The Number Field Sieve

Steven Byrnes

Math 129

May 18, 2005

1 Introduction

1.1 Prime factorization and the Number Field Sieve

One of the most important and widely-studied questions in computational number theory is how to efficiently compute the prime factorizations of large integers. Among other applications, fast prime-factorization algorithms would break the widely-used RSA cryptosystem, and be of great interest in complexity theory. In particular, there is no algorithm which can factor an integer n in polynomial time with respect to $\log n$. Indeed, a number can be proven prime or composite in polynomial time [1]; unfortunately, the proof of compositeness does not generate a factor.

Of the many sophisticated prime-factoring algorithms, the one which is (conjecturally) asymptotically the fastest is called the "Number Field Sieve." Its conjectured run-time for factoring a general integer n is

$$\exp\left[(\log n)^{1/3}\left(\left(\frac{64}{9}+o(1)\right)\log\log n\right)^{2/3}\right]$$

([3], Conjecture 11.2.) Indeed, as recently as May 9, 2005, this algorithm was used to

factor "RSA-200," a 200-decimal-digit number, into its two 100-digit prime factors (the number was posed as a challenge by the cryptography company RSA Laboratories).

In this paper, we will explain how and why the Number Field Sieve works.

1.2 Square Roots of One

When using the Number Field Sieve to factor a number n, the end goal (as with many other factoring algorithms) is to "randomly" generate square roots of unity (mod n). This pursuit is justified by the following theorem:

Theorem 1. *Let $n \in \mathbb{Z}$ be odd. Then the set of solutions to $x^2 \equiv 1$ (mod n) forms a vector space V of dimension m over F_2, where m is the number of distinct prime factors of n. Moreover, given a nontrivial solution x, one can immediately recover a nontrivial factorization of n.*

Proof. First we define the vector-space operations. Vector-space addition corresponds to mod n multiplication. Scalar multiplication in the vector space is mod n exponentiation. Then it is easy to verify that the set of solutions is indeed a vector space over F_2. Now write $n = p_1^{a_1} \cdots p_m^{a_m}$. Let S be the set of m-tuples with entries in ± 1. By the Chinese Remainder Theorem, for each $s = (s_1, \ldots, s_m) \in S$ there is a unique value (x_s mod n) satisfying $x_s \cong s_i \pmod{p_i^{a_i}}$ for each $1 \leq i \leq m$. Each x_s is a square-root of 1, again by CRT. And we will now show that every square root of 1 is of this form. By CRT, it suffices to show that ± 1 are the only square-roots of 1 (mod p^a) for p odd. If $x^2 \equiv 1 \pmod{p^a}$, then $p^a | (x+1)(x-1)$. Since p is odd, it cannot divide both $x+1$ and $x-1$, so all the powers of p are in one of the two factors. But then either $(x+1)$ or $(x-1) \equiv 0 \pmod{p^a}$, as desired. So the square-roots of unity (mod n) correspond precisely to the elements of S; in particular, there are 2^m of them, so the dimension of the vector space is m.

Now given a nontrivial solution to $x^2 \equiv 1 \pmod{n}$, let $s \in S$ be the corresponding m-tuple. Then by definition of x_s, $\gcd(x-1, n)$ is the product of $p_i^{a_i}$ over all i with $s_i = 1$, and $\gcd(x+1, n)$ is the product of $p_i^{a_i}$ over all i with $s_i = -1$. Since $x \not\equiv \pm 1$, the factorization $n = \gcd(x-1, n) \cdot \gcd(x+1, n)$ is nontrivial. □

A prime power, of course, could not be factored by means of the above theorem; but this is a minor detail, since it is quite easy to identify and factor perfect powers. One could, for example, start by taking a numerical square root of n (say, by Newton's method) and rounding to the nearest odd integer, then try squaring that integer to see whether the result is n. If not, then n is not a square; in the same way, test whether n is a perfect kth power for $2 < k \leq [\log_3 n]$. This whole test takes only $O((\log n)^3)$ time.

2 Rational Sieve

We illustrate the main idea of the number field sieve by means of a simple special case thereof: the "rational sieve." The description will be qualitative (for example, terms like "large number" and "fast algorithm" will not be defined), but we will fill in the details when we describe the more general algorithm in the next section.

Let n be the integer we are trying to factor. We pick a subset of $(\mathbb{Z}/n\mathbb{Z})^*$ as a "factor base," here $P = \{(p \bmod n) : p < B\}$ for an appropriate integer B (we assume that no element of P is a divisor of n; otherwise we are already done). A number is called *B-smooth* iff all its prime factors lie below B. We search for numbers z such that both z and $z+n$ are B-smooth. Each such z gives a multiplicative relation among the elements of P. Once we have found a few more of these relations than $\#P$, we organize the relations into a matrix, with a relation $\prod_{p\in P} p^{a_p} \equiv 1$ corresponding to a row with entries $a_1, \ldots, a_r$. We reduce this matrix (mod 2), and since it has more rows than columns, we will find several linear dependencies, which correspond to equations of the form $\prod_{i\in S}\prod_{p\in P} p^{a_{ip}} \equiv 1 \pmod n$ with $\sum_{i\in W} a_{ip}$ even for each p. Each such equation gives us the element

$$\prod_{p\in P} p^{(\frac{1}{2}\sum_{i\in W} a_{ip})},$$

which is a square-root of unity (mod n). If we are lucky, at least one such dependency will be nontrivial, and by Theorem 1 this will give us a nontrivial factorization of n. If n has more than two prime factors, we can try again with more dependencies until we find other nontrivial factorizations, and taking gcd's will hopefully give the complete factorization of n into prime powers.

