and for all problems the cost to minimize was the number of comparisons. In this context, some of the future directions for research in this area are as follows.

6.1 Comparison Systems

Through this thesis we only consider the case where a comparison acted on two elements; however, a more complex comparison system can receive more than 2 elements as input. To illustrate, consider a comparison system called $k-$sort which receives k elements and outputs them in sorted order. Note that, for $k = 2$ this is the same system as the three-way comparison. Moving from equality-test comparison to three-way can naturally suggest other comparison systems. For example, although three-way seems to be a complete comparison system in real numbers for it is trichotomous, considering a comparison model in the *middle* of equality-test and three-way in which the result can be either $>$ or $\leq$ can be interesting. It is clear that in the latter system, the two comparisons $x : y$ and $y : x$ can provide us with the same information as in $=, >, <$, but the question is whether we can do better than this. Interestingly, if we increase the dimension then there can be a four-(or more) way comparison system. For instance, in computational geometry and specifically

$\mathbb{R}^2$, the result of comparing (x_1, y_1) and (x_2, y_2) can take several forms which may result in different number of comparisons needed to answer queries such as determining the convex hull of n points.

6.2 Cost Models

The focus here was on the total number of comparisons where the cost of using each element in a comparison was assumed to be the same for every element. However, one can consider a weighted comparison problem, in which the costs of comparing elements are different. A real-world instance of this problem is when data are distributed among different servers with different responding time/cost.

References

[1] M. Aigner, G. De Marco, and M. Montangero, *The plurality problem with three colors*, in Annual Symposium on Theoretical Aspects of Computer Science, Springer, 2004, pp. 513–521.

[2] ——, *The plurality problem with three colors and more*, Theoretical Computer Science, 337 (2005), pp. 319–330.

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

[4] M. Blum, R. W. Floyd, V. R. Pratt, R. L. Rivest, and R. E. Tarjan, *Time bounds for selection*, J. Comput. Syst. Sci., 7 (1973), pp. 448–461.

[5] S. Boucheron, G. Lugosi, and P. Massart, *Concentration inequalities: A nonasymptotic theory of independence*, Oxford university press, 2013.

[6] G. E. Box, W. G. Hunter, J. S. Hunter, et al., *Statistics for experimenters*, (1978).

[7] R. S. Boyer and J. S. Moore, *MJRTY–a fast majority vote algorithm*, in Automated Reasoning, Springer, 1991, pp. 105–117.

[8] W. Cunto and J. I. Munro, *Average case selection*, J. ACM, 36 (1989), pp. 270–279.

[9] D. Dor and U. Zwick, *Selecting the median*, SIAM Journal on Computing, 28 (1999), pp. 1722–1758.

[10] ——, *Median selection requires (2+ϵ)n comparisons*, SIAM Journal on Discrete Mathematics, 14 (2001), pp. 312–325.

[11] Z. Dvořák, V. Jelínek, J. Kynčl, and M. Saks, *Three optimal algorithms for balls of three colors*, in Annual Symposium on Theoretical Aspects of Computer Science, Springer, 2005, pp. 206–217.

[12] M. J. Fischer and S. L. Salzberg, *Finding a majority among n votes: Solution to problem 81-5*, tech. rep., Yale University, 1982.

[13] R. W. Floyd and R. L. Rivest, *Expected time bounds for selection*, Commun. ACM, 18 (1975), pp. 165–172.

[14] L. R. Ford Jr and S. M. Johnson, *A tournament problem*, The American Mathematical Monthly, 66 (1959), pp. 387–389.

[15] V. Jayapaul, J. I. Munro, V. Raman, and S. S. Rao, *Finding modes with equality comparisons*, Theor. Comput. Sci., 704 (2017), pp. 28–41.

[16] D. Král, J. Sgall, and T. Tichỳ, *Randomized strategies for the plurality problem*, Discrete Applied Mathematics, 156 (2008), pp. 3305–3311.

[17] J. I. Munro and P. M. Spira, *Sorting and searching in multisets*, SIAM J. Comput., 5 (1976), pp. 1–8.

[18] M. Paterson, *Progress in selection*, Scandinavian Workshop on Algorithm Theory (SWAT), 1097 (1996).

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

[20] E. Seneta et al., *A tricentenary history of the law of large numbers*, Bernoulli, 19 (2013), pp. 1088–1121.

[21] C. P. Sison and J. Glaz, *Simultaneous confidence intervals and sample size determination for multinomial proportions*, Journal of the American Statistical Association, 90 (1995), pp. 366–369.

[22] R. Tempo, E.-W. Bai, and F. Dabbene, *Probabilistic robustness analysis: Explicit bounds for the minimum number of samples*, Systems & Control Letters, 30 (1997), pp. 237–242.

[23] E. Upfal and M. Mitzenmacher, *Probability and computing: randomized algorithms and probabilistic analysis*, vol. 160, Cambridge University Press, 2005.

[24] S. Yang, *The randomized majority*, Master's thesis, University of Waterloo. (Preprint).