Factoring integers with the number field sieve

J. P. Buhler, H. W. Lenstra, Jr., and Carl Pomerance

Abstract. In 1990, the ninth Fermat number was factored into primes by means of a new algorithm, the "number field sieve", which was invented by John Pollard. The present paper is devoted to the description and analysis of a more general version of the number field sieve. It should be possible to use this algorithm to factor arbitrary integers into prime factors, not just integers of a special form like the ninth Fermat number. Under reasonable heuristic assumptions, the analysis predicts that the time needed by the general number field sieve to factor n is $\exp((c+o(1))(\log n)^{1/3}(\log\log n)^{2/3})$ (for $n\to\infty$), where $c=(64/9)^{1/3}\doteq 1.9223$. This is asymptotically faster than all other known factoring algorithms, such as the quadratic sieve and the elliptic curve method. There does not yet exist an implementation of the number field sieve for general integers, so that a practical comparison cannot yet be made.

Key words: factoring integers, algebraic number fields.

1991 Mathematics subject classification: 11Y05, 11Y40.

Acknowledgements. The authors wish to thank Dan Bernstein, Arjeh Cohen, Michael Filaseta, Andrew Granville, Arjen Lenstra, Victor Miller, Robert Rumely, and Robert Silverman for their helpful suggestions. The authors were supported by NSF under Grants No. DMS 90-12989, No. DMS 90-02939, and No. DMS 90-02538, respectively. The second and third authors are grateful to the Institute for Advanced Study (Princeton), where part of the work on which this paper is based was done.

1. Introduction

In 1988 John Pollard circulated a manuscript [30] that described a new method for factoring integers. The procedure required the use of an algebraic number field tailored for the specific number n to be factored. In [23] a practical version of this idea was presented, dubbed by the authors "the number field sieve". This method has had several noteworthy successes in factoring numbers of the form $n = b^c \pm 1$, where b is small, from the Cunningham project (see [5]) The most spectacular of these factorizations was that of the ninth Fermat number $F_9 = 2^{2^9} + 1$, which has 155 decimal digits (see [22]).

The number field sieve has, so far, only been applied to factor numbers where certain desiderata were easily met. These include a monic irreducible polynomial $f \in \mathbf{Z}[X]$ of "small, but not too small" degree d, with "small" coefficients, and an integer $m \approx n^{1/d}$ such that $f(m) \equiv 0 \bmod n$. Further, if α is a zero of f, it is convenient for the ring of integers $\mathcal{O}$ of the number field $K = \mathbf{Q}(\alpha)$ to be not too much larger than $\mathbf{Z}[\alpha]$, for $\mathcal{O}$ to have class number one, and for the units of $\mathcal{O}$ to be easily computable.

For example, in the case $n = F_9$ the polynomial $f = X^5 + 8$ and the integer $m = 2^{103}$ were used; note that $f(m) = m^5 + 8 = 2^{515} + 8 \equiv 0 \bmod n$. More generally, for several numbers $n = b^c \pm 1$, with b small and c large, it has been fairly easy to meet the list of desiderata and to use the number field sieve to factor n. For numbers of this form it was suggested in [23] that the number field sieve takes time at most $L_n[\frac{1}{3}, (32/9)^{1/3} + o(1)]$ to factor n as n goes to infinity, where

$$L_n[u, v] = \exp(v(\log n)^u(\log\log n)^{1-u}).$$

The exponent $u = \frac{1}{3}$ in the number field sieve is the new and exciting aspect of this complexity function since all other known algorithms, such as the quadratic sieve or the elliptic curve method, have complexity, heuristic or probabilistic, at least $L_n[\frac{1}{2}, 1 + o(1)]$ for n tending to infinity through an infinite sequence of numbers.

Can the number field sieve be extended to general integers? It is to this question that this paper is addressed. We show that the method can be modified so that an arbitrary integer n can be factored with heuristic complexity $L_n[\frac{1}{3}, (64/9)^{1/3} + o(1)]$ for $n \to \infty$. We will call the new algorithm the number field sieve; if we need to specifically refer to the earlier algorithm we will refer to it as the special number field sieve.

The reason the constant $(64/9)^{1/3} \doteq 1.922999$ for the general case is larger than the constant $(32/9)^{1/3} \doteq 1.526285$ for the special number field sieve is that the coefficients of the polynomial f we construct below are about $n^{1/d}$. This is in a rough sense asymptotically best possible for general n, as we shall see in 12.10. For special values of n it may be possible to choose the coefficients of f much smaller, which makes the algorithm faster.

Is the number field sieve practical? Since it involves the same underlying sieving operations as, for instance, the quadratic sieve and the special number field sieve, it is our guess that this algorithm will eventually be the method of choice for sufficiently large integers. At the moment, its crossover with the "state-of-the-art" algorithm for factoring, namely the quadratic sieve, seems to be about 125 digits. This is so high that it is very difficult to factor a general number of this size with either method. The current record with the quadratic sieve is 116 decimal digits (see [24]). However, time is on the side of the number field sieve. It is reasonable to expect that hardware will improve and that the number field sieve will be refined and polished as it becomes better understood. Of course it is impossible to predict the future; some other faster factoring algorithm may be discovered that will supplant the quadratic sieve before theoretical and practical advances give the number field sieve its day in the sun.

If we compare the relative predicted performance of the number field sieve and the quadratic sieve on the basis of the somewhat questionable assumption that the "$o(1)$" terms in the heuristic complexity estimates can be ignored, then we find that the predicted number of operations for both are within a factor of about 3 for numbers between 100 and 150 decimal digits. This suggests that a small change in the implementation of either algorithm may have a large effect on the location of the crossover point.

Our description of the number field sieve incorporates the idea of Adleman [1] of using 'character columns', described in Section 8. In our original formulation of the number field sieve we had used a more awkward technique instead of character columns, which initially achieved only $L_n[\frac{1}{3}, 9^{1/3} + o(1)]$ as $n \to \infty$ for the heuristic complexity of the number field sieve, where $9^{1/3} \doteq 2.080084$; and it was only at the expense of considerable additional complications that we could obtain the bound $L_n[\frac{1}{3}, (64/9)^{1/3} + o(1)]$ with this technique. Adleman's idea achieves the latter bound with much less effort, and it simplifies the description of the algorithm in several ways. In addition it likely moves the number field sieve closer to being a practical factoring algorithm for arbitrary integers.

Another improvement to be mentioned is that of Coppersmith [10]. His idea reduces the complexity estimate even further, namely to $L_n[\frac{1}{3}, c + o(1)]$ for $n \to \infty$, where

$$c = \frac{(92 + 26\sqrt{13})^{1/3}}{3} \doteq 1.901884.$$

However, it is unlikely that this method will be practical for numbers of reasonable size (of fewer than 1000 digits, say).

The idea underlying the number field sieve has also been applied to the discrete logarithm problem. For this, we refer to [14] and [34].

The structure of this paper is as follows. Section 2 contains an outline of the number field sieve. In Section 3 we describe an algorithm for selecting the number field to be used by the algorithm. Section 4 is devoted to a description of a well-known sieving technique for constructing squares in the field of rational numbers. In Section 5 we carry this technique over to the algebraic number field. It turns out that we have to deal with certain obstructions, which are described and analyzed in Section 6. Two algebraic facts that are used in Sections 5 and 6 are proved in Section 7. We overcome the obstructions in Section 8, by using the character columns that were suggested by Adleman. In Section 9 we discuss a problem that has not appeared in earlier factoring algorithms, namely that of taking square roots in algebraic number fields. In Section 10 we state a heuristic principle that can be used to obtain running time estimates for a surprisingly wide class of factoring algorithms. Section 11 summarizes the entire algorithm and gives a heuristic running time analysis. Finally, in Section 12 we describe a modification of the number field sieve that should improve its practical performance.

2. The idea of the number field sieve

A very old factoring strategy going back to Fermat and Legendre is to write n as a difference of two squares. More generally, it suffices to find a solution to $x^2 \equiv y^2 \bmod n$. One might then obtain a factorization of n by finding the greatest common divisor of $x - y$ and n. In fact, it is easy to prove that if n is divisible by at least two distinct odd primes then for at least half of the pairs $x \bmod n$, $y \bmod n$ with $x^2 \equiv y^2 \bmod n$ and $\gcd(xy, n) = 1$, we have $1 < \gcd(x - y, n) < n$. There are many factoring algorithms that exploit this idea by trying to construct such pairs x, y in a random or pseudo-random manner. These algorithms include the continued fraction method [29]. the random squares method [11], the quadratic sieve [32], and, of course, the special number field sieve.

Before we see how the number field sieve attempts to find a solution to $x^2 \equiv y^2 \bmod n$ we say a few words about the ring in which the number field sieve operates. Suppose $f \in \mathbf{Z}[X]$ is monic and irreducible of degree $d > 1$. We shall work with the ring $\mathbf{Z}[\alpha]$ that is generated by a zero α of f. It makes no difference whether one thinks of $\mathbf{Z}[\alpha]$ as a subring of the field of complex numbers or as the ring $\mathbf{Z}[X]/f\mathbf{Z}[X]$, with $\alpha = (X \bmod f)$; all that matters is that each element of $\mathbf{Z}[\alpha]$ can in a unique way be written in the form $\sum_{i=0}^{d-1} a_i \alpha^i$, with $a_0, a_1, \ldots, a_{d-1} \in \mathbf{Z}$. Thus, each element of $\mathbf{Z}[\alpha]$ can be represented as a vector with d integral coordinates a_i. The addition in the ring is then just vector addition. To multiply two polynomial expressions in α, one first multiplies them as polynomials, and next uses the relation $f(\alpha) = 0$ to reduce the result to a polynomial expression of degree less than d in α. If we let, in a completely analogous way, the a_i range over the field $\mathbf{Q}$ of rational numbers rather than over $\mathbf{Z}$, then we obtain the field of fractions $\mathbf{Q}(\alpha)$ of $\mathbf{Z}[\alpha]$.

Coming back to the number field sieve, let us now assume that $m \in \mathbf{Z}$ satisfies $f(m) \equiv 0 \bmod n$. Then there is a natural ring homomorphism $\varphi\colon \mathbf{Z}[\alpha] \to \mathbf{Z}/n\mathbf{Z}$ induced by $\varphi(\alpha) = (m \bmod n)$; so $\varphi(\sum_i a_i\alpha^i) = (\sum_i a_i m^i \bmod n)$. Suppose we can find a non-empty set S of pairs (a, b) of relatively prime integers with the following two properties:

$$\prod_{(a,b)\in S} (a + bm) \quad \text{is a square in } \mathbf{Z}, \tag{2.1}$$

$$\prod_{(a,b)\in S} (a + b\alpha) \quad \text{is a square in } \mathbf{Z}[\alpha]. \tag{2.2}$$

Let $x \in \mathbf{Z}$ be a square root of the square in (2.1) and let $\beta \in \mathbf{Z}[\alpha]$ be a square root of the element of $\mathbf{Z}[\alpha]$ in (2.2). Since $\varphi(a + b\alpha) = (a + bm \bmod n)$, we have $\varphi(\beta^2) = (x^2 \bmod n)$. Let $y \in \mathbf{Z}$ be such that $\varphi(\beta) = (y \bmod n)$. Then $y^2 \equiv x^2 \bmod n$, and we have constructed our congruent squares and so may attempt to factor n by computing $\gcd(y - x, n)$.

There are several questions that are raised by the above outline:

(i) How are the polynomial f and the integer m to be constructed?

(ii) How is the set S of coprime integer pairs that satisfies (2.1) and (2.2) to be found?

(iii) How is an element $\beta \in \mathbf{Z}[\alpha]$ to be found such that β^2 is the square in (2.2)?

(iv) How much time do these steps take?

The overall plan of this paper is to gradually answer these questions until we can finally state a precise version of the algorithm and attempt to analyze its complexity.

Remark. The basic goal of most "combination of congruence" factoring algorithms, including the number field sieve, can be encapsulated algebraically by saying that we have a ring R and a ring homomorphism $\psi\colon R \to \mathbf{Z}/n\mathbf{Z} \times \mathbf{Z}/n\mathbf{Z}$ together with a means of generating many elements of R whose image under ψ lies in the diagonal $\{(x, x) : x \in (\mathbf{Z}/n\mathbf{Z})^*\}$. We then hope to combine these elements multiplicatively to obtain squares in R whose square roots have an image under ψ not lying in $\{(x, \pm x) : x \in (\mathbf{Z}/n\mathbf{Z})^*\}$. In the case of the quadratic sieve we have $R = \mathbf{Z} \times \mathbf{Z}$. In the case of the number field sieve we have $R = \mathbf{Z} \times \mathbf{Z}[\alpha]$ and $\psi(r, \beta) = (r \bmod n, \varphi(\beta))$, and we consider elements of the form $(a + bm, a + b\alpha)$. It is tempting to consider more general rings, e.g., $R = \mathbf{Z}[\alpha] \times \mathbf{Z}[\alpha']$, or $R = \mathbf{Z}[\alpha]$ where f has *two* zeroes modulo n, but so far we have not found a way to exploit this.

3. Finding a polynomial

Given a positive integer n that is not a prime power, the first step of the number field sieve algorithm is to find a polynomial f with integer coefficients and an integer m such that $f(m)$ is a multiple of n. In the basic version of the number field sieve that we will present, the following particularly simple method is used to find a polynomial; this algorithm will be referred to as the "base m" method.

Suppose that we are given positive integers n and d with $d > 1$ and $n > 2^{d^2}$. Set $m = [n^{1/d}]$, and write n to the base m:

$$n = c_d m^d + c_{d-1} m^{d-1} + \ldots + c_0 \tag{3.1}$$

where the "digits" c_i satisfy, as usual, the inequality $0 \leq c_i < m$. The output of the base m algorithm consists of the integer m and the polynomial $f = X^d + c_{d-1}X^{d-1} + \ldots + c_1 X + c_0$. Note that we have $f(m) = n$.

Proposition 3.2. *The leading coefficient c_d of f is equal to 1, and $c_{d-1} \leq d$.*

Proof. From our assumption $n > 2^{d^2}$ we have $\binom{d}{i} \leq 2^d - 2 \leq n^{1/d} - 2 \leq m - 1$. Therefore the digits of $(m+1)^d$ in the base m are the binomial coefficients $\binom{d}{i}$, and the proposition follows from the inequalities $m^d \leq n < (m+1)^d$.

For the d that we will recommend later, n will be much larger than 2^{d^2}.

The polynomial f produced by the base m algorithm may be reducible. However, since our interest lies in factoring n, this event would be fortunate. Indeed, if $f = gh$ is a non-trivial factorization of f in $\mathbf{Z}[X]$ then $g(m)h(m) = f(m) = n$ is a non-trivial splitting of n in $\mathbf{Z}$. This result follows from the proofs in [4], where we need only the easier case $m \geq 3$. We note that f can be factored in time $(\log n)^{O(1)}$, by means of the algorithm of [21].

In a weak asymptotic sense, the base m algorithm, simple as it may be, cannot be improved for use in the number field sieve, although for practical purposes there is still room for improvement. This is further discussed in 12.10 and 12.15.

The following estimate will be needed later in this paper. We let f be as produced by the base m algorithm, with $d > 1$, $n > 2^{d^2}$.

Lemma 3.3. *The discriminant Δ of f satisfies $|\Delta| < d^{2d}n^{2-3/d}$.*

Proof. The discriminant of the monic polynomial f is, up to sign, equal to the resultant of f and its derivative, which in turn is equal to the determinant of the corresponding Sylvester matrix (see [36, Sections 34 and 35]). The non-zero entries of each of the first $d-1$ rows of that matrix are the coefficients of f, and the non-zero entries of each of the remaining d rows are the coefficients of f'. To estimate the determinant, we divide each of the last d rows, corresponding to f', by d, and we divide each of the last $2d-3$ columns by m; those are the columns involving a c_i with $i < d-1$. Finally, we subtract c_{d-1} times the first column from the second column. This results in a matrix of which all entries are at most 1 in absolute value. Each of the first $d-1$ row vectors of that matrix has Euclidean length at most $\sqrt{d+1}$, and each of the last d row vectors has Euclidean length at most $\sqrt{d}$. Thus from Hadamard's determinant bound we obtain

$$|\Delta| \le d^d m^{2d-3}(d+1)^{(d-1)/2} d^{d/2} < d^{2d} n^{2-3/d},$$

using $m^d \le n$ and $d > 1$ for the last inequality. This proves 3.3.

4. The rational sieve

We let n and d be integers with n, $d > 1$, and we let $f \in \mathbf{Z}[X]$ a monic irreducible polynomial of degree d. We let m be an integer with the property $f(m) \equiv 0 \bmod n$. By α we denote a zero of f, as explained in Section 2. We write $\mathbf{Z}[\alpha]$ for the ring generated by α.

As suggested above, the heart of the number field sieve lies in constructing a non-empty set S of coprime integer pairs for which we have

$$\prod_{(a,b)\in S} (a+bm) \quad \text{is a square in } \mathbf{Z}, \tag{4.1}$$

$$\prod_{(a,b)\in S} (a+b\alpha) \quad \text{is a square in } \mathbf{Z}[\alpha]. \tag{4.2}$$

Basically, the construction of S proceeds in two steps. First, one uses a sieve to find a set T of pairs (a,b) such that both $a+bm$ is smooth (i. e., factors into small primes), and $a+b\alpha$ is smooth (in a similar sense, to be defined later) in $\mathbf{Z}[\alpha]$. Next, one uses linear algebra over the field with two elements to locate $S \subset T$.

Let u be a large positive number to be chosen later, depending on n. Our overall universe of possible pairs, from which the sets T and S will be chosen, is

$$U = \{(a,b) : a, b \in \mathbf{Z},\ \gcd(a,b) = 1,\ |a| \le u,\ 0 < b \le u\}. \tag{4.3}$$

We will need to choose the parameter u sufficiently large so that U contains a non-empty set S satisfying (4.1) and (4.2).

Initially, we will discuss conditions (4.1) and (4.2) separately. That is, in the present section we focus on the "rational" side of the number field sieve, i. e., finding a set S satisfying (4.1). Next we shall concentrate on the "algebraic" side (4.2). Finally, we shall see how to achieve (4.1) and (4.2) simultaneously.