The main reason that the Rational Sieve described above is prohibitively slow is that we need to find numbers of order n (namely, $z + n$) which are B-smooth. These are extremely rare, unless B is chosen to be so large that the matrix-reduction step becomes unworkable. The purpose of the Number Field Sieve is to modify the algorithm so that only numbers of order $n^{o(1)}$ have to be tested for smoothness.

3 Special Number Field Sieve

3.1 Special and General Number Field Sieve

The term "Number Field Sieve" can refer to one of two algorithms. The *Special Number Field Sieve* only works for numbers of the form $r^e - s$, with $r, |s|$ small. The *General Number Field Sieve* was a later extension of this algorithm to arbitrary integers. We will describe both algorithms in some detail, starting with the Special Field Sieve in this section, followed by the General Number Field Sieve in Section 4.

We will leave out, for the purpose of clarity and brevity, some details of implementation. These details can be found, for example, in [6].

3.2 Idea of algorithm

In the Number Field Sieve, we pick a monic irreducible polynomial $f \in \mathbb{Z}[x]$, and an element $m \in \mathbb{Z}/n\mathbb{Z}$ such that $f(m) \equiv 0 \pmod{n}$. Let α be a complex root of f, and note that there is a homomorphism of rings $\varphi : \mathbb{Z}[\alpha] \to \mathbb{Z}/n\mathbb{Z}$ with $\alpha \mapsto m$. For ease of exposition, we assume that $\mathbb{Z}[\alpha]$ is a unique factorization domain; we will drop this assumption later (Section 3.6). As before, we set up a factor base in $\mathbb{Z}$ of primes below some number B_1, and we also set up an analogous factor base in $\mathbb{Z}[\alpha]$ consisting of a generating set for the group of units in $\mathbb{Z}[\alpha]$, along with a set of maximal set of non-associate primes with norm below some number B_2 (we actually only need a subset of these – see Section 3.4). Then, by multiplicativity of the norm, every element of $\mathbb{Z}[\alpha]$ whose norm is B_2-smooth will be uniquely expressible as a product of elements of the $\mathbb{Z}[\alpha]$ factor base. We look for pairs of relatively-prime integers (a, b) such that $a+bm \in \mathbb{Z}$ is B_1-smooth and $a + b\alpha \in \mathbb{Z}[\alpha]$ has a norm which is B_2-smooth. Then, applying φ to the factorization of $a+b\alpha$, and setting that equal to the factorization of $a+bm$, we get a multiplicative relation among the elements in a fixed factor base in $\mathbb{Z}/n\mathbb{Z}$. With enough such pairs (a, b), we can proceed as in Section 2 to find square roots of 1 in $\mathbb{Z}/n\mathbb{Z}$, which give rise to factorizations of n.

The advantage of this method is that m can be chosen to be much smaller than n, as can the degree and coefficients of f. Then the numbers we are checking for smoothness will be small enough that the smooth ones will not be too hard to come by, so B_1 and B_2 can be chosen far lower than they could for the rational sieve, and the algorithm will have a better chance of working in reasonable time.

3.3 Picking the Number Field

The first step of the algorithm is choosing a number field to work in. Recall that we want to factor the number $n = r^e - s$. We first pick the degree d of the extension. Optimizing the conjectured run-time for the algorithm gives

$$d \approx \left(\frac{(3 + o(1)) \log n}{\log \log n} \right)^{1/3}. \tag{1}$$

This follows primarily from computing the density of B-smooth numbers – for details, see ([6], §6.3, and [3], Conjecture 11.2). For example, $d = 3$ was used for factoring the seventh Fermat number $2^{2^7} + 1$, and $d = 5$ for the ninth. Having chosen d, we then find the smallest integer k such that $kd \geq e$, we let $t = s \cdot r^{kd-e}$, and we define f and m by

$$f = X^d - t, \quad m = r^k.$$

Note that $f(m) = r^{dk} - sr^{dk-e} = nr^{dk-e} \equiv 0 \pmod{n}$. There are simple criteria to check whether polynomials of this form are irreducible (see [6], p15); if f is not irreducible, we can factor it (e.g. through trial division) and plugging in m, we will either get a factorization of n, or we can replace f by an irreducible factor, such that $f(m) \equiv 0 \pmod{n}$ still holds. We now let α be a complex root of f, so that we can define the ring $\mathbb{Z}[\alpha] \cong \mathbb{Z}[X]/f$. Since $f(m) \equiv 0 \pmod{n}$, there is a well-defined ring homomorphism $\varphi : \mathbb{Z}[\alpha] \to \mathbb{Z}/n\mathbb{Z}$ with $\alpha \mapsto m$.

