# The Number Field Sieve Factoring Algorithm

A thesis submitted to
Indian Institute of Science Education and Research Pune
in partial fulfillment of the requirements for the
BS-MS Dual Degree Programme

Thesis Supervisor: Ayan Mahalanobis

by
Rahul Kumar
April, 2012


Indian Institute of Science Education and Research Pune
Sai Trinity Building, Pashan, Pune India 411021

This is to certify that this thesis entitled "The Number Field Sieve Factoring Algorithm" submitted towards the partial fulfillment of the BS-MS dual degree programme at the Indian Institute of Science Education and Research Pune, represents work carried out by Rahul Kumar under the supervision of Ayan Mahalanobis.

Rahul Kumar

Thesis committee:
Ayan Mahalanobis
Baskar Balasubramanyam

A. Raghuram
Coordinator of Mathematics

# Acknowledgments

First of all I would like to thank my guide Dr. Ayan Mahalanobis for his guidance and motivation, both academic and otherwise, especially with programming in magma. Without his efforts this project would not have been a successes.

I would also like to thank Dr. Baskar Balasubramanyam for teaching me the algebraic number theory required in this thesis. I would like to thank IISER pune for providing me good working environment and nice library.

Lastly, I would like to thank whole of the 2007 batch of IISER Pune. Who haven been there by my side through thick and thin.

# Abstract


## The Number Field Sieve Factoring Algorithm

by Rahul Kumar

Integer factorization has been interesting problem for mathematicians since centuries. Integer factorisation lies in the heart of Number Theory. There has been many algorithms for factorisation such as Dixon's factorisation, continued fractions and Quadratic Sieve Factoring Algorithm. Many of the encryption algorithms in cryptography are based on the "hardness" in factoring large composite numbers with no small prime factors Number Field Sieve is the best known factoring algorithm. It works best with large numbers, for small one Quadratic Sieve is the best algorithm because of its low requirement of storage. Time complexity of GNFS (General Number Field Sieving) algorithm is $L_n[\frac{1}{3}, \sqrt[3]{\frac{64}{3}}]$(explanation of L-notation is given in appendix) and that of quadratic sieve algorithm is $L_n[\frac{1}{2}, 1]$.

# Contents

**Abstract** **vii**

**1 Introduction** **1**
1.1 Motivation . . . . . 1
1.2 Historical Perspective . . . . . 2

**2 Mathematical Preliminaries** **7**
2.1 Basic Properties of Integers($\mathbb{Z}$) . . . . . 7
2.2 Properties of Polynomial over $\mathbb{Z}$ and $\mathbb{Q}$ . . . . . 8
2.3 Basic Algebraic Number Theory . . . . . 8

**3** **13**
3.1 Steps involved in the algorithm . . . . . 13
3.2 Setting Up the Factor bases . . . . . 17
3.3 Finding the exponents . . . . . 20
3.4 Quadratic Character . . . . . 23
3.5 Square root extraction . . . . . 24

**4 Implementation of algorithm** **25**

**5 Appendix** **31**
5.1 Code for the General Number Field Sieve written in MAGMA . . . . 31
5.2 L-Notation . . . . . 39

# Chapter 1

# Introduction

## 1.1 Motivation

Most of the modern factoring algorithm use the concept of the difference of squares as described under Legendre's method. All these algorithm uses concept of factor base, smoothness, sieving,and reducing matrices over $\mathbb{Z}_2$. The major difference between Quadratic Sieve and Dixon's Algorithm is the basic function used in construction of smooth elements. This eventually results in faster collection of relations. The success of General Number Field Sieve Algorithm relies on the realization that unlike Quadratic Sieve and Dixon's Algorithm we can use a higher degree polynomial for smooth element collection.

There is one more reason for thinking about this algorithm,which arises from the fact that there could be ring on which the notion of smoothness can be imposed,similar to $\mathbb{Z}$. This ring could possibly have more smooth elements than that in $\mathbb{Z}$. If we have some kind of natural mapping between $\mathbb{Z}/n\mathbb{Z}$ and these rings then we could possibly arrive at the difference of squares.

### 1.1.1 Shaping the Idea

In this section we will be discussing on ways to try some other ring instead of $\mathbb{Z}/n\mathbb{Z}$. The main idea is, to consider the function chosen for smooth element generation. In Quadratic sieve we have

$$f(x) = r_i^2 - n$$

this function can be thought of as

$$f : \mathbb{Z} \to \mathbb{Z}/n\mathbb{Z}$$

this function maps product of all the smooth $r_i^2$ - $n$, a square in $\mathbb{Z}$ to a square in $\mathbb{Z}/n\mathbb{Z}$. Once we have a square in $\mathbb{Z}_n$, we can use the difference in square congruence. Similarly if we have a different ring with the same kind of map, then we can use it to generate congruent squares.

$$f : Ring \to \mathbb{Z}_n$$

## 1.2 Historical Perspective

The idea of this algorithm was given by Pollard. He circulated a paper to all the eminent mathematicians of that time describing this algorithm. Then H.A.Lenstra, Pollard and couple of their colleagues factored Fermats ninth number [4],that is $2^{2^9}+1$ using Number Field Sieve. But the algorithm developed by them was only for certain kind of numbers. Then Pomerance and few more came up with a general algorithm which works for all numbers. Recently, this algorithm has factored a 663-bit RSA challenge number.

The problem of factorization was there since the time of Euclid. Before Fermat this problem was solved by trail division method.

In 1640, Fermat introduced some unique idea of factorization which are still used. His idea was to write the composite number $n$ in form of $x^2 - y^2$ from where we can easily see the factor $(x + y)(x - y)$. If the factors of $n$ are close this is good method.

In 1750, Euler gave an algorithm for special type of integers, the integers which can be expressed in the form $n = a^2 + \mathbf{D}b^2$ in two different ways with the same $\mathbf{D}$. He used the method to successfully factor large numbers [5].

In 1798, Legendre presented an idea which will revolutionize the factoring game. If we have integer x and y which satisfies this congruence

$$x^2 = y^2 \mod n, \quad 0 \leq x, y \leq n; x \neq y; x + y \neq n$$

then $GCD(n, x - y)$ and $GCD(n, x + y)$ is non trivial factor of $n$. If $n = pq$,

$$\begin{gathered} x^2 = y^2 \mod n \\ \Rightarrow pq \mid x^2 - y^2 \\ \Rightarrow pq \mid (x - y)(x + y) \end{gathered}$$

$$\Rightarrow p \mid (x-y) \text{ or } p \mid (x+y)$$
$$\Rightarrow q \mid (x-y) \text{ or } q \mid (x+y)$$

The importance of integer factorization gained new heights with use of hardness of factorization for security. Some of the cryptographic algorithm which rely on hardness of factorization are listed below

### 1.2.1 RSA Cryptosystem

The RSA system, named after its inventors Ron Rivest, Adi Shamir and Len Adleman, was the first public key cryptosystem and is still the most widely used cryptosystem [12] Its security is closely related to difficulty of factoring the large composite number. Alice wants to send message to Bob. The algorithm goes like this:

- Bob generates randomly and independently two large composite number $p, q$ and find $n = pq$. Bob also chooses an integer $e$ with
  $1 < e < \phi(n) = (p-1)(q-1)$ and $GCD(e, (p-1)(q-1)) = 1$

  (**NOTE**: Given $n$ of the form $p.q$ where $p$ and $q$ are primes, the order of the group $\mathbb{Z}_n$ is $(p-1)(q-1)$)

  .

- Bob computes an integer $d$ with $1 < d < (p-1)(q-1)$ and
  $de \equiv 1 \; mod \; (p-1)(q-1)$.

- ENCRYPTION: A plaintext $m$ is encrypted by computing $c = m^e \; mod \; n$. The chipertext is $c$. If Alice knows the public key $(n, e)$ she can encrypt.

- DECRYPTION: Bob knows the exponent $d$ which he has already chosen. $c^d \; mod \; n = m^{ed} \; mod \; n = m \; mod \; n$

  (as ed = 1 mod $\phi$(n)).

### 1.2.2 Rabin Encryption

This encryption method is almost the same as RSA. The difference between them is that breaking Rabin Encryption system [13] is equivalent to efficiently factoring integers. It is still not known that breaking RSA is equivalent to efficiently factoring integers.

- Alice chooses randomly two large prime numbers $p, q$ with

$$p \equiv q \equiv 3 \ mod \ 4$$

Alice calculates $n = pq$. Her public key is $n$ and her private key is $(p, q)$.

- As in RSA the plaintext is in set $\{0, 1, ..., n-2, n-1\}$ Bob uses public key $n$ and calculates

$$c = m^2 \ mod \ n$$

This $c$ is the ciphertext.

- Alice computes the plain text by calculating the square root of $c$.

$$m_p = c^{(p+1)/4} \ mod \ p, \qquad m_q = c^{(q+1)/4} \ mod \ q$$

Then $\pm m_p + p\mathbb{Z}$ are the two square roots of $c + p\mathbb{Z}$ in $\mathbb{Z}_p$ and $\pm m_q + p\mathbb{Z}$ are the two square roots of $c + q\mathbb{Z}$ in $\mathbb{Z}_q$ Now one of these four is plaintext. But the problem is how does one decide which one is plain text? Alice can choose the candidate which looks most likely, but this might not always work. If the plaintext is chosen is such that some bits in the starting are same as the bits at the end(we are adding some bits in front and back of the message bit), then the output will also have the the similar pattern. This output will be the required square root.

Now we will look into some of the factoring Algorithms:

### 1.2.3 Trail Division

Let the number we want to factor be $n$. To find the small prime factor of n. This is the most primitive algorithm. We have a table containing primes numbers below a bound $B$. Pick a prime divide the number $n$ by maximum possible power of prime. Keep on doing this until all the primes are exhausted. This will lead to a factor of $n$. In fact we will have complete contribution from prime below $B$.

### 1.2.4 $p-1$ method

There are certain algorithm which works good for special type of composite number. This algorithm was given by Pollard. The $(p-1)$ method [14] works good for composite number having prime factor such that $p-1$ has only small prime divisors, where $n$ is the composite number to be factored and $p$ is the prime factor of $n$. The algorithm is based on the fact that given the above condition it is possible to determine a multiple $k$ of $p-1$ without knowing $p-1$ as the product of powers of small prime numbers. Then Fermat's little theorem implies that

$$a^k \equiv 1 \; mod \; p$$

for all integers $a$ that are not divisible by $p$. This means that $p$ divides $a^k - 1$. If $a^k - 1$ is not divisible by $n$, a factor of $n$ is found.

### 1.2.5 Kraitchik's Scheme

There are large number of factoring algorithm which shares a common stratergy. If $n$ is a number to be factored with no small factor and if we can find $x$ and $y$ which satisfies following condition:

- $x^2 \equiv y^2 \bmod n$
- $x \not\equiv \pm y \bmod n$

Then $g = gcd(x-y, n)$ will be a non trivial factor of n.
All sieveing algorithm differ in method and approach of finding $x$ and $y$.

