数体ふるい法による素因数分解[1]

立教大学理学部　木田 祐司

数体ふるい法 (Number Field Sieve method) は Pollard が基本的アイディアを出してから、今年で満10年を迎える。

この間、特殊な数にだけ適用可能だったものが一般の合成数にまで適用できるように発展した。この講演では新しいアイディアの発掘を目指して、素因数分解のアルゴリズムの勘所を解説する。

[1]この小文は 1998年1月7日に東京大学で行われた研究集会「代数幾何・数論及び符号・暗号」のOHP原稿に加筆修正したものである。
この研究は 1997年度立教大学研究奨励助成金 の援助を受けた。
第1.0版 98/3/14
第1.1版 98/3/18
第1.2版 98/3/29
第1.3版 98/8/16
第1.4版 99/7/28
第1.5版 00/2/4(2/28)
第1.6版 00/4/6
第1.7版 01/3/21
第1.8版 02/1/16
第1.9版 02/1/24
第1.10版 03/6/25
第1.11版 03/12/15

1　素因数分解の発展

以下には素因数分解が何桁までできるかの表を掲げる。

たとえば **1999** のところは 1999 年には GNFS 法で 155 桁の合成数が素因数分解されたことを表している。この方法の特質としてどんな 155 桁の数もほぼ同じ手間で分解される。[1990B][1997B] などは分解する数の特質を利用しているため同じ桁数の他の数には適用できない。

1983 50 桁 by CFRAC(continued fraction), QS(quadratic sieve)

1984 60 桁 by QS

1985 70 桁 by MPQS(multiple polynomial QS)

1986 80 桁　”

1987 90 桁　”

1988 102 桁　”

1990 116 桁 by MPQS

1990B 148(155) 桁 by NFS(number field sieve)
$F_9 = 2^{2^9} + 1 = \mathrm{P7} \times \mathrm{C148} = \mathrm{P7} \times \mathrm{P49} \times \mathrm{P99}$

1994 129 桁 (RSA129) by MPQS

1996 130 桁 (RSA130) by GNFS(general NFS)

1999 140 桁 (RSA140) by GNFS
155 桁 (512bit)(RSA155) by GNFS

1999B 211 桁 by NFS(not General)　$(10^{211} - 1)/9$

2000B 233 桁 by NFS(not General)　$(2^{773} + 1)/3$

2002 158 桁 (a factor of $2^{953} + 1$) by GNFS

2003B 244 桁 (a factor of $2^{809} - 1$) by NFS(not General)

2003 160 桁 (RSA160) by GNFS

2003 174桁(RSA576) by GNFS

素因数分解の限界のもうひとつの見方として 全体の桁数に制約されない方法 を用いて「何桁の素因数が見つかったか」という方向もある。これは 2003年に 58桁の素因数が Roger Backstrom によってECM(Elliptic Curve Method) で見つけられたのが最高である。

2　不思議な方法　ρ 法

群論系にもふるい系にも属さない。説明に困る方法に ρ 法がある。乱数（らしきもの）を用いるのでモンテカルロ法とも呼ばれる。

一般に、与えられた合成数の素因数分解をするには、5桁の素因数までは試し割り算で行い、つぎに ρ 法を適用するのが定石である。10桁 までの素因数は容易に見つけることができる。

3　群論的方法(その１) $p-1$ 法

mod p の既約剰余類群を利用。今では、特殊な形の数であることが分かっている場合以外は用いられない。次の楕円曲線法の原理の説明用としての意義しかなくなってしまった。

4　群論的方法(その２) 楕円曲線法

mod p の楕円曲線の有理点の群を利用。20桁までの素因子は比較的楽に見つかるであろう。理論的にもっと研究の余地があると思われる。

5　平方差法の原理

これから最後までは『ふるい(sieve)』系の素因数分解法の解説である。

Carl Pomerance : A Tale of Two Sieves, Notices of AMS, Dec. 1996 のイントロを拝借する。

Pomerance は high school 時代に出た数学コンテストで次の問題に遭遇したという：

A Contest Problem　8051 を素因数分解せよ。

An Answer

$$8051 = 8100 - 49 = 90^2 - 7^2$$

に気がつけば

$$8051 = (90 - 7)(90 + 7) = 83 \cdot 97$$

という正解を得る。これより n の素因数分解には

$$x^2 - n = y^2$$

となる x, y を探すことになる。

x を $\sqrt{n}$ の近辺を動かして捜す。

$n = 8059$ の場合は $x = 90,\ 90 \pm 1,\ 90 \pm 2,\ 90 \pm 3,\ \ldots$ として $Q(x) = x^2 - n$ が平方数になるものを探すと、幸い最初から