We assume for now that $\mathbb{Z}[\alpha]$ is a principal ideal ring, a unique factorization ring, and equal to the ring of integers of $\mathbb{Q}(\alpha)$. We will drop these assumptions in Section 3.6.

3.4 Factoring in $\mathbb{Z}[\alpha]$

We next explain how factorizations are performed in $\mathbb{Z}[\alpha]$. We want all elements $a + b\alpha$ whose norm is B_2-smooth for some B_2 to be factorizable in $\mathbb{Z}[\alpha]$ as a product of elements in a factor base. By our assumptions on $\mathbb{Z}[\alpha]$, it would clearly suffice to take a generating set U for the group of units (finite by Dirichlet's Unit Theorem), along with a set G consisting of one generator for each prime ideal of norm $\leq B_2$. In fact, it turns out that we need only take generators for *first-order prime ideals*, that is, ideals with prime integer norms:

Theorem 2. *Suppose $\mathbb{Z}[\alpha]$ is the ring of integers of $\mathbb{Q}(\alpha)$. If a prime ideal $\mathfrak{p}$ contains an element of the form $a + b\alpha$ with $(a, b) = 1$, then $\mathfrak{p}$ is a first-order prime ideal.*

Proof. Suppose that $\mathfrak{p}$ contains $a + b\alpha$. Let $x \mapsto \overline{x}$ be the reduction map from $\mathbb{Z}[\alpha] \to \mathbb{Z}[\alpha]/\mathfrak{p}$. Note that $\mathbb{Z}[\alpha]/\mathfrak{p}$ is a finite field, of cardinality equal to $\mathrm{Norm}(\mathfrak{p})$. Let p be its characteristic, and define the subgroup F_p of $\mathbb{Z}[\alpha]/\mathfrak{p}$ to be, as usual, the roots of $x^p - x$ in the field. Since $a + b\alpha \in \mathfrak{p}$, we get $\overline{a} + \overline{b}\overline{\alpha} = 0$. Since $a, b \in \mathbb{Z}$ and $p \in \mathfrak{p}$, we get by Fermat's Little Theorem $a, b \in F_p$. Moreover, $\overline{b} \neq 0$, because if $\overline{b} = 0$ then $\overline{a} = -\overline{b}\overline{\alpha} = 0$, and then a and b would have the common factor p, a contradiction. Hence $\overline{\alpha} = -\overline{a}/\overline{b} \in F_p$. Then $\overline{\mathbb{Z}[\alpha]} \subset F_p$, so $\mathbb{Z}[\alpha]/\mathfrak{p} = F_p$, and hence $\mathrm{Norm}(\mathfrak{p}) = p$. □

Now the set of first-order prime ideals of norm p is in one-to-one correspondence with the set of solutions ($c \bmod p$) to $f(c) \equiv 0 \pmod{p}$, with the map given by letting c be the image of α in the mod-$\mathfrak{p}$ reduction map. Consequently, we will write first-order prime ideals in the form (p, c). The number $a + b\alpha$ is in the ideal (p, c) iff $0 = \overline{a + b\alpha} = \overline{a + bc}$, i.e. iff $a + bc \equiv 0 \pmod{p}$. Hence, it is easy to give a prime ideal factorization of $(a + b\alpha)$ in $\mathbb{Z}[\alpha]$: one needs to first prime-factor its norm $a^d - t(-b)^d$, and then for each prime-power factor p^k compute the unique ($c \bmod p$) for which $a + bc \equiv 0 \pmod{p}$; then $a + b\alpha$ contains the kth power of (p, c).

3.5 Finding relations

3.5.1 Computing the factor base

First we choose parameters B_1 and B_2. This is often done empirically, although the heuristic argument mentioned in Section 3.3 indicates (conjecturally) that an asymptotically optimal choice would be

$$B_1 \approx B_2 \approx \exp((2/3)^{2/3}(\log n)^{1/3}(\log\log n)^{2/3})$$

(see [6], §6.2, and [3], Conjecture 11.4).

Having chosen B_1 and B_2, we want to generate the set P of prime integers $\leq B_1$, a set U that generates the units of $\mathbb{Z}[\alpha]$, and a set G containing a generator for each first-order prime ideal of $\mathbb{Z}[\alpha]$ with norm $\leq B_2$. P can be computed easily with the Sieve of Eratosthenes. As a first step in computing G, the appropriate first-order prime ideals are computed. This step is also easy: By the preceding section, this is equivalent to finding solutions to $f(c) \equiv 0 \pmod{p}$ for primes $p < B_2$, and for each p this takes only $O(\log p)$ time [2]. That still leaves the task of finding a generator for each such ideal, and finding generators for the units. This can be done essentially through trial and error; see ([6],§3.1).

