# ELEMENTARY ANALYSIS OF A CLASSICAL PROBABILISTIC ALGORITHM

CAROLINA MEJIA, J. ANDRES MONTOYA.

Abstract. In this short paper we analyse Freivald's randomized algorithm for palindrome recognition, we base our analysis on elementary facts of number theory

Palindrome recognition could be considered as a *Toy Model* of *Stringology,* it is the case because the language of palindromes has the following singular feature: there is a perfect matching between upper and lower bounds concerning the time and space complexity of palindrome Recognition, on almost any model of computation. Consider the following list of results:

(1) There exists a one tape Turing machine which recognizes palindromes in time $O\left(n^2\right)$. The Theorem of Henning [6] says that any one tape Turing machine recognizing palindromes requires cuadratic time.
(2) Any multitape Turing machine recognizing palindromes requires, at least, *real time*. A theorem of Galil [4] shows that there exists a multitape Turing Machine recognizing palindromes in real time.
(3) Biedl et al [2] proved that for all $m \geq 1$ and for nay $m$-dimensional one tape Turing machine $\mathcal{M}$ recognizing the language of palindromes, the machine $\mathcal{M}$ has a running time $\Omega\left(\frac{n^2}{\log^{m-1}(n)}\right)$. On the other hand, Biedl et al showed that for any $m \geq 1$ there exists a $m$-dimensional one tape Turing Machine recognizing palindromes in time $O\left(\frac{n^2}{\log^{m-1}(n)}\right)$.
(4) Any Turing machine recognizing palindromes requires logarithmic space, and there exists a Turing machine which recognizes palindromes using logarithmic space (see [7]).
(5) There exists a parallel algorithm which recognizes palindromes in time $O\left(\log\left(\log\left(n\right)\right)\right)$ using $O\left(\frac{n}{\log(\log(n))}\right)$ processors [1], the running time of this algorithm matches the trivial lower bound for the number of processors employed .
(6) Yao proved in [8] that any probabilistic one tape Turing machine recognizing palindromes requires *pseudolinear time.* On the other hand Freivald [3] proved that there exists a probabilistic one tape Turing Machine recognizing palindromes in time $O\left(n\log\left(n\right)\right)$.

In this short contribution we analyze Freivald's algorithm. Specifically, we compute the error probability of this algorithm using completely elementary facts of number theory. Some times, the analysis of elementary probabilistic algorithms requires nonelementary tools, Freivald claims in [3] that the estimation of the error

*Key words and phrases.* Probabilistic Algorithms, Palindrome Recognition, Prime Number Theorem.

probability of his algorithm requires the use of *Cebotarev density theorem.* We estimate this probability using elementary arguments which are based on some basic facts of number theory.

## 1. Freivald's Algorithm

Freivald's algorithm, which we will denote with the symbol $\mathcal{F}$, is a probabilistic algorithm that recognizes palindromes in time $O(n\log(n))$. Freivald's algorithm is optimal on one-tape probabilistic Turing machines, since its running time matches the lower bound for palindrome recognition on this type of machines, (Yao proved in [8] that any one-tape probabilistic Turing machine recognizing palindromes has running time $\Omega(n\log(n))$).

In the presentation of Freivald's algorithm we have chosen some specific values for the parameters defining it, those specific values allow us to prove our upper bound on the error probability. If we would like to obtain a smaller upper bound, we could use the standard *probability amplification techniques* or we could choose another set of parameters.

Freivald's algorithm is the algorithm defined by:

On input $x = x_1...x_n$, algorithm $\mathcal{F}$ works in the following way

(1) $\mathcal{F}$ decides if $n$ is an even number.
(2) If $n = 2m$, $\mathcal{F}$ computes $y, z \in \{0,1\}^m$ such that $x = y\overline{z}$. If $n = 2m+1$, $\mathcal{F}$ computes $y, z \in \{0,1\}^m$ and $a \in \{0,1\}$ such that $x = ya\overline{z}$.
(3) $\mathcal{F}$ chooses, uniformly at random, $64\log(m)$ numbers from the interval $\{1, ..., m^4\}$.
(4) Given $n_1, ..., n_{64\log(m)}$ the numbers generated at step three, $\mathcal{F}$ computes for any $i \leq 64\log(m)$ the numbers: $k_i = y \bmod n_i$ and $l_i = z \bmod n_i$.
(5) $\mathcal{F}$ decides if there exists $i \leq 64\log(m)$ such that $k_i \neq l_i$. If it is the case $\mathcal{F}$ rejects $x$, otherwise $\mathcal{F}$ accepts $x$.

In [3] Freivald claims that the error probability of $\mathcal{F}$ is upperbounded by a number which is smaller than $\frac{1}{2}$, but he claims that the verification of this fact requires the use of Cebotarev density theorem. In this short contribution we prove the theorem below, using completely elementary facts of number theory.