### 1.2.6 Quadratic Sieve

Quadratic sieve works best uptill 115 digit number. The idea of Quadratic sieve was given by *Carl Pomerance* in 1990. The time complexity of this algorithm is $e^{(logn)^{1/2}(loglogn)^{2/3}}$ In nutshell, Quadratic sieve is a method of finding $x$ and $y$. The main steps are as follows:

- Select a factor base $F(B) = \{p \in \mathbb{P} : p \leq B\} \cup \{-1\}$
- Compute $m = \lfloor n \rfloor$
- Choose a sieving interval $S = \{-c, \cdots, c\}$ and do the following:

- Compute $f(s) = (x+m)^2 - n$, and test by dividend the number of $F(B)$ weather this is $B$-smooth or not.
- If some $f(s)$ is $B$-smooth then

$$f(s) = \prod_{j=1}^{t} p_j^{ij} \tag{1.2.1}$$

and define $v_i = (v_i 1, \cdots, v_i t)$, where $v_i j = e_i j \bmod 2$.
$\underline{NOTE:}$ $f(s)$ is called an auxiliary number.

- Use linear algebra over $\mathbb{Z}_2$ to find a non-empty set of $T = \{i\}$ such that $\sum v_i = 0$.
- Compute $x = \prod_{i \in T} \sqrt{f(s) + n} \bmod n$.
- For each $j$, $1 \leq j \leq t$, compute $l_j = \frac{(\sum_{i \in T} e_i j)}{2}$
- Compute $y = \prod_{j=1}^{t} p_j^{l_j} \bmod n$.
- If this $x$ and $y$ satisfies the Kraitchik's criteria then $d = gcd(x - y, n)$ will be a non trivial factor of $n$ else find another set $T$ such that $\sum v_i = 0$ and compute another $x$ and $y$.

# Chapter 2

# Mathematical Preliminaries

In this chapter, we will be discussing some of the basic mathematics which is essential to understand the basic working of the algorithm.

## 2.1 Basic Properties of Integers($\mathbb{Z}$)

In this section we will discuss the basic properties of $\mathbb{Z}$. We will be defining basic things which is necessary for the implementation of algorithm [1].

**Definition 2.1.1** (Greatest Common Divisor). *Greatest Common Divisor of $a, b \in \mathbb{Z}$, denoted by $gcd(a, b)$, is the largest positive integer which divides $a$ and $b$.*

**Definition 2.1.2** (Relative Prime). *Two numbers $a, b \in \mathbb{Z}$, are called relatively prime if $gcd(a, b) = 1$.*

**Definition 2.1.3** (Fundamental theorem of Algebra). *$a \in \mathbb{Z}$ and $a > 1$ then either $a$ is a prime or can be expressed as product of finitely many primes.*

**Definition 2.1.4** (Euler phi function). *Given a positive integer $n$. $\phi(n)$ equals the number of positive integers less than or equal to $n$ and are relatively prime to $n$.*

**Definition 2.1.5** (B-Smooth). *A positive integer is called $B-smooth$ if all of it's factors are less than $B$.*

**Theorem 2.1.6** (Euler's Theorem). *If $n$ is a positive integer with $gcd(a, n) = 1$, then $a^{\phi(n)} \equiv 1(mod)\ n$.*

**Theorem 2.1.7** (Fermat's Little Theorem). *If $p$ is a prime number and $a$ is a integer and $p \not| a$ then $a^{p-1} \equiv 1\ mod\ p$.*

## 2.2 Properties of Polynomial over $\mathbb{Z}$ and $\mathbb{Q}$

In this section we will discuss about the polynomial over integer and rational. The basic form of polynomial is

$$a_n x^n + a_{n-1}x^{n-1} + ... + a_1 x + a_0$$

**Definition 2.2.1.** *Divisibility in $\mathbb{Q}[x]$. Let $f, g \in \mathbb{Q}[x]$, $f$ is called divisor of $g$ if there exists $h \in \mathbb{Q}[x]$, such that $g = f.h$. Otherwise we say $f$ is not a divisor of $g$ and is denoted by $f \not| g$.*

**Definition 2.2.2.** *Irreducibility and Prime in $\mathbb{Q}[x]$. An element $f$ of $\mathbb{Q}[x]$ is called a zero polynomial if $f = 0$. An element $f$ of $\mathbb{Q}[x]$ is called a unit if $f/1$. An element $f$ of $\mathbb{Q}[x]$ is called irreducible if given $f = g.h$, where $g, h \in \mathbb{Q}[x]$ then either $f$ or $g$ is a unit. An element $f$ of $\mathbb{Q}[x]$ is called prime if given $f/g.h$, where $g, h \in \mathbb{Q}[x]$ then $f/g$ or $f/h$ or both. If none of the above then the element is called irreducible.*
*Associates :* *$f, g \in \mathbb{Q}[x]$ are called associates if $f = gu$, where $u$ is unit.*

## 2.3 Basic Algebraic Number Theory

**Definition 2.3.1** (Number Field)**.** *If $r$ is an algebraic number of degree $n$ , then the totality of all expressions that can be constructed from $r$ by repeated additions,subtractions, multiplications and divisions is called a number field (or an algebraic number field) generated by $r$ , and is denoted by $F[r]$.Formally, a number field is a finite extension $\bar{Q}$ of the field $Q$ of rational numbers.*

**Definition 2.3.2** (Algebraic Number)**.** *If $r$ is a root of a nonzero polynomial equation $a_n x^n + a_{n-1}x^{n-1} + ... + a_0$,*
*where the $a_i$'s are rational numbers and satisfies no similar equation of degree $< n$ ,then $r$ is said to be an algebraic number of degree $n$.*

**Definition 2.3.3** (Algebraic Integer)**.** *An element $\delta$ is an algebraic integer if it is a root of some monic polynomial with coefficient in $\mathbb{Z}$. If Number Field(2.3.1) is $K$, then ring of algebraic integer is denoted by $O_K$. Note that, in general case $O_K \neq \mathbb{Z}[\alpha]$, where $\mathbb{Z}[\alpha]$ is $\mathbb{Z}[x]/f(x)$, $f(x)$ is irreducible polynomial over $\mathbb{Z}[x]$.*

**Definition 2.3.4** (Norm Map)**.** *We can view the Algebraic Number Field as a finite dimensional vector space over $Q$. Then if $\alpha \in Q$, the map form $K$ to $K$ defined*

*by $\phi_\alpha : v \longrightarrow \alpha v$ define a linear operator on $K$ and we define the norm map as determinant of that map $N_K := det(\phi_\alpha)$.*
*we define the norm of an non zero ideal in $O_K$ to be its index in $O_K$.*

**Definition 2.3.5** (Noetherian Ring)**.** *A ring is called Noetherian if every ascending chain $\mathcal{A}_1 \subseteq \mathcal{A}_2 \subseteq \mathcal{A}_3 \subseteq ...$ of ideals terminates,i.e., if there exists $n$ such that $\mathcal{A}_n = \mathcal{A}_{n+k}$ for all $k \geq 0$.*

**Definition 2.3.6** (Dedekind Domain)**.** *A commutative integral domain which satisfies these conditions is called a Dedekind domain:*

- *Integrally Closed.*
- *Every non zero prime ideal is maximal.*
- *It should be Noetherian.*

**Definition 2.3.7** (First Degree Prime Ideal)**.** *A first degree prime ideal $\mathcal{P}$ of a Dedekind Domain $D$ is a prime ideal of $D$ such that $N(\mathcal{P})= p$ for some integer $p$.*

**Lemma 2.3.8.** *If $R$ is a commutative ring with $1_R$, $S$ is a commutative ring with identity $1_S$, and $\phi : R \longrightarrow S$ ring epimorphish then $\phi(1_R) = 1_S$.*

*Proof.* Let $y \in$ S. As $\phi$ is a ring epimorphism there exists $x \in$ R such that $\phi(x) = y$. Then $y.\phi(1_R)= \phi(1_R).y = \phi(1_R).\phi(x) = \phi(1_R.x)= \phi(x) = y$ hence $\phi(1_R) = 1_S$.

□

**Theorem 2.3.9.** *Let $R$ be a commutative ring with identity. Then:*

*a) $\mathcal{M}$ is a maximal ideal if and only if $R/\mathcal{M}$ is a field.*

*b) $\mathcal{P}$ is a prime ideal if and only if $R/\mathcal{P}$ is an integral domain.*

*c) $\mathcal{A}$ and $\mathcal{B}$ be ideals of $R$. If $\mathcal{P}$ is a prime ideal containing $\mathcal{AB}$, then $\mathcal{P} \supseteq \mathcal{A}$ or $\mathcal{P} \supseteq \mathcal{B}$.*

*d) If $\mathcal{P}$ is a prime ideal containing the product $\mathcal{A}_1\mathcal{A}_2...\mathcal{A}_r$ of $r$ ideals of $R$, then $\mathcal{P} \supseteq \mathcal{A}_i$, for some $i$.*

*Proof.* a) We know there is a bijection between the ideals of R containing $\mathcal{M}$ and ideals of $R/\mathcal{M}$. By this correspondence $R/\mathcal{M}$ has a non trivial ideal if and only if there is an ideal $\mathcal{A}$ of $R$ strictly between $\mathcal{M}$ and $R$. Thus, $\mathcal{M}$ is maximal, $\Rightarrow R/\mathcal{M}$ has no non trivial ideals, $\Rightarrow R/\mathcal{M}$ is a field.

b) Let $\mathcal{P}$ be a prime ideal means, If $ab \in \mathcal{P}$ then either $a$ or $b \in \mathcal{P}$. we know $ab + \mathcal{P} = 0 + \mathcal{P}$ in $R/\mathcal{P}$. $\Rightarrow a + \mathcal{P} = 0 + \mathcal{P}$ or $b + \mathcal{P} = 0 + \mathcal{P}$ in $R/\mathcal{P}$, $\Rightarrow R/\mathcal{P}$ has no zero divisors, $\Rightarrow R/\mathcal{P}$ is an integral domain.

c) Suppose that $\mathcal{P} \supseteq \mathcal{AB}$, $\mathcal{P} \not\supseteq \mathcal{A}$. Let $a \in \mathcal{A}$, $a \notin \mathcal{P}$. We know $a\mathcal{B} \in \mathcal{P}$ for all b $\in \mathcal{B}$ since $a\mathcal{B} \subseteq \mathcal{P}$. But, $a \notin \mathcal{P}$. Thus $b \in \mathcal{P}$ for all $b \in \mathcal{B}$, since $\mathcal{P}$ is prime. Thus $\mathcal{B}$ $\subseteq \mathcal{P}$.