Applying φ to $P \cup U \cup G$ gives our "factor base" in $\mathbb{Z}/n\mathbb{Z}$.

3.5.2 Full Relations

We then search for pairs (a, b) of integers for which

$$\gcd(a, b) = 1, \;\; a + bm \text{ is } B_1\text{-smooth}, \;\; \text{and } \operatorname{Norm}(a + b\alpha) \text{ is } B_2\text{-smooth}. \tag{2}$$

Since we know which pair (p, c) each generator corresponds to, we can use the algorithm in Section 3.4 to factor $a + b\alpha$ in terms of the elements of U and G; and it is easy to factor $a + bm$ in terms of the elements of P; so applying φ to both of these factorizations gives the desired relation among the elements of the factor base.

It is worth noting that the reason the algorithm is called a "sieve" is because in this step, smooth pairs are found through sieving. For example, a sieve to find pairs with $a + bm$ smooth would work as follows: A list of possible (a, b) are generated; for each prime below B_2, all the numbers $a + bm$ which are multiples of that prime are divided by it; and the entries which are 1 after this process are the ones where $a + bm$ was B_1-smooth. In practice, the algorithm is a bit different: for example, by only keep track of the approximate size of entries, speed is greatly enhanced at a small price in thoroughness. The initial list of possible (a, b) is also important to choose appropriately – based on the density of smooth numbers, a reasonable choice is to take all relatively prime pairs (a, b) with $\max(|a|, |b|)$ on the order of B_1 and B_2 (see [6], §6.2).

3.5.3 Partial Relations

In order to lower B_1 and B_2, we expand our search to include additional pairs (a, b) besides those satisfying (2). Instead of forcing $a + bm$ to be B_1-smooth, we allow it to have at most one prime factor p_1 above B_1, which must also be below a higher bound B_3. Likewise, instead of forcing $\text{Norm}(a + b\alpha)$ to be B_2-smooth, we allow it to have at most one prime ideal factor $\mathfrak{p}_2$ above B_2, which then must have norm below a higher bound B_4. These additional pairs (a, b) are called *partial relations*, since they do not in themselves give a relation among the elements of the factor base. But we can combine two or more partial relations into a "full relation." For example, if (a_1, b_1) and (a_2, b_2) yield the same pair of large primes p_1 and $\mathfrak{p}_2$, then in the ratio of their factorizations, these large primes drop out, giving a full relation among the elements of the factor base. More generally, we can represent each partial relation as a line on a graph between the nodes p_1 and $\mathfrak{p}_2$ (or p_1 and 1 if there is no large prime ideal, or 1 and $\mathfrak{p}_2$ if there is no large prime integer), and then each cycle in the graph easily yields a full relation.

In practice, cycles account for the majority of full relations found in the Special Number Field Sieve. For example, in the factoring of the ninth Fermat number [5], about 80% of relations came from cycles. The larger the prime or prime ideal is, the less likely it is to contribute to a cycle; empirically, it seems to be counterproductive to set B_3 any higher than about $B_1^{1.4}$, or likewise to set B_4 any higher than about $B_2^{1.4}$ ([6], §4.6).

3.5.4 Free Relations

A limited number of extra, easy-to-compute relations are those that follow from the arithmetic of $\mathbb{Z}[\alpha]$. Each prime $p < \min(B_1, B_2)$ that is a product of d factors in $\mathbb{Z}[\alpha]$ (through splitting and/or ramification) is a product of first-order prime ideals (since the Norm is multiplicative); hence each such prime yields a "free relation" in the factor base. Ignoring the finite number of ramified primes, a prime splits completely iff it has a trivial Frobenius element; the density of such primes, by the Chebotarev Density Theorem and Galois Theory, is one over the degree of the splitting field of f. For example, if d is 3, then adjoining the two primitive cube-roots of unity to $\mathbb{Z}[\alpha]$ results in an extension of degree 6, so we would get roughly $\pi(\min(B_1, B_2))/6$ relations this way.

3.6 If $\mathbb{Z}[\alpha]$ is not a unique factorization domain

We next summarize how these algorithms are modified in the case that $\mathbb{Z}[\alpha]$ is not necessarily a unique factorization domain, and not necessarily the entire ring of integers. We first extend φ to the whole ring of integers $\mathcal{O}$ of $\mathbb{Q}(\alpha)$. $\mathcal{O}$ can be computed via any number of algorithms; see e.g. [7]. Then since $\mathbb{Z}[\alpha]$ is an order, there is an integer e

for which $e\mathcal{O} \subseteq \mathbb{Z}[\alpha]$, and by Proposition 7.2.4 and the remarks following 7.2.2 in the course textbook, e divides the discriminant of f, which one can directly compute to be $-(-d)^d t^{d-1}$. So as long as $\gcd(drs, n) = 1$ (which holds, or else we get a factorization of n), we conclude that $\varphi(e)$ is a unit in $\mathbb{Z}/n\mathbb{Z}$, and therefore we can extend φ to $\mathcal{O}$ by $\varphi(a/e) = \varphi(a)\varphi(e)^{-1}$.

