FACTORING WITH CUBIC INTEGERS

J. M. Pollard

SUMMARY. We describe an experimental factoring method for numbers of form $x^3 + k$; at present we have used only $k = 2$. The method is the cubic version of the idea given by Coppersmith, Odlyzko and Schroeppel (Algorithmica 1 (1986), 1–15), in their section 'Gaussian integers'. We look for pairs of small coprime integers a and b such that:

i. the integer $a + bx$ is smooth,

ii. the algebraic integer $a + bz$ is smooth, where $z^3 = -k$. This is the same as asking that its norm, the integer $a^3 - kb^3$ shall be smooth (at least, it is when $k = 2$).

We used the method to repeat the factorisation of F_7 on an 8-bit computer ($2F_7 = x^3 + 2$, where $x = 2^{43}$).

Introduction

We consider the case $k = 2$ throughout. We denote by $\mathbf{Z}$ the set of rational integers (ordinary integers) and by S the set of algebraic integers:

$$[a, b, c] = a + bz + cz^2, \qquad (a, b, c \text{ in } \mathbf{Z}).$$

These constitute the algebraic integers of the field generated by z, and possess the property of unique factorisation (neither statement true for general k, see e. g. [2]). According to [1], such methods are still possible when unique factorisation fails.

We also write:

$$\{a, b, c\} = a + bx + cx^2.$$

When ii. holds, we have some factorisation:

$$[a, b, 0] = [d, e, f] \cdot \ldots,$$

into units and primes of S (defined shortly). Then also:

$$\{a, b, 0\} \equiv \{d, e, f\} \cdot \ldots \pmod{n}.$$

But by i. we have also a factorisation:

$$\{a, b, 0\} = p \cdot q \cdot \ldots,$$

into small primes of $\mathbf{Z}$. So we have a congruence (mod n) involving rational integers from two small sets. From a sufficient number of such congruences, we obtain some equations:

$$X^2 \equiv Y^2 \pmod{n},$$

and hopefully the factorisation of n.

1991 *Mathematics Subject Classification.* Primary 11Y05, 11Y40.

Key words and phrases. Factoring algorithm, algebraic number fields.

Properties of the set S

The norm of a member $[a, b, c]$ of S is the rational integer:

$$N(a, b, c) = a^3 - 2b^3 + 4c^3 + 6abc.$$

This is a multiplicative function, i. e. the equation

$$[a, b, c] = [d, e, f] \cdot [g, h, i] \tag{1}$$

implies:

$$N(a, b, c) = N(d, e, f) \cdot N(g, h, i).$$

Given an equation (1), we say that $[d, e, f]$ divides $[a, b, c]$.

The norm can be zero only when $a = b = c = 0$. Numbers with norm $+1$ or -1 are called *units*. There are an infinity of units, namely all the numbers:

$$\pm U^i \quad (i = 0, \pm 1, \pm 2, \ldots),$$

where $U = [1, 1, 0]$. We give a table of the small powers of U:

i	U^i	U^{-i}
0	$[\ 1, 0, 0]$	$[\ 1, 0, 0]$
1	$[\ 1, 1, 0]$	$[-1, 1, -1]$
2	$[\ 1, 2, 1]$	$[\ 5, -4, 3]$
3	$[-1, 3, 3]$	$[-19, 15, -12]$
4	$[-7, 2, 6]$	$[\ 73, -58, 46]$

A unit divides any integer. If $[d, e, f]$ in (1) is a unit, then the other two numbers are termed *associates*; clearly this means that:

$$N(a, b, c) = \pm N(g, h, i),$$

but the converse statement is false as we shall see.

A number $[a, b, c]$ is termed *prime* if any factorisation (1) contains a unit (and an associate). A number of norm $\pm p$ (p prime) is certainly a prime; but not all primes are of this form.

A rational prime p need not be a prime of S. We have $N(p, 0, 0) = p^3$, so perhaps p can have prime factors of norm $\pm p$ or $\pm p^2$. Indeed it can. There are four cases (see [2, p. 186]):

1. The primes $p = 2$ and 3. These factor as a unit and the cube of a prime of norm p:

$$\begin{aligned} 2 &= \quad -1 \quad \cdot [\ 0, 1, 0]^3, \\ 3 &= [1, 1, 0] \cdot [-1, 1, 0]^3. \end{aligned}$$

2. Primes p of form $6m + 1$, with -2 a cubic residue (mod p):

$$p = 31,\ 43,\ 109,\ 127,\ 157,\ \ldots.$$

There are three nonassociated factors of norm p. For example:

$$31 = [5, -4, 3] \cdot [-1, -2, 1] \cdot [-9, -6, 1] \cdot [3, 0, 1].$$

(The first factor on the right is a unit.)