d)We will prove this by induction. For $r$=1 it is trivial. Let $r > 1$ and $a_1a_2...a_{r-1}a_r$ $\subseteq \mathcal{P}$. By (c), $\mathcal{A}_1\mathcal{A}_2...\mathcal{A}_{r-1} \subseteq \mathcal{P}$ or $\mathcal{A}_r \subseteq \mathcal{P}$. If $\mathcal{A}_1\mathcal{A}_2...\mathcal{A}_{r-1} \subseteq \mathcal{P}$ then the induction hypothesis implies that $\mathcal{A}_i \subseteq \mathcal{P}$ for some $i \in \{1, 2, ..., r-1\}$. In either case, $\mathcal{A}_i \subseteq \mathcal{P}$ for some $i \in \{1, 2, ..., r\}$. □

**Theorem 2.3.10.** *For any commutative ring with identity, the following are equivalent*

*a) R is Noetherian;*

*b) every nonempty set of ideals contains a maximal element; and*

*c) every ideal of R is finitely generated.*

*Proof.* a) $\Rightarrow$ b) Given R is Noetherain Suppose S is non-empty set of ideals of R that does not contain a maximal element. Let $\mathcal{A}_1 \in$ S. $\mathcal{A}_1$ is not maximal in S. So there exists an ideal $\mathcal{A}_2$ in $S$ such that $\mathcal{A}_1 \subset \mathcal{A}_2$. Now $\mathcal{A}_2$ is not maximal element of S. So there exists an ideal $\mathcal{A}_3$ in S such that $\mathcal{A}_2 \subset \mathcal{A}_3$. Continuing this we get a infinite ascending chain of ideals in R. This contradicts our assumption of R being Noetherian. Hence, every nonempty set of ideals contains a maximal element.

b) $\Rightarrow$ c) Let $\mathcal{B}$ be an ideal of R. Let A be the set of ideals contained in $\mathcal{B}$ which are finitely generated. A is nonempty as (0) $\in A$. Thus, by b) $A$ has a maximum element, say $\mathcal{A} = (x_1, x_2, ..., x_{n-1}, x_n)$. if $A \neq \mathcal{B}$, then $\exists\ x \in \mathcal{B}$ but $\notin A$. But then $A + (x) = (x_1, x_2, ..., x_{n-1}, x_n, x)$ is a larger finitely generated ideal contained in $\mathcal{B}$, this contradicts the maximality of $A$. Thus, $\mathcal{B} = A$. Thus $\mathcal{B}$ is finitely generated.

c) $\Rightarrow$ a) Let $\mathcal{A}_1 \subseteq \mathcal{A}_2 \subseteq \mathcal{A}_3 \subseteq ...$ be a ascending chain of ideals of R. Thus $\cup_{i=1}^{\infty}\mathcal{A}_i$ is also an ideal of R, and so is finitely generated. say $\mathcal{A} = (x_1, x_2, ..., x_{n-1}, x_n, x)$.

Then $x_1 \in \mathcal{A}_{i_1}, ..., x_n \in \mathcal{A}_{i_n}$. Let $m = max(i_1, ...i_n)$. Then $\mathcal{A} \subseteq \mathcal{A}_m$, so $\mathcal{A} = \mathcal{A}_m$. Thus $\mathcal{A}_m = \mathcal{A}_{m+1} = ...$ and the chin does terminate. Thus, R is Noetherian. □

**Theorem 2.3.11.** *Any ideal of $O_K$ is can be written as a product of prime ideals uniquely.*

*Proof.* **Existence:** Let S be a set of ideals of $O_K$ that cannot be written as a product of prime ideals. Now we will try to prove that this set is empty and this completes the proof. We know $O_K$ is Noetherian, and assuming that this set,S, is non-empty we have a maximal element in the set, say $a$. Then $a \subseteq p$ for some maximal ideal $\mathcal{P}$, as $O_K$ is Noetherian. From [2.17] Every maximal ideal of $O_K$ is prime. And hence $\mathcal{P}$ is a prime ideal. Therefore $\mathcal{A} \neq \mathcal{P}$ and $\mathcal{A}$ is not prime or else this contradicts our assumption. Before going ahead we see a lemma.
*FractionalIdeal :* A *fractional ideal* $\mathcal{A}$ of $O_K$ is an $O_K$ module contained in $K$ such that there exists $m \in \mathbb{Z}$ with $m\mathcal{A} \subseteq O_K$.([2] page 57). Let $\mathcal{P}$ be a prime ideal. Define

$$\mathcal{P}^{-1} = \{x \in K : x\mathcal{P} \subseteq O_K\}$$

*Lemma :* Let $\mathcal{P}$ be a prime ideal of $O_K$. Then $\mathcal{P}^{-1}$ is a fractional ideal and $\mathcal{P}\mathcal{P}^{-1}$= $O_K$.([2] page 58) Going back to proof, Consider $\mathcal{P}^{-1}\mathcal{A}$. $\mathcal{P}^{-1}\mathcal{A} \subset \mathcal{P}^{-1}\mathcal{P}$= $O_K$. Since $\mathcal{A} \subset \mathcal{P}$,

$$\mathcal{P}^{-1}\mathcal{A} \subseteq \mathcal{P}^{-1}\mathcal{P} = O_K$$

Since for any $x \in \mathcal{P}$ but not in $\mathcal{A}$,

$$\mathcal{P}^{-1}x \subseteq \mathcal{P}^{-1}\mathcal{A} \Rightarrow x \in \mathcal{P}\mathcal{P}^{-1}a = O_K\mathcal{A} = \mathcal{A},$$

which is not true. Means that $\mathcal{P}^{-1}\mathcal{A}$ is a proper ideal of $O_K$, and contains $\mathcal{A}$ properly since $\mathcal{P}^{-1}$ contains $O_K$ properly. Thus, $\mathcal{P}^{-1}\mathcal{A} \notin S$, since $\mathcal{A}$ is a maximal element in S. Thus, $\mathcal{P}^{-1}\mathcal{A} = \mathcal{P}_1...\mathcal{P}_r$, for some prime ideals $\mathcal{P}_i$. Then, $\mathcal{P}\mathcal{P}^{-1}\mathcal{A} = \mathcal{P}\mathcal{P}_1...\mathcal{P}_r$, so $\mathcal{A} = \mathcal{P}\mathcal{P}_1...\mathcal{P}_r$. But then $\mathcal{A} \notin S$, this is a contradiction. Thus, S is empty, so every ideal of $O_K$ can be written as a product of prime ideals. □

Uniqueness:Assume that this ideal factorization is not unique. So we can write $\mathcal{A} = \mathcal{P}_1...\mathcal{P}_r = \mathcal{P}_1'...\mathcal{P}_s'$. Take $\mathcal{P}_1$, as $\mathcal{P}_1$ is an ideal of $O_K$ we have

$$\mathcal{P}_1' \supseteq \mathcal{P}_1'...\mathcal{P}_s' = \mathcal{P}_1...\mathcal{P}_r$$

so, $\mathcal{P}_1' \supseteq \mathcal{P}_i$ for some i, say $\mathcal{P}_1' \supseteq \mathcal{P}_1$. But $\mathcal{P}_1$ is maximal (as primes are maximal in $O_K$), so $\mathcal{P}_1' = \mathcal{P}_1$. Now multiplying both side by $(\mathcal{P}_1')^{-1}$, we get,

$$\mathcal{P}_2^{'}...\mathcal{P}_s^{'}= \mathcal{P}_2^{'}...\mathcal{P}_r^{'}$$

Continuing the same thing completes the proof.

# Chapter 3

# Description of number field sieve factoring algorithm

One question that everyone asks about the *number field sieve* algorithm is: Why is this algorithm faster than all other known algorithms?

The reason for superior performance of *general* number field sieve (GNFS) over quadratic field sieve is that in Quadratic Sieve we require auxiliary numbers of size $O(\sqrt{n})$ and in GNFS we need smaller auxiliary number(for explanation see (1.2.1)) to be smooth they are of the size

$$\exp\left(c'(\log n)^{2/3}(\log\log n)^{1/3}\right) \tag{3.0.1}$$

where $c^{'}$ is 2.77 approximately. In other words in GNFS the length of these numbers is not half of the length of $n$, but it is only the $2/3 - rd$ power of length of $n$ [3].

## 3.1 Steps involved in the algorithm

- Given a number $n$ to be factored, first check that it is neither prime, nor a perfect square. For this we can use primality testing [15].

- Once these conditions are checked calculate the degree of the polynomial by

$$d = Round(3log(n)/log(log(n)))^{1/3} \tag{3.1.1}$$

  Where $round()$ is integer round off function.

- Calculate $m = (n)^{1/d}$. Now write base $m$ expansion of $n$ i.e.

$$n = m^d + a_{d-1}m^{d-1} + ... + a_1 m + a_0$$

Define $f(x)$ as:

$$f(x) = x^d + a_{d-1}x^{d-1} + ... + a_1 x + a_0 \tag{3.1.2}$$

- We can assume that this polynomial is irreducible because if it is reducible there is no point in using this algorithm. Let

$$f(x) = l(x)v(x)$$

then, by substituting $x$ by $m$ we get

$$f(m) = l(m)v(m) = n$$

so the number $n$ is factored without using Number Field Sieve Factoring Algorithm.

- Assume $\alpha$ be a root of this polynomial. As $f(x)$ is irreducible over $\mathbb{Z}[x]/f(x)$ so this $\alpha$ has to be complex root of $f(x)$. Decide upon the bound, say B for factor base construction.
With $f(x)$( (3.1.2)) in hand we have a natural homomorphism in that is

$$\phi : \mathbb{Z}[\alpha] \to \mathbb{Z}/n\mathbb{Z} \tag{3.1.3}$$

This maps take polynomial $f(\alpha)$ to $f(m)$. And we try to construct square in $\mathbb{Z}[\alpha]$ using smoothness(2.1.5) and take the image of it under $\phi$ and use congruent squares to find a factor of $n$.
This is exactly similar to what we do in Quadratic Sieve(1.2.6).
Algebraic factor base is the factor base which is used to find the smooth elements(2.1.5) over $\mathbb{Z}/n\mathbb{Z}$.
Rational factor base is the factor base which is used to find the smooth elements(2.1.5) over $\mathbb{Z}[\alpha]$.

- Construct the Algebraic factor base(AFB): Take primes upto bound B and construct the set.

- Construct the Rational Factor Base(RFB): Decide on bound $B^{'}$ and construct a set say E Pick prime $p$ from the set E now find the roots of $f(x)$ inside $F_p$, say $r_i$ be the set of roots w.r.t. $p$ then store $(r_1, p)$, $(r_2, p)$,...$(r_k, p)$. Now pick another element and repeat the process again until all the elements of E are exhausted.