Second, we note that Theorem 2 need not hold in this case. If p is relatively prime to the index $[\mathcal{O} : \mathbb{Z}[\alpha]]$, then by a theorem we proved in class (Lemma 5.2.1), Theorem 2 does still apply to primes $\mathfrak{p}$ lying over p. But if p does divide $[\mathcal{O} : \mathbb{Z}[\alpha]]$, then it is possible for a non-first-order prime ideal $\mathfrak{p}$ lying over p to intersect $\mathbb{Z}[\alpha]$ in a first-order prime ideal, and hence to contain a number $a + b\alpha$ with $\gcd(a, b) = 1$. We call such prime ideals *exceptional primes*. Note that, by the previous paragraph, the prims p which divide $[\mathcal{O} : \mathbb{Z}[\alpha]$ also divide drs, so we need only worry about the ideals over some small, fixed set of primes. Both determining exceptional ideals and testing divisibility by them is computationally quick – see ([7], Theorem 4.9ff).

The next difficulty is the possibility that $\mathcal{O}$ is not a principal ideal domain, and hence that it may be impossible to pick generators for all the prime ideals (as we did when forming G earlier). Note that there is a Minkowski bound M such that every ideal class contains an integral ideal of norm at most M. Then for each first-order or exceptional prime ideal $\mathfrak{p}$ satisfying $M < \mathrm{Norm}(\mathfrak{p}) \leq B_2$, we can find an element $\pi_\mathfrak{p} \in \mathfrak{p}$ such that $|\mathrm{Norm}(\pi_\mathfrak{p})| < M\,\mathrm{Norm}(\mathfrak{p})$ (multiply $\mathfrak{p}$ by an ideal with norm at most M in the ideal class $[\mathfrak{p}]^{-1}$). We let G' be the set of such $\pi_\mathfrak{p}$, and we let H be the group of nonzero elements $\beta \in \mathbb{Q}(\alpha)$ for which the fractional ideal $\beta\mathcal{O}$ contains only prime ideals of norm at most M in its (fractional) prime factorization. Then every B_2-smooth number is a product of elements of G' and one element of H (this is analogous to the fact that in the simpler case above, every B_2-smooth number was a product of elements of G and one unit of $\mathbb{Z}[\alpha]$). Finally, we need a set U' of independent generators for H; a number of algorithms for this purpose are possible – see ([6],§3.8). We extend φ to U' as we did for $\mathcal{O}$ before, and finally we have a new factor base $\varphi(P \cup G' \cup U')$.

3.7 Finishing the algorithm

Following the recipe in Section 3.5, we compute enough relations such that there are a small excess k of relations, relative to the size of the factor base (we will specify k below). As in Section 2, we put these relations into a matrix, one row for each relation, and search for (mod 2) linear dependences among the rows. This search can be done, for example, by standard Gaussian elimination, although there are faster techniques that take advantage of the sparseness of the matrix (see e.g. [5], §7). Each dependency thus found, again as in Section 2, gives a solution to $x^2 \equiv 1 \pmod{n}$, and hence, by Theorem 1, a possible factorization of n.

Each dependency, assuming that it generates a random square-root of 1, has prob-

ability at least 1/2 of generating a nontrivial factorization of n (by Theorem 1). And by linear algebra, if there are k more rows than columns, then there will be at least k linearly-independent dependencies. Hence, even taking k as small as 10, it is still reasonable to expect a 99.9% chance of finding a fruitful dependency. In practice, it is chosen to be a few orders of magnitude higher, to be sure of a fruitful dependency and to buffer against hardware errors.

4 General Number Field Sieve

4.1 Introduction to the General Number Field Sieve

The General Number Field Sieve is an extension of the ideas in Section 3 to general integers (i.e. integers not necessarily of the form $r^e - s$ with $r, |s|$ small).

The algorithm starts as before, using equation (1) to pick the degree d of the extension. The choice of m and f is somewhat different. We pick m by $m = [n^{1/d}]$, and we write n in base m with $n = a_d m^d + \cdots + a_0$, $0 \leq a_i < m$. Then f is defined by $f(x) = a_d x^d + \cdots + a_0$. Note that, as required, $f(m) = n \equiv 0 \pmod{n}$.

Unlike the fs generated for the Special Number Field Sieve, the above polynomials may have large coefficients (as high as $n^{1/d}$) and large discriminants, and consequently the number fields generated may be very hard to perform computations in. In particular, attempting to find generators for the units and for prime ideals through exhaustive search, as in Section 3.5.1, would take too long, and even storing these elements explicitly would take up too much space. For example, the Minkowski bound M is a multiple of the square root of the discriminant of the number field, so the elements of U' and G' (as in Section 3.6) would be quite large.