The procedure for finding a square in $\mathbf{Z}$ by sieving is standard; we recall the idea. First a parameter $y = y(n)$ is chosen, and by sieving one finds a subset

$$T_1 = \{(a,b) \in U : a+bm \text{ is } y\text{-smooth}\},$$

where we say that an integer is y-smooth if all of its prime divisors are less than or equal to y. The sieving procedure works as follows. For each fixed integer b with $0 < b \le u$ an array is initialized with the integers $a + bm$ for $-u \leq a \leq u$. For each prime number $p \leq y$ the numbers in the array corresponding to values of a with $a \equiv -bm \bmod p$ are retrieved one at a time, divided by the highest power of p that divides them, and the quotient is replaced in the same array at the same location from which the number was retrieved. At the end of this procedure the number in the ath location is, up to sign, the largest divisor of $a + bm$ that is coprime to the primes up to y. Any location that contains the number 1 or -1 at the end of the procedure corresponds to a number $a + bm$ that is y-smooth. If $\gcd(a, b) = 1$, we have thus detected a member of T_1.

In practice various devices can be used to speed up the sieving. For instance, it is more efficient to replace the numbers in the array by their approximate logarithms (say to base 2), to initialize the array with 0 instead of the logarithms of the numbers $|a + bm|$, to add the logarithm of p instead of dividing by p, to ignore small primes, to ignore higher powers of p, and to inspect, at the end of the procedure, all values of a for which the ath location contains a number exceeding a certain bound independent of a.

Remark. The primes less than or equal to y are said to be in the "factor base" of the sieve. The precise choice of the parameters y and u will be given later as part of the complexity analysis of the final algorithm, see Section 11.

Suppose the parameters u and y are chosen so that $\#T_1 > \pi(y) + 1$, where $\#T_1$ denotes the cardinality of the set T_1 and $\pi(y)$ denotes the number of primes up to y. It is well-known that by using linear algebra over the field $\mathbf{F}_2$ with two elements one can find a non-empty subset S of T_1 for which (4.1) holds; again we recall the idea.

Let $B = \pi(y)$, let p_j denote the jth prime, for $1 \leq j \leq B$, and let $p_0 = -1$. For a y-smooth integer

$$w = \prod_{j=0}^{B} p_j^{e_j}$$

we define the exponent vector $e(w) \in \mathbf{F}_2^{B+1}$ by

$$e(w) = (e_0 \bmod 2,\ e_1 \bmod 2,\ \ldots,\ e_B \bmod 2).$$

We may form such a vector $e(a + bm)$ for each $(a, b) \in T_1$. Since the number of such vectors exceeds the dimension of the $\mathbf{F}_2$-vector space $\mathbf{F}_2^{B+1}$, there is a non-trivial linear dependence relation with coefficients 0 and 1, and hence a non-empty subset $S \subset T_1$ such that

$$\sum_{(a,b)\in S} e(a+bm) = 0 \in \mathbf{F}_2^{B+1}.$$

Therefore

$$\prod_{(a,b)\in S} (a+bm) \quad \text{is a square in } \mathbf{Z}.$$

Thus we have "solved" (4.1) by combining smooth elements.

5. The algebraic sieve

The notation and hypotheses in this section are as in Section 4. In addition, we write K for the field of fractions $\mathbf{Q}(\alpha)$ of $\mathbf{Z}[\alpha]$ (see Section 2) and $\mathcal{O}$ for the ring of algebraic integers in K. The multiplicative group of K is indicated by K^*, and $N: K \to \mathbf{Q}$ is the norm map of the extension $\mathbf{Q} \subset K$. For background on algebraic number theory we refer to [18; 39].

In order to find a square in $\mathbf{Z}[\alpha]$, i.e., find a set satisfying (4.2), we attempt to mimic the well-worn strategy described in the previous section. If the ring $\mathbf{Z}[\alpha]$ is a unique factorization domain this would be fairly easy, though problems with units would still remain. We note that in only a few of the applications so far of the special number field sieve, $\mathbf{Z}[\alpha]$ has been a unique factorization domain, but in the remaining cases where it has not, the full ring of integers $\mathcal{O}$ in K has been. Since we certainly cannot count on this being true for arbitrary numbers, we will describe a strategy for solving (4.2) that does not depend on special properties of $\mathbf{Z}[\alpha]$.

Define an element $\beta \in \mathbf{Z}[\alpha]$ to be y-smooth if its norm $N(\beta) \in \mathbf{Z}$ is y-smooth. We can calculate the norm of an element of the form $a+b\alpha$ by substituting a, b in the homogeneous polynomial $(-Y)^d f(-X/Y)$; that is, if a, $b \in \mathbf{Z}$ then

$$(5.1) \qquad N(a+b\alpha) = a^d - c_{d-1}a^{d-1}b + \ldots + (-1)^d c_0 b^d$$

where $f = X^d + c_{d-1}X^{d-1} + \ldots + c_0$.

A modification of the earlier sieving idea can be used to find the set

$$T_2 = \{(a,b) \in U : a+b\alpha \text{ is } y\text{-smooth}\},$$

where U is as in (4.3). Namely, for each prime p let the set of zeroes of f mod p be denoted by $R(p)$, i. e., $R(p) = \{r \in \{0, 1, \ldots, p-1\} : f(r) \equiv 0 \bmod p\}$. Then for any fixed integer b with $0 < b \leq u$ and $b \not\equiv 0 \bmod p$, the integers a with $N(a + b\alpha) \equiv 0 \bmod p$ are those with $a \equiv -br \bmod p$ for some $r \in R(p)$. Note that if $b \equiv 0 \bmod p$, then there are no integers a with $(a, b) \in U$ and $N(a + b\alpha) \equiv 0 \bmod p$.

For each fixed b initialize an array with the numbers $N(a + b\alpha)$ for $-u \leq a \leq u$. For each prime $p \leq y$ that does not divide b and each choice of $r \in R(p)$ the positions corresponding to a that are congruent to $-br \bmod p$ are identified, the numbers in these positions are retrieved and divided by the highest power of p that divides them and then the quotient is replaced in the array as before. At the end of this process the locations containing ± 1 correspond to y-smooth values of $a + b\alpha$ with $\gcd(a, b) = 1$, and hence to elements of T_2. We can make this procedure more efficient by using the techniques mentioned in the previous section, including the use of approximate logarithms.

Remark 5.2. Note that for each prime p we might sieve as many as d residue classes modulo p; however, heuristically the average size of $R(p)$ is about 1 (see [18, Chapter VIII, Section 4]). (This would even be provable if we were to choose y large enough.)

The next step is to apply linear algebra over the field with two elements, but here some complications arise. In the previous section we combined the numbers $a + bm$, for $(a, b) \in T_1$, into a square by using their exponent vectors. Similarly, we can now use the exponent vectors of the numbers $N(a + b\alpha)$ for $(a, b) \in T_2$ and proceed with them in the same way. However, this leads only to a subset $S \subset T_2$ for which the *norm* of the product $\prod_{(a,b)\in S}(a + b\alpha)$ is a square (in $\mathbf{Z}$). This is a necessary condition for the product *itself* to be a square in $\mathbf{Z}[\alpha]$ (or even just in K), but it is very far from being sufficient. It turns out that we can overcome this problem almost completely by keeping track, for each prime number p dividing $N(a + b\alpha)$, of the value $r \in R(p)$ that is "responsible" for the fact that p divides $N(a + b\alpha)$.

More explicitly, let a, $b \in \mathbf{Z}$ satisfy $\gcd(a, b) = 1$. Further let p be a prime number and r an element of the set $R(p)$ defined above. Then we define $e_{p,r}(a + b\alpha)$ by

$$e_{p,r}(a + b\alpha) = \begin{cases} \operatorname{ord}_p(N(a + b\alpha)) & \text{if } a + br \equiv 0 \bmod p \\ 0 & \text{otherwise,} \end{cases}$$

where $\mathrm{ord}_p(k)$ is the number of factors p in k. Clearly we have

$$N(a+b\alpha) = \pm \prod_{p,r} p^{e_{p,r}(a+b\alpha)},$$

the product ranging over all pairs p, r with p prime and $r \in R(p)$. The following result justifies the introduction of the numbers $e_{p,r}(a+b\alpha)$.

Proposition 5.3. *Let S be a finite set of coprime integer pairs (a,b) with the property that $\prod_{(a,b)\in S}(a+b\alpha)$ is the square of an element of K. Then for each prime number p and each $r \in R(p)$ we have*

$$\sum_{(a,b)\in S} e_{p,r}(a+b\alpha) \equiv 0 \bmod 2.$$

This proposition is proved below.

For the number field sieve we are really interested in the converse of the proposition: if the congruence in 5.3 holds for all pairs p, r, does it follow that $\prod_{(a,b)\in S}(a+b\alpha)$ is a square? The answer is "no", as is shown by the example $S = \{(-1,0)\}$, if K does not contain a square root of -1. However, we shall see, using the results in Section 7, that the extent to which the converse fails can be measured, that it is quite small (see Theorem 6.7), and that the failure of the converse can be overcome by the use of quadratic characters (see Section 8).

In order to prove 5.3 it is convenient to recall some basic facts about the non-zero prime ideals, or "primes" as we shall call them, of the ring $\mathbf{Z}[\alpha]$. If $P \subset \mathbf{Z}[\alpha]$ is a prime, then $\mathbf{Z}[\alpha]/P$ is a finite field, and P contains a unique prime number p (see Section 7). The *norm* $\mathrm{N}P$ of a prime P is the number of elements $\mathrm{N}P = \#\mathbf{Z}[\alpha]/P$ of its residue class field, and the *degree* of P is the degree of $\mathbf{Z}[\alpha]/P$ as a field extension of its prime field $\mathbf{F}_p$. If P is a *first degree* prime, then $\mathbf{Z}[\alpha]/P$ is isomorphic to $\mathbf{F}_p$, we have $\mathrm{N}P = p$, and the map $\mathbf{Z}[\alpha] \to \mathbf{F}_p$ with kernel P sends α to a zero $r \bmod p$ of $f \bmod p$. Hence, a first degree prime P gives rise to a pair p, r as considered above. Conversely, if p is a prime number and $r \in R(p)$, then there is a unique ring homomorphism $\mathbf{Z}[\alpha] \to \mathbf{F}_p$ that maps α to $r \bmod p$, and its kernel is a first degree prime P of $\mathbf{Z}[\alpha]$. Thus there is a one-to-one correspondence between pairs p, r with $r \in R(p)$ and first degree primes $P \subset \mathbf{Z}[\alpha]$; the ideal P corresponding to p, r is generated by p and $\alpha - r$.

We shall interpret the number $e_{p,r}(a+b\alpha)$ defined above as the "number of factors P in $a+b\alpha$", where P corresponds to p, r. If $\mathbf{Z}[\alpha]$ is equal to the full ring of integers $\mathcal{O}$ of K then it is clear what we mean by this: it is a standard fact from algebraic number theory that non-zero ideals of $\mathcal{O}$ factor uniquely into primes, and $e_{p,r}(a+b\alpha)$ is the exponent of P in the factorization of the ideal $(a+b\alpha)\mathcal{O}$. In order to generalize this to the case in which $\mathbf{Z}[\alpha] \neq \mathcal{O}$ we need the following result.

Proposition 5.4. *There is, for each prime P of $\mathbf{Z}[\alpha]$, a group homomorphism $l_P \colon K^* \to \mathbf{Z}$, such that the following hold:*

(a) $l_P(\beta) \geq 0$ *for all* $\beta \in \mathbf{Z}[\alpha]$, $\beta \neq 0$;

(b) *if* $\beta \in \mathbf{Z}[\alpha]$, $\beta \neq 0$, *then* $l_P(\beta) > 0$ *if and only if* $\beta \in P$;

(c) *for each* $\beta \in K^*$ *one has* $l_P(\beta) = 0$ *for all but finitely many* P, *and*

$$\prod_P (\mathrm{N}P)^{l_P(\beta)} = |N(\beta)|,$$

where P ranges over the set of all primes of $\mathbf{Z}[\alpha]$.

If $\mathbf{Z}[\alpha] = \mathcal{O}$, it suffices to take $l_P(x)$ equal to the exponent to which P appears in the prime ideal factorization of the ideal $x\mathcal{O}$. The proof of 5.4 for the general case is given in Section 7. It does not use algebraic number theory, but depends on the Jordan-Hölder theorem.

Corollary 5.5. *Let a and b be coprime integers and let P be a prime of $\mathbf{Z}[\alpha]$. If P is not a first degree prime, then $l_P(a+b\alpha) = 0$. If P is a first degree prime, corresponding to a pair p, r, then $l_P(a+b\alpha) = e_{p,r}(a+b\alpha)$.*

Proof. Let P be a prime of $\mathbf{Z}[\alpha]$ with $l_P(a+b\alpha) > 0$, and let p be the prime number contained in P. By 5.4(b), the element $a+b\alpha$ maps to 0 under the map $\mathbf{Z}[\alpha] \to \mathbf{Z}[\alpha]/P$. If p divides b, then $b\alpha$ also maps to 0, so the same is true for a, and therefore p divides a; this contradicts that $\gcd(a,b) = 1$. It follows that b maps to a non-zero element of $\mathbf{Z}[\alpha]/P$. Denote by b' the inverse of the image of b; it belongs to the prime field $\mathbf{F}_p$ of $\mathbf{Z}[\alpha]$. Since

$a + b\alpha$ maps to 0, the element α maps to $-ab'$, which belongs to $\mathbf{F}_p$. Therefore all of $\mathbf{Z}[\alpha]$ maps to $\mathbf{F}_p$, which proves that P is a first degree prime. This implies the first assertion of 5.5. If P corresponds to p, r, then r is determined by $a + br \equiv 0 \bmod p$. This shows that P is the unique prime of $\mathbf{Z}[\alpha]$ containing p and $a + b\alpha$. Now the last statement of 5.5 follows if one compares the power of p on both sides of 5.4(c). This proves 5.5.

We can now prove Proposition 5.3. Let $\prod_{(a,b)\in S}(a + b\alpha) = \gamma^2$, and let P be the first degree prime corresponding to p, r. Since l_P is a homomorphism, we have

$$\sum_{(a,b)\in S} e_{p,r}(a+b\alpha) = \sum_{(a,b)\in S} l_P(a+b\alpha) = l_P\Big(\prod_{(a,b)\in S}(a+b\alpha)\Big)$$
$$= l_P(\gamma^2) = 2l_P(\gamma) \equiv 0 \bmod 2.$$

This proves 5.3.

6. Four obstructions

We retain the previous notation and remind the reader that we are trying to find a square in $\mathbf{Z}[\alpha]$ by finding a non-empty subset S of

$$T_2 = \{(a,b) \in U : a + b\alpha \text{ is } y\text{-smooth}\}$$

such that the product, over all $(a,b) \in S$, of $a + b\alpha$ is a perfect square in $\mathbf{Z}[\alpha]$.

Suppose there are exactly B' first degree primes P of $\mathbf{Z}[\alpha]$ of norm at most y. (We expect B' to be close to $\pi(y)$—see Remark 5.2.) If $\#T_2 > B'$ the linear algebra described in Section 4 can be modified to give us a non-empty set $S \subset T_2$ such that

$$\text{(6.1)} \qquad \sum_{(a,b)\in S} l_P(a+b\alpha) \equiv 0 \bmod 2 \qquad \text{for all } P.$$

This is weaker than we want. In fact there are four obstructions that may prevent a set S that satisfies (6.1) from satisfying (4.2):

(6.2) The ideal $\prod_{(a,b)\in S}(a + b\alpha)\mathcal{O}$ of $\mathcal{O}$ may not be the square of an ideal, since we work with primes of $\mathbf{Z}[\alpha]$ rather than with primes of $\mathcal{O}$.

(6.3) Even if $\prod_{(a,b)\in S}(a + b\alpha)\mathcal{O} = I^2$ for some ideal I of $\mathcal{O}$, the ideal I need not be principal.

(6.4) Even if $\prod_{(a,b)\in S}(a+b\alpha)\mathcal{O}=\gamma^2\mathcal{O}$ for some $\gamma\in\mathcal{O}$, it is not necessary that $\prod_{(a,b)\in S}(a+b\alpha)=\gamma^2$.

(6.5) Even if $\prod_{(a,b)\in S}(a+b\alpha)=\gamma^2$ for some $\gamma\in\mathcal{O}$, we need not have $\gamma\in\mathbf{Z}[\alpha]$.

We remark that if $\mathbf{Z}[\alpha]=\mathcal{O}$ then the obstructions (6.2) and (6.5) cannot occur. Further, if $\mathcal{O}$ has class number one, and is hence a principal ideal domain, then obstruction (6.3) cannot occur. Finally, if $\mathcal{O}$ is a principal ideal domain and we have an explicit basis for the unit group of $\mathcal{O}$ then we can handle the obstruction (6.4) by linear algebra by including a system of generating units in our factor base. However, in general we cannot make any of these assumptions.

First we note that the fourth obstruction can be dealt with very easily. Namely, if

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

with $\gamma\in K$, then $\gamma\in\mathcal{O}$ and $\gamma f'(\alpha)\in\mathbf{Z}[\alpha]$ (see [39, Proposition 3-7-14]), so

$$f'(\alpha)^2\cdot\prod_{(a,b)\in S}(a+b\alpha)\quad\text{is the square of an element of }\mathbf{Z}[\alpha]. \tag{6.6}$$

Thus we may replace (4.2) with (6.6) in our factoring algorithm if we also multiply (4.1) by $f'(m)^2$. Indeed, if f and m are chosen by the base m algorithm then $1<f'(m)<n$ so that we can assume that $\gcd(f'(m),n)=1$ (since otherwise n would be factored); thus multiplying (4.1) by $f'(m)^2$ will not affect our chance of factoring n.