**Theorem 1.** *There exists $N_0 \geq 150$ such that for all $n \geq N_0$ and for all $x \in \{0,1\}^i$, with $i \in \{2n, 2n+1\}$, we have that*

- *If $x \in Pal$, then $\Pr[\mathcal{F}$ accepts $x] = 1$.*
- *If $x \notin Pal$, then $\Pr[\mathcal{F}$ accepts $x] \leq \frac{7}{16}$*

## 2. Computing the error probability of Freivald's algorithm

In this section we estimate the error probability of $\mathcal{F}$, and we prove theorem 1.

**Lemma 1.** *There exists $N_0 \geq 150$ such that given $n \geq N_0$ and given $\epsilon \in (0,1]$, if we choose uniformly at random $\frac{2\log(n)}{\epsilon}$ numbers from the interval $\{1, ..., n\}$, then the probability that at least one of those numbers is a prime number is bigger than or equal to $1-\epsilon$.*

*Proof.* Given $n \in \mathbb{N}$ we define

$$P(n) = \{p \leq n : p \text{ is prime}\}$$

*Chebyshev Prime Number Theorem* says that, if $\pi(n)$ is equal to $|P(n)|$, then

$$\lim_{n\to\infty}\left(\frac{\pi(n)}{\frac{n}{\log(n)}}\right)=1$$

It implies that there exists a number $N_0 \geq 150$ such that for any $n \geq N_0$

$$\frac{1}{2\log(n)} \leq \Pr_{i\leq n}[i \text{ is prime}] \leq \frac{2}{\log(n)}$$

Let $n$ be a number bigger than $N_0$ and let $(\mathbb{N}^+, \mu_n)$ be the probability space defined by the distribution

$$\mu_n(m) = \left(1-\frac{\pi(n)}{n}\right)^{m-1}\frac{1}{\pi(n)}$$

We consider the random variable $X : \mathbb{N}^+ \to \mathbb{R}^+$ defined by:

$$\text{For any } m \geq 1,\ X(m) = m$$

Note that $E[X]$, the expected value of $X$, is equal to the expected quantity of numbers we have to pick out of $\{1,...,n\}$, before selecting a prime number. We have that $E[X] = \frac{n}{\pi(n)} \leq 2\log(n)$. Given $\epsilon \in (0,1]$ we have

$$\Pr_{n_1,...,n_{\frac{2\log(n)}{\epsilon}}\in\{1,...,n\}}\left[\text{for any } i \leq \frac{2\log(n)}{\epsilon},\ \text{the number } n_i \text{ is not prime}\right] \leq$$

$$\Pr\left[X \geq \frac{n}{\pi(n)\epsilon}\right] \leq \epsilon$$

Thus, we have that if $n \geq N_0$ and $\epsilon \in (0,1]$, the probability of choosing at least one prime number, when we pick $\frac{2\log(n)}{\epsilon}$ elements out of $\{1,...,n\}$, is bigger than or equal to $1-\epsilon$ □

Let $n$ be a natural number and let $x, y$ be two numbers within the interval $\{1,...,2^n\}$.

**Lemma 2.** *Let $A$ be a set of prime numbers. Suppose that $x \neq y$ and suppose that for any $p \in A$ we have $x \bmod p = y \bmod p$. Then, $|x-y| \geq \prod_{p\in A} p$.*

*Proof.* Suppose that for all $p \in A$ we have that $x \bmod p = y \bmod p$, then

$$x \bmod \prod_{p\in A} p_i = y \bmod \prod_{p\in A} p_i$$

Therefore, we have that $|x-y| \geq \prod_{p\in A} p_i$. □

Given $x, y \in \{1,...,2^n\}$, we define

$$A_{xy} = \left\{p \in P\left(n^4\right) : x \bmod p = y \bmod p\right\}$$

We want to prove that if $x \neq y$, then $A_{xy}$ is a set of small size. The proof of the following lemma can be found in [5].

**Lemma 3.** *Given $n \geq 150$, we have that $\prod_{a\in P(n^2)} a \geq n!2^n$.*

From now on, we will use the symbol $n$ to denote natural numbers which are bigger than or equal to $N_0$.