- Construct the Quadratic Character Base(QCB): Pick a prime greater than $B^{'}$ and repeat the above process. Take the next prime until we get a set half the size of AFB.
  Relation We call $(a, b)$ a relation if $a + bm$ is smooth over the AFB and $a + b\alpha$ is smooth over RFB.

- After this we collect relations $(a, b)$ which satisfies given conditions:

$$\prod_{(a,b)} a + bm \qquad is\ \ a\ \ square\ \ in\ \ \mathbb{Z} \tag{3.1.4}$$

$$\prod_{(a,b)} a + b\alpha \qquad is\ \ a\ \ square\ \ in\ \ \mathbb{Z}[\alpha] \tag{3.1.5}$$

  We first sieve over RFB and then check those for smoothness over AFB. It is quite clear that for QCB we just need to check $B$ smoothness of $a + bm$. The question is how do we check for smoothness over AFB? For this we use norm map. If the Norm of $a + b\alpha$ is smooth, we call that $a + b\alpha$ is smooth. This information can be used only to construct a product

$$\prod_{(a,b)} a + b\alpha$$

  with square Norm. A square in $\mathbb{Z}[\alpha]$ implies the norm of it will be a square but converse is far from being true. for example to demonstrate this:
  **Example:** In the ring of Gaussian integers $\mathbb{Z}[i]$ we have

$$N(3 + 4i) = 3^2 + 4^2 = 5^2 = N(5).$$

  Now $3 + 4i = (2 + i)^2$ is a square but $5 = (2 + i)(2 - i)$ is not. To provide a better level check we use Quadratic character.

- We use quadratic character to refine the sieve

$$\left(\frac{\gamma}{\mathcal{P}}\right) := \begin{cases} +1 & \text{if } \gamma \text{ is a non zero square modulo } \mathcal{P}, \\ -1 & \text{if } \gamma \text{ is a not a square modulo } \mathcal{P} \end{cases}$$

  where $\mathcal{P}$ is an element of QCB, and $\gamma \in O_K$. Using this we emphasize on $\prod_{(a,b)\in S}(a+bm)$ is a square in $\mathbb{Z}$ and that $\prod_{(a,b)\in S}(a+b\alpha)$ generates a square ideal in $O_K$ and $\left(\frac{\gamma}{\mathcal{P}}\right) = 1$ for $i = 1, ...s$ where $s$ is the number of elements in quadratic character base.

- After finding enough number of relations we do sieving and construct a matrix similar to what we do in Quadratic Sieve(1.2.6). We reduce that matrix(mostly we use gausian elimination method) to find out the null space of it.

- After getting the null space we pick up a vector from it and using that vector we calculate the product of $(a+mb)$ and $(a+\alpha b)$. Do the square root extraction on Algebraic part and we get $\nu$ and square root of rational part say $\mu$.

- Use the natural homomorphism

$$\phi \ : \ \mathbb{Z}[\alpha] \ \longrightarrow \ \mathbb{Z}/n\mathbb{Z}$$

  where $x^d + a_{d-1}\alpha^{d-1} + ... + a_1\alpha + a_0$ goes to $x^d + a_{d-1}m^{d-1} + ... + a_1 m + a_0$ to find the image of $\nu$ in $\mathbb{Z}/n\mathbb{Z}$, lets call it as $u$ now we have $\mu^{\,2} = u^{\,2} \bmod n$.

- Use Kraitchik scheme(1.2.5) and find $GCD(\mu - u, n)$ if its non trivial we get a factor of $n$. If it is non trivial repeat the process.

This algorithm is called General Number Field Sieve factoring algorithm because this can be used to factor any general $n$. Earlier version of the algorithm which was used to factor "Fermat's Number", had a big assumption i.e. $\mathbb{Z}[\alpha] = O_K$(2.3.3). In the general case we may not have any of the nice property that we demand from the ring in which we want to work. In this section we will discuss some of the theorems which will be used as the basic building block of the algorithm. Our sole aim here will be to describe how we can mimic the sieving done in the quadratic sieve in this algebraic set up.After that we will handle some of the problem that arises in mimicking the sieve. And at last we will state the obstacle that is there in implementing the

algorithm for general set up. Before stating the first theorem of the section we need to define
Canonical notation of ideal multiplication: Given two ideals $\mathcal{A}, \mathcal{B}$ of ring $R$, their multiplication $\mathcal{AB} = \{a_i b_j \mid a_i \in \mathcal{A}, b_j \in \mathcal{B}\}$

## 3.2 Setting Up the Factor bases

In Quadratic Sieve, we have factor base which is a collection of primes in $\mathbb{Z}$ upto some bound. If we want to mimic the same idea in $\mathbb{Z}[\alpha]$ we collect prime elements of ring $\mathbb{Z}[\alpha]$. But the idea is to collect the prime ideals instead of primes in $\mathbb{Z}[\alpha]$. The major reason behind this is group of units, in earlier implementations they have chosen such a $n$, for which $\mathbb{Z}[\alpha]$ has finitely generated unit set.
The first question is how do we decompose any ideal into prime ideals? Following theorem will take care of it.

**Theorem 3.2.1.** *Given a monic, irreducible polynomial $f(x)$ of degree $d$ with integer coefficients and a root $\alpha \in \mathbb{C}$ of $f(x)$, then the ring of algebraic integers(2.3.3) $O_K$ forms a Dedekind Domain.*

- *The ring $O_K$ is Noetherian.*
- *Prime ideals of $O_K$ are maximal ideals of $O_K$.*
- *Using the canonical notations of ideal multiplication, ideals of $O_K$ can be uniquely factored, up to order,into prime ideal of $O_K$*

*Proof.* [2][Chapter 5](page.56-59) □

**Theorem 3.2.2.** *Given a monic, irreducible polynomial $f(x)$ of degree $d$ with rational coefficients and a root $\alpha \in \mathbb{C}$ of $f(x)$, then the norm map maps element of $\mathbb{C}(\alpha)$ to $\mathbb{C}$. Furthermore, algebraic integers in $\mathbb{Q}(\alpha)$ are mapped to elements in $\mathbb{Z}$. In fact we can say that it sends elements of $\mathbb{Z}[\alpha]$ to elements of $\mathbb{Z}$.*

*2.* [Chapter 4](page.40) □

This theorem is important because this tells us that the ideals of $\mathbb{Z}[\alpha]$ are mapped to $\mathbb{Z}$ by norm function. We will use this theorem when we will do sieving on rational part.

**Theorem 3.2.3.** *Let $f(x)$ be a monic irreducible polynomial of degree d with rational coefficient and $\alpha \in \mathbb{C}$ Then the norm function maps ideal of $O_K$ to positive integer. If $\theta \in O_K$ then*
*$N(\langle\theta\rangle) = \mid N(\theta) \mid$ where $N(\langle\theta\rangle)$ is the norm of ideal generated by $\theta$ and $\mid N(\theta) \mid$ modulus of norm of element $\theta$*

*Proof.* [**?**][Chapter 4](page 40) □

**Proposition 3.2.4.** *Let $D$ be a Dedekind Domain. If $\mathcal{P}$ is a prime ideal of $D$ with $N(\mathcal{P}) = p$ for some prime integer $p$, then $\mathcal{P}$ is a prime ideal of $D$. Conversely, if $\mathcal{P}$ is a prime ideal of $D$ then $N(\mathcal{P}) = p^e$ for some prime integer $p$ and positive integer $e$.*

Given any element $\beta \in O_K$ from above proposition we can say that $\langle\beta\rangle$ of $O_K$ factors uniquely as

$$\langle\beta\rangle = \mathcal{P}_1^{e_1}\mathcal{P}_2^{e_2}\mathcal{P}_3^{e_3}\cdots\mathcal{P}_k^{e_k}$$

for distict prime ideals $\mathcal{P}_i$ of $O_K$ and positive integer $e_i$ with $1 \leq i \leq k$. Also,

$$\begin{aligned}
\mid N(\beta) \mid = N(\langle\beta\rangle) &= N(\mathcal{P}_1^{e_1}\mathcal{P}_2^{e_2}\mathcal{P}_3^{e_3}\cdots\mathcal{P}_k^{e_k}) && (3.2.1)\\
&= N(\mathcal{P}_1^{e_1})N(\mathcal{P}_2^{e_2})N(\mathcal{P}_3^{e_3})\cdots N(\mathcal{P}_k^{e_k}) && (3.2.2)\\
&= (p_1^{f_1})^{e_1}(p_2^{f_2})^{e_2}(p_3^{f_3})^{e_3}\cdots(p_k^{f_k})^{e_k} && (3.2.3)\\
&= (p_1^{f_1+e_1})(p_2^{f_2+e_2})(p_3^{f_3+e_3})\cdots(p_k^{f_k+e_k}) && (3.2.4)
\end{aligned}$$

Here $\mathcal{P}_i$ is prime ideals, $e_i$ is the exponent of the ideal corresponding to $\mathcal{P}_i$ and the norm of $\mathcal{P}_i$ is $p_i^{f^i}$.
NOTE: Any ideal $\mathcal{P}$ of a ring $R$ with $N(\mathcal{P}) = p$ for some prime integer $p$ is necessarily a prime ideal of $R$. Because $[R : \mathcal{P}] = p$ implies $R/\mathcal{P} \cong \mathbb{Z}/p\mathbb{Z}$ which is a field hence $\mathcal{P}$ is maximal ideal and in Dedekind Domain maximal ideals are prime.
Here instead of prime ideals of $O_K$ we will consider the prime ideals of $\mathbb{Z}[\alpha]$ because we can easily handle the first degree prime ideals of $\mathbb{Z}[\alpha]$ and with these type of ideals only in the factor base it is easier to find smooth $a + b\alpha$. Next theorem talks about how we can view the prime ideals in $\mathbb{Z}[\alpha]$, that is easier to work with, easier to store in computer and easily manageable.