We could have dealt with the first obstruction by working with the primes P of $\mathcal{O}$ rather than those of $\mathbf{Z}[\alpha]$. There is an efficient algorithm for constructing the functions l_P for those primes, given in [6] (cf. [26, Theorem 4.9]). In practice—or perhaps in the application of the number field sieve to the discrete logarithm problem in a finite field as in [14; 34]—it may be better to use the algorithm from [6]. However, it turns out that the techniques we have to use anyway, in order to cope with obstructions (6.3) and (6.4), also can be used to get around the difference between $\mathbf{Z}[\alpha]$ and $\mathcal{O}$. Thus for simplicity we do not use the algorithm of [6] in what follows.

In Section 8 we describe how to deal with (6.2), (6.3) and (6.4); in the remainder of this section we show that these obstructions are, in a suitable sense, "small" obstructions.

Denote by V the multiplicative group of those $\beta \in K^*$ with the property that $l_P(\beta) \equiv 0 \bmod 2$ for all primes P of $\mathbf{Z}[\alpha]$. Since each l_P is a group homomorphism, we have $K^{*2} \subset V$. The quotient V/K^{*2} is a vector space over $\mathbf{F}_2$ in a natural way. We can readily produce elements of V but would like elements of K^{*2}; we can measure our obstructions precisely by bounding the dimension of the quotient.

Theorem 6.7. *Let n, d be integers with $d \geq 2$ and $n > d^{2d^2}$, and let m, f be as produced by the base m algorithm in Section 3. Let $K = \mathbf{Q}(\alpha)$ be as in Section 5, and V as defined above. Then we have $\dim_{\mathbf{F}_2} V/K^{*2} < (\log n)/\log 2$.*

Note that this is equivalent to $[V : K^{*2}] < n$. Note also that the bound $n > d^{2d^2}$ supersedes the bound $n > 2^{d^2}$ required in Section 3.

We prove 6.7. Define

$$W = \{\gamma \in K^* : \gamma\mathcal{O} = I^2 \text{ for some fractional } \mathcal{O}\text{-ideal } I\}.$$

In Section 7 we shall prove that

$$V \supset W, \qquad [V : W] \leq [\mathcal{O} : \mathbf{Z}[\alpha]]. \tag{6.8}$$

Let $Y = \mathcal{O}^* K^{*2}$, where $\mathcal{O}^*$ denotes the group of units of $\mathcal{O}$. Note that the chain of subgroups

$$V \supset W \supset Y \supset K^{*2}$$

corresponds exactly to the first three obstructions.

The index of W in V is bounded by (6.8). Next we consider W/Y. If $\gamma \in W$, then $\gamma\mathcal{O} = I^2$ for some fractional $\mathcal{O}$-ideal I, and the map that sends γ to the ideal class of I in the ideal class group of $\mathcal{O}$ clearly has Y as its kernel. We conclude that if h is the order of the class group of K, then

$$[W : Y] \leq h.$$

Finally, Y/K^{*2} is isomorphic to $\mathcal{O}^*/\mathcal{O}^{*2}$, of which the $\mathbf{F}_2$-dimension is equal to the rank of the unit group $\mathcal{O}^*$ plus one (accounting for the roots of unity). Thus from Dirichlet's unit theorem we have

$$[Y : K^{*2}] = 2^{d-s},$$

where s is one-half the number of non-real embeddings of K in the field of complex numbers. Combining the estimates, we find that

$$[V : K^{*2}] \le [\mathcal{O} : \mathbf{Z}[\alpha]] \cdot h \cdot 2^{d-s}.$$

Let Δ_K denote the discriminant of K. From [26, Theorem 6.5, Remark] we have that

$$h \le M \cdot \frac{(d-1+\log M)^{d-1}}{(d-1)!},$$

where $M = (d!/d^d)(4/\pi)^s\sqrt{|\Delta_K|}$ is the Minkowski constant of K. Let Δ denote, as in 3.3, the discriminant of f. Then we have

$$M \le \sqrt{|\Delta_K|} \le \sqrt{|\Delta_K|} \cdot [\mathcal{O} : \mathbf{Z}[\alpha]] = \sqrt{|\Delta|} < d^d n^{1-3/(2d)}.$$

The equality follows from [8, Chapter I, Section 3, Proposition 4(i) and Section 4, Proposition 6(ii)], and the last inequality is Lemma 3.3. From $d \ge 2$ and $n > d^{2d^2}$ one deduces that

$$d - 1 + d\log d < \frac{3}{2d}\log n, \qquad 2d \cdot (2\log n)^{d-1} < n^{3/(2d)}.$$

Combining all this, we obtain

$$\begin{aligned}
[V : K^{*2}] &\le [\mathcal{O} : \mathbf{Z}[\alpha]] \cdot \frac{d!}{d^d}\left(\frac{4}{\pi}\right)^s \sqrt{|\Delta_K|} \cdot \frac{(d-1+\log\sqrt{|\Delta|})^{d-1}}{(d-1)!} \cdot 2^{d-s} \\
&= \frac{\sqrt{|\Delta|}}{d^{d-1}} \cdot 2^d \cdot (d-1+\log\sqrt{|\Delta|})^{d-1} \cdot \left(\frac{2}{\pi}\right)^s \\
&< n^{1-3/(2d)} \cdot d \cdot 2^d \cdot \left(d - 1 + d\log d + \left(1 - \frac{3}{2d}\right)\log n\right)^{d-1} \\
&< n^{1-3/(2d)} \cdot 2d \cdot (2\log n)^{d-1} < n,
\end{aligned}$$

as required. This proves Theorem 6.7.

7. Algebraic interlude

This section is devoted to the proof of 5.4 and (6.8); it can be skipped by the reader who is willing to take those assertions for granted. Our fundamental tool is the Jordan-Hölder theorem. One can also prove these results using some of the machinery of commutative algebra; for instance, some of the facts proved here can be extracted, with some work, from Appendices A1–3 in [12].

We denote by K an algebraic number field, i. e., a finite field extension of the field $\mathbf{Q}$ of rational numbers, and by K^* its multiplicative group. We let A be an *order* in K, i. e., a subring (with 1) of the ring of integers $\mathcal{O}$ of K with the property that the index of the additive group of A in that of $\mathcal{O}$ is finite. The case of interest in 5.4 is $A = \mathbf{Z}[\alpha]$. In $\mathcal{O}$ one has unique factorization of ideals into prime ideals; in the present section we develop a substitute for A that meets the needs of the number field sieve.

Let $N: K \to \mathbf{Q}$ be the norm map. For each $x \in K$, the norm $N(x)$ of x equals the determinant of the $\mathbf{Q}$-linear map $K \to K$ that sends each $y \in K$ to xy. It follows that for each non-zero element $x \in A$ we have $\#A/xA = |N(x)|$. This implies that A/I is *finite* for each non-zero ideal I of A. The cardinality of A/I is called the *norm* of I, denoted $\mathrm{N}I$. In particular, if P is a non-zero prime ideal of A, then A/P is a finite integral domain, and therefore a field. Hence every such P is a maximal ideal of A and contains a unique prime number p; the *degree* of P is the degree of A/P as a field extension of its prime field $\mathbf{F}_p$. In the sequel, by a "prime of A" we will mean a non-zero prime ideal of A.

The following result clearly contains 5.4 as the special case $A = \mathbf{Z}[\alpha]$.

Proposition 7.1. *There is, for each prime P of A, a group homomorphism $l_P: K^* \to \mathbf{Z}$, such that the following hold:*

(a) $l_P(x) \geq 0$ *for all* $x \in A$, $x \neq 0$;

(b) *if x is a non-zero element of A, then $l_P(x) > 0$ if and only if $x \in P$;*

(c) *for each $x \in K^*$ one has $l_P(x) = 0$ for all but finitely many P, and*

$$\prod_P (\mathrm{N}P)^{l_P(x)} = |N(x)|,$$

where P ranges over the set of all primes of A.

Proof. First we construct the functions l_P. Let P be a prime of A and let $x \in A$, $x \neq 0$. Since xA is of finite index in A, there is a finite chain

$$A = I_0 \supset I_1 \supset I_2 \supset \ldots \supset I_{t-1} \supset I_t = xA$$

of distinct ideals of A that cannot be refined, in the sense that there is no ideal properly between I_{i-1} and I_i, for $1 \leq i \leq t$. We now define $l_P(x)$ to be the number of $i \in \{1, 2, \ldots, t\}$ for which $I_{i-1}/I_i \cong A/P$ as A-modules. It follows from the Jordan-Hölder theorem (see [36, Section 51]) that $l_P(x)$ is well-defined in the sense that it does not depend on the choice of the chain of ideals I_i. (In terms of commutative algebra, $l_P(x)$ can be defined as the length of the module A_P/xA_P over the local ring A_P.)

If x, y are non-zero elements of A, then a chain $I_0, I_1, \ldots, I_t$ as above can be combined with a similar chain $J_0, J_1, \ldots, J_u$ for y into a chain $I_0, I_1, \ldots, I_t = xJ_0, xJ_1, \ldots, xJ_u$ for xy. This proves that we have $l_P(xy) = l_P(x) + l_P(y)$. Therefore we can extend the map l_P to a well-defined group homomorphism $K^* \to \mathbf{Z}$ by putting $l_P(x/z) = l_P(x) - l_P(z)$ for any two non-zero elements x, $z \in A$. This completes the construction of the homomorphisms l_P. It is clear that (a) holds.

To prove the "if" part of (b), it suffices to observe that one can take $I_1 = P$ if $x \in P$. For the "only if" part, suppose that $x \notin P$. Since P is maximal, the ideal $xA + P$ equals A, so $xy + z = 1$ for certain $y \in A$, $z \in P$. Then $z \equiv 1 \bmod xA$, so multiplication by z induces the identity map $A/xA \to A/xA$. Hence $z \cdot (I_{i-1}/I_i) = I_{i-1}/I_i$, which by $z \in P$ implies that I_{i-1}/I_i cannot be isomorphic to A/P.

It suffices to prove (c) in the case that $x \in A$. Let the I_i be as above, so that

$$|N(x)| = \#A/xA = \prod_{i=1}^{t} \#I_{i-1}/I_i.$$

Thus to prove (c) it suffices to show that for each i there is a unique prime P of A with $I_{i-1}/I_i \cong A/P$. Let $y \in I_{i-1}$, $y \notin I_i$. Since there is no ideal properly in between I_i and I_{i-1}, we have $yA + I_i = I_{i-1}$, so multiplication by y induces a surjective map $A \to I_{i-1}/I_i$. Therefore $A/P \cong I_{i-1}/I_i$ for some ideal P, and since this module has no non-trivial submodules the ideal P must be maximal. Also, P is the annihilator of the A-module I_{i-1}/I_i, so it is uniquely determined. This proves 7.1.

Remark. We remark that the functions l_P are uniquely determined by the properties listed in 7.1. To prove this, let l'_P, for each prime P of A, be a homomorphism $K^* \to \mathbf{Z}$, such that (a), (b), (c) hold with l'_P instead of l_P. Let P be a prime of A, and p the prime number with $p \in P$. Let $x \in A$, $x \neq 0$. To prove that $l'_P(x)$ is uniquely determined we proceed as follows. From the definition of l_P we see that $P^m J \subset pxA$, where

$$m = l_P(px), \qquad J = \prod_{Q \neq P} Q^{l_Q(px)}.$$

From $P^m + J = A$ and the Chinese remainder theorem it follows that there exist y, $z \in A$ with $y \equiv x \bmod P^m$, $y \equiv 1 \bmod J$, $z \equiv 1 \bmod P^m$, $z \equiv x \bmod J$. Then $yz \equiv x \bmod pxA$, so $yz = wx$ with $w \equiv 1 \bmod pA$. From z, $w \notin P$ one obtains $l'_P(x) = l'_P(y)$. We have $y \notin P'$ for any $P' \neq P$ that is of p-power norm, since each such P' divides J. Hence $l'_P(y)$ can be read off from (c). This proves the uniqueness.

From the uniqueness it follows that in the case $A = \mathcal{O}$ the functions l_P coincide with the normalized exponential valuations corresponding to the primes of $\mathcal{O}$; in other words, $l_P(x)$ is the exponent of the exact power of P dividing the ideal $x\mathcal{O}$. One can also see this by writing the ideal $x\mathcal{O}$ as a product of prime ideals, $x\mathcal{O} = P_1 P_2 \cdots P_t$, and choosing $I_\imath = P_1 P_2 \cdots P_\imath$.

We now turn to the proof of (6.8). In the rest of this section A and B denote orders in K with $A \subset B$; for (6.8), we shall take $A = \mathbf{Z}[\alpha]$, $B = \mathcal{O}$. If Q is a prime of B, then $P = Q \cap A$ is a prime of A. In this case we say that Q lies over P, notation: $Q|P$. If Q lies over P, then the finite field B/Q is a field extension of A/P, and we denote the degree of this field extension by $f(Q/P)$. In order to avoid confusion we shall write $l_{P,A}$ for what we denoted by l_P above.

Proposition 7.2. *Let P be a prime of A. Then we have*

$$l_{P,A}(x) = \sum_{Q|P} f(Q/P) l_{Q,B}(x)$$

for each $x \in K^*$, the sum ranging over the primes Q of B that lie over P.

Proof. It is convenient, in this proof, to introduce the following notation. If M is a finite A-module, then we let $l_{P,A}(M)$ be the number of composition factors of M that are isomorphic to A/P. With this notation, we have $l_{P,A}(x) = l_{P,A}(A/xA)$ for every non-zero element $x \in A$. Note that $l_{P,A}(M) = l_{P,A}(L) + l_{P,A}(M/L)$ whenever L is a submodule of M.

It clearly suffices to prove the formula in 7.2 for $x \in A$. Multiplication by x shows that the A-modules B/A and xB/xA are isomorphic, so $l_{P,A}(B/A) = l_{P,A}(xB/xA)$. Therefore we have

$$\begin{aligned} l_{P,A}(x) = l_{P,A}(A/xA) &= l_{P,A}(B/xA) - l_{P,A}(B/A) \\ &= l_{P,A}(B/xA) - l_{P,A}(xB/xA) = l_{P,A}(B/xB). \end{aligned}$$

Hence the formula in 7.2 is equivalent to the statement that for $M = B/xB$ we have

$$l_{P,A}(M) = \sum_{Q|P} f(Q/P) l_{Q,B}(M).$$

We prove this formula for *any* finite B-module M. Choosing a composition series for M we immediately reduce to the case that M is a *simple* B-module, which means that M has exactly two B-submodules ($\{0\}$ and itself). In that case $M \cong B/Q'$ for some prime Q' of B, and $l_{Q,B}(M)$ equals 1 or 0 according as $Q = Q'$ or $Q \neq Q'$. Let $P' = Q' \cap A$. As an A-module, $M = B/Q'$ is a direct sum of $f(Q'/P')$ copies of A/P', so that $l_{P,A}(M)$ equals $f(Q'/P')$ or 0 according as $P = P'$ or $P \neq P'$. Thus the above formula follows by inspection. This proves 7.2.

Note that it follows from 7.2 that for each P the set of primes Q of B lying over P is finite and non-empty. We now prove that for all but finitely many P it is true that there is exactly *one* Q lying over P, and that it satisfies $f(Q/P) = 1$.

Proposition 7.3. *For all but finitely many primes P of A we have $\sum_{Q|P} f(Q/P) = 1$. In addition, the integer*

$$\prod_{P} (\mathrm{N}P)^{-1+\sum_{Q|P} f(Q/P)},$$

with P ranging over all primes of A, divides the index $[B : A]$ of A in B.

Proof. Let T be any finite set of primes of A, and let U be the set of primes of B lying over the primes in T. Let the A-ideal I be the intersection of the primes $P \in T$, and let the B-ideal J be the intersection of the primes $Q \in U$. Then $I = J \cap A$, so A/I is a subring of B/J, and the index of A in B is divisible by the index of A/I in B/J. By the Chinese remainder theorem, we have $A/I \cong \prod_{P\in T} A/P$, and therefore

$$\#A/I = \prod_{P\in T} \mathrm{N}P.$$

Likewise we have

$$\#B/J = \prod_{Q\in U} \mathrm{N}Q = \prod_{P\in T} (\mathrm{N}P)^{\sum_{Q|P} f(Q/P)}.$$

It follows that $[B : A]$ is divisible by

$$(\#B/J)/(\#A/I) = \prod_{P\in T} (\mathrm{N}P)^{-1+\sum_{Q|P} f(Q/P)}.$$

Therefore the number of $P \in T$ for which $\sum_{Q|P} f(Q/P) \neq 1$ is bounded independently of T, which implies the first assertion of 7.3. Taking for T the set of all P with $\sum_{Q|P} f(Q/P) \neq 1$ we obtain the second. This proves 7.3.

In our final result in this section, we write

$$V_A = \{x \in K^* : l_{P,A}(x) \equiv 0 \bmod 2 \text{ for all primes } P \text{ of } A\}.$$

In the notation of (6.8) we clearly have $V_{\mathbf{Z}[\alpha]} = V$ and $V_{\mathcal{O}} = W$. Hence (6.8) is an immediate consequence of the following proposition.

Proposition 7.4. *If $A \subset B$ are orders of K, then $V_B \subset V_A$, and $[V_A : V_B] \leq [B : A]$.*

Proof. The inclusion $V_B \subset V_A$ is clear from 7.2. To bound $[V_A : V_B]$, we choose for each prime P of A a set S_P of primes Q of B lying over P, as follows. If $f(Q/P)$ is even for each prime Q of B lying over P, then we let S_P be the set of all Q lying over P. If there is at least one Q lying over P for which $f(Q/P)$ is odd, then we choose one such prime, Q_0 (say), and we let S_P consist of all primes $Q \neq Q_0$ that lie over P. Since $f(Q/P) \geq 2$ if $f(Q/P)$ is even, we have

$$\#S_P \leq -1 + \sum_{Q|P} f(Q/P)$$

for all P. In particular, S_P is empty for almost all P. Let S be the union of the sets S_P, with P ranging over the primes of A. We have

$$2^{\#S} \leq \prod_P (\mathrm{N}P)^{\#S_P} \leq \prod_P (\mathrm{N}P)^{-1+\sum_{Q|P} f(Q/P)} \leq [B : A],$$

by 7.3. Thus to prove 7.4, it suffices to show that the group V_A/V_B embeds in the group $(\mathbf{Z}/2\mathbf{Z})^S$. To do this, map $x \in V_A$ to the element $(l_{Q,B}(x) \bmod 2)_{Q\in S}$ of $(\mathbf{Z}/2\mathbf{Z})^S$. If x is in the kernel of this map, then $l_{Q,B}(x)$ is even for all $Q \in S$. Since also all $l_{P,A}(x)$ are even, it follows from 7.2 and the choice of S_P that $l_{Q,B}(x)$ is even for *all* Q, so that $x \in V_B$. This proves 7.4.