$$90^2 - n = 7^2$$

となるということである。これを Fermat の difference-of-squares 法という。

基本原理　一般に n の素因数分解には

$$x^2 \equiv y^2 \bmod n$$

となる x, y を見つければ良い。

もちろん最初から $x \equiv \pm y$ とすればこの合同式は得られるが無意味である。

$$x_i \equiv y_i \bmod n \quad \text{かつ} \quad x_i \neq y_i$$

となる x_i, y_i を集めて組み合わせて平方数を作るのである。 mod n での代表元を異なるようにとってその分解の違いを利用する。

例

$$\begin{aligned} 14 \cdot 67 &\equiv 3 \bmod 187 \\ 31 \cdot 67 &\equiv 20 \bmod 187 \\ 14 \cdot 31 &\equiv 60 \bmod 187 \end{aligned}$$

より

$$(14 \cdot 31 \cdot 67)^2 \equiv 60^2 \bmod 187$$

となる。

代表元を変えるというのは「加法的」な性質、素因数分解は「乗法的」な性質、それらをブレンドさせるところがポイントと言えるかもしれない。

ポイント　mod n での代表元を二通りにとってその素因数分解の違いを利用する。

問題　一般にたくさんの整数 $r_1, r_2, \ldots, r_s$ が与えられたとき、その中から組み合わせて平方数になるものを見つけるにはどうするか。

• まず適当な個数の素数の集合 $\{p_1, p_2, \ldots, p_B\}$ を決める。factor base という。

• 次にその factor base で完全に素因数分解できてしまうもの、smooth な数という、つまり

$$r_i = \prod_{j=1}^{B} p_j^{e(i,j)}$$

となる r_i を $B+1$ 個以上捜す。これらをあらためて $r_0, r_1, \ldots, r_B$ としよう。

• これらのべき指数部分に注目し、さらにそれを $\bmod 2$ で考えて次のような($\mathbf{Z}/2\mathbf{Z}$ 上の) B 次元ベクトルにする。

$$v_i = (e(i,1) \bmod 2,\ e(i,2) \bmod 2,\ \ldots,\ e(i,B) \bmod 2)$$

• 自明でない関係式を求める。

$$c_0 v_0 + c_1 v_1 + \cdots + c_B v_B \equiv 0 \bmod 2$$

• 元に戻して

$$r_0^{c_0} \cdot r_1^{c_1} \cdots r_B^{c_B} \text{ は平方数}$$

ポイント　$r_1, r_2, \ldots, r_s$ の絶対値が小さいほど良い。

6　直接法

アイディア　分解したい数 n に対して $1, 2, 3, \ldots$ に n を加えて代表元を変える。

例　$n = 703$ の場合

$$\begin{aligned}
1 &\equiv 704 = 2^6 \cdot 11 \\
2 &\equiv 705 = 3 \cdot 5 \cdot 47 \\
3 &\equiv 706 = 2 \cdot 353 \\
4 &\equiv 707 = 7 \cdot 101 \\
5 &\equiv 708 = 2^2 \cdot 3 \cdot 59 \\
6 &\equiv 709 = 709 \\
7 &\equiv 710 = 2 \cdot 5 \cdot 71 \\
8 &\equiv 711 = 3^2 \cdot 79 \\
9 &\equiv 712 = 2^3 \cdot 89 \\
10 &\equiv 713 = 23 \cdot 31
\end{aligned}$$