A good solution to these problems is to give up keeping track of explicit factorizations, and just focus on generating a pair (a, b) with $a + bm$ a perfect square in $\mathbb{Z}$ and $a + b\alpha$ a perfect square in $\mathbb{Z}[\alpha]$. Once this pair is generated, we compute the square roots of $a + bm$ and $a + b\alpha$, which easily give us a square root of 1 in $\mathbb{Z}/n\mathbb{Z}$. The principal advantage of this approach is that by not having to keep track of explicit factorizations, we do not need to write down U', G', or even $\mathcal{O}$. The principal disadvantage is that one must compute the square root of the (generally) large algebraic integer $a + b\alpha$, one of the most difficult parts of the algorithm. We will explain here the method of "quadratic characters," perhaps the most elegant and important aspect of the algorithm, which is used to generate squares in $\mathbb{Z}[\alpha]$ without knowing their complete factorizations. We will leave out other more minor aspects of the algorithm – for a more thorough account, see [3].

Before starting, however, we will mention one small detail of the modified algorithm

that will be relevant later. It may be the case that a product of numbers of the form $(a + b\alpha)$ will be a perfect square in $\mathcal{O}$, but not in $\mathbb{Z}[\alpha]$. As it turns out, multiplying the product by $f'(\alpha)^2$ gives a perfect square in $\mathbb{Z}[\alpha]$ ([3],§6). If we correspondingly multiply the product of $(a + bm)$ by $f'(m)^2$, the algorithm can run as before, but with no risk of producing a square root in $\mathcal{O} - \mathbb{Z}[\alpha]$. The only thing to check is that $\gcd(f'(m), n) = 1$: by definition of f, $1 < f'(m) < n$, so either this holds, or we have found a factor of n.

4.2 Quadratic Characters

Following [3], we introduce quadratic characters via an analogy. Suppose that we had a finite set X of integer primes, and an integer $r \neq 0$, and we knew that every prime outside X divided r to an even power. But suppose we could not look at the sign of r or the exponent of the primes in X. Then how can we tell if r is a perfect square? An effective way is to take a bunch of primes outside X, and test whether r is a perfect square modulo all of them. If not, r is definitely not a perfect square; if so, there is a good chance that it is.

To use this idea in the present case, we start with a definition:

Definition 1. Quadratic Characters. *Let (q, s) be a first-order prime ideal, as in Section 3.4, with q an odd prime, and suppose $(q, s) \nmid (c_0 + c_1\alpha + \cdots + c_{d-1}\alpha^{d-1})$ (equivalently, $c_0 + c_1 s + \cdots + c_{d-1}s^{d-1} \not\equiv 0 \pmod q$). Then we define the quadratic character $\chi_{(q,s)}$ in terms of the Legendre symbol as follows:*

$$\chi_{(q,s)}(c_0 + c_1\alpha + \cdots + c_{d-1}\alpha^{d-1}) = \left(\frac{c_0 + c_1 s + \cdots + c_{d-1}s^{d-1}}{q}\right).$$

Note that $\chi_{(q,s)}$ is a multiplicative homomorphism, since it is the composition of modding out by (q, s) and applying the Legendre symbol. Note also that $\chi_{(q,s)}$ can be quickly computed by quadratic reciprocity. Next we prove a theorem relating these characters to perfect squares in $\mathbb{Z}[\alpha]$.

Theorem 3. *Let S be a set of pairs of relatively prime integers (a, b) such that $\prod_{(a,b)\in S}(a + b\alpha)$ is a perfect square in $\mathbb{Q}(\alpha)$. Let (q, s) be a first-order prime ideal such that $a + bs \not\equiv 0 \pmod q$ for all $(a, b) \in S$, and $f'(s) \not\equiv 0 \pmod q$ (equivalently, $f'(\alpha) \notin (q, s)$). Then*

$$\prod_{(a,b)\in S} \chi_{(q,s)}(a + b\alpha) = 1.$$

Proof. Recall that there is a $z \in \mathbb{Z}[\alpha]$ such that $f'(\alpha)^2 \prod_{(a,b)\in S}(a + b\alpha) = z^2$. No factor on the left side is in (q, s) by hypothesis, so neither is z. Since $\chi_{(q,s)}$ is a multiplicative function, we apply it to both sides to prove the theorem. □

Next we look at the converse. For how many ideals (q, s) do we need to check $\prod \chi_{(q,s)}(a + b\alpha) = 1$ before we can say with some certainty that $\prod(a + b\alpha)$ is a square in $\mathbb{Q}(\alpha)$? To answer this, we define the multiplicative group V as follows:

$$V = \{z \in \mathbb{Q}(\alpha)^* \mid \text{ ord}_{\mathfrak{p}}(z\mathcal{O}) \text{ is even for all prime ideals } \mathfrak{p} \subset \mathbb{Z}[\alpha]\}$$

where $\text{ord}_{\mathfrak{p}} I$ is the exponent of $\mathfrak{p}$ in the prime ideal factorization of I.