**Lemma 4.** *Let $A_n$ be a subset of $P\left(n^4\right)$ whose size is bounded below by $\frac{\pi\left(n^4\right)}{4}$, we have that $\prod_{a\in A_n} a \gneqq 2^n$*

*Proof.* Let $B$ be equal to the set of the first $\frac{\pi\left(n^4\right)}{4}$ prime numbers. It is clear that $\prod_{a\in A_n} a \geq \prod_{a\in B} a$. Also, it is sufficient to show that $\prod_{a\in B} a \geq 2^n$. We claim that $P\left(n^2\right) \subset B$. It is the case because $\pi\left(n^2\right) \leq \frac{n^2}{\log(n)} \leq \frac{n^4}{32\log(n)} \leq \frac{\pi\left(n^4\right)}{4}$. Thus, we have

$$\prod_{a\in A_n} a \geq \prod_{a\in B} a \geq \prod_{p\in P(n^2)} a \geq n!2^n \gneqq 2^n$$

□

**Corollary 1.** *Given $x, y \in \{1, ..., 2^n\}$, if $x \neq y$ we have that $|A_{xy}| \leq \frac{\pi\left(n^4\right)}{4}$.*

*Proof.* Suppose that $|A_{xy}| \gneqq \frac{\pi\left(n^4\right)}{4}$, we have that

$$|x - y| \geq \prod_{p\in A_{xy}} p \gneqq 2^n.$$

Last inequality is not possible, since $x, y \in \{1, ..., 2^n\}$ □

**Corollary 2.** *Given $n \geq N_0$ and given $x, y \in \{1, ..., 2^n\}$ such that $x \neq y$, we have that*

$$\Pr_{p\in P(n^4)}\left[x \bmod p \neq y \bmod p\right] \geq \frac{3}{4}$$

*Proof.* Last corollary says that $|A_{xy}| \leq \frac{\pi\left(n^4\right)}{4}$, and it implies that

$$\frac{3}{4} \leq \Pr_{p\in P(n^4)}\left[p \notin A_{xy}\right] = \Pr_{p\in P(n^4)}\left[x \bmod p \neq y \bmod p\right]$$

□

**Lemma 5.** *Given $n \geq N_0$ and given $x, y \in \{0,1\}^n$, if we choose uniformly at random $64\log(n)$ numbers from the interval $\left\{1, ..., n^4\right\}$, we have that*

$$\Pr_{n_1,...,n_{64\log(n)}}\left[\exists i \leq 64\log(n)\left(x \bmod n_i \neq y \bmod n_i\right)\right] \geq \frac{9}{16}$$

*Proof.* Lemma 1 implies that

$$\Pr_{n_1,...,n_{64\log(n)}}\left[\exists i \leq 64\log(n)\left(n_i \in P\left(n^4\right)\right)\right] \geq \frac{3}{4}$$

Corollary 2 implies that

$$\Pr_{p\in P(n^4)}\left[x \bmod p \neq y \bmod p\right] \geq \frac{3}{4}$$

Therefore, we have

$$\Pr_{n_1,...,n_{64\log(n)}}\left[\exists i \leq 64\log(n)\left(x \bmod n_i \neq y \bmod n_i\right)\right] \geq \frac{9}{16}$$

□

Now, we can obtain theorem 1 as an easy corollary of lemma 5.

**Corollary 3.** *Given $n \geq N_0$ and given $x \in \{0,1\}^i$, with $i \in \{2n, 2n+1\}$, we have that*

- *If $x \in Pal$, then $\Pr[\mathcal{F} \text{ accepts } x] = 1$.*
- *If $x \notin Pal$, then $\Pr[\mathcal{F} \text{ accepts } x] \leq \frac{7}{16}$.*

**Agradecimientos 1.** *Thanks to VIE-UIS and thanks to Colciencias research project 111518925292.*

## Referencias

[1] A. Apostolico, D Breslauer, Z. Galil. Parallel Detection of all the Palindromes in a String. Theoretical Computer Science 141(1&2):163-173 (1995).
[2] T. Biedl, J. Buss, E. Demaine, M. Demaine, M. Hajiaghayi, M. Vinar. Palindrome Recognition Using a Multidimensional Tape. Theoretical Computer science 302(1-3): 475-480 (2003).
[3] R. Freivald. Fast computation by probabilistic Turing machines. *Theory of Algorithms and Programs*, Latvian State University 2:201-205 (1975).
[4] Z. Galil. Palindrome Recognition in Real Time. Journal of Computers and Systems Sciences 16(2):140-157 (1978).
[5] G. Hardy, E Wright. *An Introduction to the Theory of Numbers.* Oxford University Press, Oxford UK, 1960.
[6] F. Hennie. Crossing Sequences and Off-line Turing Machines. FOCS 1965:168-172.
[7] J. Hopcroft, J. Ullman. *Introduction to Automata Theory, Languages, and Computation.* Addison-Wesley 1979.
[8] A. Yao. A Lower Bound for Palindrome Recognition by Probabilistic Turing Machines. Technical report #77-647, Standford University 1977.

Universidad Industrial de Santander
*E-mail address*: juamonto@uis.edu.co, caromejia@uis.edu.co,