こういう違いが出る。両辺を組み合わせて平方数を作りたいがこの範囲では不可能である。

ポイント　smooth になってほしい数は n 程度の大きさ。

7　定数倍法

アイディア　分解したい数 n に対して $n/2$ の近くの数を 2 倍して n を引けば代表元が "ずれる"。

例　$n = 253$ の場合

$$\begin{array}{lll}
(1) & 2 \cdot 127 & = 2 \cdot 127 \quad \equiv 1 \\
(2) & 2 \cdot 128 & = 2^8 \quad \equiv 3 \\
(3) & 2 \cdot 129 & = 2 \cdot 3 \cdot 43 \quad \equiv 5 \\
(4) & 2 \cdot 130 & = 2^2 \cdot 5 \cdot 13 \quad \equiv 7 \\
(5) & 2 \cdot 131 & = 2 \cdot 131 \quad \equiv 9 \\
(6) & 2 \cdot 132 & = 2^3 \cdot 11 \quad \equiv 11 \\
(7) & 2 \cdot 133 & = 2 \cdot 7 \cdot 19 \quad \equiv 13 \\
(8) & 2 \cdot 134 & = 2 \cdot 2 \cdot 67 \quad \equiv 15 \\
(9) & 2 \cdot 135 & = 2 \cdot 3^3 \cdot 5 \quad \equiv 17 \\
(10) & 2 \cdot 136 & = 2^4 \cdot 17 \quad \equiv 19
\end{array}$$

これより Gauss 消去をして (2),(4),(7),(9),(10) を組み合わせると良いことが分かる。実際

$$2^8\ 2^2\ 5\ 13\ 2\ 7\ 19\ 2\ 3^3\ 5\ 2^4\ 17 \equiv 3\ 7\ 17\ 192^{16}\ 3^3\ 5^2\ 7\ 17\ 19 \equiv 3\ 7\ 17\ 19$$

より

$$2^{16} \cdot 3^2 \cdot 5^2 = (2^8 \cdot 3 \cdot 5)^2 \equiv 1$$

これから次を得る。

$$\mathrm{GCD}(2^8 \cdot 3 \cdot 5 - 1, 253) = 11$$

ポイント　smooth になってほしい数は $n/2$ 程度の大きさ。

8　2次ふるい法

アイディア　分解したい奇数 n に対して $\sqrt{n}$ の近くの数を 2 乗して n を引けば代表元が "ずれる"。

$$x = [\sqrt{n}] \pm m,\ m = 1, 2, \ldots$$

に対して

$$Q(x) = x^2 - n \sim \pm 2m\sqrt{n}$$

を考えるということ。

$n = 3937$ の場合

$$\begin{aligned}
63^2 - n &= 32 &&= 2^5 \\
64^2 - n &= 159 &&= 3 \cdot 53 \\
65^2 - n &= 288 &&= 2^5 \cdot 3^2 \\
66^2 - n &= 419 &&= 419 \\
67^2 - n &= 552 &&= 2^3 \cdot 3 \cdot 23 \\
&\cdots
\end{aligned}$$

の 63, 65 を組み合わせて $(63 \cdot 65)^2 \equiv (2^5 \cdot 3)^2 \bmod n$ を得る。これより $\mathrm{GCD}(63 \cdot 65 - 2^5 \cdot 3,\ n) = 31$。

ポイント　smooth になってほしい数は $H\sqrt{n}$ 程度の大きさ。ただし m を $\pm H$ の範囲で動かすとする。

9　数体ふるい法

アイディア　代数方程式の複素数体での解を θ、 mod n での解を M、とすると θ と M の式は mod n で異なる代表元を与える。

正確に言うと、多項式 $f(X)$ に対して

$$\begin{cases} f(\theta) &= 0 & ,\theta \in \mathbf{C} \\ f(M) &\equiv 0 \bmod n & ,M \in \mathbf{Z} \end{cases}$$

のとき、次の ϕ は環準同型である

$$\begin{array}{rccl} \phi: & \mathbf{Z}[\theta] & \longrightarrow & \mathbf{Z}/n\mathbf{Z} \\ & g(\theta) & \mapsto & g(M) \bmod n \end{array}$$

これによって次の図式を2通りに進んで、その分解の違いを利用する。この図式は $\mathbf{Z}/n\mathbf{Z}$ で考えれば可換だが、$\mathbf{Z}$ では可換とは限らない。

$$\begin{array}{ccc} a+b\theta & \longrightarrow & \mathbf{Z}[\theta] \text{での分解} \\ \phi\downarrow & & \downarrow\phi \\ a+bM & \longrightarrow & \mathbf{Z} \text{での分解} \end{array}$$