**Theorem 3.2.5.** *Let $f(x)$ be a monic, irreducible polynomial with integer coefficient and let $\alpha \in \mathbb{C}$ be a root of $f(x)$. The set of pairs $(r, p)$ where $p$ is a prime integer and*

*$r \in \mathbb{Z}/p\mathbb{Z}$ with $f(r) = 0 \bmod p$ is in bijective correspondence with the set of all first degree prime ideals of $\mathbb{Z}[\alpha]$.*

*Proof.* Let $\mathcal{P}$ be a first degree prime ideal of $\mathbb{Z}[\alpha]$, so we have $[\mathbb{Z}[\alpha] : \mathcal{P}] = p$, for some prime $p$.
This implies that $\mathbb{Z}[\alpha]/\mathcal{P}$ is isomorphic to $\mathbb{Z}/p\mathbb{Z}$. Which gives us that there is function $\phi : \mathbb{Z}[\alpha] \to \mathbb{Z}[\alpha]/\mathcal{P}$ such that $ker(\phi) = \mathcal{P}$.
$\Rightarrow \phi : \mathbb{Z}[\alpha] \longrightarrow \mathbb{Z}/p\mathbb{Z}$ such that $ker\ \phi = \mathcal{P}$. As, $\phi(1) = 1$,$\phi(a)= a \bmod p$ means constants are fixed under homomorphism $\phi$ where $a \in \mathbb{Z}$.Now, Let $r= \phi(\alpha) \in \mathbb{Z}/p\mathbb{Z}$. If $f(x) = x^d + a_{d-1}x^{d-1} + ... + a_1x + a_0$ with $a_i \in \mathbb{Z}$, for $0 \leq i < d$, then $\phi(f(\alpha)) \equiv 0 \bmod p$ , since $f(\alpha) = 0$.
implies that $0 \equiv \phi(\alpha^d + a_{d-1}\alpha^{d-1} + ... + a_1\alpha + a_0)$
$\equiv \phi(\alpha)^d + a_{d-1}\phi(\alpha)^{d-1} + ... + a_1\phi(\alpha) + a_0$
$\equiv (r^d + a_{d-1}r^{d-1} + ... + a_1r + a_0)$
$\equiv f(r) \bmod p$
so that $r$ is a root of $f(x) \bmod p$ and the ideal $\mathcal{P}$ determines the unique pair $(r, p)$.

**Convesely,** Let $p$ be a prime integer and $r \in \mathbb{Z}/p\mathbb{Z}$ with $f(r) \equiv 0 \bmod p$. Then there exists a homomorphism that maps $\alpha$ to $r$ and polynomials in $\alpha$ to polynomial in $r$.
In particular, $\phi(a) \equiv a \bmod p$, $\forall a \in \mathbb{Z}$. Let $\mathcal{P}= ker(\phi)$. So, $\mathcal{P}$ is an ideal in $\mathbb{Z}[\alpha]$. Since $\phi$ is onto and $ker\phi= \mathcal{P}$ it follows that $\mathbb{Z}[\alpha]/\mathcal{P} \equiv \mathbb{Z}/p\mathbb{Z}$ and hence $[\mathbb{Z}[\alpha] : \mathcal{P}] = p$ and $\mathcal{P}$ therefore is first degree prime ideal of $\mathbb{Z}[\alpha]$. Thus the pair $(r, p)$ determines a unique prime ideal $\mathcal{P}$. This establishes the bijection. Now once we have collection of ideals ready, we can look at divisibility of ideals. □

**Definition 3.2.6.** *For $\mathcal{A}$,$\mathcal{B}$ ideals of $O_K$, we say $\mathcal{A}$ divides $\mathcal{B}$ (denoted by $\mathcal{A} \mid \mathcal{B}$), if $\mathcal{A} \supseteq \mathcal{B}$.*

**Definition 3.2.7.** *Let $e_{\mathcal{P}_i}$: $\mathbb{Q}(\alpha)^*$ (non zero elements of number field $\bar{Q}$ 2.3.1) $\to \mathbb{Z}$ be a function for a fixed prime ideal $\mathcal{P}_i$ and is homomorphism with following properties:*

- *$e_{\mathcal{P}_i}(\beta) \geq 0$ for all $\beta \in \mathbb{Q}(\alpha)^*$.*
- *$e_{\mathcal{P}_i}(\beta) > 0$ if and only if the ideal $\mathcal{P}_i$ divides the principal ideal $\langle\beta\rangle$.*
- *$e_{\mathcal{P}_i}(\beta) = 0$ for all but a finite number of prime ideal $\mathcal{P}_i$ of $O_K$ and $\mid N(\beta) \mid = \prod N(\mathcal{P}_i)^{e_{\mathcal{P}_i}}$*

by Jordan-Holder Theorem( [16])

**Proposition 3.2.8.** *For every prime ideal $\mathcal{P}_i$ of $\mathbb{Z}[\alpha]$, there exists a group homomorphism $l_{\mathcal{P}_i}$:$\mathbb{Q}(\alpha)_* \longrightarrow \mathbb{Z}$ that posses the following properties:*

- *$l_{\mathcal{P}_i}(\beta) \geq 0$ for all $\beta \in \mathbb{Q}(\alpha)^*$.*
- *$l_{\mathcal{P}_i}(\beta) > 0$ if and only if the ideal $\mathcal{P}_i$ divides the principal ideal $\langle\beta\rangle$.*
- *$l_{\mathcal{P}_i}(\beta) = 0$ for all but a finite number of prime ideal $\mathcal{P}_i$ of $\mathbb{Z}[\alpha]$ and $\mid N(\beta) \mid = \prod N(\mathcal{P}_i)^{l_{\mathcal{P}_i}}$ for all prime ideals $\mathcal{P}_i$ of $\mathbb{Z}[\alpha]$.*

*Proof.* For proof ([6]) □

This above homomorphism works as exponent map.

## 3.3 Finding the exponents

**Theorem 3.3.1.** *Given an element $\beta \in \mathbb{Z}[\alpha]$ of the form $\beta = a+b\alpha$ for co-prime integers $a$ and $b$ and a prime ideal $\mathcal{P}$ of $\mathbb{Z}[\alpha]$, then the homomorphism $l_{\mathcal{P}}$, corresponding to $\mathcal{P}$ has $l_{\mathcal{P}}(\beta)=0$ if $\mathcal{P}$ is not a first degree prime ideal of $\mathbb{Z}[\alpha]$. Furthermore, if $\mathcal{P}$ is a first degree prime ideal of $\mathbb{Z}[\alpha]$ corresponding to the pair $(r, p)$ then*

$$l_{\mathcal{P}}(\beta) = \begin{cases} ord_p(N(\beta)) & \text{if } a \equiv -br \mod p \\ 0 & \text{otherwise} \end{cases}$$

*where $ord_p(N(\beta))$ denote the exponent of the prime integer $p$ occurring in the unique factorization of the integer $N(\beta)$ into distinct primes.*

*Proof.* Let $\mathcal{P}$ be a prime ideal of $\mathbb{Z}[\alpha]$ with $l_{\mathcal{P}}(a + b\alpha) > 0$. Thus $\mathcal{P}$ serves as kernel of epimorphism $\phi : \mathbb{Z}[\alpha] \longrightarrow \mathbb{Z}[\alpha]/\mathcal{P}$.

Now $\mathbb{Z}[\alpha]/\mathcal{P} \equiv \mathbb{F}_{p^e}$ where $p$ is a prime, $e$ is a positive integer and $\mathbb{F}_{p^e}$ denote the finite field with $p^e$ elements. This means that $\mathbb{Z}[\alpha]/\mathcal{P}$ must contain an isomorphic copy of $\mathbb{Z}/p\mathbb{Z}$. Now we will try to show that $Image(\phi) = \mathbb{Z}/p\mathbb{Z}$ then by first isomorphism theorem $\mathbb{Z}[\alpha]/ker(\phi) \equiv Image(\phi)$ and $ker(\phi) = \mathcal{P}$ This implies $\mathbb{Z}[\alpha]/\mathcal{P} \equiv \mathbb{Z}/p\mathbb{Z}$ and $\mathcal{P}$ is the first degree prime ideal of $\mathbb{Z}[\alpha]$. Since $\phi$ is an epimorphism of rings $\phi(1) = 1 \in \mathbb{Z}/p\mathbb{Z}$ and hence $\phi(m) \equiv m \ mod \ p \ \forall m \in \mathbb{Z}$. If $l_{\mathcal{P}}(a+b\alpha)$, $\mathcal{P} \mid \langle a+b\alpha\rangle \Rightarrow a+b\alpha \in \mathcal{P}$. But since $ker(\phi) = \mathcal{P}$ it follows that $\phi(a + b\alpha) \equiv 0 \ mod \ p$. Now, Suppose $b \in \mathbb{Z}$ and $p \mid b$. It follows from $\phi(a + b\alpha) \equiv 0 \ mod \ p$ and $\phi(b) \equiv b \equiv 0 \ mod \ p$ that

$$0 \equiv \phi(a + b\alpha) \equiv a + b\phi\alpha \equiv a \ mod\ p \text{ and hence } p \mid a$$

This contradicts our assumption of a and b being coprime. Hence $b$ cannot be divisible by $p$. And since $b$ is not divisible by $p$ there exists a inverse of $b$, say $b^{-1}$ in $\mathbb{Z}/p\mathbb{Z}$. from above equation, a+b$\phi(\alpha)$ *equiv* 0 *mod p.* $\Rightarrow \phi(\alpha) \equiv$ -ab$^{-1}$ *mod p.* $\Rightarrow \phi(\alpha) \in \mathbb{Z}/p\mathbb{Z}$ Hence, $\mathbb{Z}/p\mathbb{Z} \subseteq \phi(\mathbb{Z}[\alpha]) \subseteq \mathbb{Z}/p\mathbb{Z}$. Therefore, $Image(\phi) = \mathbb{Z}/p\mathbb{Z}$. For Second Part, We first prove that $l_{\mathcal{P}}(a+b\alpha) > 0$ for first degree prime ideal $\mathcal{P}$ with pair $(r, p)$ if and only if $a \equiv -br \ mod\ p$. Let $l_{\mathcal{P}}(a+b\alpha) \Rightarrow a + b\alpha \in \mathcal{P}$. Now, $\mathcal{P} = ker(\phi)$ where $\phi : \mathbb{Z}[\alpha] \longrightarrow \mathbb{Z}/p\mathbb{Z}$.
$\phi(\alpha) = r \ mod\ p$ and $\phi(a) = a \ mod\ p \ \forall \ a \in \mathbb{Z}$. But then $0 \equiv \phi(a + b\alpha) = a + br \ mod\ p \Rightarrow a \equiv -br \ mod\ p$.