3. Primes of form $p = 6m + 5$:

$$p = 5,\ 11,\ 17,\ 23,\ 29,\ \ldots.$$

There is one factor of norm p and one of norm p^2. For example:

$$5 = [1, 0, 1] \cdot [1, -2, -1].$$

4. Primes p of form $6m + 1$, with -2 a cubic nonresidue (mod p):

$$p = 7,\ 13,\ 19,\ 37,\ \ldots.$$

There is no factorisation: p is a prime of norm p^3.

Operations on the integers of S

It is easy to add, subtract and multiply numbers $[a, b, c]$; when multiplying, we use $z^3 = -2$, to remove the cube and fourth power terms in z. As for division, we do not need a Euclidean algorithm (does it exist?), but only to test whether:

$$[d, e, f] | [a, b, c]? \qquad (2)$$

and, if so, to find the quotient, $[g, h, i]$ in (1).

We shall use:

$$N(a, b, c) = [a, b, c] \cdot C(a, b, c),$$

where

$$C(a, b, c) = [a^2 + 2bc, -ab - 2c^2, b^2 - ac].$$

This can be easily verified; in fact C is the product of the conjugates of $[a, b, c]$, obtained by replacing z by the other two cube roots of -2. To test (2) we multiply both sides by $C(d, e, f)$; this gives the question

$$N(d, e, f) | [A, B, C]?$$

This holds when each of A, B and C is divisible by the integer $N = N(d, e, f)$: if so the required quotient is $[A/N, B/N, C/N]$.

Factorisation of F_7

An obvious test case for our method is $F_7 = 2^{128} + 1$, first factored by Brillhart and Morrison in 1970 [3]. We have:

$$2F_7 = x^3 + 2, \qquad \text{where } x = 2^{43}.$$

We describe the method in detail for this number.

Step 1. Compute the factor base.

The first part of the factor base, FB_1, consists of the first 500 primes:

$$2, 3, 5, \ldots, 3571.$$

It is also convenient to compute $x \pmod p$ for each such prime.

The second part FB_2 consists of primes of S arising from the factorisations of the rational primes p of FB_1. Only primes of norm $\pm p$ are used. Those of norm p^2 in case 3 cannot divide numbers $[a, b, 0]$, $\gcd(a, b) = 1$, and are not needed.

Cases	Times	Primes
1	2	2
2	81	243
3	252	252
4	165	0
Totals	500	497

We also included in FB_2 three units: -1, U and $1/U$, making 500 members in all, like FB_1 (a coincidence!). The choice of FB_2 was dictated by convenience, and is larger than necessary. We have at once 81 equations connecting FB_1 and FB_2, and one more involving the units.

Step 2. Run the sieve.

We want to find numbers $a + bx$ composed of the primes of FB_1, except perhaps for one larger prime. Our program is like that for the Quadratic Sieve of Pomerance [4], but simpler; b is held constant while a varies over an interval of width up to 12,000. Only coprime pairs (a, b) are saved.

$$\begin{aligned} \text{Range for } b &= 1 \ldots 2000 \\ \text{Range for } a &= -4800 \ldots 4800 \\ \text{Limit for large prime} &= 10{,}000 \\ \text{Integers sieved} &= 1.92(7) \\ \text{Successes} &= 40{,}762 \end{aligned}$$

Note:

1. My sieve represents an integer m by the nearest integer to $2\log_2(m)$. This means that the limit on the large prime is very rough.

2. We do not know, at this stage, how many of these successes involve a large prime.

Step 2A. Look for smooth values of the norm.

For each pair, we compute the norm:

$$N = N(a,b,0) = a^3 - 2b^3.$$

We factor by trial division, using only the norms of primes of FB_2 (there are 335 distinct values). When N factors completely over these primes (no 'large' prime allowed here), compute the large prime q, if any, implicit in Step 2.

$$\begin{aligned} \text{Number of } A\text{-solutions (no large prime)} &= \ \ 538 \\ \text{Number of } B\text{-solutions (large prime)} &= 1133 \end{aligned}$$

Step 3. Pair the large primes.

When the large prime q arises $l > 2$ times, we count $l - 1$ pairs.

$$\begin{aligned} \text{Number of } A\text{-solutions} &= \ \ 538 \\ \text{Pairs of } B\text{-solutions} &= \ \ 399 \\ \text{Equations already known} &= \ \ \ \ 81 \\ \text{Total} &= 1018 \end{aligned}$$

Step 4. Obtain factorisations of $a + bx$, $a + bz$.