$a+b\theta$ と $a+bM$ が同時に smooth な a,b のペアがほしい。$a+b\theta$ の smoothness は $\mathcal{N}(a+b\theta)$ のsmoothness と同じである。

ポイント　smooth になってほしい数は $f(x)$ の次数を d、係数の最大値を s、変化させる a,b の最大値を H とすると

- $a+bM$ は $n^{1/d}H$ 程度、
- $\mathcal{N}(a+b\theta)$ は $s \cdot H^d$ 程度

の大きさである。d の増加に対して逆向きに反応する。

例

$$\begin{cases} f(X) &= X^5+X^4-4X^3-3X^2+3X+1 \\ M &= 8706988105675235069106 \\ n &= f(M) \\ &= 500425873512465873676795340482322031719914256320239965518193747923744495655033466611365861148377775959 \\ & \quad 5958157019 \quad (110\text{digit}) \end{cases}$$

とし、

$\mathbf{Z}$ での factor base(RFB) の個数 $=$ 68100
$\mathbf{Z}[\theta]$ での factor base(AFB) の個数 $=$ 68171

とし

$$-210944 \le a < 210944, \quad 1 \le b \le 126677$$

で動かして十分なだけの smooth pair を探す。

つまり、b を固定して a を動かし、$a+bM$ を $p \in$ RFB で分解する。同時に $a+b\theta$ を $P \in$ AFB で分解する。このとき実際の割り算をする前に "エラトステネスのふるい" と同様な "ふるい" によって候補を絞り込むので "数体ふるい" という名がある。

実際にこの範囲で必要なだけの smooth なデータが得られた。ただし半数をやや超えるだけを複数個の組み合わせで得ている。また一割は free relation と呼ばれるものにより最初から分かっている smooth data である。

あとはべき指数の行列を作って関係式を探すことになる。

ではどの程度の大きさの数を調べているのだろうか。概算してみよう。

- $a+bM$ の大きさは平均して $126677 \cdot M/2 \sim n^{0.24}$ であり、
- $\mathcal{N}(a+b\theta)$ の大きさは平均 $(210944/2)^5 \sim n^{0.23}$ である。
- 合わせて $n^{0.47}$ 以下の大きさの数を調べていることに相当する。

2次ふるいではおよそ $2^{19}n^{0.5} \sim n^{0.52}$ 程度の大きさの数を調べることになる。実際に計算機を動かすと数体ふるい法では19時間で可能であった（ふるいの時間のみ）が、2次ふるい法では過去の実験から約1年強かかると推定される。

ただし、ここまで速いのは **多項式の係数が1桁** という特殊な事情による。上の評価において係数が6桁になればもう優位性はなくなる。

$f(X) = X^5 + X^4 - 4X^3 - 3X^2 + 3X + 1$ による 特殊数体ふるい法の実行時間　on PentiumII 300MHz							
桁数	60桁	70桁	80桁	90桁	100桁	110桁	120桁
#FB	4K*2	7K*2	14K*2	24K*2	40K*2	67K*2	106K*2
Ha	12K	24K	40K	68K	128K	206K	315K
Hb	7K	13K	22K	38K	64K	124K	194K
時間	4分	12分	38分	2.5時間	7.5時間	19時間	50時間

注意　ここでの実行時間はふるいの部分だけのものである。

特殊な形をしていない数にも適用できるように改良したものが後述の一般数体ふるい法であるが、これよりはもっとずっと遅くなる。

補足　この数の特殊性は三つある。

・$f(x)$ の係数の絶対値の最大が 4 に過ぎない。一般数体ふるい法では n の 5乗根ないし 6乗根 程度の大きさになる。

・$f(x) = 0$ の解 5個すべてが実数解である。

・$f(x)$ のガロア群は位数 5 の巡回群である。そのため free relations と呼ばれる特殊な関係式が代数的 factor base の個数の 1/5 も存在する。多くの場合ガロア群は5次対称群であり、free relations は 1/120 しか存在しない。