**Conversely:** Suppose $a \equiv -br \ mod\ p$ for first degree prime ideal $\mathcal{P}$ with pair $(r, p)$. Then $0 \equiv a + br \ mod\ p = \phi(a + b\alpha)$.
Hence, $a + b\alpha \in ker(\phi) = \mathcal{P}$ which implies that $\mathcal{P}$ divides $\langle a + b\alpha \rangle$ and hence $l_{\mathcal{P}}(a + b\alpha) > 0$.
Next, it will be shown that for a first degree prime ideal $\mathcal{P}$ of $\mathbb{Z}[\alpha]$ with pair $(r, p)$ that $N(a + b\alpha)$ is divisible by $p$ if and only if $a \equiv -br \ mod\ p$.
Combining this with what we have proved just now implies $l_{(a+b\alpha)}$ if and only if $p \mid N(a + b\alpha)$.
We Know, $N(a + b\alpha) = \sigma_1(a + b\alpha).\sigma_2(a + b\alpha)... \ \sigma_d(a + b\alpha)$

$$(a + b\alpha_1).(a + b\alpha_2)....(a + b\alpha_d)$$
$$(-b)^d) \ (-a/b - \alpha_1).(-a/b - \alpha_2).... \ (-a/b - \alpha_d)$$
$$(-b)^d) \ f(-a/b)$$

So, $p \mid N(a + b\alpha)$ iff $p \mid (-b)^d$ or $p \mid f(-a/b)$ Now we have already proved that $p \not| (-b)^d$.
$\Rightarrow f(-a/b) \equiv 0 \ mod\ p$. Hence, $a \equiv -br \ mod\ p$, for some root $r$ of $f(x) \ mod\ p$.
The value of $r$ taken together with $p$ determines a first degree prime ideal for which $l_{\mathcal{P}}(a + b\alpha) > 0$, and vice versa.
Suppose $l_{\mathcal{P}}(a + b\alpha) > 0$ for some first degree prime ideal $\mathcal{P}$ of $\mathbb{Z}[\alpha]$ with pair $(r, p)$.
Suppose another first degree prime ideal $\mathcal{P}_2$ exists with pair $(r_2, p)$ such that $l_{\mathcal{P}_2}(a + b\alpha) > 0$.
$\Rightarrow a \equiv -br \ mod\ p$ and $a \equiv -br_2 \ mod\ p$ but the later implies that

$$r \equiv r_2 \ mod\ p$$

$\Rightarrow \mathcal{P}$ and $\mathcal{P}_2$ corresponds to the same pair and hence represent the same ideal. This

means that there can be only one first degree prime ideal $\mathcal{P}$ with pair $(r, p)$ and has $l_{\mathcal{P}}(a + b\alpha) > 0$.
Hence, $l_{\mathcal{P}}(a + b\alpha) = Ord_p N(a + b\alpha)$. □

This theorem above shows that we only need to consider the elements of the form $a + b\alpha$. We can easily follow the algorithm and get set U of pair of integer with the desired property(3.1.5 and 3.1.4). But still we have some problem due to the following theorem.

**Theorem 3.3.2.** *If $U$ is a set of pairs of integers $(a, b)$ such that the product of all elements $a + b\alpha \in \mathbb{Z}[\alpha]$ is a perfect square $\alpha^2 \in \mathbb{Q}(\theta)$, then*

$$\sum_{(a,b)\in U} l_{\mathcal{P}_i}(a + b\alpha) \equiv 0 \ (mod \ 2)$$

*for all prime ideals $\mathcal{P}_i$ of $\mathbb{Z}[\alpha]$.*

*Proof.* Let $\mathcal{P}_i$ prime ideal of $\mathbb{Z}[\alpha]$ as $l_{\mathcal{P}_i}$ is homomorphism. We get

$$\sum_{(a,b)\in U} l_{\mathcal{P}_i}(a + b\alpha) = l_{\mathcal{P}_i} \sum_{(a,b)\in U} (a + b\alpha)$$
$$= l_{\mathcal{P}_i}(\alpha^2) = 2l_{\mathcal{P}_i}(\alpha) \equiv \ 0 \ mod \ 2$$

. □

NOTE: This is not sufficiency condition.
**Example:** Consider the number field $Q(\sqrt{-5})$. Take $\mathbb{Z}[\sqrt{-5}$, we have $\langle 2, 1 + \sqrt{-5}\rangle$ prime ideal as its index in $\mathbb{Z}[\sqrt{-5}$ is two. Now consider the element $2 \in Q(\sqrt{-5})$ the ideal generated by 2 is square of the ideal generated by $\langle 2, 1 + \sqrt{-5}\rangle$.
Now we need to show that this is not a square of an element in $\mathbb{Z}[\sqrt{-5}$. We proceed by contradiction. Assume that $2 = (a + b\sqrt{-5})^2$ this means that

$$a^2 - 5b^2 + 2ab\sqrt{-5} = 2$$

Comparing the integer part we get, $a^2 - 5b^2 = 2$ and $2ab\sqrt{-5} = 0$. These two conditions are not possible simultaneously. So, 2 is not a square.
Here we talk about necessary conditions for set $U$ to give a square root in $\mathbb{Q}(\theta)$ we didn't talk about $\mathbb{Z}[\alpha]$ for which we need Quadratic Character.

When a set $U$ of pairs of integer (a,b) has been found such that

$$\sum_{(a,b)\in U} l_{\mathcal{P}_i}(a+b\alpha) \equiv 0 \ (mod \ 2)$$

a further test is needed to determine whether or not the product of corresponding elements $a+b\alpha \in \mathbb{Z}[\alpha]$ is a perfect square in $\mathbb{Z}[\alpha]$.

## 3.4 Quadratic Character

To tackle the above problem we use concept of Quadratic character.

**Theorem 3.4.1.** *Let $U$ be the set of (a,b) pairs such that*

$$\prod_{(a,b)\in U} (a+b\alpha) = \theta^2$$

*for some $\theta \in \mathbb{Q}(\alpha)$. Give a first degree prime ideal $\mathcal{V}$ corresponding to pair $(s,v)$ that doesn't divide $\langle a+b\alpha\rangle$ for any pair $(a,b)$ and for which $f'(s) \neq 0 \ (mod\ v)$,it follows that*

$$\prod_{(a,b)\in U} \left(\frac{a+bs}{v}\right) = 1$$

*Proof.* Let $\phi : \mathbb{Z}[\alpha] \longrightarrow \mathbb{Z}/p\mathbb{Z}$ such that $\phi(\alpha) = s \ mod \ v$. Then $\mathcal{V} = ker\phi$ is the first degree prime ideal corresponding to $(s,v)$. Now if we restrict the map $\phi$ to $\mathbb{Z}[\alpha] \setminus \mathcal{V}$ we get a onto map from $\mathbb{Z}[\alpha] \setminus \mathcal{V}$ to non zero elements of $\mathbb{Z}/p\mathbb{Z}$ and using this map we can define map $\chi : \mathbb{Z}[\alpha] - \mathcal{V} \longrightarrow \{1,-1\}$ given by

$$\chi_q(\gamma) = \left(\frac{\phi(\gamma)}{v}\right) \tag{3.4.1}$$

We know from algebraic number theory that there exists $\beta = f^{'}(\alpha).\theta \in \mathbb{Z}[\alpha]$ satisfies

$$f^{'}(\alpha)^2 . \prod_{(a,b)\in U} (a+b\alpha) = \beta^2 \tag{3.4.2}$$

As we have assumed that $\langle a+b\alpha\rangle$ is not divisible by ideal $V$ we get $a+b\alpha \notin V$. We have also assumed that $f^{'}(s)$ is not divisible by $V$, we get $f^{'}(\alpha)^2 \notin V$. Thus we have, $\langle\beta\rangle$ and $\langle\beta^2\rangle$ hence $\chi_V$ is defined at both $\beta$ and $\beta^2$.

$$\chi_V(\beta^2) = \left(\frac{\phi(\beta)^2}{v}\right) = \left(\frac{\phi(\beta)\phi(\beta)}{v}\right) = \left(\frac{\phi(\beta)}{v}\right)^2 = 1 \tag{3.4.3}$$

and we also have, $\chi_V(f^{'}(\alpha)^2) = 1$.
Now from 3.4.3 we have,

$$1 = \chi_V(\beta^2) = \chi_V\left(f^{'}(\alpha)^2.\prod_{(a,b)\in U}(a+b\alpha)\right) = \left(\frac{f^{'}(\alpha)^2.\prod_{(a,b)\in U}(a+b\alpha)}{v}\right)$$

$$= \left(\frac{\phi(f^{'}(\alpha)^2).\phi(\prod_{(a,b)\in U}(a+b\alpha))}{v}\right) = \left(\frac{\phi(f^{'}(\alpha)^2)}{v}\right).\left(\frac{\prod_{(a,b)\in U}\phi(a+b\alpha)}{v}\right)$$

$$= \chi_V(f^{'}(\alpha)^2).\left(\frac{\prod_{(a,b)\in U}\phi(a+bs)}{v}\right) = 1.\prod_{(a,b)\in U}\left(\frac{a+bs}{v}\right)$$

Hence Proved. □

## 3.5 Square root extraction

Now the final step of the algorithm is to find the square root of the polynomial constructed in $\mathbb{Z}[\alpha]$. For general number field sieve, square root extraction can be done by computing the root of polynomial $x^2 - \gamma^2$ in $\mathbb{Q}_\alpha$. Where $\gamma$ is

$$f^{'}(\alpha)^2.\prod_{(a,b)\in U}(a+b\alpha) \tag{3.5.1}$$

There exists a lot of standard methods to do this but the most widely used method is successive approximation using Hensel's lemma ([17]). Here in my implementation I have used SquareRoot() command which directly computes the square root.

# Chapter 4

# Implementation of algorithm

We chose a number randomly and tried to factor it using our algorithm [1, Chapter 2] the code can be found in Appendix. We took $n = 56442324723497$.

- Using 3.1.1 calculate the degree of the polynomial. And using this degree find the value of $m$. The value of $m$ turns out to be 38359.
- Using this $m$ find the polynomial which turns out to be

$$X^3 + 10376X + 8234 \tag{4.0.1}$$

- Construct the factor bases, which turns out to be

## Rational factor base:

{ 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89,97, 101, 103, 107, 109, 113, 127, 131, 137, 139, 149, 151, 157, 163, 167, 173,179, 181, 191, 193, 197, 199, 211, 223, 227, 229, 233, 239, 241, 251, 257, 263, 269, 271, 277,281, 283, 293, 307, 311, 313, 317, 331, 337, 347, 349, 353, 359, 367, 373, 379, 383, 389, 397, 401, 409, 419, 421, 431, 433, 439, 443, 449, 457, 461, 463, 467, 479, 487, 491, 499, 503, 509, 521, 523, 541, 547, 557, 563, 569, 571, 577, 587, 593, 599, 601, 607, 613, 617, 619, 631, 641, 643, 647,653,659,661, 673, 677, 683, 691, 701, 709, 719, 727, 733, 739, 743, 751, 757, 761, 769, 773, 787, 797, 809, 811, 821, 823, 827, 829, 839, 853, 857, 859, 863, 877, 881, 883, 887, 907, 911, 919,929,937,941,947, 953, 967, 971, 977, 983, 991, 997, 1009, 1013, 1019, 1021, 1031, 1033, 1039, 1049, 1051, 1061, 1063, 1069, 1087, 1091, 1093, 1097, 1103, 1109, 1117, 1123, 1129, 1151, 1153, 1163, 1171, 1181, 1187, 1193, 1201, 1213, 1217, 1223, 1229, 1231, 1237, 1249, 1259, 1277, 1279, 1283, 1289, 1291, 1297, 1301, 1303, 1307, 1319, 1321, 1327, 1361, 1367, 1373, 1381, 1399, 1409, 1423, 1427, 1429, 1433, 1439, 1447, 1451, 1453, 1459, 1471, 1481, 1483, 1487, 1489, 1493, 1499, 1511, 1523, 1531, 1543, 1549, 1553, 1559, 1567, 1571, 1579, 1583, 1597, 1601, 1607, 1609, 1613, 1619, 1621, 1627, 1637, 1657, 1663, 1667, 1669, 1693, 1697, 1699, 1709, 1721, 1723, 1733, 1741, 1747, 1753, 1759, 1777, 1783, 1787, 1789, 1801, 1811, 1823, 1831, 1847, 1861, 1867, 1871, 1873, 1877, 1879, 1889, 1901, 1907, 1913, 1931, 1933, 1949, 1951, 1973, 1979, -1 }