8. Quadratic characters

In this section the notation and hypotheses are as in Sections 4 and 5. We assume in addition that $n > d^{2d^2}$, and that m, f have been produced by the base m method of Section 3.

In our original version of the number field sieve we handled the three obstructions (6.2), (6.3), (6.4) as follows. We dealt with the first obstruction, which is due to the difference between the rings $\mathbf{Z}[\alpha]$ and $\mathcal{O}$, by using the algorithm of [6], as mentioned in Section 6. To overcome the second obstruction, we proposed that the linear algebra on the algebraic side be done over $\mathbf{Z}$ rather than over $\mathbf{F}_2$ (cf. [23, Extended abstract, Section 7]). This allowed the construction of integers $s(a,b)$ for pairs $(a,b) \in T_2$ such that

$$(8.1) \qquad \prod_{(a,b)\in T_2} (a+b\alpha)^{s(a,b)}\mathcal{O} = (1).$$

Thus $\prod(a+b\alpha)^{s(a,b)}$ is a unit. The third obstruction was overcome by means of lattice basis reduction methods on the logarithmic embedding in Euclidean space of the units arising (see [14]). Thus several equations of the form (8.1) could be combined to find integers $s'(a,b)$ such that

$$\prod_{(a,b)\in T_2} (a+b\alpha)^{s'(a,b)} = 1.$$

By then combining these ideas with the sieve on the rational side as discussed in Section 4, we could find integers $s''(a,b)$ for each pair $(a,b) \in T_1 \cap T_2$ such that we have

$$\prod_{(a,b)\in T_1\cap T_2} (a+bm)^{s''(a,b)} \quad \text{is a square in } \mathbf{Z},$$

$$\prod_{(a,b)\in T_1\cap T_2} (a+b\alpha)^{s''(a,b)} = 1.$$

These equations could then be used in place of (2.1) and (2.2) to attempt to factor n.

In addition to being inelegant and complicated, the linear algebra step over $\mathbf{Z}$ in the above scenario became a bottleneck in the complexity argument. In fact the heuristic run time of the above version of the number field sieve is $L_n[\frac{1}{3}, 9^{1/3} + o(1)]$ for $n \to \infty$ rather than the bound we advertised above; the latter could be achieved only at the expense of considerable additional complications.

It was at this point that Adleman [1] suggested using quadratic characters to overcome the second and third obstructions. As we shall see this allows the linear algebra on the algebraic side to be done over $\mathbf{F}_2$, greatly simplifying the algorithm. In fact we use this same idea to also overcome the first obstruction.

In order to explain the idea behind "character columns", we start by considering a simpler situation. Suppose that X is a finite set of primes and that $l \in \mathbf{Z}$, $l \neq 0$, has the property that in the factorization of l into primes, the exponent of each prime not in X is even. Is l a square? The answer of course depends on the sign of l and the exponent of each prime $p \in X$ in the factorization of l. If these quantities are inaccessible for some reason then we can still *test* l for squareness by the following probabilistic device: if p is a prime number that is not in X and p does not divide $2l$, then test the Legendre symbol $(\frac{l}{p})$ to see if it is equal to 1. If the symbol is ever equal to -1 then l is not a square; if the symbol is always equal to 1 for a number of primes p significantly exceeding $\#X$ then we become convinced that l is a square. Specifically, if V_X denotes the multiplicative group of non-zero rational numbers that are squares outside X as above, then $V_X/\mathbf{Q}^{*2}$ is an $\mathbf{F}_2$-vector space of dimension $\#X + 1$. The Legendre symbol corresponding to each "test" prime p is a presumably random linear function on this vector space. Our test for l being a square is ironclad if the characters corresponding to the primes p that we choose span the dual space of $V_X/\mathbf{Q}^{*2}$.

Lemma 8.2. *Let k, r be non-negative integers, and let E be a k-dimensional $\mathbf{F}_2$-vector space. Then the probability that $k + r$ elements that are independently drawn from E, with the uniform distribution, form a spanning set for E is at least $1 - 2^{-r}$.*

Proof. For any hyperplane H of E, the probability that each of the $k + r$ vectors lies in

H is 2^{-k-r}. Since each hyperplane is the kernel of a uniquely determined non-zero linear function $E \to \mathbf{F}_2$, the number of hyperplanes of E is $2^k - 1$. Thus the probability that the $k + r$ vectors all lie in *some* hyperplane is at most

$$(2^k - 1)2^{-k-r} < 2^{-r}.$$

However, the $k + r$ vectors do not span E if and only if they lie in some hyperplane. Thus the lemma follows.

Remark. If one picks random elements of E, independently, and from a uniform distribution, until one has a set of generators, then the expectation of the number of elements drawn is equal to $k + \sum_{i=1}^{k}(2^i - 1)^{-1}$. For $k \to \infty$, the sum tends to a limit c where $c \doteq 1.606695$. Thus for any k, the expectation is less than $k + 2$.

If we had some method of choosing Legendre characters that in the above scenario corresponds to choosing elements of the dual space of $V_X/\mathbf{Q}^{*2}$ independently and from a uniform distribution, then we could develop a virtually certain test for squareness for the integer l. In what follows, we replace $\mathbf{Z}$ with $\mathbf{Z}[\alpha]$ and make the heuristic assumption that choosing Legendre characters corresponding to small primes outside the factor base suffices for a squareness test.

The following result shows how Legendre symbols provide us with a necessary condition for a product of elements $a + b\alpha$ to be a square. The set $R(q)$ is as defined after (5.1).

Proposition 8.3. *Let S be a finite set of coprime integer pairs (a, b) with the property that $\prod_{(a,b)\in S}(a + b\alpha)$ is the square of an element of K. Further let q be an odd prime number and $s \in R(q)$, such that*

$$a + bs \not\equiv 0 \bmod q \qquad \textit{for each } (a.b) \in S,$$

$$f'(s) \not\equiv 0 \bmod q.$$

Then we have

$$\prod_{(a,b)\in S} \left(\frac{a + bs}{q}\right) = 1.$$

Proof. Let $\mathbf{Z}[\alpha] \to \mathbf{F}_q$ be the ring homomorphism mapping α to $s \bmod q$, and let Q be its kernel; this is the first degree prime corresponding to q, s. Define the map $\chi_Q\colon \mathbf{Z}[\alpha] - Q \to \{\pm 1\}$ to be the composition of $\mathbf{Z}[\alpha] - Q \to \mathbf{F}_q - \{0\}$ with the Legendre symbol $\mathbf{F}_q - \{0\} \to \{\pm 1\}$. Clearly, we have $\chi_Q(a + b\alpha) = \left(\frac{a+bs}{q}\right)$.

As we saw in (6.6), we have

$$f'(\alpha)^2 \cdot \prod_{(a,b)\in S} (a + b\alpha) = \delta^2$$

for some $\delta \in \mathbf{Z}[\alpha]$. By hypothesis, the factors on the left are not in Q, so we have $\delta \notin Q$. The proposition follows if we apply χ_Q to the equation.

As with 5.3, it is really the converse to 8.3 that we are interested in, and in this case it does hold: if an element $\beta \in \mathbf{Z}[\alpha] - \{0\}$ satisfies $\chi_Q(\beta) = 1$ for all first degree primes Q with $2\beta \notin Q$, or even for all such Q with finitely many exceptions, then β is a square in K.

In the actual algorithm, we use both the functions $e_{p,r}$ and the Legendre symbols to produce the square that we need, as follows. Let $T = T_1 \cap T_2$, so that

$$T = \{(a, b) : \gcd(a, b) = 1,\ |a| \le u,\ 0 < b \le u,\ (a + bm)N(a + b\alpha) \text{ is } y\text{-smooth}\}.$$

Define

$$B = \pi(y),$$
$$B' = \#\{(p, r) : p \text{ is a prime number},\ p \le y,\ r \in R(p)\},$$
$$B'' = [3(\log n)/\log 2].$$

We define the factor base on the rational side to be the set of all prime numbers up to y, call them p_1, p_2, $\ldots$, p_B. Define the factor base on the algebraic side to be the set of pairs (p_1, r_1), (p_2, r_2), $\ldots$, $(p_{B'}, r_{B'})$ as in the definition of B'. Let (q_1, s_1), (q_2, s_2), $\ldots$, $(q_{B''}, s_{B''})$ be the first B'' pairs consisting of a prime number $q > y$ and an integer $s \in R(q)$ with $f'(s) \not\equiv 0 \bmod q$, ordered by increasing q.

We now define a map e from T to $\mathbf{F}_2^{1+B+B'+B''}$. Say $(a, b) \in T$. The first coordinate of $e(a, b)$ is determined by the sign of $a + bm$; it is 0 if $a + bm > 0$ and 1 if $a + bm < 0$ (we cannot have $a + bm = 0$ if $m > u$, which will be the case with our choice of parameters; see Section 11). The next B coordinates are given by $\operatorname{ord}_p(a + bm) \bmod 2$ as p runs over

p_1, p_2, …, p_B. The next B' coordinates are given by $e_{p,r}(a+b\alpha) \bmod 2$ as (p,r) runs over (p_1,r_1), (p_2,r_2), …, $(p_{B'},r_{B'})$. The last B'' coordinates of $e(a,b)$ are determined by $\left(\frac{a+bs}{q}\right)$ as (q,s) runs over (q_1,s_1), (q_2,s_2), …, $(q_{B''},s_{B''})$. For a particular (q,s) it is 0 if $\left(\frac{a+bs}{q}\right)=1$ and 1 if $\left(\frac{a+bs}{q}\right)=-1$. Note that the reason for the special treatment of the first coordinate and the last B'' coordinates is to turn a multiplicative structure into an additive structure.

If $\#T > 1+B+B'+B''$ then the vectors $e(a,b)$ for $(a,b) \in T$ are linearly dependent. Thus there is a non-empty subset S of T such that $\sum_{(a,b)\in S} e(a,b)$ is the zero vector in $\mathbf{F}_2^{1+B+B'+B''}$. It is clear that such a set satisfies (4.1), and we conjecture that it satisfies (6.6) as well.

To support this conjecture, we make the following remarks. Let V be the subgroup of K^* defined before Theorem 6.7. If Q is any first degree prime of $\mathbf{Z}[\alpha]$ with $f'(\alpha) \notin Q$, then the function χ_Q defined in the proof of 8.3 induces a group homomorphism $V/K^{*2} \to \{\pm 1\}$, again to be denoted by χ_Q; namely, one can show that any $\beta \in V$ can be written as $\beta = \beta_1\beta_2^2$, with $\beta_1 \in \mathbf{Z}[\alpha] - Q$ and $\beta_2 \in K^*$, and that $\chi_Q(\beta_1)$ is independent of this representation, so that we can put $\chi_Q(\beta) = \chi_Q(\beta_1)$. The Čebotarev density theorem (see [18, Chapter VIII, Section 4]) implies that if Q ranges over all first degree primes of $\mathbf{Z}[\alpha]$ with $f'(\alpha) \notin Q$, ordered by increasing norm, then the elements χ_Q are asymptotically equally distributed over $\mathrm{Hom}(V/K^{*2}, \{\pm 1\})$. This suggests that the B'' functions χ_Q that the algorithm employs may be viewed as random homomorphisms $V/K^{*2} \to \{\pm 1\}$, so that Theorem 6.7 and Lemma 8.2 make it overwhelmingly likely that these functions χ_Q span $\mathrm{Hom}(V/K^{*2}, \{\pm 1\})$. If they do, then for an element $\beta \in V$ to be a square it would be necessary and sufficient that $\chi_Q(\beta) = 1$ for each of the B'' primes Q, which would imply the conjecture. A rigorous proof of the conjecture along these lines would require a very strong effective version of the Čebotarev density theorem, which presently appears to be completely out of reach. It may be possible to deduce a weak form of the conjecture—with B'' replaced by a larger value—from the generalized Riemann hypothesis (cf. [2]). In addition, it may be possible to rigorously prove a *random* version of the above, where the B'' primes Q are independently and uniformly chosen from all the first degree primes of $\mathbf{Z}[\alpha]$ in some reasonable range.

Remark. One can also make use of Legendre symbols that are defined for primes Q of odd norm that have degree greater than 1. However, there is a certain danger involved in using these primes. For example, if $d = 2$, then the base m method of Section 3 leads to an imaginary quadratic field, and one can show that in that case $\chi_Q(u) = 1$ for every unit u of $\mathcal{O}$ and every prime Q of odd norm of degree greater than 1; this means that the quadratic characters associated to such primes are not sufficient to deal with obstruction (6.4). First degree primes do not suffer from this shortcoming.

9. Finding square roots

We retain the notation and hypotheses from the last section.

Now that we have produced presumed squares in $\mathbf{Z}$ and $\mathbf{Z}[\alpha]$ we need to find their square roots. In $\mathbf{Z}$ this is easy. If $f'(m)^2 \prod_{(a,b)\in S}(a+bm)$ is a square, then since the prime factorization of each $a+bm$ is known it is an easy matter to compute the square root. We are ultimately only interested in the result $\bmod n$, so all of the arithmetic can be done with integers of the size of n.

Next we address the problem of finding the square root in the number field. This is a component of the number field sieve that has no analogue in earlier factoring algorithms, including the special number field sieve. In the known solutions to this problem one cannot work "$\bmod n$", as we did in $\mathbf{Z}$, which means that one has to deal with numbers of a truly gigantic size. More precisely, the number of digits of the numbers that we work with are about $\sqrt{C}$, where C is the running time of the entire number field sieve (see 9.3 and Section 11). (In all other components of the number field sieve we work only with numbers of $C^{o(1)}$ digits, for $n \to \infty$.) Thus we have to be very careful when performing arithmetic operations on these numbers, and methods depending on the fast Fourier transform become important. In this section we discuss the problem from a theoretical point of view. Practical experiments that are being conducted by D. J. Bernstein indicate that the method that we shall suggest actually works in practice.

Let $\gamma = f'(\alpha)^2 \prod_{(a,b)\in S}(a+b\alpha)$ be the presumed square in $\mathbf{Z}[\alpha]$. To find its square root, we can first multiply out the product and represent γ as a polynomial in α of degree less than d, and next apply one of the algorithms that have been proposed for factoring polynomials over algebraic number fields (see [37; 38; 17; 20]) to the polynomial $X^2 - \gamma \in K[X]$. It is important to bear in mind that, when all parameters of the number field sieve are chosen optimally, the cardinality of the set S and the coefficients of γ as a polynomial in α are very large (see 9.3 and Section 11). This implies that just *computing* γ is already very time consuming, and factoring $X^2 - \gamma$ even more so. In order to be able to analyze the complexity of this step we consider what the algorithms of [37; 38; 17; 20] come down to in our case.

There is no essential difference between the algorithms proposed in [37; 38; 17; 20] if an odd prime number q is available for which $f \bmod q$ is irreducible in $\mathbf{F}_q[X]$; so let this now first be assumed. Then $\mathbf{Z}[\alpha]/q\mathbf{Z}[\alpha]$ is isomorphic to $\mathbf{F}_q[X]/(f \bmod q)$, which is a field of cardinality q^d. Hence the ideal $Q = q\mathbf{Z}[\alpha]$, which consists of all elements $\sum_{i=0}^{d-1} a_i\alpha^i$ for which each of the integer coefficients a_i is divisible by q, is a prime of $\mathbf{Z}[\alpha]$ of degree d. From the irreducibility of $f \bmod q$ it follows that $f'(\alpha) \notin Q$, and for each $(a, b) \in S$ we have $a + b\alpha \notin Q$ since $\gcd(a, b) = 1$. Therefore the product γ of all these elements does not belong to Q either. Taking the coefficients of γ modulo q, and applying an algorithm for taking square roots in the finite field $\mathbf{Z}[\alpha]/Q$ (see [19; 16, Section 4.6.2, Exercise 15]), we find an element $\delta_0 \pmod{Q}$ such that $\delta_0^2\gamma \equiv 1 \bmod Q$; this $\delta_0 \bmod Q$ is unique up to sign. (If one finds, unexpectedly, that $X^2 - \gamma$ is actually irreducible modulo Q, so that δ_0 cannot be found, then γ is not a square in $\mathbf{Z}[\alpha]$, and we have hit upon a counterexample to the conjecture stated in Section 8. In this case more character columns might be tried.) Note that δ_0 is the *inverse* of a square root of $\gamma \bmod Q$; this is in order to avoid divisions in the iteration to follow. Starting from δ_0, we apply a Newton iteration

$$\delta_j \equiv \frac{\delta_{j-1}(3 - \delta_{j-1}^2\gamma)}{2} \bmod Q^{2^j}$$

to find $\delta_1, \delta_2, \ldots$, such that $\delta_j^2\gamma \equiv 1 \bmod Q^{2^j}$. Notice that working modulo Q^{2^j} means that the coefficients a_i in the expressions $\sum_i a_i\alpha^i$ are taken modulo q^{2^j}, so that one may take $|a_i| < q^{2^j}/2$. One continues the Newton iteration until q^{2^j} is at least twice as large as an upper bound that one is able to prove for the absolute values of the coefficients of a true square root β of γ in $\mathbf{Z}[\alpha]$. Then β can be calculated from $\beta \equiv \delta_j\gamma \bmod Q^{2^j}$. If we wish, we can now verify that $\beta^2 = \gamma$, and thus free ourselves from having to rely on the unproved conjecture of Section 8; but in the context of the number field sieve it is more efficient to just assume that $\beta^2 = \gamma$, and to proceed immediately to the calculation of $\varphi(\beta)$ (as in Section 2) in an attempt to factor n.