For those solutions to be used, whether A or B, obtain the complete factorisations of $a+bx$ by trial division (by working (mod p), we eliminate unsuccessful trials).

Factorising the numbers $[a,b,0]$ is slightly more complicated. Again compute the norm $N(a,b,0)$ and factor by trial division. When a prime p is found, divide out a prime of norm $\pm p$ from the number $[a,b,0]$ (we may need to try up to three such primes). We should finish with a unit, $[d,e,f]$ say. From a table of powers of U and $1/U$, we can recognise it as $\pm U^i$.

Two questions arise here:

1. How large a table is needed? I took $i = -8 \ldots 8$, the largest I could compute in single length arithmetic (32 bits).
2. Perhaps this process could still cause an overflow? But it didn't. Numbers with small norm *can* have large coefficients (even units).

Step 5. Obtain linearly dependent sets.

Just as in QS (or CF). My QS program needed only trivial changes. 31 sets were obtained.

Step 6. Complete the factorisation.

Again just as in QS. Add up the powers of each member of FB_1 and FB_2 in all the solutions in the set (take the quotient of each pair of B-solutions). The totals should be even! Replacing each number $[a,b,c]$ by $\{a,b,c\}$, compute integers X and Y with

$$X^2 \equiv Y^2 \pmod{F_7}.$$

Compute

$$\gcd(X - Y, F_7) = 59\,64958\,91274\,97217 \qquad \text{(1st set!)}.$$

This agrees with [3], and with the well-known mnemonic for Fermat factors.

Computer and program details

The 8-bit computer used is the Philips P2012, with 64k of store, and two disc drives (640k each). There are seven separate programs, one for each step described above. The programs are in Pascal, with a small amount of machine code (for multi-length arithmetic and logical operations). My programs have much in common with those written for QS (only six!), with which I have factored numbers up to 51 digits.

Program 1. Single length working. Rational primes are generated by a sieve. My method to find their factors in S is crude. Generate all numbers $[a, b, c]$ of S with small coefficients (I used $-15 \dots 15$), saving those with norm ± 1 or $\pm p$. By experiment, find a fundamental unit U. Sort the others on p and remove associates to obtain FB_2.

Program 2. Similar to QS, e. g. [4]. I use a sieve array of 8 bit elements. Multilength arithmetic not essential (only used if exact values of large primes are to be found at once).

Program 2A. Requires multilength working for $a+bx$ and double length working for the values of the norm. Inefficient on an 8-bit computer without division—unlike Program 2, which doesn't need them. Probably capable of improvement—but the author prefers to keep to Pascal if at all possible!

Program 3. Simple sorting program using Treesort3, CACM algorithm 245.

Program 4. Requires multilength arithmetic.

Program 5. Requires bit operations. Binary matrix is processed in a series of passes between two disc files (largest matrix used so far (for QS) was $\sim$ 1700 rows).

Program 6. Requires multilength arithmetic and bit operations.

The running times quoted below are very rough since the programs have been run over a period of some weeks, and in some cases have gone through several versions.

Program	Time (hours)
1	1
2	5
2A	10
3	0.5
4	1.5
5	2
6	0.1
Total	20.1

Example of an A-solution

$$
\begin{aligned}
x &= 2^{43}, \qquad a = 1693, \qquad b = 749. \\
a + bx &= 6\,58827\,36736\,35485 \\
&= 5 \cdot 13 \cdot 19 \cdot 449 \cdot 1567 \cdot 2477 \cdot 3061 \\
a^3 - 2b^3 &= 40121\,80059 \\
&= -1 \cdot -3 \cdot -43 \cdot 157 \cdot -397 \cdot 499 \\
[a, b, 0] &= [-1, -2, -1] \cdot [-1, 1, 0] \cdot [-3, 2, 0] \cdot [5, 0, 2] \cdot [-7, 3, 0] \cdot [5, 1, 4]
\end{aligned}
$$

The first factor on the right is $-U^2$; the others are primes (written as in FB_2, with highest nonzero coefficient positive).

References

1. D. Coppersmith, A. M. Odlyzko, R. Schroeppel, *Discrete logarithms in $GF(p)$*, Algorithmica **1** (1986), 1–15.
2. I.N. Stewart, D.O. Tall, *Algebraic number theory*, second edition, Chapman and Hall, London, 1987.
3. M. Morrison, J. Brillhart, *A method of factoring and the factorization of F_7*, Math. Comp. **29** (1975), 183–205.
4. J.L. Gerver, *Factoring large numbers with a quadratic sieve*, Math. Comp. **41** (1983), 287–294.

Tidmarsh Cottage, Manor Farm Lane, Tidmarsh, Reading, Berkshire, RG8 8EX, England