Theorem 4. *Each $\chi_{(q,s)}$ with q odd and $f'(\alpha) \notin (q, s)$ induces a nontrivial group homomorphism from the quotient group $V/\mathbb{Q}(\alpha)^{*2}$ to $\{\pm 1\}$.*

Proof. First we show that each coset contains an element of $\mathbb{Z}[\alpha] - (q, s)$. Fix a coset H, and pick any $h \in H$. If $\text{ord}_{(q,s)}(h\mathcal{O}) \neq 0$, we pick an ideal I in the same class group as (q, s) but with $\text{ord}_{(q,s)}(I) = 0$, pick a generator g of $(q, s)I^{-1}$, and multiply or divide h by g^2 as necessary (by definition of V, the initial value of $\text{ord}_{(q,s)}(h\mathcal{O})$ is even). In this way, we can assume WLOG that $\text{ord}_{(q,s)}(h\mathcal{O}) = 0$. Now for each prime ideal $\mathfrak{p}$ in the denominator of (h), we multiply h by a generator for some positive power of $\mathfrak{p}$ (this generator exists by finiteness of the class group). This way, we can assume WLOG that $h \in \mathcal{O} - (q, s)$. Finally, we multiply h by $f'(\alpha)^2$, to get an element of $\mathbb{Z}[\alpha]$ (see [3], p61); note that $(q, s) \nmid (f'(\alpha))$ by hypothesis. Therefore, every coset contains an element in $h \in \mathbb{Z}[\alpha] - (q, s)$. We set $\chi_{(q,s)}(H) = \chi_{(q,s)}(h)$.

It remains to show that this function is independent of the choice of h. If $h_1, h_2 \in H$ are both in $\mathbb{Z}[\alpha] - (q, s)$, then $f'(\alpha)^2 h_1 h_2 = k^2$ for some $k \in \mathbb{Q}(\alpha)$; it is clear that in fact $k \in \mathbb{Z}[\alpha] - (q, s)$. Hence $\chi_{(q,s)}(f'(\alpha))^2 \chi_{(q,s)}(h_1)\chi_{(q,s)}(h_2) = \chi_{(q,s)}(k^2) = 1$, so indeed $\chi_{(q,s)}(h_1) = \chi_{(q,s)}(h_2)$.

We have now proven that $\chi_{(q,s)}$ is a well-defined homomorphism from $V/\mathbb{Q}(\alpha)^{*2}$ to $\{\pm 1\}$. The fact that it is nontrivial is simple: since q is odd, there is at least one quadratic nonresidue ($a \bmod q$), and then $\chi_{(q,s)}(a) = -1$. □

Note that $V/\mathbb{Q}(\alpha)^{*2}$ can be written as a vector space over F_2. With some additional work ([3], Theorem 6.7), one can also prove:

$$\dim_{F_2}(V/\mathbb{Q}(\alpha)^{*2}) < \log_2 n. \tag{3}$$

Now if we pick enough $\chi_{(q,s)}$ to span the space $\text{Hom}(V/\mathbb{Q}(\alpha)^{*2}, \{\pm 1\})$, then doing the test from Theorem 3 with each will give a completely reliable test for whether $\prod(a + b\alpha)$ (in V and satisfying the hypotheses of the theorem) is a perfect square in $\mathbb{Q}(\alpha)$. Moreover, by the Chebotarev density theorem, the $\chi_{(q,s)}$ (for q odd, $f'(\alpha) \notin (q, s)$) are asymptotically evenly distributed among all nontrivial elements of $\text{Hom}(V/\mathbb{Q}(\alpha)^{*2}, \{\pm 1\})$. Assuming that the same distribution holds among first-order primes with small q (this is only an assumption, and its accuracy should first be tested in different fields), we can regard different $\chi_{(q,s)}$ as picked randomly from the nontrivial elements of $\text{Hom}(V/\mathbb{Q}(\alpha)^{*2}, \{\pm 1\})$. Given this, how many $\chi_{(q,s)}$'s are necessary? To answer this we prove the following Lemma:

Lemma 1. *Let $k, \ell > 0$ be integers, and let W be a k-dimensional vector space over F_2. Let P be the probability that ℓ randomly-picked nontrivial elements of W span W. Then $P > 1 - 2^{k-\ell}$.*

Proof. The probability that these elements do *not* span W is the probability that they all lie on a $(k-1)$-hyperplane. Each such hyperplane is the kernel of a different nonzero element of the dual space of W; hence there are $2^k - 1$ of them. And given such a hyperplane, the probability that all ℓ elements lie on it is $2^{-\ell}$ (actually a bit less, since ℓ is constrained to be nontrivial). It follows that

$$1 - P < (2^k - 1)2^{-\ell} < 2^{k-\ell}.$$

□

Note that $\mathrm{Hom}(V/\mathbb{Q}^{*2}, \{\pm 1\})$ can be regarded as the dual space of $V/\mathbb{Q}^{*2}$, and hence has the same dimension over F_2. Thus, combining the above lemma with (3), we find that if we pick, say, $B = ([\log_2 n] + 30)$ pairs (q, s) we can be quite confident that the $\chi_{(q,s)}$ will give an accurate test of squareness for elements in V.