There are several refinements and modifications that might affect the practical performance of this scheme. For example, one can apply fast multiplication techniques in the iteration; one can go up by powers Q^j instead of Q^{2^j} of Q; and one can stop the iteration as soon as the coefficients of $\delta_j\gamma \bmod Q^{2^j}$ do not change for a few successive values of j. One may also wonder whether there is a method that does not start by multiplying out the product that defines γ.

In the above description we made the assumption that an odd prime number q is available for which $f \bmod q$ is irreducible. One can attempt to find such a prime number q by trying $q = 3, 5, 7, \ldots$ in succession. (Of course, the prime numbers that are norms of first degree primes of $\mathbf{Z}[\alpha]$ can be left out.) For each q, one can test $f \bmod q$ for irreducibility by applying an irreducibility test in $\mathbf{F}_q[X]$ (see [19]). As we shall see below, one may for most n expect to be successful fairly soon. However, there are cases in which not a single prime number q exists for which $f \bmod q$ is irreducible. This occurs, for example, when $d = 4$ and $n = m^4 + 1$. The question arises how to proceed when this happens.

One solution of this problem is based on the remark that, in a sense that can be made precise, most monic polynomials f of degree d in $\mathbf{Z}[X]$ have the property that the Galois group of f is the full symmetric group S_d of order $d!$ (see [13]). If f satisfies this condition, then the Čebotarev density theorem implies that the density, inside the set of all prime numbers, of the set of prime numbers q for which $f \bmod q$ is irreducible is equal to the probability that a random permutation of $\{1, 2, \ldots, d\}$ is a single d-cycle (cf. the proof of 9.1 below), which is equal to $1/d$. Since d will be chosen quite small with respect to n (see Section 11), this is fairly large, so that for most values of n we expect that there are many suitable prime numbers q and that it will be easy to find one. It may be possible to make this loose argument perfectly rigorous. If, for whatever reason, a good q is difficult to find, then one has the option of changing f (and hence the number field), for example by adding a polynomial that is divisible by $X - m$ to f, or by choosing a different value of m in the base m algorithm. However, there are situations in which it is very undesirable to change f, for example when f has particularly small coefficients. In that case one may not be able to work with primes q for which f mod q is irreducible.

We briefly discuss what one can do if no odd prime number q is available for which f mod q is irreducible. The approach of [38] is then to do a similar Newton iteration modulo powers of an odd prime number q. At the start of the iteration, the ideal $q\mathbf{Z}[\alpha]$ is not prime, so that the inverse square root δ_0 of $\gamma \pmod{q}$ is not unique up to sign. Instead, one must take the inverse square root of γ modulo each of the primes Q containing q, and combine them into an inverse square root modulo $q\mathbf{Z}[\alpha]$; or if q is small, one can try all $(q^d - 1)/2$ non-zero elements of $\mathbf{Z}[\alpha]/q\mathbf{Z}[\alpha]$, up to sign. If there are t primes Q containing q, then this gives rise to 2^{t-1} different starting values δ_0 for the Newton iteration. If we choose q as indicated below, then we have $t \leq d/2$, and it turns out that, with our choice of parameters, a factor $2^{[d/2]-1}$ does not greatly affect the running time; so the algorithm of [38] may be feasible for our purposes.

The polynomial time algorithm of [17; 20] does a Newton iteration modulo the powers of a *single* prime Q containing q. To recover the square root of γ from $\delta_j\gamma$, for large j, one then needs to apply a basis reduction algorithm to the ideal Q^{2^j}. This is, with our choice of parameters, not attractive (see 9.3). Another possibility is the algorithm of [37], but we have not investigated its merits for use in the number field sieve. A final possibility is to make use of the "infinite" prime, as was pointed out to us by V. S. Miller and R. D. Silverman. In this case, one chooses an element of $K = \mathbf{Q}(\alpha)$ that under each embedding σ of K in the field of complex numbers is close to a square root of $\sigma(\gamma)$. and one next applies a Newton iteration in $\mathbf{Q}(\alpha)$, where one works with the coefficients a_i as real numbers that are rounded to rationals. For this algorithm, the number of different starting values to be tried is 2^{d-s-1}, where s is one-half the number of non-real embeddings of K into the field of complex numbers. For each of these methods, the applicability of the refinements mentioned above is to be considered. Which method is the best one for practical purposes remains to be tested.

If one decides to use the algorithm of [38], then the choice of an appropriate prime number q is still important, since the method requires that the algebraic integer γ be coprime to q. This is guaranteed if f mod q factors into distinct irreducible non-linear factors. Indeed, if f mod q is squarefree, then q is relatively prime to $f'(\alpha)$, and if f mod q has no linear factors then there is no first degree prime of norm q, so that by 5.5 each $a+b\alpha$ is coprime to q. One may wonder whether primes q with the properties just mentioned exist. The following result answers this question affirmatively, and in addition it asserts that there are so many of them that in practice it should not be hard to find one.

Proposition 9.1. *Let $f \in \mathbf{Z}[X]$ be an irreducible monic polynomial of degree d, with $d > 1$. Then the density, inside the set of all prime numbers, of the set of prime numbers q for which f mod q factors in $\mathbf{F}_q[X]$ into distinct irreducible non-linear factors exists and is at least $1/d$.*

Proof. Let G be the Galois group of f over $\mathbf{Q}$, viewed as a permutation group of the set Ω of zeroes of f. For each prime number q that does not divide the discriminant of f, there is a Frobenius element $\sigma_q \in G$, which is well-defined up to conjugacy in G, and which has the property that the degrees of the irreducible factors of f mod q are the same as the lengths of the cycles of the permutation σ_q. Hence, we are interested in those q for which σ_q acts without fixed points on Ω. The Čebotarev density theorem [18, Chapter VIII, Section 4] implies that for every subset $C \subset G$ that is a closed under conjugation by G, the set of prime numbers q for which σ_q belongs to C has a density, and that this density equals $\#C/\#G$. Hence, the proposition follows from the following fact in group theory, which was kindly proved for us by A. M. Cohen (see [9; 3]).

Lemma 9.2. *Let G be a finite group that acts transitively on a finite set Ω, with $\#\Omega = d > 1$. Then there are at least $(\#G)/d$ elements of G that act without fixed points on Ω.*

Proof. We recall that if G acts on a finite set X, then the number of orbits of X under G is given by the formula

$$\frac{1}{\#G}\sum_{\sigma\in G}\#X^{\sigma},$$

where $X^{\sigma} = \{x \in X : \sigma x = x\}$ (see [15, Kapitel V, Satz 13.4]). We first apply this formula to $X = \Omega$, which by hypothesis has one orbit under G. Writing f_i for the number of $\sigma \in G$ that have exactly i fixed points on Ω, we get

$$\sum_{i=0}^{d} i f_i = \#G.$$

Next we apply it to $X = \Omega \times \Omega$, with G acting componentwise. The diagonal is transformed into itself by G, and there are also off-diagonal points, because $d > 1$. Hence X has at least two orbits under G, so that we obtain

$$\sum_{i=0}^{d} i^2 f_i \geq 2\#G.$$

Finally, we have the trivial relation

$$\sum_{i=0}^{d} f_i = \#G.$$

Since the number $i^2 - (d+1)i + d = (i-1)(i-d)$ is non-positive for $1 \leq i \leq d$, and equal to d for $i = 0$, we now find that

$$df_0 \geq \sum_{i=0}^{d} (i^2 - (d+1)i + d) f_i \geq (2 - (d+1) + d) \cdot \#G = \#G,$$

as desired. This completes the proof of 9.2 and 9.1.

9.3. *Complexity.* The complexity analysis of the square root algorithm that we described in this section is entirely straightforward. As we shall see in Section 11, the parameter y will be chosen as a function of n and d to satisfy

$$\log y = (\tfrac{1}{2} + o(1))\Big(d \log d + \sqrt{(d \log d)^2 + 4 \log(n^{1/d}) \log\log(n^{1/d})}\Big)$$

for $n \to \infty$ and the running time of other steps in the algorithm will (heuristically) be bounded by $y^{2+o(1)}$. In addition, we shall have $\#T = y^{1+o(1)}$, so the same expression is an upper bound for $\#S$ as well, and it is unlikely that $\#S$ is much smaller. Thus an upper bound for the absolute value of the integers involved in the computation of a square root of γ is $\exp(y^{1+o(1)})$. In these circumstances, the calculation of the square root of γ as described in this section takes time at most $y^{1+o(1)}$ if one employs fast multiplication techniques, and $y^{2+o(1)}$ if one uses traditional algorithms for the arithmetic operations. Thus if one does not use fast multiplication techniques then the running time of the square root algorithm may dominate the running time of the entire number field sieve. If we replace [38] by [17;

20] in the square root algorithm, then one has to perform a basis reduction algorithm, and the running time bounds become $y^{2+o(1)}$ and $y^{3+o(1)}$, with fast and traditional arithmetic respectively; the numbers one works with are bounded by $\exp(y^{1+o(1)})$, as before. Thus it is not attractive to use the methods of [17; 20].

Remark. To make the above algorithm more efficient, we can attempt to replace the element γ of which we take the square root by an element that has smaller coefficients when expressed as a polynomial in α. This can possibly be achieved by means of the following idea, which bears some resemblance to the square root algorithm of [29]. Suppose $S = \{(a_1, b_1), \ldots, (a_s, b_s)\}$, where $\#S = s$. We inductively define two sequences $(\mu_i)_{i=0}^{s}$ and $(\nu_i)_{i=0}^{s}$ of elements of $\mathbf{Z}[\alpha]$. First let $\mu_0 = \nu_0 = 1$. Suppose $1 \le i \le s$ and μ_{i-1}, ν_{i-1} have been defined. If $a_i + b_i\alpha$ divides μ_{i-1} in $\mathbf{Z}[\alpha]$, we let $\mu_i = \mu_{i-1}/(a_i + b_i\alpha)$ and we let $\nu_i = \nu_{i-1}(a_i + b_i\alpha)$. Otherwise, we let $\mu_i = \mu_{i-1}(a_i + b_i\alpha)$ and $\nu_i = \nu_{i-1}$. We have the identity

$$\gamma = f'(\alpha)^2 \prod_{i=1}^{s} (a_i + b_i\alpha) = f'(\alpha)^2 \mu_s \nu_s^2,$$

so that if γ is a square in $\mathbf{Z}[\alpha]$, so is $f'(\alpha)^2\mu_s$. Thus, instead of taking a square root of γ, it suffices to take a square root of $f'(\alpha)^2\mu_s$ and to multiply this square root by ν_s. In addition, our factoring algorithm does not need ν_s itself, but only its image $\varphi(\nu_s)$ in $\mathbf{Z}/n\mathbf{Z}$, which one can calculate by only doing arithmetic with integers the size of n.

To test if some non-zero $a + b\alpha$ divides some μ in $\mathbf{Z}[\alpha]$ and compute the quotient if it does, we divide $a + bX$ into f to get $f = (a + bX)g + f(-a/b)$, where $g \in \mathbf{Q}[X]$. Then $a + b\alpha$ divides μ in $\mathbf{Z}[\alpha]$ if and only if $\mu/(a + b\alpha) = -\mu g(\alpha)/f(-a/b)$ belongs to $\mathbf{Z}[\alpha]$.

By using exponent vectors, one can often see very cheaply that $a_i + b_i\alpha$ does not divide μ_{i-1} in $\mathbf{Z}[\alpha]$. Let (p_1, r_1). $(p_2, r_2), \ldots, (p_{B'}, r_{B'})$ be the factor base on the algebraic side, and for $1 \le i \le s$ let $v_i \in \mathbf{Z}^{B'}$ be the integer vector whose coordinates are the numbers $e_{p,r}(a_i + b_i\alpha)$ as (p, r) runs over $(p_1, r_1), (p_2, r_2), \ldots, (p_{B'}, r_{B'})$. Define the vectors $w_i \in \mathbf{Z}^{B'}$ inductively by $w_0 = 0$, $w_i = w_{i-1} - v_i$ if $a_i + b_i\alpha$ divides μ_{i-1}, and $w_i = w_{i-1} + v_i$ otherwise. From Proposition 7.1 we see that w_i is the exponent vector of μ_i and that it has non-negative coordinates. This gives an easily checked necessary condition for $a_i + b_i\alpha$ to divide μ_{i-1}, namely that $w_{i-1} - v_i$ has non-negative coordinates. If $w_{i-1} - v_i$ has a negative coordinate, we do not have to compute $\mu_{i-1}/(a_i + b_i\alpha)$.

The condition that $w_{i-1} - v_i$ has non-negative coordinates is not a sufficient condition for $a_i + b_i\alpha$ to divide μ_{i-1}, but it is nearly so. That is, if $w_{i-1} - v_i$ has non-negative coordinates, then the only prime numbers that can divide the denominators of the coefficients of $\mu_{i-1}/(a_i + b_i\alpha)$ are the prime numbers $p \leq y$ that divide $[\mathcal{O} : \mathbf{Z}[\alpha]]$. From Lemma 3.3 it follows that there are only a few such prime numbers, namely not more than $o(\log n)$ for $n \to \infty$. We can modify the procedure described above by always putting $\mu_i = \mu_{i-1}/(a_i + b_i\alpha)$ when $w_{i-1} - v_i$ has non-negative coordinates. Then we have to keep track of the exponents to which those few prime numbers occur in the denominator of μ_i.

The use of exponent vectors suggests that it may be advantageous to order the set S in such a way that the event that $w_i - v_{i-1}$ has non-negative coordinates is frequent. One possible ordering is the one which puts the smoother elements of S first. There may be better orderings than this, but we are not sure what to suggest.

The practical value of these ideas is unclear; the final verdict must await an implementation.

10. Analytic interlude

In this section we prove a theorem in analytic number theory that is helpful in the complexity analysis of many factoring algorithms, including the number field sieve.

For $x \geq 1$, $y \geq 1$ let $\psi(x, y)$ denote the number of y-smooth positive integers up to x. Suppose x, y are positive integers and consider a process where we choose random integers with the uniform distribution from $[1, x]$ and stop when we have chosen y not necessarily distinct numbers that are y-smooth. The probability that we choose a y-smooth number on one draw is $\psi(x, y)/x$. Thus the expected number of draws to choose y numbers that are y-smooth is $xy/\psi(x, y)$. We now ask for the value of y that minimizes an expression slightly more general than this expectation. Recall the definition of $L_x[u, v]$ from Section 1.

Theorem 10.1. *Suppose g is a function defined for all $y \geq 2$ that satisfies $g(y) \geq 1$ and $g(y) = y^{1+o(1)}$ for $y \to \infty$. Then as $x \to \infty$,*

$$\frac{xg(y)}{\psi(x, y)} \geq L_x[\tfrac{1}{2}, \sqrt{2} + o(1)]$$

uniformly for all $y \geq 2$. In addition,

$$\frac{xg(y)}{\psi(x,y)} = L_x[\tfrac{1}{2}, \sqrt{2} + o(1)]$$

for $x \to \infty$ if and only if $y = L_x[\frac{1}{2}, \sqrt{2}/2 + o(1)]$ for $x \to \infty$.

Proof. We shall use the following result from [7]. For any $\epsilon > 0$ we have

$$(10.2) \qquad \psi(x, x^{1/w}) = x/w^{(1+o(1))w} \qquad \text{for} \quad w \to \infty,$$

uniformly in the region $x \geq w^{(1+\epsilon)w}$.

We first show that if $y \leq L_x[\frac{1}{2}, \frac{1}{4}]$ or $y \geq L_x[\frac{1}{2}, 2]$, then

$$(10.3) \qquad \frac{xg(y)}{\psi(x,y)} \geq L_x[\tfrac{1}{2}, 2 + o(1)] \qquad \text{for} \quad x \to \infty.$$

Indeed, if $y \leq L_x[\frac{1}{2}, \frac{1}{4}]$, then (10.2) implies that

$$\frac{xg(y)}{\psi(x,y)} \geq \frac{x}{\psi(x,y)} \geq \frac{x}{\psi(x, L_x[\frac{1}{2}, \frac{1}{4}])} = L_x[\tfrac{1}{2}, 2 + o(1)]$$

for $x \to \infty$. If $y \geq L_x[\frac{1}{2}, 2]$, then it is clear that (10.3) holds since $x/\psi(x,y) \geq 1$.

Note that (10.2) implies that if $y = L_x[\frac{1}{2}, \vartheta]$, then

$$(10.4) \qquad \frac{xg(y)}{\psi(x,y)} = L_x[\tfrac{1}{2}, \vartheta + 1/(2\vartheta) + o(1)] \qquad \text{for } x \to \infty$$

uniformly for ϑ in any compact subset of the set of positive real numbers. Further $\vartheta + 1/(2\vartheta)$ has its minimum value for $\vartheta > 0$ at $\vartheta = \sqrt{2}/2$ and nowhere else. This minimum value is $\sqrt{2}$, which proves the theorem.

Theorem 10.1 is useful in the analysis of many factoring algorithms. For example, suppose an algorithm factoring n produces auxiliary numbers up to $x = x(n)$ and hopes to find $y^{1+o(1)}$ (for $n \to \infty$) auxiliary numbers that are y-smooth. If these auxiliary numbers are just as likely to be y-smooth as random integers up to x, then we expect to examine $xy^{1+o(1)}/\psi(x,y)$ of these to find the y-smooth integers that we need. If the time to test a single auxiliary number for y-smoothness is $y^{o(1)}$, the expected time for this stage of the factoring algorithm is $xy^{1+o(1)}/\psi(x,y)$. Theorem 10.1 tells us how to choose y so as to minimize this running time, namely $y = L_x[\frac{1}{2}, \sqrt{2}/2 + o(1)]$. Further, this running time would be $y^{2+o(1)} = L_x[\frac{1}{2}, \sqrt{2}+o(1)]$. Thus if other steps in the algorithm, such as processing a matrix, also take time at most $y^{2+o(1)}$, then $L_x[\frac{1}{2}, \sqrt{2} + o(1)]$ is the running time of the complete algorithm. This leads to the following heuristic principle: *if x is a bound on the numbers that "would be smooth" in a factoring algorithm, then the running time of the algorithm is $L_x[\frac{1}{2}, \sqrt{2} + o(1)]$.*

For some factoring algorithms, this outline of a complexity analysis can be used as the backbone of a completely rigorous analysis, such as with Examples 10.5, 10.6 and 10.7 below. For other factoring algorithms, the above argument is supplemented with various heuristic assumptions, one of which is often that the auxiliary numbers that "would be smooth" are just as likely to be smooth as random integers of the same approximate magnitude.