## Algebraic factor base

{

(0,2),(2,7),(3,7),(8,13), (3,17), (7,17), (2,19), (0,23), (13,29), (1,37), (4,37), (32,37), (41,47), (30,53), (5,59), (25,59), (29,59), (37,61), (24,67), (12,71), (27,71), (32,71), (34,89), (50,97), (94,101), (2,109), (75,113), (67,127), (54,131), (92,131), (116,131), (81,139), (87,139), (110,139), (82,149), (77,151), (148,157), (44,163), (54,163), (65,163), (157,167), (34,173), (0,179), (56,181), (22,191), (33,193), (58,199), (164,199), (176,199), (98,227), (167,229), (121,233), (137,239), (44,241), (76,269), (206,269), (256,269), (53,271), (99,277), (7,281), (97,281), (177,281), (257,293), (197,313), (132,317), (3,331), (257,337), (101,347), (114,349), (310,359), (19,373), (163,379), (83,383), (134,383), (166,383), (360,421), (376,431), (127,433), (349,443), (115,457), (30,467), (407,467), (466,479), (146,487), (1,503), (142,503), (360,503), (144,509), (200,521), (187,541), (114,547), (442,547), (538,547), (99,557), (116,557), (342,557), (19,569), (39,571), (113,577), (470,577), (571,577), (90,593), (162,593), (341,593), (123,599), (480,601), (312,607), (142,617), (135,619), (529,631), (306,641), (342,641), (634,641), (22,647), (336,653), (482,659), (4,673), (157,673), (512,673), (326,691), (412,701), (294,719), (546,733), (341,751), (355,757), (319,761), (270,773), (55,787), (354,787), (378,787), (503,797), (168,809), (172,809), (469,809), (147,823), (181,823), (495,823), (39,827), (227,829), (65,839), (165,853), (199,853), (489,853), (331,857), (101,859), (805,859), (812,859), (632,863), (533,881), (38,883), (59,887), (372,907), (630,907), (812,907), (78,919), (518,937), (12,947), (55,947), (880,947), (21,953), (428,953), (504,953), (478,983), (719,991), (615,997), (982,1009), (995,1013), (5,1021), (345,1031), (449,1039), (233,1049), (687,1051), (815,1063), (596,1091), (687,1091), (899,1091), (533,1093), (725,1097), (583,1103), (590,1103), (1033,1103), (384,1109), (309,1117), (1136,1151), (10,1153), (548,1163), (760,1171), (520,1181), (339,1187), (1015,1201), (732,1213), (768,1213), (926,1213), (707,1229), (69,1237), (410,1237), (758,1237), (325,1249), (836,1277), (648,1279), (926,1279), (984,1279), (94,1283), (204,1291), (140,1297), (517,1297), (640,1297), (549,1303), (169,1307), (306,1307), (832,1307), (669,1319), (474,1327), (1036,1327), (1144,1327), (455,1367), (527,1381), (90,1409), (442,1409),

(877,1409), (718,1423), (182,1429), (1280,1429), (1396,1429), (778,1433), (158,1439), (870,1451), (918,1451), (1114,1451), (1000,1453), (837,1459), (832,1481), (970,1487), (869,1489), (967,1493), (357,1511), (294,1523), (1522,1553), (976,1559), (406,1571), (1258,1571), (1478,1571), (276,1579), (431,1583), (765,1601), (649,1607), (23,1609), (273,1609), (1313,1609), (87,1613), (326,1619), (140,1621), (581,1627), (781,1637), (1542,1657), (334,1663), (199,1667), (1356,1693), (1067,1699), (1455,1709), (1672,1721), (309,1723), (627,1733), (1067,1747), (1713,1759), (1393,1777), (845,1783), (1235,1783), (1486,1783), (566,1787), (811,1811), (1169,1811), (1642,1811), (425,1823), (1245,1831), (121,1861), (316,1873), (1006,1877), (727,1879), (1389,1879), (1642,1879), (949,1889), (790,1901), (1291,1901), (1721,1901), (644,1907), (1325,1907), (1845,1907), (679,1913), (1230,1931), (870,1933), (1455,1933), (1541,1933), (154,1949), (350,1949), (1445,1949), (836,1951), (1153,1951), (1913,1951), (1899,1979), (815,1997), (1964,1999), (33,2003), (443,2003), (1527,2003), (1679,2011), (1935,2029), (1337,2039), (233,2053), (534,2053), (1286,2053), (633,2063), (217,2069), (846,2081), (1566,2081), (1750,2081), (1394,2083), (661,2087), (1450,2087), (2063,2087), (2046,2099), (211,2111), (688,2111), (1212,2111), (56,2113), (934,2113), (1123,2113), (1972,2129), (1025,2137), (2142,2143), (1626,2153), (923,2161), (627,2179), (660,2179), (892,2179), (733,2207), (1552,2207), (2129,2207), (386,2213), (825,2221), (1585,2221), (2032,2221), (71,2237), (880,2237), (1286,2237), (1683,2243), (163,2273), (2252,2281), (1731,2287), (896,2293), (76,2297), (774,2309), (1864,2311), (1147,2333), (1639,2341), (372,2347), (241,2351), (555,2357), (1738,2371), (544,2377), (570,2377), (1263,2377), (656,2381), (1729,2381), (2377,2381), (1849,2389), (327,2393), (482,2399), (1545,2411), (883,2417), (1586,2417), (2365,2417), (227,2423), (1669,2459), (1009,2467), (1174,2473), (1526,2477), (1737,2521), (905,2531), (1831,2531), (2326,2531), (721,2539), (813,2543), (1220,2549), (799,2551), (428,2557), (755,2557), (1374,2557), (1243,2579), (1493,2579), (2422,2579), (1263,2593), (53,2609), (464,2609), (2092,2609), (452,2617), (960,2617), (1205,2617), (1561,2621), (1343,2633), (749,2647), (1360,2657), (112,2677), (667,2683), (2224,2687), (1156,2689), (2002,2689), (2220,2689), (1572,2693), (596,2699), (2250,2699), (2552,2699), (1615,2711), (1335,2719), (1686,2729), (1839,2731), (869,2741), (601,2749), (1890,2753), (223,2767), (1159,2777), (1997,2777), (2398,2777), (2775,2791), (2767,2797), (1144,2801), (2407,2833), (1345,2837), (1550,2837), (2779,2837), (635,2843), (2518,2851), (1368,2861), (1659,2861), (2695,2861), (225,2887), (1141,2887), (1521,2887), (1752,2897), (1781,2903), (2478,2909), (395,2969), (2755,2969), (2788,2969)

}

## Quadratic character base

{

(1443,2999), (740,3011), (744,3011), (1527,3011), (735,3019), (301,3023), (813,3023), (1909,3023), (2534,3079), (733,3083), (2663,3083), (2770,3083), (684,3089), (1310,3109), (1947,3109), (2961,3109), (2904,3119), (644,3121), (395,3163), (1071,3163), (1697,3163), (1279,3167), (476,3169), (1112,3169), (1581,3169), (895,3187), (1130,3191), (1706,3203), (2685,3251), (1327,3253), (1960,3253), (3219,3253), (1176,3257), (905,3259), (959,3259), (1395,3259), (2578,3299), (782,3307), (889,3313), (2438,3319), (1190,3323), (2456,3323), (3000,3323), (3266,3329), (1371,3331), (1244,3343), (579,3347), (50,3361), (1724,3371), (858,3391), (2682,3391), (3242,3391), (2481,3407), (2609,3413), (235,3433), (2197,3449), (2160,3457), (1094,3463), (1830,3467), (477,3491), (738,3491), (2276,3491), (485,3511), (931,3517), (8,3529), (1351,3533), (1512,3539), (1848,3541), (370,3557), (3429,3559), (1710,3581), (2430,3583), (1915,3593), (2263,3593), (3008,3593), (1451,3607), (2846,3607), (2917,3607), (3376,3617), (418,3623),

(735,3623), (2470,3623), (1053,3631), (3286,3637), (1560,3643), (1300,3659), (2702,3659), (3316,3659), (2056,3671), (1467,3673), (568,3677), (3356,3677), (3430,3677), (2665,3691), (1524,3697), (3361,3709), (1939,3719), (1445,3767), (2795,3767), (3294,3767), (2235,3769), (1288,3793), (923,3797), (2098,3803), (1543,3821), (630,3823), (1857,3833), (1653,3847), (3358,3851), (2066,3853), (1331,3877), (2420,3881), (929,3889), (3803,3907), (3241,3917), (3730,3919), (3834,3923), (1364,3929), (2455,3943)
}

Using these factor bases we collected smooth relations. Using those relations and factor bases we constructed the matrix and after calculating the null space of it.
the value of $f^{'}(m) = 4414249019$
and the value of $\sqrt{\prod_{(a,b)\in S}(a+bm)}$ is 47460456068820.

$$u = f^{'}(m).\sqrt{\prod_{(a,b)\in S}(a+bm)} = 209502271643081281487580. \tag{4.0.2}$$

Then we calculate the value of $f^{'}(\alpha)$ and $f^{'}(\alpha)$ which turns out to be

$$3\alpha^2 + 10376 \quad and \quad -31128\alpha^2 - 74106\alpha + 107661376$$

respectively and the value of

$$\nu = \sqrt{f'(\alpha).\prod_{(a,b)\in S}(a+b\alpha)} = \tag{4.0.3}$$

-71698655736614058051872222240663869012512861398405905499722665210556609760710820489997792449780478415285821524191889745206539317890686560
254707674042263748923681640659211523282732296214466855932462967210241256969878557020213291352645851891276162254607766967275494505033753548
804924551716499675604204203691872455128368783773008461235535667 2$\alpha^2$+
111536589308245578848856438895889427673374088636175303801078944190378977711324243538165437633584015906718371492574708297142104219629068763
605462441661642395818920406456563262015595991528832670096901465474177885056559278675633853379201858764579676098956062156647822050168853396
934375874519171554287549908292043741093496327132394135747857758208$\alpha$+
889572961906148392580677101616124712791023699035365651140603288640341891473273526923273544327828844879163393574020521419472403670593558196
724418684214077571578516594824967376297808013785419469716456171077366925060151850135683105042392239244388911632181097682023880063442894661
030482420943984364540782331891654118682292957453371959759917568 00

calculating the value of $u$ which is the image of $\nu$ under $\phi$. we get, $u$ = 96687834939610.