4.3 Summary of the Algorithm

We are now ready to summarize the algorithm for the General Number Field Sieve. We start with a sieving process to generate a set T of integer pairs (a, b) such that $\gcd(a, b) = 1$, $b > 0$, $(a + bm)$ is B_1-smooth, and $\mathrm{Norm}(a + b\alpha)$ is B_2-smooth. Let $\pi(B_1)$ be the number of primes below B_1 and let $\pi'(B_2)$ be the number of first-order prime ideals of $\mathbb{Z}[\alpha]$ with norm below B_2. We put all the elements of T into a matrix with $\pi(B_1) + \pi'(B_2) + B + 1$ columns and entries in $\mathbb{Z}/2\mathbb{Z}$, with one row for each $(a, b) \in T$, as follows. The first $\pi(B_1)$ entries are the exponent (mod 2) of p dividing $(a + bm)$, for each prime p below B_1. The next $\pi'(B_2)$ entries are $(\mathrm{ord}_{(q,s)}(a + b\alpha) \bmod 2)$, for each first-order prime ideal (q, s) with norm below B_2. The next B entries are $\nu(\chi_{(q,s)}(a+b\alpha))$, for B fixed choices of (q, s) with $q > B_2$ and $f'(\alpha) \notin (q, s)$; here, ν maps -1 to 1 and 1 to 0 – it merely converts the multiplicative representation of $\mathbb{Z}/2\mathbb{Z}$ to an additive one. And finally, the last entry is ν applied to the sign of $(a + bm)$.

We find (mod 2) dependencies in this matrix via some algorithm; the end result is a number of sets $S_i \subset T$, such that the sum of the matrix rows corresponding to the S_i gives 0 (mod 2). Since the first $\pi(B_1)$ and last entry are 0, we conclude that $\prod_{(a,b)\in S}(a + bm)$ is a square in $\mathbb{Z}$. Since the second $\pi'(B_2)$ entries are 0 (and $\mathrm{Norm}(a + b\alpha)$ is B_2-smooth), we get that $z := \prod_{(a,b)\in S_i}(a + b\alpha) \in V$. And since the next B entries are 0, we get that none of the chosen quadratic characters contradict z being a square in $\mathbb{Q}(\alpha)$; so by the discussion in Section 4.2, z is indeed almost certainly a square in $\mathbb{Q}(\alpha)$. It follows that $f'(\alpha)^2 z$ is the square of an element in $\mathbb{Z}[\alpha]$.

We compute the square root of $f'(m)^2 \prod_{(a,b)\in S_i}(a+bm)$ in $\mathbb{Z}$, which we denote by x – this step is easy, since we know the factorization of each $a+bm$. We also compute the square root of $f'(\alpha)^2 z = f'(\alpha)^2 \prod_{(a,b)\in S_i}(a+b\alpha)$ in $\mathbb{Z}[\alpha]$, which we denote by y – this step is computationally difficult, but not impossible – see ([3], §9) and [4]. We have $\varphi(x^2) \equiv \varphi(y^2) \pmod{n}$, so $(\varphi(x)/\varphi(y))^2 \equiv 1 \pmod{n}$. Hence each set S_i gives us a square root of unity. If we generate enough, we will likely find a nontrivial one, and by Theorem 1, this will give us a factorization of n.

References

[1] M. Agrawal, N. Kayal, and N. Saxena, "PRIMES is in P," Ann. of Math. **160** (2004), 781–793.

[2] E. R. Berlekamp, "Factoring Polynomials Over Large Finite Fields," Math. Comp. **24** (1970), 713–735. See also Andrei Jorza's section notes on this topic.

[3] J. P. Buhler, H. W. Lenstra, Jr., and C. Pomerance, "Factoring Integers With the Number Field Sieve," in A. K. Lenstra and H. W. Lenstra, Jr. (eds.) *The Development of the Number Field Sieve*, Lecture Notes in Mathematics 1554, Springer-Verlag, New York, 1993, pp. 50–94. [Section 4 in my paper largely followed this article.]

[4] J.-M. Couveignes, "Computing a Square Root for the Number Field Sieve," in A. K. Lenstra and H. W. Lenstra, Jr. (eds.) *The Development of the Number Field Sieve*, Lecture Notes in Mathematics 1554, Springer-Verlag, New York, 1993, pp. 95–102.

[5] A. K. Lenstra, H. W. Lenstra, Jr., M. S. Menasse, and J. M. Pollard, "The Factorization of the Ninth Fermat Number," Math. Comp. **61** (1993), 319–349. [The idea for Section 2 in my paper was from this article.]

[6] A. K. Lenstra, H. W. Lenstra, Jr., M. S. Manasse, and J. M. Pollard, "The Number Field Sieve," in A. K. Lenstra and H. W. Lenstra, Jr. (eds.) *The Development of the Number Field Sieve*, Lecture Notes in Mathematics 1554, Springer-Verlag, New York, 1993, pp. 11–42. [Section 3 in my paper largely followed this article.]

[7] H. W. Lenstra, Jr., "Algorithms in Algebraic Number Theory," Bull. Amer. Math. Soc, **26** (1992), 211–244.