Example 10.5. In the random squares algorithm of Dixon (see [11]) the bound for the auxiliary numbers that would be smooth is $x = n$. The running time of the algorithm thus turns out to be $L_n[\frac{1}{2}, \sqrt{2} + o(1)]$ (see [33]). Here, and in the next two examples, we use the elliptic curve smoothness test (see [27; 33]) so that most y-smooth numbers can (rigorously) be recognized to be y-smooth in time $y^{o(1)}$.

Example 10.6. In [35], Vallée modified the random squares method so that the bound for the auxiliary numbers that would be smooth is $x = n^{2/3+o(1)}$. Thus the running time for her algorithm is $L_n[\frac{1}{2}, \sqrt{4/3} + o(1)]$.

Example 10.7. In the class group relations method [27] the size of the numbers that would be smooth is $n^{1/2+o(1)}$, and its running time is $L_n[\frac{1}{2}, 1 + o(1)]$.

Example 10.8. In the quadratic sieve method [32] the size of the numbers that would be smooth is $n^{1/2+o(1)}$ and so its heuristic running time is $L_n[\frac{1}{2}, 1 + o(1)]$. Here sieving replaces the elliptic curve method as a smoothness test.

The heuristic even works for the elliptic curve factoring method [25]. Here the auxiliary numbers that would be smooth are near the least prime factor p of n. We only need to find one y-smooth auxiliary number, but the time to process one trial is not $y^{o(1)}$ but $y^{1+o(1)}$. Thus the heuristic expected time is still $xy^{1+o(1)}/\psi(x,y)$ where $x = p$. Hence Theorem

10.1 applies and we find that the heuristic running time of the elliptic curve method to factor n is $L_p[\frac{1}{2}, \sqrt{2}+o(1)]$ arithmetic operations with integers the size of n.

A sixth example is provided by the number field sieve. Its heuristic complexity analysis, which is given in Section 11, depends on the two final results of this section.

Lemma 10.9. *Define, for real numbers $k \geq e$, $l \geq 1$, the number $v = v(k, l)$ by*

$$\frac{v^2}{\log v} = kv + l, \qquad v \geq e.$$

Then we have

$$2v = (1+o(1))\big(k \log k + \sqrt{(k \log k)^2 + 2l \log l}\big)$$

as $k + l \to \infty$.

Proof. From $v((v/\log v) - k) = l$ one sees that v is well-defined and that $v \to \infty$ as $k+l \to \infty$. To prove the lemma. we shall show that we can transform the defining equation

$$v^2 = kv \log v + l \log v \tag{10.10}$$

into the quadratic equation

$$v^2 = (1+o(1))\big(kv \log k + \frac{l \log l}{2}\big) \qquad \text{as} \quad k+l \to \infty. \tag{10.11}$$

We distinguish two cases. First suppose that $kv \geq l$, say $kv = cl$ with $c \geq 1$. Then from $kv \leq v^2/(\log v) \leq 2kv$ it follows that $k \to \infty$ and $\log v = (1+o(1)) \log k$ as $k+l \to \infty$. Hence the first term on the right of (10.10) is $(1+o(1))kv \log k$. Using that $l = kv/c$ and that $\log c = O(c)$, we see that the second term is

$$l \log v = \frac{l \log l}{2} + \frac{kv}{2c}(\log v - \log k + \log c) = \frac{l \log l}{2} + o(kv \log k).$$

This gives (10.11). In the second case we have $l \geq kv$, say $l = ckv$ with $c \geq 1$. Then from $l \leq v^2/\log v \leq 2l$ we obtain $\log v = (\frac{1}{2}+o(1)) \log l$. The second term on the right of (10.10) is then $(1+o(1))(l \log l)/2$, and the first is

$$kv \log v = kv \log k + \frac{l}{c}(2 \log v - \log l + \log c) = kv \log k + o(l \log l).$$

This gives again (10.11). Solving the quadratic equation we obtain the lemma.

Lemma 10.12. *Let, for each pair of positive integers n, d satisfying $n > d^{2d^2} > 1$, real numbers $u = u(n,d) \geq 2$ and $y = y(n,d) \geq 2$ be given, with the property that the number*

$$x = x(n,d) = 2dn^{2/d}u^{d+1}$$

satisfies

$$\frac{u^2\psi(x,y)}{x} \geq g(y) \tag{10.13}$$

for some function g satisfying $g(y) \geq 1$ and $g(y) = y^{1+o(1)}$ as $y \to \infty$. Then we have

$$2\log u \geq (1+o(1))\Big(d\log d + \sqrt{(d\log d)^2 + 4\log(n^{1/d})\log\log(n^{1/d})}\Big)$$

for $n \to \infty$, uniformly in d.

Proof. In the proof, all $o(1)$'s are for $n \to \infty$, uniformly in d. From $x^x > n$ we see that $x \to \infty$ as $n \to \infty$. Hence Theorem 10.1 implies that

$$u^2 \geq \frac{xg(y)}{\psi(x,y)} \geq L_x[\tfrac{1}{2}, \sqrt{2} + o(1)].$$

Taking the square of the logarithm on both sides we obtain

$$2(\log u)^2 \geq (1+o(1))\log x \log\log x.$$

Dividing each side by its logarithm, and using that $t/\log t$ is an increasing function of t for $t \geq e$, we find that

$$\frac{(\log u)^2}{\log\log u} \geq (1+o(1))\log x = (1+o(1))(\frac{2}{d}\log n + (d+1)\log u).$$

Applying 10.9, with $k \geq (1+o(1))(d+1)$ and $l \geq (2+o(1))\log(n^{1/d})$, we obtain the lemma.

11. Summary of the number field sieve and a heuristic analysis

We are finally in a position to list the steps of the number field sieve with some precision and to analyze its running time.

Algorithm 11.1. Given a positive integer n, together with parameters d, u, and y satisfying $d > 1$ and $n > d^{2d^2}$, this algorithm attempts to find a non-trivial factor of n or to prove that n is prime; it halts whether or not it is successful.

Step 1. Test whether n is a power of a prime (see [22, Section 2]) or is divisible by a prime that is less than or equal to y. In either case, output the prime and stop.

Step 2. Apply the base m algorithm (see Section 3) to find an integer m and a monic polynomial $f \in \mathbf{Z}[X]$ of degree d such that $f(m) \equiv 0 \bmod n$. Factor f into irreducible factors in $\mathbf{Z}[X]$ by the algorithm of [21]. If f is found to be reducible, with non-trivial factor g, output the non-trivial factor $g(m)$ of n and stop. Assume now that f is irreducible, and denote by α a zero of f. Compute $\gcd(f'(m), n)$. If this is a non-trivial factor of n output this factor and then stop.

Step 3. As described in Sections 4 and 5, use a sieve to find all members of the set

$$T = \{(a,b) \in \mathbf{Z}^2 : \gcd(a,b) = 1,\ |a| \le u,\ 0 < b \le u,\ (a+bm)N(a+b\alpha) \text{ is } y\text{-smooth}\}.$$

Step 4. Form the matrix whose rows are the $\mathbf{F}_2$-vectors $e(a,b)$, as defined in Section 8, for $(a,b) \in T$. Use the Wiedemann coordinate recurrence algorithm (see [40]) to find a non-trivial linear dependence relation on the rows of the matrix. If this is unsuccessful, stop. If it is successful, let S be the set of pairs (a,b) for which $e(a,b)$ occurs in the dependence relation.

Step 5. Express the algebraic integer $\gamma = f'(\alpha)^2 \prod_{(a,b)\in S}(a+b\alpha)$ as a polynomial in α of degree less than d. Attempt to find a square root $\beta = \sum_{i=0}^{d-1} b_i \alpha^i$ of γ by the method of [38] (see Section 9). If this is unsuccessful, stop.

Step 6. For c an integer with $c^2 = f'(m)^2 \prod_{(a,b)\in S}(a+bm)$, find the residue $c \bmod n$.

Step 7. Compute $\gcd(c - \sum_{i=0}^{d-1} b_i m^i,\ n)$. If this is a non-trivial factor of n, output the result and stop. Otherwise, remove an element of S from T and start again at Step 4.

This completes the description of the algorithm.

The following conjectural result describes the optimal choice of the parameters d, u, and y, and the running time of the algorithm for this choice.

Conjecture 11.2. *For each integer n with $n > 256$, one can choose d, u, and y, such that*

$$d = (3^{1/3} + o(1))(\log n/\log\log n)^{1/3}, \qquad n > d^{2d^2} > 1,$$
$$u = y = L_n[\tfrac{1}{3}, (\tfrac{8}{9})^{1/3} + o(1)]$$

for $n \to \infty$, and such that Algorithm 11.1, *on input n, d, u, and y, succeeds either in finding a non-trivial factor of n or in proving that n is prime, in time at most*

$$L_n[\tfrac{1}{3}, (64/9)^{1/3} + o(1)] \tag{11.3}$$

for $n \to \infty$. Moreover, this is optimal in the sense that for general n and for all choices of d, u and y satisfying $n > d^{2d^2} > 1$ for which the algorithm is successful, the expression (11.3) *is a lower bound for the time taken by the algorithm.*

The adjective "general" in the last assertion of the conjecture is meant to express that we allow for exceptional integers n, for which the algorithm takes less time. For example, if n is a power of a prime number, then Algorithm 11.1 terminates in Step 1 in time much less than (11.3), independently of the choice of d, u, and y. Likewise, if n has a relatively small prime factor, then there may be a choice of y for which the algorithm terminates in Step 1 in time less than (11.3). Next, there is a very small class of integers that for a suitable choice of d are factored in Step 2 with very little effort. Finally, if the coefficients of the polynomial f constructed in Step 2 are, for a suitable value of d, much smaller than their upper bound $n^{1/d}$, then it is reasonable to suppose that one can factor n in time less than (11.3), with values for u and y that may not be those in the conjecture. This occurs, for example, if the special number field sieve [23] can be applied. We do not know whether further categories of exceptional integers n exist, but we believe that most integers divisible by at least two distinct primes and not divisible by any small primes are in the class of "general" integers for which (11.3) is a lower bound for the time taken by Algorithm 11.1 to factor them.

The following more general conjecture describes the optimal choice of u and y for given n and d.

Conjecture 11.4. *For any two positive integers n and d satisfying $n > d^{2d^2} > 1$, one can choose u and y such that each of u and y is*

$$(11.5) \qquad \exp\Big((\tfrac{1}{2} + o(1))\Big(d \log d + \sqrt{(d \log d)^2 + 4 \log(n^{1/d}) \log\log(n^{1/d})}\Big)\Big)$$

and such that Algorithm 11.1, *on input n, d, u, and y, succeeds either in finding a nontrivial factor of n or in proving that n is prime, in time at most*

$$(11.6) \qquad \exp\Big((1 + o(1))\Big(d \log d + \sqrt{(d \log d)^2 + 4 \log(n^{1/d}) \log\log(n^{1/d})}\Big)\Big),$$

where the $o(1)$'s are for $n \to \infty$, uniformly in d. Moreover, this is optimal in the sense that for general n, for all d in the region $n > d^{2d^2} > 1$, and for all choices of u, y for which the algorithm is successful, the time taken by the algorithm is at least (11.6).

To deduce 11.2 from 11.4 it suffices to choose d so as to minimize (11.6). It is easy to see that we have to make $(d \log d)^2$ and $\log(n^{1/d}) \log\log(n^{1/d})$ of the same order of magnitude, which occurs when d has the same order of magnitude as $(\log n / \log\log n)^{1/3}$. Putting $d = \delta(\log n / \log\log n)^{1/3}$ and optimizing δ we find that the optimal choice of d satisfies $\delta = 3^{1/3} + o(1)$ for $n \to \infty$. This immediately leads to 11.2.

We now present a heuristic argument for the correctness of Conjecture 11.4. We begin with the last assertion of the conjecture, which states that (11.6) is, in general, a lower bound for the running time. We deduce this from Lemma 10.12. If we assume that the algorithm does not terminate in Step 1 or in Step 2, then the running time is at least the total number of locations in the sieve from Section 4 that is used in Step 3, which is at least u^2. The lower bound for u^2 that is given by Lemma 10.12 thus leads immediately to the lower bound (11.6) for the running time, provided that we check that condition (10.13) is satisfied. We shall deduce (10.13), heuristically, from a constraint that is implicit in Step 4 of the algorithm, namely, the condition that the number of rows of the matrix in this step is at least of the same order of magnitude as the number of columns; otherwise Step 4 is unlikely to be successful in finding a set S. The number of columns is at least the number of primes in the factor base on the rational side. This is $\pi(y)$, which is $y^{1+o(1)}$ for $y \to \infty$, as required for the right side of (10.13). To estimate the number of rows, we first discuss a bound on the magnitude of the auxiliary numbers generated in Step 3 that "would be smooth". For $|a| \le u$ and $0 < b \le u$, the integer $(a + bm)N(a + b\alpha)$ has absolute value at most

$$(u + um)(d + 1)mu^d \le 2dm^2u^{d+1} \le 2dn^{2/d}u^{d+1},$$

since the coefficients of f are bounded by m and $m \le n^{1/d}$. Hence the number $x = 2dn^{2/d}u^{d+1}$ defined in Lemma 10.12 is a bound on the auxiliary numbers that would be smooth. A random positive integer up to x is y-smooth with probability $\psi(x, y)/[x]$. The number of integers that we try is the number of pairs of integers a, b satisfying $|a| \le u$, $0 < b \le u$, and $\gcd(a, b) = 1$, which is about cu^2 for $c = 12/\pi^2$. Thus we might naively think that a good approximation to the cardinality of the set T in Step 3 is given by $cu^2\psi(x, y)/x$. This belief then leads to (10.13), the constant c being absorbed in the factor $y^{o(1)}$ that we allow on the right hand side of (10.13).

We do not know to what extent the naive belief on which the above argument relies is justified. However, we feel that it is reasonable to suppose that for "general" n, d our approximation to the cardinality of T is correct within an exponent $1 + o(1)$ for $y \to \infty$ (as allowed by (10.13)), at least for the values of u and y that are relevant for the algorithm.

Next we turn to the first assertion of 11.4. Our heuristic argument for this is based on the same naive belief as above. Inspecting the case in which equality is achieved in Lemma 10.12, we find in a straightforward way that the numbers

$$u_0 = y_0 = \exp\Big(\tfrac{1}{2}\Big(d\log d + \sqrt{(d\log d)^2 + 4\log(n^{1/d})\log\log(n^{1/d})}\Big)\Big),$$
$$x_0 = 2dn^{2/d}u_0^{d+1}$$

satisfy

$$\frac{u_0^2\psi(x_0, y_0)}{x_0} = y_0^{1+o(1)}, \tag{11.7}$$

the $o(1)$'s here and in the rest of the argument being for $n \to \infty$, uniformly in d. We shall choose u and y a little larger. Specifically, let ϵ be a positive real number, and put

$$u = y = \exp\Big(\frac{1+\epsilon}{2}\Big(d\log d + \sqrt{(d\log d)^2 + 4\log(n^{1/d})\log\log(n^{1/d})}\Big)\Big),$$
$$x = 2dn^{2/d}u^{d+1}.$$

Note that these numbers tend to infinity with n, and that we have $\log n = y^{o(1)}$ and $(\log y)/\log x = o(1)$. From $y = y_0^{1+\epsilon}$, $x \le x_0^{1+\epsilon}$ we see that $(\log x)/\log y \le (\log x_0)/\log y_0$, so (10.2) gives

$$\frac{\psi(x,y)}{x} \ge \left(\frac{\psi(x_0,y_0)}{x_0}\right)^{1+o(1)}.$$

Combining this with (11.7) we obtain

$$\frac{u^2\psi(x,y)}{x} \ge \left(\frac{u_0^{2(1+\epsilon)}\psi(x_0,y_0)}{x_0}\right)^{1+o(1)} = \left(u_0^{2\epsilon}y_0\right)^{1+o(1)},$$

which by $u_0 = y_0 = y^{1/(1+\epsilon)}$ implies that

$$\frac{u^2\psi(x,y)}{x} \ge y^{(1+o(1))(1+2\epsilon)/(1+\epsilon)}. \tag{11.8}$$

From this inequality we shall deduce, heuristically, that there is a constant $n(\epsilon)$ such that for $n > n(\epsilon)$, with the above choices of u and y, the number of rows in the matrix in Step 4 is at least the number of columns in the matrix plus an upper bound for the number of times that we cycle through Steps 4 to 7.

As above, we estimate the number of rows to be $(u^2\psi(x,y)/x)^{1+o(1)}$. The number of columns is, in the notation of Section 8, equal to $1 + B + B' + B''$. We have

$$B = \pi(y) < y, \qquad B' \le dy, \qquad B'' \le 5\log n, \qquad d < \log n = y^{o(1)},$$

and therefore

$$1 + B + B' + B'' = y^{1+o(1)}.$$

Finally, the number of times that we cycle through Steps 4 to 7 is one more than the number of times that we find a trivial factor of n in Step 7, which is heuristically bounded by $(\log n)^{O(1)} = y^{o(1)}$. Thus our assertion follows, heuristically, from (11.8).

We conclude that every time that Step 4 is performed, it finds a non-trivial linear relation between the rows of the matrix. The linear relations found by the algorithm are linearly independent, so it is reasonable to conjecture that ultimately one of these relations will give rise to a non-trivial factor of n in Step 7. Letting ϵ tend to 0 for $n \to \infty$ we find that we can indeed choose u and y such that each of them is given by (11.5) and such that the algorithm is likely to be successful on input n, d, u, y. Then we have $u = u_0^{1+o(1)}$, $y = y_0^{1+o(1)}$, so (11.7) is also true with u_0, y_0, x_0 replaced by u, y, x.