補足　110 桁の数の分解にかかった全時間を記しておく。CPU は PentiumII-300MHz(1 台) である。ここで用いる整数環は UFD であるが、このプログラムではそのことは用いていない。そのため最後の step での平方根の計算に比較的時間がかかっている。

```
1. Initialize
f(X) = X^5 + X^4 - 4*X^3 - 3*X^2 + 3*X + 1
M    = 8706988105675235069106
n = f(M) = 500425873512465873676795340482322031719914256320239965518193747923744495655033466611365861148377775959581570 19
#RFB = 68100  #AFB = 68171  #Qsymbol = 81
Ha = 210944  Hb = 126677
free relations = 13634

2. Do sieving                                  18h 47m 27s
 FF =  74282
 PF = 429036
 FP = 132479
 PP = 770717

3. Process large primes of FP, PF and PP       10m 21s
 new FF = 42298 from PF and PP
 new FF = 10401 from FP and PP
 total 140615 FF (= 13634+74282+42298+10401 >= 136352)

3' Make matrix data                             2h 35m 59s

4. Reduce to a smaller matrix                   2h 03m 56s
 full size = 136352  -> reduced size = 19936

5. Gaussian elimination                         1h 12m 12s
  655 relations

6. Get primes for square root                    1m 41s
  40906 primes

7. Get square root and result                   57m 51s
n = 22847118173403112372383046530687657066241905721903238286525949739448686039184398564209032992276528007677267866 7
```

10　数体ふるい法の発展

ここからは数体ふるい法がどう発展してきているかを述べる。

復習

$$\begin{cases} f(M) & \equiv & 0 \bmod n \\ f(\theta) & = & 0 \end{cases}$$

を考え、$K = \mathbf{Q}(\theta)$ とする。

オリジナルな数体ふるい法では仮定として次を置いた。

- $\mathbf{Z}[\theta]$ は UFD(素元分解環) である。

n の大きさに応じて $\mathbf{Z}$ の素数の集合 F と O_K の素元と単数の集合 G を適当にとる。

a, b を互いに素な有理整数で $a + bM$ は F-smooth とする。つまり

$$a + bM = \prod_{p \in F} p^{e(p)}$$

であり、かつ $a + b\theta$ は G-smooth とする。

$$a + b\theta = \prod_{\pi \in G} \pi^{e(\pi)}$$

と完全に素元分解されるものとする。

このようなペア (a, b) を $\#F + \#G$ 個より多く集める。すると各べき指数を mod 2 で考えたベクトル

$$\left(\ (e(p) \bmod 2)_{p \in F}, \ (e(\pi) \bmod 2)_{\pi \in G} \ \right)$$

は線形従属となり、適当に組み合せると、つぎのような関係式が出る。

$$\prod(a_i + b_i M) = \left(\prod_{p \in F} p^{e(p)}\right)^2$$

と $\mathbf{Z}$ で平方数になり、かつ

$$\prod(a_i + b_i \theta) = \left(\prod_{\pi \in G} \pi^{e(\pi)}\right)^2$$

と O_K で平方数になる。そこで

$$\prod(a_i + b_i \theta) = h(\theta)^2$$

であったとしてこれを準同型

$$\begin{aligned} \phi : \mathbf{Z}[\theta] &\longrightarrow \mathbf{Z}/N\mathbf{Z} \\ g(\theta) &\mapsto g(M) \bmod n \end{aligned}$$