So find the factor of n using

$$GCD(209502271643081281487580 - 96687834939610, 56442324723497) \quad (4.0.4)$$

So we get 47107 as a factor of $n$ and the other factor is 1198172771.

# Chapter 5

# Appendix

## 5.1 Code for the General Number Field Sieve written in MAGMA

The code goes like this:

```
h1:=n;

/* Polynomial construction started*/

d:= Round(Root(3*Log(h1)/Log(Log(h1)),3));
m := Iroot( h1, d );
print "m := ", m, ";";
coeffs := [];
for i := d to 0 by -1 do
        temp2 := m^i;
        coeff := h1 div temp2;
        coeffs[i+1] := coeff;
        h1 -:= coeff*temp2;
end for;
P<X> := PolynomialRing( IntegerRing() );
f := P!coeffs;
print f;
```

```
/*Polynomial constructed*/

 J<alpha> := NumberField(f);
derivativef := Derivative(f);

/*Finding Smoothness Bound*/

dlogd:=d*Log(d);
tempo:=1.0/d * Log(n);
e:=dlogd + Sqrt(dlogd^2 + 4*tempo*Log(tempo));
k1:=Round(Exp(0.5*e));
k:=2*Floor(k1);

/* k is the smoothness bound */

/* constructing the Factor Bases*/

RFB:=PrimesUpTo (k);
RFB:=Append(RFB,-1);
/*AFB*/

localset:=PrimesUpTo(Floor(1.5*k));
AFB:= AssociativeArray();
cg:=0;
just:=1;
for ko in [1 .. #localset] do
         x:=localset[ko];
         S:=GF(x);
         T:=PolynomialRing(S);
         v1:=[];
        set1:=[];
         set1:=Roots( f , S);
        for j in [1.. #set1] do
```

```
                tt:=IntegerRing()!set1[j][1];
  cg:=cg+1;
                vv:=IntegerRing()! x;
               AFB[just] := [*tt,vv*];

               just:=just+1;
        end for;
end for;

/*AFB done */

/*QCB*/

 tempset:=PrimesInInterval(Floor(1.51*k) , Floor(2*k));
QCB:= AssociativeArray();
cg1:=0;
just1:=1;
for kp in [1 .. #tempset] do
         x:=tempset[kp];
         S:=GF(x);
         T:=PolynomialRing(S);
        v2:=[];
        set2:=[];

        set2:=Roots(f , S);

        for jq in [1.. #set2] do
                 tj:=IntegerRing()!set2[jq][1];
                 cg1:=cg1+1;
                vv:=IntegerRing()! x;
                QCB[just1] := [*tj,vv*];

                just1:=just1+1;
        end for;
```

```
end for;
print cg1;

/*QCB done*/

/* main body of the program */
a:=AssociativeArray();
e:=AssociativeArray();
b:=0;
rels:=[];
req:= #RFB + cg1+ cg + 20;
got:= 0;
while (got lt req) do
         b:=b+1;
         c:=Floor(k1);
         for i in [-c .. c] do
                 a[i]:= i+b*m;
         end for;
        for y1 in [-c .. c] do
                 for j3 in [1 .. #RFB-1] do
                         if (y1 mod RFB[j3]) eq (-b*m mod RFB[j3]) then
                                 while ((a[y1] ne 1) and ((a[y1] mod RFB[j3]) eq 0)) do
                                         a[y1]:=a[y1] div RFB[j3];
                                 end while;
                        end if;
                end for;
         end for;
          for i3 in [-c .. c] do
          e[i3]:= Numerator( Norm(i3+(b*alpha)));

          end for;
         for y2 in [-c .. c] do
         /* print "y2 loop is running";*/
                 for j9 in [1 .. cg] do
```

```
                        /*  print "j9 loop is running";*/
                       if (y2 mod AFB[j9][2]) eq (-b*(AFB[j9][1]) mod AFB[j9][2])
                                       while ((e[y2] ne 1) and ((e[y2] mod AFB[j9

                                               e[y2]:= e[y2] div AFB[j9][2];
                                        end while;
                                  end if;
                 end for;
        end for;
         for i5 in [-c .. c] do
                  if (a[i5] eq e[i5]) and (a[i5] eq 1) and (Gcd(i5,b) eq 1) then
                        rels := Append(rels, [i5,b]);

                        got:=got+1;
                  end if;
         end for;
end while;

/*relation collected*/

/*constructing the matrix from relations*/

 M := RMatrixSpace (IntegerRing(),#rels, req-20)!0;
gf2:= GF(2);
Mprime := RMatrixSpace (gf2, #rels, req-20)!M;


for i21 in [1 .. #rels] do
        if (rels[i21][1] + (rels[i21][2]*m) lt 0) then
        Mprime[i21][#rels]:= 1;
        end if;
end for;

for i21 in [1 .. #rels] do
```

```
          tte:= rels[i21][1]+(rels[i21][2]*m);
        for i22 in [2 .. #RFB-1 ] do
                dummy:=0;
                while ((tte mod RFB[i22]) eq 0) do
                        tte:= tte div RFB[i22];
                        dummy := dummy+1;
                end while;
                M[i21][i22]:= dummy;
                dummy:=dummy mod 2;
                Mprime[i21][i22]:= dummy;
        end for;
end for;



for j21 in [1 .. #rels] do
     tte1:= Abs(Numerator((-rels[j21][2])^d * Evaluate(f,-rels[j21][1]/rels[j21][2])));
        for j22 in [1 .. cg] do
                dummy1:=0;
                while ((tte1 mod AFB[j22][2]) eq 0) do
                        tte1:=tte1 div AFB[j22][2];
                        dummy:=dummy+1;
                end while;
                M[j21][#RFB+j22]:= dummy;
                dummy:=dummy mod 2;
                Mprime[j21][#RFB+j22]:= dummy;
        end for;
end for;


/* inserting the QCB part */

for k21 in [1 .. #rels] do
        for k22 in [1 .. cg1] do
```

```
                tte3:= rels[k21][1]+(rels[k21][2]*QCB[k22][1]);
                lsymbol:= LegendreSymbol(tte3,QCB[k22][2]);
                        if (lsymbol ne 1) then
                                M[k21][#RFB+cg+k22]:=1;
                                Mprime[k21][#RFB+cg+k22]:=1;
                        end if;
        end for;
end for;

/* matrix construction done */

/* finding the null space */

 nullmat := NullSpace(Mprime);

for j41 in [1 .. Dimension(nullmat)] do
        sol1:= ChangeUniverse (Eltseq (nullmat.j41), IntegerRing());
        exponent := RSpace(IntegerRing(),req-20 )!0;
        for i41 in [1 .. #rels] do
                if sol1[i41] eq 1 then
                        exponent := exponent + M[i41];
                end if;
        end for;

/* calculating the integer square root */

        num := 1;
        for i44 in [1 .. #RFB] do
                e := exponent[i44];
                if e gt 0 then
                        num:=(num * (RFB[i44]^(e div 2) ) ) mod n;
                end if;
        end for;
```

```
num := (num * Evaluate(derivativef , m));
list := [];
for i51 in [1 .. #rels] do
        if sol1[i51] eq 1 then
                Append (~list, rels[i51][1]+(alpha* rels[i51][2]));

        end if;
end for;
Append ( ~list, Evaluate(derivativef , alpha)^2 );
prod_alg := 1;

for i61 in [1 .. #list] do
        prod_alg:=(J!( prod_alg*list[i61]));
end for;
if IsSquare(prod_alg) eq false then
print "not square";
continue;
else;
prod_alg:= Sqrt(prod_alg);
prod_alg:=(J! prod_alg);
prod_alg;
list:= Eltseq (prod_alg);
yu:=0;
elt := 0;
        for j in [1 .. #list] do
              num_coeff:= Numerator(list[yu+1]) mod n;
              den_coeff:= Denominator(list[yu+1]) mod n;
 elt := elt + ((num_coeff * Modinv(den_coeff, n) mod n) * Modexp(m, yu,n)mod n);
                yu:=yu+1;
        end for;

       v:= elt;
        v;
        factor:= GCD(v-num,n);
```

```
                print factor;
        end if;
end for;
```

Note: This code is not at all optimised. Magma is a software package designed for the computation in Algebra, number theory, algebraic number theory, algebraic geometry and algebraic combinatorics. [18]

## 5.2 L-Notation

$L_n[\alpha, c] = e^{(c+o(1))(ln\ (n))^{\alpha}(ln\ (ln\ (n)))}$
where $c$ is positive constant and $\alpha$ is a constant $0 \leq \alpha \geq 1$

# Bibliography


[1] David S. Dummit, Richard M. Foote, Abstract Algebra, Wiley- India, 1999.

[2] Jody Esmonde, M. Ram Murty, Problems in Algebraic Number Theory, Springer, 1999.

[3] Peter Stevenhagen, The Number Field Sieve, Algorithmic Number Theory, 2008

[4] A.K. Lenstra, H.W. Lenstra jr., M.S. Manasse, J. M. Pollard, The Factorization of Ninth Fermat's Number, Mathematical Subjects Classification, 1990.

[5] Johannes A. Buchmann, Introduction to Cryptography, Springer, 2001.

[6] Joshua Baron, A Description of the Number Field Sieve, 2003.

[7] Carl Pomerance, The Number Field Sieve, Proceeding and Symposia in Applied Mathematics, 1994.

[8] Carl Pomerance,A tale of two sieve, Notices of the AMS, 1996.

[9] Steven Bryens, The Number Field Sieve, 2005.

[10] A.K. Lenstra, H.W. Lenstra jr., M.S. Manasse, J. M. Pollard, The Number Field Sieve,1995.

[11] Arjen K. Lenstra, Hendrik W.Jr. Lenstra, The Development of the Number Field Sieve, springer, 1993.

[12] Ron L. Rivest, Adi Shamir, and Len Adleman, A method for obtaining digital signatures and public-key cryptosystems, Communications of the ACM 21 (1978), 120126.

[13] Martin O. Rabin, Digitalized signatures and public-key functions as in- tractable as factorization, MIT Technical Report TR-212 (1979).

[14] John M. Pollard, Theorems on factorization and primality testing., Proceed- ings of the Cambridge Philosophical Society 76 (1974), 521528.

[15] M. Agrawal, N. Kayal, and N. Saxena, Primes is in p, 2002.

[16] Serge Lang, Algebra, Springer, 2002.

[17] J.P. Bulher, S.Wagon, Basic algorithms in number theory, surveys in algorithm number theory, new york press, 2008.

[18] http://magma.maths.usyd.edu.au/magma/