It remains to estimate the running time of the algorithm with this choice of parameters. It is easy to see that the time taken by Step 3 equals $u^{2+o(1)}$, which is the length of the sieve multiplied by a lower order factor; this gives rise to the expression (11.6). It is clear that Steps 1, 2, 6 and 7 are negligible compared with Step 3. To estimate the running time of the Wiedemann coordinate recurrence method in Step 4, we note that the matrix formed in this step has $y^{1+o(1)}$ columns and about as many rows. In addition, the number of non-zero entries in each row is $O(\log n) = y^{o(1)}$. Thus the number of non-zero entries in the matrix is $y^{1+o(1)}$ and the running time of Step 4 is $y^{2+o(1)}$. This is the same as our bound for the running time of Step 3. In Section 9 we saw that the running time for Step 5 is $y^{2+o(1)}$ if we use naive arithmetic and $y^{1+o(1)}$ if we use fast arithmetic subroutines. Thus either way this step too is dominated by Step 3. Finally, as we saw above, the number of times that we cycle through Step 4 to 7 is likely to be $y^{o(1)}$.

This concludes our heuristic argument supporting Conjectures 11.2 and 11.4.

We note that the bound for the numbers that "would be smooth" is

$$x = \exp\Big(\big(\tfrac{1}{2} + o(1)\big)\Big(d^2 \log d + 4\log(n^{1/d}) + d\sqrt{(d\log d)^2 + 4\log(n^{1/d})\log\log(n^{1/d})}\Big)\Big)$$

for $n \to \infty$, uniformly in d, when u is chosen as in Conjecture 11.4, and

$$x = L_n[\tfrac{2}{3}, (64/3)^{1/3} + o(1)] \qquad \text{for} \quad n \to \infty$$

when d and u are chosen as in Conjecture 11.2.

We make a final remark concerning the numbers $(a+bm)N(a+b\alpha)$ in Step 3 that are examined for y-smoothness. We have assumed above that these integers are just as likely to be y-smooth as random integers of the same magnitude. In fact, the alert reader may have noticed that these numbers, since they already factor into the two smaller numbers $a+bm$ and $N(a+b\alpha)$, perhaps have a *greater* chance of being y-smooth than a random integer. For practical purposes this may be true. Asymptotically, however, an argument similar to the one above, but taking this factorization into account, can be worked out, and it gives exactly the same results. That is, any differences in the two analyses are absorbed in the expression "$o(1)$". It may be of interest to point out that when $(a+bm)N(a+b\alpha)$ is y-smooth and a, b are coprime, then the numbers $a+bm$ and $N(a+b\alpha)$ are coprime too. Indeed, if a prime p divides both, it divides $f(m)=n$. However, after Step 1 we are assured that n has only prime factors greater than y.

12. Homogeneous polynomials

In this section we discuss a modification of the number field sieve, in which the one-variable polynomial f is replaced by a homogeneous polynomial in two variables. This has the advantage that its coefficients can be taken a bit smaller, which may improve the practical performance of the algorithm. We first describe the algorithm, and then provide additional explanations for some of the individual steps.

Algorithm 12.1. Given an integer $n>1$, which is not a power of a prime number (see 11.1, Step 1), together with parameters d, u, and y, which are positive integers, this algorithm attempts to find a non-trivial factor of n.

Step 1. Find integers m_1, m_2, and a dth degree homogeneous polynomial

$$f = c_d X^d + c_{d-1}X^{d-1}Y + \ldots + c_1 XY^{d-1} + c_0 Y^d \in \mathbf{Z}[X,Y]$$

such that m_1, m_2, and the coefficients c_i are "small" and such that we have $f(m_1,m_2)\equiv 0 \bmod n$, $f(m_1,m_2)\neq 0$. See 12.2 for methods to select f, m_1, m_2.

Step 2. Check that f is irreducible in $\mathbf{Z}[X,Y]$, so that in particular $\gcd(c_0,c_1,\ldots,c_d)=1$, and that $f\neq X$, $f\neq Y$. Further check that each of m_2, c_d, and

$$f_X(m_1,m_2)=\sum_{i=1}^{d} i c_i m_1^{i-1} m_2^{d-i}, \qquad \text{where } f_X=\frac{\partial f}{\partial X},$$

is coprime to n. See 12.5 for more information on this step.

Step 3. For each prime number $p\leq y$, determine the set $R'(p)$ of elements $(r_1:r_2)$ of the projective line $\mathbf{P}^1(\mathbf{F}_p)$ over $\mathbf{F}_p$ for which $f(r_1,r_2)=0$. Note that if we identify $\mathbf{P}^1(\mathbf{F}_p)$ with $\mathbf{F}_p\cup\{\infty\}$ by identifying $(r_1:r_2)$ with r_1/r_2, then $R'(p)$ consists of those $r=r_1/r_2\in\mathbf{F}_p$ for which $f(r,1)=0$, together with ∞ if $c_d\equiv 0 \bmod p$.

Step 4. Find all members of the set

$$T = \{(a, b) \in \mathbf{Z}^2 : \gcd(a, b) = 1,\ |a| \le u,\ 0 < b \le u,\ (am_2 - bm_1)f(a, b) \text{ is } y\text{-smooth}\}.$$

This is done with a sieve, as described in Sections 4 and 5. Note that, for coprime integers a, b, and a prime number p, the number $f(a, b)$ is divisible by p if and only if $(a \bmod p : b \bmod p) \in R'(p)$.

Step 5. For each $(a, b) \in T$, form the $\mathbf{F}_2$-vector $e(a, b)$ that is defined as follows. The first coordinate of $e(a, b)$ is determined by the sign of $am_2 - bm_1$, as in Section 8 (we cannot have $am_1 - bm_2 = 0$ if $\max\{|m_1|, |m_2|\} > u$, which will be the case with our choice of parameters). The next $B = \pi(y)$ coordinates are given by $\operatorname{ord}_p(am_2 - bm_1) \bmod 2$, as p runs over the prime numbers $\le y$. Next there are B' coordinates, where $B' = \sum_{p \le y} \# R'(p)$, the sum ranging over prime numbers p. For each prime number $p \le y$ and each $r \in R'(p)$, the (p, r)th coordinate of $e(a, b)$ is equal to $e_{p,r}(a, b) \bmod 2$, where $e_{p,r}(a, b)$ equals $\operatorname{ord}_p(f(a, b))$ if $(a \bmod p : b \bmod p) = r$ and $e_{p,r}(a, b) = 0$ otherwise. Each of the following B'' coordinates corresponds to a prime number $q > y$ and a pair of numbers s_1, s_2 with $(s_1 \bmod q : s_2 \bmod q) \in R'(q)$; see 12.7 for the choice of B'' and the triples q, s_1, s_2. The (q, s_1, s_2)th coordinate of $e(a, b)$ is 0 mod 2 if the Legendre symbol $(\frac{as_2 - bs_1}{q})$ equals 1, and 1 mod 2 if it equals -1. Finally, $e(a, b)$ has a last coordinate that is equal to 1 mod 2. (Dan Bernstein points out that this last coordinate can be omitted if $m_2 = 1$ and $m_1 > u$, since then it is equal to the first coordinate, which gives the sign of $am_2 - bm_1$.) Thus $e(a, b) \in \mathbf{F}_2^{2+B+B'+B''}$.

Step 6. Use the Wiedemann coordinate recurrence algorithm (see [40]) to find a nontrivial linear dependence relation between the vectors $e(a, b)$, $(a, b) \in T$. If this is unsuccessful, stop. If it is successful, let S be the set of pairs $(a. b)$ for which $e(a, b)$ occurs in the dependence relation. Note that $\#S$ is even, due to the presence of the last coordinate.

Step 7. Let the algebraic integer ω be a zero of the polynomial $f(X, c_d)$. Express the algebraic integer

$$\gamma = (f_X(\omega, c_d)/c_d)^2 \prod_{(a,b) \in S} (c_d a - b\omega)$$

(with f_X as in Step 2) as a polynomial in ω of degree less than d. Attempt to find a square root β of γ by the method of [38] (see Section 9). If this is unsuccessful, stop. Otherwise, if $\beta = \sum_{i=0}^{d-1} b_i \omega^i$, with $b_i \in \mathbf{Z}$, calculate an integer v with $v \equiv \sum_{i=0}^{d-1} b_i c_d^i m_1^i m_2^{d-1-i} \bmod n$.

Step 8. For w an integer with $w^2 = \prod_{(a,b)\in S}(am_2 - bm_1)$, find the residue $w \bmod n$. In addition, calculate integers h, l with

$$h \equiv c_d^{d-2+\#S/2} \cdot f_X(m_1, m_2) \bmod n, \qquad l \equiv m_2^{\#S/2} \bmod n.$$

Step 9. Compute $\gcd(hw - lv, n)$. If this is a non-trivial factor of n, output the result and stop. Otherwise, remove an element of S from T and start again at Step 6.

This completes the description of the algorithm. We now discuss some of the individual steps.

12.2. *Selecting f and m_1, m_2.* If we insist on choices for which $m_2 = c_d = 1$, then Algorithm 12.1 reduces to 11.1, except for the last coordinate that was appended to the vectors $e(a, b)$. We now discuss three methods for choosing f, m_1, m_2. In the first method we allow $c_d \neq 1$, in the second method we allow $m_2 \neq 1$, and in the third method we allow both.

In the first method we take $m_2 = 1$, and we let m_1 be the least integer exceeding $n^{1/(d+1)}$. We obtain f by expanding n in base m_1, so that

$$n = c_d m_1^d + c_{d-1} m_1^{d-1} + \ldots + c_1 m_1 + c_0, \qquad 0 \leq c_i < m_1.$$

With this method, we have $|m_i| \leq n^{1/(d+1)} + 1$, $|c_i| \leq n^{1/(d+1)}$. Of course, we can modify this method by changing m_1 a little, by allowing some of the digits c_i to be negative, or by replacing n by a small multiple.

In the second method we take $c_d = 1$. To find the other c_i and m_1, m_2, we proceed as follows. For m_1 one tries several values with $m_1 \approx n^{1/(d+1)}$, until one discovers a value for which $n - m_1^d$ is found to have a divisor m_2 with $m_2 \approx n^{1/(d+1)}$; for example, by trial division or by the elliptic curve method one may discover so many small factors of $n - m_1^d$ that it is easy to multiply some of them together in order to obtain a suitable m_2. Note that $\gcd(m_1, m_2)$ divides n and is generally much smaller than n; so we may assume that $\gcd(m_1, m_2) = 1$. Next one determines small coefficients c_i such that

$$\frac{n - m_1^d}{m_2} = c_{d-1} m_1^{d-1} + \ldots + c_1 m_1 m_2^{d-2} + c_0 m_2^{d-1}. \qquad (12.3)$$

One can do this either by going from the right, determining c_0, c_1, ... successively by looking modulo m_1 and requiring that $|c_i| \leq m_1/2$ (or $0 \leq c_i < m_1$); or similarly from the left and finding c_{d-1}, c_{d-2}, ... from congruences modulo m_2; or by determining some from the right and some from the left. In all cases, the final c_j to be determined is forced by equation (12.3). If d is small in comparison with $n^{1/(d+1)}$, as it will be in practice, then the order of magnitude of $|c_j|$ will not be much larger than $n^{1/(d+1)}$. Again, this method allows several refinements. For example, one might choose m_1, m_2 to be $\approx (n/d)^{1/(d+1)}$; a judicious choice of non-negative values for the c_i may then result in a smaller value for the final c_j.

In the third method we allow both $m_2 \neq 1$ and $c_d \neq 1$. Although we do not know how to exploit this freedom in order to obtain substantially better results, it is still of interest to see how one can proceed. Namely, one can first choose arbitrary coprime integers m_1, m_2 that are $\approx n^{1/(d+1)}$. Next one needs to determine the c_i such that

$$kn = c_d m_1^d + c_{d-1} m_1^{d-1} m_2 + \ldots + c_0 m_2^d \tag{12.4}$$

for some small non-zero integer k. One can either do this by first choosing k (for example, $k = 1$), and next determining the c_i by one of the methods that we indicated for solving (12.3). Alternatively, one can consider the subgroup

$$L = \{(x_i)_{i=0}^d : \sum_{i=0}^{d} x_i m_1^i m_2^{d-i} \equiv 0 \bmod n\}$$

of $\mathbf{Z}^{d+1}$. A basis of L is given by $(0, 0, \ldots, 0, n)$ together with the d vectors

$$(0, \ldots, 0, 1, -t, 0, \ldots, 0),$$

where $t \in \mathbb{Z}$ is such that $tm_2 \equiv m_1 \bmod n$ (here we assume that $\gcd(m_2, n) = 1$; see 12.5). One can apply a lattice basis reduction algorithm (see [21]) to find a basis of L that consists of relatively short vectors. At least one of the vectors $(x_i)_{i=0}^d$ in the reduced basis satisfies $\sum_{i=0}^d x_i m_1^i m_2^{d-i} \neq 0$, and a solution to (12.4) is then given by $c_i = x_i$. Also for this algorithm one expects the c_i to be of order of magnitude $n^{1/(d+1)}$.

In the above we attempted to minimize the absolute values of m_1, m_2, and the coefficients of f. It should be kept in mind, however, that other properties of f also influence the running time of the algorithm. For example, one may want to choose f in such a way that $f \bmod p$ has many linear factors in $\mathbf{F}_p[X]$ for several small prime numbers p. This increases the smoothness probability of the numbers $f(a,b)$.

12.5. *Irreducibility testing.* With any reasonable choices that are made in Step 1, each of m_2 and c_d will be much less than n in absolute value. Hence if any of $\gcd(m_2,n)$, $\gcd(c_d,n)$ is found to be different from 1 then it is a non-trivial divisor of n, and the algorithm can stop. Assume now that $\gcd(m_2,n) = \gcd(c_d,n) = 1$. The content $\operatorname{cont} f = \gcd(c_0, c_1, \ldots, c_d)$ of the polynomial f divides the multiple $f(m_1,m_2)$ of n, and it is coprime to n because it divides c_d. Therefore the polynomial $f^* = f/\operatorname{cont} f$ still has the property that $f^*(m_1,m_2)$ is divisible by n. Thus, replacing f by f^* if necessary, we may assume that $\operatorname{cont} f = 1$. We can now factor f into irreducible factors in $\mathbf{Z}[X,Y]$ with the algorithm of [21]; note that the factorization of f can easily be obtained from the factorization of the one-variable polynomial $f(X,1)$. Suppose first that f is found to be reducible, $f = gh$ (say). Then we have $f(m_1,m_2) = g(m_1,m_2)h(m_1,m_2)$, which leads to a splitting of n. For most reasonable choices of the parameters it is very likely that $g(m_1,m_2)$ and $h(m_1,m_2)$ are less than n in absolute value, so that this is a non-trivial splitting. If nevertheless the splitting is trivial, then one of $g(m_1,m_2)$, $h(m_1,m_2)$ is divisible by n, say the first one. Then we can replace f and d by g and $\deg g$. It is easy to see that this replacement improves the algorithm. Let it next be assumed that f is irreducible. Again, in most cases the number $f_X(m_1,m_2)$ will be less than n, so that $\gcd(f_X(m_1,m_2),n)$ is a non-trivial divisor of n if it is not 1; and if it ever happens that $\gcd(f_X(m_1,m_2),n) = n$ then one has the option of replacing f and d by f_X and $d-1$. Finally, the conditions $f \neq X$, $f \neq Y$ are satisfied if $|m_1|$, $|m_2| < n$, which is very likely to be true.

12.6. *First degree primes.* In Section 5 we saw that the pairs consisting of a prime number p and an element $r \in R(p)$ correspond to the first degree primes of the ring $\mathbf{Z}[\alpha]$. The pairs consisting of a prime number p and an element $r \in R'(p)$ that occur in Algorithm 12.1 can be interpreted in a similar manner. We introduce some notation.

Let $\alpha = \omega/c_d$, where ω is. as in Step 7, a zero of $f(X, c_d)$; so α is a zero of $f(X,1)$. Note that α is not an algebraic integer unless $c_d = \pm 1$; but ω is an algebraic integer, since $f(X, c_d)/c_d$ is a monic polynomial with integral coefficients. Let the elements $\beta_0, \ldots, \beta_{d-1} \in \mathbf{Z}[\alpha]$ be defined by $f(X,1)/(X-\alpha) = \sum_{i=0}^{d-1} \beta_i X^i$; so $\beta_i = c_d\alpha^{d-1-i} + c_{d-1}\alpha^{d-2-i} + \ldots + c_{i+1}$. Further let $A = \mathbf{Z} + \sum_{i=0}^{d-2} \mathbf{Z}\beta_i$. A simple computation shows that A is closed under multiplication, so that A is an order in the number field $K = \mathbf{Q}(\alpha)$, in the sense of Section 7 (cf. [26, 2.10]). We have $\mathbf{Z}[\omega] \subset A \subset \mathbf{Z}[\alpha]$, where $\mathbf{Z}[\omega]$ is also an order in K, but $\mathbf{Z}[\alpha]$ is not (unless $c_d = \pm 1$). The discriminant of A is equal to the discriminant Δ of $f(X,1)$, and the discriminant of $\mathbf{Z}[\omega]$ equals $c_d^{(d-1)(d-2)}\Delta$. It is of interest to observe that the ring A does not change if $f(X,Y)$ is replaced by $f(Y,X)$ and α by α^{-1}; so we also have $A \subset \mathbf{Z}[\alpha^{-1}]$, and in fact one can show that $A = \mathbf{Z}[\alpha] \cap \mathbf{Z}[\alpha^{-1}]$.

With this notation, the pairs consisting of a prime number p and an element $(r_1 : r_2) \in R'(p)$ are in bijective correspondence with the first degree primes P of A. If $r_2 \neq 0$, then P is the intersection of A and the kernel of the ring homomorphism $\mathbf{Z}[\alpha] \to \mathbf{F}_p$ that sends α to r_1/r_2. If $r_2 = 0$, then P is the intersection of A and the kernel of the ring homomorphism $\mathbf{Z}[\alpha^{-1}] \to \mathbf{F}_p$ that sends α^{-1} to 0. Each prime P of A gives rise to a function l_P as in 7.1.