で写せば

$$\left(\prod_{p \in F} p^{e(p)}\right)^2 \equiv h(M)^2 \bmod n$$

となり、ふるい法の目標である $x^2 \equiv y^2 \bmod n$ 型の合同式が得られる。

問題点　$\mathbf{Z}[\theta]$ が UFD でない場合はどうするか。

11　一般数体ふるい法

$\mathbf{Z}[\theta]$ が UFD でない場合も扱おうというのが General NFS である。

ただし以下の説明では簡単のため $O_K = \mathbf{Z}[\theta]$ は仮定する。

素元分解ができなくても一般に素イデアル分解は可能である。

そこで O_K については素元ではなく素イデアルの集合 G を適当にとる。

a, b を互いに素な有理整数で $a + bM$ は F-smooth とする。つまり

$$a + bM = \prod_{p \in F} p^{e(p)}$$

であり、かつ単項イデアル $(a + b\theta)$ は G-smooth とする。つまり

$$(a + b\theta) = \prod_{\mathcal{P} \in G} \mathcal{P}^{e(\mathcal{P})}$$

と完全に素イデアル分解されるものとする。

これをたくさん集め、適当に組み合せると、つぎのような関係式が出る。

$$\prod (a_i + b_i M) = \left(\prod_{p \in F} p^{e(p)} \right)^2$$

と $\mathbf{Z}$ で平方数になり、かつ

$$\left(\prod (a_i + b_i \theta) \right) = \left(\prod_{\mathcal{P} \in G} \mathcal{P}^{e(\mathcal{P})} \right)^2$$

と O_K で平方イデアルになる。しかし「平方イデアルを生成する数」と「平方数」の間にはギャップがある。

アイディア　平方剰余記号でチェックする。

有理素数 p で K で 1 次であるものをいくつかとり、それを法とする平方剰余記号を χ_p とする。この平方剰余記号を最初から何個か factor base に入れておく。

すると $\prod(a_i + b_i\theta)$ は高い確率で O_K の平方数になる。実際

$$\prod(a_i + b_i\theta) = h(\theta)^2$$

であったとすれば後は Special NFS の場合とまったく同様である。

しかし

問題点　$\prod(a_i + b_i\theta)$ の平方根 $h(\theta)$ を求めるのが難しい。

従来型の代数的数の平方根を求める手法では係数膨張が起こりかなり難しい。実際に必要なのは $\bmod n$ での値にすぎない。そこで中国の剰余定理を用いる。奇数次の場合は符号を考えることにより各々の法での二つの平方根のうちどちらをとるべきかを決めることができる。しかし偶数次の場合はそれができていない。

補足　奇数次の場合に中国の剰余定理により小さな数の計算で可能だといっても必要とする素数は非常に多い。factor base の個数個程度の 32bit 素数が必要になることもあり、所用時間はかなりのものになる。ただし各素数ごとに独立した処理が可能なので複数の機械を投入して実際の処理時間を短縮することができる。

12　数体ふるい法の計算量

定義　整数 r の素因数がすべて Y 以下のとき r は Y-smooth であるという。

素因数分解で出てくるような場合は次の定理でおおよその smoothness を見積もることができる。

記号

$$L_x[\nu, \lambda] = \exp(\lambda(\log x)^{\nu}(\log\log x)^{1-\nu})$$

例

$$\begin{array}{rll} x & = \exp(\log x) & = L_x[1,1] \\ x^{\lambda} & = \exp(\lambda \log x) & = L_x[1,\lambda] \\ (\log x)^{\lambda} & = \exp(\lambda \log\log x) & = L_x[0,\lambda] \end{array}$$

定理　$L_x[\nu, \lambda]$ 以下の自然数が $L_x[\omega, \mu]$-smooth である確率は

$$L_x[\nu-\omega,\ -(\nu-\omega)\lambda/\mu + o(1)] \quad \text{as } x \to \infty$$

である。

一般に分解したい数 n が与えられたとき、

適当に次数 d をとり

$$M^{d+1} > n$$

となるように M を定めれば n を M 進展開して d 次の多項式

$$f(X) = c_d X^d + c_{d-1} X^{d-1} + \cdots + c_1 X + c_0 \text{ such that } f(M) = n$$

を作ることができる。このとき $f(X)$ の係数は $n^{1/(d+1)}$ 以下である。

したがって

$$\begin{aligned} c_d\mathcal{N}(a+b\theta) &= (-b)^d f(-a/b) \\ &= c_d a^d + c_{d-1}a^{d-1}(-b) + \cdots + c_r a^r(-b)^{d-r} + \cdots + c_0(-b)^d \end{aligned}$$

の絶対値はおよそ $\max(a,b)^d\, n^{1/(d+1)}$ である。

一方 $a+bM$ の絶対値はおよそ $\max(b)\, n^{1/(d+1)}$ である。

これらの積を最小にするには次数 d を次のようにとる。

$$d = \lambda^{-1}\left(\frac{\log n}{\log\log n}\right)^{1/3}$$

factor base の個数、ふるいの範囲を共に

$$L_n[1/3,\, 2\lambda^2]$$

とすれば

$$\begin{aligned} &|\mathcal{N}(a+b\theta)| \\ &= L_n[1/3,\, 2\lambda^2]^d \cdot n^{1/(d+1)} \\ &= \exp\left(2\lambda(\log n)^{2/3}(\log\log n)^{1/3}\right) \cdot \exp\left(\lambda(\log n)^{2/3}(\log\log n)^{1/3}\right) \\ &= L_n[2/3,\, 3\lambda] \end{aligned}$$

であり

$$\begin{aligned} &\max(b)M = L_n[1/3,\, 2\lambda^2] \cdot n^{1/(d+1)} \\ &= L_n[1/3,\, 2\lambda^2] \cdot \exp\left(\lambda(\log n)^{2/3}(\log\log n)^{1/3}\right) \\ &= L_n[2/3,\, \lambda] \end{aligned}$$

である。

定理によればこれらが同時に $L_n[1/3,\, 2\lambda^2]$-smooth になる確率は

$$L_n[1/3,\, -3(1/3)/(2\lambda)] \cdot L_n[1/3,\, -(1/3)/(2\lambda)] = L_n[1/3,\, -2/(3\lambda)]$$

となる。

smooth data がこの a-b 区間内に factor base の個数個以上なければならないから

$$L_n[1/3,\, -2/(3\lambda)]\, L_n[1/3,\, 2\lambda^2]^2 \geq L_n[1/3,\, 2\lambda^2]$$

である。したがって

$$\lambda^3 \geq 1/3$$

smooth data は bactor base の個数個以上集めるのだから

$$(\text{factorbase の個数})/(\text{確率}) = L_n[1/3,\, 2\lambda^2{+}(2/3)/\lambda] \geq L_n[1/3,\, (64/9)^{1/3}]$$

だけの計算量になる。この第一因子の $1/3$ が数体ふるい法の革命的な点であった。従来の二次ふるい法や楕円曲線法では $1/2$ であった。

ちなみにこの式から計算した d の値は
$n = 10^{26}$ で 3.5、
$n = 10^{67}$ で 4.5、
$n = 10^{139}$ で 5.5、
$n = 10^{254}$ で 6.5 である。

13　複数多項式数体ふるい法

問題点　多項式の係数が大きすぎる。

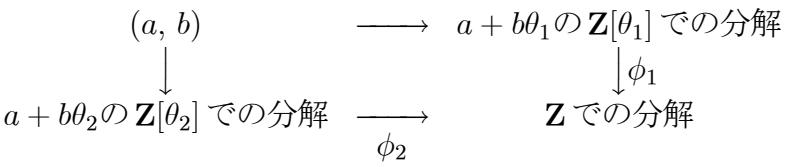
ターゲットの合成数 n に対し、複数の多項式 $f_t(x)$ $(t = 1, 2, \ldots)$ と共通の整数 M を選び

$$f_t(M) \equiv 0 \bmod n \quad \text{for } \forall t$$

となるようにする。このとき $f_t(x)$ の根の1つを θ_t とし、$K_t = Q(\theta_t)$ とする。

このとき $\theta_1, \theta_2, \ldots$ は M を媒介として $\bmod n$ で結びつく。

$t = 2$ の場合を図示すれば



そこでふるいは

$$a + b\theta_1,\ a + b\theta_2, \ldots$$

に対して行うことになる。つまり M がふるいの対象から消えてしまう。

ポイント　M の大きさが $n^{1/(d+1)}$ 程度という制約がなくなる。

しかし一般の次数に対して、この M の自由度を生かして多項式の係数を最適化することはまだできていないようである。

もしこれが革命的に良い結果となれば計算量が $L_n[1/4, *]$、あるいはもっと良く、なることがあるかもしれない。

参考文献　数体ふるい法の全貌は

A.K.Lenstra and H.W.Lenstra, Jr. *The development of the number field sieve.* Lecture Notes in Mathematics 1554(1993), Springer-Verlag, Berlin

にまとめられている。これに次を加えればまず十分であろう。

Marije Elkenbrach-Huizing *A Multiple Polynomial General Number Field Sieve*, Lecture Notes in Computer Science 1122(1996), 99–114, Springer-Verlag, Berlin