Let a, b be a pair of coprime integers. Then the following analogue of 5.5 is valid. First, if P is a prime of A of degree greater than 1, then $l_P(a - b\alpha) = 0$. Next, let P be a first degree prime of A, corresponding to a pair p, $r \in R'(p)$. Then the number $e_{p,r}(a,b)$ that is defined as in Step 5 is equal to the number of composition factors of the A-module $(A + A\alpha)/A(a - b\alpha)$ that are isomorphic to A/P; explicitly speaking, one has

$$e_{p,r}(a,b) = \begin{cases} l_P(a - b\alpha) & \text{if } r \neq \infty, \\ l_P(a - b\alpha) + \operatorname{ord}_p c_d & \text{if } r = \infty. \end{cases}$$

(Note that this is consistent with 7.1(c), since $f(a,b) = c_d N(a - b\alpha)$.) It follows that the analogue of 5.3 holds, provided that we restrict attention to sets S for which $\#S$ is even.

12.7. *Making squares.* It is the purpose of Steps 5 and 6 to find a non-empty subset $S \subset T$ of even cardinality such that $\prod_{(a,b)\in S}(am_2 - bm_1)$ is a square in $\mathbf{Z}$ and $\prod_{(a,b)\in S}(a - b\alpha)$ is a square in K. Clearly, the condition that S be even is taken care of by the last coordinate of the vectors $e(a,b)$, and the condition that the product of the elements $am_2 - bm_1$ be a square by the first $1 + B$ coordinates. The B' coordinates that correspond to the pairs p, r guarantee, by 12.6, that the set S found in Step 6 is such that the product of the elements $a - b\alpha$, for $(a, b) \in S$, belongs to the subgroup V_A of K^* defined in Section 7. One has $V_A \supset K^{*2}$, and depending on the algorithm used for Step 1 one can mimic the proof of Theorem 6.7 and find a constant c for which the obstruction group V_A/K^{*2} has $\mathbf{F}_2$-dimension at most $c \log n$. To overcome this obstruction group, one can use quadratic characters for, say, $B'' = [(c + 2/\log 2) \log n]$ first degree primes Q of A. As in Section 8, one can choose these primes to be the first B'' primes of norm exceeding y that do not contain $f_X(\omega, c_d)$. Explicitly, one can take the first B'' triples q, s_1, $s_2 = 1$ for which q is a prime number not dividing c_d with $q > y$, and $s_1 \pmod q$ is such that $f(s_1, 1) \equiv 0 \bmod q$, $f_X(s_1, 1) \not\equiv 0 \bmod q$, and use these in Step 5.

12.8. *The final congruence.* Suppose that $\prod_{(a,b)\in S}(a - b\alpha)$ is a square in K and that $\#S$ is even. Multiplying by $c_d^{\#S}$ we see that the element $\prod_{(a,b)\in S}(c_d a - b\omega)$ of the order $\mathbf{Z}[\omega]$ is also a square in K. The square root is in the ring of integers of K, so $f_X(\omega, c_d)/c_d$ times the square root belongs to $\mathbf{Z}[\omega]$ (see [39, Proposition 3-7-14]). Hence the element β calculated in Step 7 has coefficients b_i in $\mathbf{Z}$.

Let now the ring homomorphism $\varphi\colon \mathbf{Z}[\alpha] \to \mathbf{Z}/n\mathbf{Z}$ be such that $\varphi(\alpha) = (m_1 \bmod n)/(m_2 \bmod n)$. Then $\varphi(m_2\omega) = (c_d m_1 \bmod n)$, so with v as in Step 7 we have

$$\varphi(m_2^{d-1}\beta) = (v \bmod n).$$

With l, w and h as in Step 8, this leads to

$$\begin{aligned}
(l^2 v^2 \bmod n) &= \varphi(m_2^{2(d-1)+\#S}\beta^2) \\
&= \varphi\Big(\big(m_2^{d-1} f_X(\omega, c_d)/c_d\big)^2 \prod_{(a,b)\in S} m_2(c_d a - b\omega)\Big) \\
&= \varphi\Big(\big(f_X(m_2\omega, m_2 c_d)/c_d\big)^2 \prod_{(a,b)\in S} c_d(a m_2 - b m_1)\Big) \\
&= \varphi((c_d^{d-2} f_X(m_1, m_2))^2 c_d^{\#S} w^2) \\
&= (h^2 w^2 \bmod n).
\end{aligned}$$

This explains the attempt in Step 9 to find a non-trivial factor of n.

12.9. *Choice of parameters u, y, d.* The heuristic analysis of Algorithm 11.1 given in Section 11 can be copied without essential changes for Algorithm 12.1. The main difference is that the factor $n^{2/d}$ in x is to be replaced by $n^{2/(d+1)}$. Since our analysis gave the optimal value for d only up to a factor $1 + o(1)$ (for $n \to \infty$), the heuristic asymptotic results for Algorithm 12.1 are the same as for Algorithm 11.1. From a practical point of view, 12.1 may be better than 11.1; see the discussion in 12.15.

12.10. *The optimal choice of the polynomial.* In Section 3 and in 12.2 we described altogether four methods for selecting f, m_1, and m_2. One may ask whether there is a better method for doing this. We present an argument that leads to a limit on the performance of any method for selecting f, m_1, and m_2. It shows that asymptotically one cannot expect to do better than the methods that we described if one wishes the algorithm to apply to all integers n. In addition, the argument suggests that for practical purposes there may still be room for improvement (see 12.15).

For a given choice of n and d, what would be a good choice of f, m_1, m_2 in Step 1 of Algorithm 12.1? Let $M = \max\{|m_1|, |m_2|\}$ and let $C = \max\{|c_0|, |c_1|, \ldots, |c_d|\}$. An upper bound on the integers $|(am_2 - bm_1)f(a.b)|$ that are examined for smoothness in Step 4 of the algorithm is $2(d+1)u^{d+1}CM$. Thus for a given n and d, a choice of f, m_1, m_2 which has the product CM small should be better for factoring n than another choice with CM large.

For example, in the base m method used in Algorithm 11.1 we have $M \leq n^{1/d}$ and $C \leq n^{1/d}$, so that $CM \leq n^{2/d}$. The methods of 12.2 achieve $CM = O(n^{2/(d+1)})$, so we would expect these methods to give an improvement over the base m method. The following result expresses that we cannot expect to get CM substantially smaller than $n^{2/(d+2)}$ for all n.

Given positive integers d, C, M, let $S(d,C,M)$ denote the set of non-zero integers of the form $f(m_1,m_2)$ where m_1, m_2 are integers with $|m_1|, |m_2| \leq M$ and $f = \sum_{i=0}^{d} c_i X^i Y^{d-i} \in \mathbf{Z}[X,Y]$ satisfies $|c_i| \leq C$ for $0 \leq i \leq d$.

Proposition 12.11. *For each $\epsilon > 0$ there is a number $N(\epsilon)$ with the following property. Suppose d, C, M, N are positive integers with $N > N(\epsilon)$. If each integer in the interval $[1,N]$ has a multiple in $S(d,C,M)$, then*

$$CM \geq \frac{1}{8} N^{(2-\epsilon)/(d+2)}.$$

Proof. It suffices to prove the proposition in the case $0 < \epsilon < 1$. Suppose d, C, M, N are positive integers and $\mathcal{S}(d, C, M)$ contains a multiple of each of the integers in $[1, N]$. We may assume that $CM \le N^{2/d}$ for otherwise there is nothing to prove. It is clear that each member of $\mathcal{S}(d, C, M)$ has absolute value at most $(d+1)CM^d$. Thus

$$N \le (d+1)CM^d \le (d+1)(CM)^d \le (d+1)N^2. \tag{12.12}$$

Let $D = \max\{\tau(j) : 1 \le j \le (d+1)N^2\}$, where $\tau(j)$ denotes the number of divisors of j. Since $\tau(j) = j^{o(1)}$ for $j \to \infty$, there is some number $N(\epsilon)$ such that if $N > N(\epsilon)$ we have

$$D \le (d+1)^{\epsilon} N^{\epsilon}. \tag{12.13}$$

By our assumption on N we have

$$N \le D \cdot \#\mathcal{S}(d, C, M) \le D(2C+1)^{d+1}(2M+1)^2 \le 3^{d+3} D C^{d+1} M^2. \tag{12.14}$$

Multiplying this by the first inequality in (12.12) we get

$$N^2 \le 3^{d+3}(d+1) D C^{d+2} M^{d+2},$$

so that using (12.13) we obtain

$$\begin{aligned} CM &\ge (3^{d+3}(d+1)D)^{-1/(d+2)} N^{2/(d+2)} \\ &\ge 3^{-(d+3)/(d+2)}(d+1)^{-(1+\epsilon)/(d+2)} N^{(2-\epsilon)/(d+2)} \\ &\ge 3^{-4/3} 4^{-2/5} N^{(2-\epsilon)/(d+2)} > \frac{1}{8} N^{(2-\epsilon)/(d+2)}. \end{aligned}$$

This completes the proof of the proposition.

If we do not require that every integer up to N have a multiple in $\mathcal{S}(d, C, M)$, but only that N does, we still have (12.12) holding, which gives $CM \ge (d+1)^{-1/d} N^{1/d} \ge \frac{1}{2} N^{1/d}$. This lower bound for CM is almost achieved in the *special* number field sieve, which accounts for its lower complexity.

12.15. *Practical considerations.* In practical circumstances, when n is fixed rather than tending to infinity, the above argument suggests that our methods for selecting f, m_1,

m_2 are not yet optimal. Namely, suppose that for a given N and d we ignore lower order factors in (12.12) and (12.14) and solve the equations $N = CM^d = C^{d+1}M^2$ for C and M. This suggests we may be able to choose M near N^s and C near N^t where

$$s = \frac{d}{(d-1)(d+2)}, \quad t = \frac{d-2}{(d-1)(d+2)}.$$

In fact, suppose we choose $M = [N^s]$, $C = [N^t]$, so that $CM \leq N^{s+t} = N^{2/(d+2)}$. It is likely that for most integers in $[1, N]$ there is a choice of f, m_1, m_2 satisfying $|c_i| \leq C$ and $|m_i| \leq M$. Indeed the total number of such triples is $(2C+1)^{d+1}(2M+1)^2$, which by our choice of C, M is somewhat larger than N. Also, the typical order of magnitude of $|f(m_1, m_2)|$ is CM^d, which is about N. But if we have a set of a little over N "pseudorandom" numbers of order of magnitude N, then it is quite likely that most integers in $[1, N]$ have a multiple in the set. Thus if we are interested in a particular number $n \leq N$, either this choice of values for C, M or perhaps slightly larger ones should suffice. Note that this imprecise argument is purely existential, and that it does not suggest a way of constructing f, m_1, m_2.

Suppose that n lies in a realistic range, like $n \approx 10^{130}$, and that we take $d = 5$. Then Algorithm 11.1 uses $s = t = \frac{1}{5}$, and therefore m and the coefficients of f each have about 26 digits. In Algorithm 12.2 we have $s = t = \frac{1}{6}$, so m_1, m_2 and the coefficients of f have about 22 digits, which is a significant improvement. The above argument suggests that the optimal values would be $s = 5/28$ and $t = 3/28$, in which case the m_i would have about 23 digits and the coefficients of f about 14 digits. Thus for practical purposes there may still be room for improvement.

12.16. *Additional improvements.* We mention two variations of the number field sieve that improve its practical performance, while not affecting the asymptotic analysis. The first is the double large prime variation, which was used in the factorization of the ninth Fermat number [22]; see also [24]. The second is the lattice sieve idea of Pollard [31].

References

1. L. M. Adleman, *Factoring numbers using singular integers*, Proc. 23rd Ann. ACM Symp. on Theory of Computing (STOC) (1991), 64–71.

2. E. Bach, *Explicit bounds for primality testing and related problems*, Math. Comp. **55** (1990), 355–380.

3. N. Boston, W. Dabrowski, T. Foguel, P. Gies, D. Jackson, J. Leavitt, D. Ose, *The proportion of fixed-point-free elements in a transitive permutation group*, to appear.

4. J. Brillhart, M. Filaseta, A. Odlyzko, *On an irreducibility theorem of A. Cohn*, Can. J. Math. **33** (1981), 1055–1059.

5. J. Brillhart, D. H. Lehmer, J. L. Selfridge, B. Tuckerman, S. S. Wagstaff, Jr., *Factorizations of* $b^n \pm 1$, $b = 2, 3, 5, 6, 7, 10, 11, 12$ *up to high powers*, second edition, Contemporary Mathematics **22**, Amer. Math. Soc., Providence, 1988.

6. J. A. Buchmann, H. W. Lenstra, Jr., *Decomposing primes in number fields*, in preparation.

7. E. R. Canfield, P. Erdős, C. Pomerance, *On a problem of Oppenheim concerning "factorisatio numerorum"*, J. Number Theory **17** (1983), 1–28.

8. J. W. S. Cassels, A. Fröhlich (eds), *Algebraic number theory*, Proceedings of an instructional conference, Academic Press, London, 1967.

9. A. M. Cohen, *On the number of fixed point free elements in a permutation group*, to appear.

10. D. Coppersmith, *Modifications to the number field sieve*, IBM Research Report #RC 16264, Yorktown Heights, New York, 1990.

11. J. D. Dixon, *Asymptotically fast factorization of integers*, Math. Comp. **36** (1981), 255–260.

12. W. Fulton, *Intersection theory*, Springer-Verlag, Berlin. 1984.

13. P. X. Gallagher, *The large sieve and probabilistic Galois theory*, in: H. G. Diamond (ed.), *Analytic number theory*, Proc. Symp. Pure Math. **24**, Amer. Math. Soc., Providence, 1973, 91–101.

14. D. Gordon, *Discrete logarithms in* GF(p) *using the number field sieve*, SIAM J. Discrete Math., to appear.

15. B. Huppert, *Endliche Gruppen I*, Springer-Verlag, Berlin, 1967.

16. D. E. Knuth, *The art of computer programming*, volume 2, second edition, Addison-Wesley, Reading, Mass., 1981.

17. S. Landau, *Factoring polynomials over algebraic number fields*, SIAM J. Comput. **14** (1985), 184–195.

18. S. Lang, *Algebraic number theory*, Addison-Wesley, Reading, Mass., 1970.

19. A. K. Lenstra, *Factorization of polynomials*, in [28], 169–198.

20. A. K. Lenstra, *Factoring polynomials over algebraic number fields*, in: J. A. van Hulzen (ed.), *Computer algebra*, Lecture Notes in Comput. Sci. **162**, Springer-Verlag, Berlin, 1983, 245–254.

21. A. K. Lenstra, H. W. Lenstra, Jr., L. Lovász, *Factoring polynomials with rational coefficients*, Math. Ann. **261** (1982), 515–534.

22. A. K. Lenstra, H. W. Lenstra, Jr., M. S. Manasse, J. M. Pollard, *The factorization of the ninth Fermat number*, in preparation.

23. A. K. Lenstra, H. W. Lenstra, Jr., M. S. Manasse, J. M. Pollard, *The number field sieve*, in preparation. Extended abstract: Proc. 22nd Ann. ACM Symp. on Theory of Computing (STOC) (1990), 564–572.

24. A. K. Lenstra, M. S. Manasse, *Factoring with two large primes*, in preparation.

25. H. W. Lenstra, Jr., *Factoring integers with elliptic curves*, Ann. of Math. **126** (1987), 649–673.

26. H. W. Lenstra, Jr., *Algorithms in algebraic number theory*, Bull. Amer. Math. Soc. **26** (1992), 211–244.

27. H. W. Lenstra, Jr., C. Pomerance, *A rigorous time bound for factoring integers*, J. Amer. Math. Soc., to appear.

28. H. W. Lenstra, Jr., R. Tijdeman (eds), *Computational methods in number theory*, Mathematical Centre Tracts **154/155**, Mathematisch Centrum, Amsterdam, 1982.

29. M. A. Morrison, J. Brillhart, *A method of factoring and the factorization of F_7*, Math. Comp. **29** (1975), 183–205.

30. J. M. Pollard, *Factoring with cubic integers*, unpublished manuscript, August 1988.

31. J. M. Pollard, *The lattice sieve*, unpublished manuscript, September 1991.

32. C. Pomerance, *Analysis and comparison of some integer factoring algorithms*, in [28], 89–139.

33. C. Pomerance, *Fast, rigorous factorization and discrete logarithm algorithms*, in: D. S. Johnson, T. Nishizeki, A. Nozaki, H. S. Wilf (eds), *Discrete algorithms and complexity*, Academic Press, Orlando, 1987, 119–143.

34. O. Schirokauer, *On pro-finite groups and on discrete logarithms*, Ph. D. thesis, University of California, Berkeley, May 1992.

35. B. Vallée, *Generation of elements with small modular squares and provably fast integer factoring algorithms*, Math. Comp. **56** (1991), 823–849.

36. B. L. van der Waerden, *Algebra*, seventh edition, Springer-Verlag, Berlin, 1966.

37. P. S. Wang, *Factoring multivariate polynomials over algebraic number fields*, Math. Comp. **30** (1976), 324–336.

38. P. J. Weinberger, L. P. Rothschild, *Factoring polynomials over algebraic number fields*, ACM Trans. Math. Software **2** (1976), 335–350.

39. E. Weiss, *Algebraic number theory*, McGraw-Hill, New York, 1963; reprinted, Chelsea, New York, 1976.

40. D. Wiedemann, *Solving sparse linear equations over finite fields*, IEEE Trans. Inform. Theory **32** (1986), 54–62.

Department of Mathematics, Reed College, Portland, OR 97202
E-mail: jpb@reed.edu

Department of Mathematics, University of California, Berkeley, CA 94720
E-mail: hwl@math.berkeley.edu

Department of Mathematics, University of Georgia, Athens, GA 30602
E-mail: carl@ada.math.uga.edu