Centre for Information
Security and Cryptography

UNIVERSITY OF
CALGARY

A New Look at an Old Equation

R. E Sawilla, DRDC, Ottawa

A. Silvester, U. of Calgary

H. Williams, U of Calgary

The Diophantine Equation

Consider

$$(*) \quad ax^2+bxy+cy^2+dx+ey+f=0,$$

where a, b, c, d, e, f are given integers.

We have two questions:

1) Does (*) have solutions in integers x, y?
2) If so, what are they?

Lagrange (1767-69)

Lagrange solved these problems by writing (*) as

$$DY^2 = (Dy+E)^2 + DF-E^2,$$

where $D = b^2 - 4ac$, $E=bd-2ae$, $F=d^2-4af$, $Y=2ax+by+d$.

Putting $N=E^2 - DF$, $X=Dy+E$, we get

$$(1) \qquad X^2-DY^2=N.$$

Thus, any solution X, Y of (1) such that

$$(2) \quad X \equiv E \ (mod\ D) \text{ and } Y \equiv b(X-E)/D + d \ (\ mod\ 2a)$$

will provide a solution of (*).

The Pell Equation

Let D be a positive integer and not a perfect square.

Solve for integers T, U

(3) $$T^2 - DU^2 = 1.$$

For example, for $D=7$, a solution is $T=8$, $U=3$.

All solutions of (3) are generated by a fundamental solution t, u by

$$T+U\sqrt{D} = (t+u\sqrt{D})^n .$$

We may assume that $\varepsilon(D) := t + u\sqrt{D} > 1$.

Solutions of (1)

When $D < 0$, (1) can be easily solved by using Cornacchia's Algorithm. We will assume, then, that $D > 0$.

If X, Y is a solution of (1), then

$$\lambda = X + Y\sqrt{D} = \mu\varepsilon(D)^n,$$

Where n is some integer, $\mu = R + S\sqrt{D}$ is from a finite set $\mathcal{S}$ and R, S is a solution of (1).

Thus, for a given μ we need to find n such that (2) holds.

Finding *n* (Dujardin, 1894)

If we are given $\mu = R+S\sqrt{D}$, in order for (2) to hold we must have either

$R \equiv E \bmod (D)$ $(n\ even)$ or $tR \equiv E\ (mod\ D)$ $(n\ odd)$.

Also, $R \equiv bS\ (mod\ 2a)$.

Furthermore,
$D \mid (dD-bE-DS+bR)/2a\ (n\ even)$
$D \mid (dD-bE-DSt+bRt)/2a\ (n\ odd)$.

Thus, only the parity of *n* needs to be determined.

Finding μ

Let $[\alpha, \beta]$ denote the module $\{x\alpha + y\beta\}$, where x and y range over the integers.

We need to find the principal ideals of $O= [1,\sqrt{D}]$ which have norm $|N|$. Such ideals have the form

$$c[a, b+ \sqrt{D}].$$

Here $c^2 \mid N$, $a=|N|/c^2$, and $b^2 \equiv D \pmod{a}$.

Let $O^*=< -1,\varepsilon >$, where ε (>1) is the fundamental unit of O.

If γ is a generator of such an ideal, then $\mu= \gamma$ or $\mu= \varepsilon\gamma$

The Regulator

We overcome the problem of large units by working with the regulator of O instead of ε. This is defined to be $R_d = log\ \varepsilon$. Here d denotes the discriminant ($4D$) of O. We also use $h(d)$ or h to represent the class number of O.

R_d is a transcendental number, so we are content to compute R'_d, a rational approximation to R_d which is within 1 of R_d.

How Big is R_d?

Analytic class number formula

$$2h(d)R_d = \sqrt{d}\,L(1,\chi_d),\chi_d(n) = \left(\frac{d}{n}\right)$$

$$L(1,\chi_d) = \sum_{n=1}^{\infty}\frac{\chi_d(n)}{n} = \prod_q \left(\frac{q}{q-(q/n)}\right)$$

$$< \frac{1}{2}\log d + 1 \qquad (\text{Hua},1942).$$

Comments

Quite frequently

$$\varepsilon > e^{\sqrt{d}}.$$

For example, if

$D = 990676090995853870156271607886$,

then x and y have in excess of $2 \cdot 10^{15}$ decimal digits each. As the average paperback contains about one million symbols this means that it would take more than two billion such volumes to record x alone.

If γ is the generator of a principal ideal, then

$$1 \leq \gamma < \varepsilon$$

Techniques for computing $\boldsymbol{R}_d$

Continued fraction method— $O(d^{1/2+\varepsilon})$

Infrastructure Method (Shanks, 1972) — $O(d^{1/4+\varepsilon})$

Lenstra's Las Vegas Algorithm (1982) — $O(d^{1/5+\varepsilon})$

Srinivasan's Las Vegas Algorithm (1998) — $O(d^{1/5+\varepsilon})$

Subexponential Methods

Buchmann (1989) and Abel(1994) have shown that it is possible to compute R'_d and $h(d)$ by an index calculus algorithm that under certain generalized Riemann hypotheses (GRH) should execute in expected time

$$exp\{(\sqrt{2}+o(1))(log\ d)^{1/2}\ (log\ log\ d)^{1/2}\}$$

(Vollmer 2002).

Example

For
D=13022194102190350410319085329793205127-
3194641328847761633615783665713790925835-
60263087397184669099836 (101D)

we get (under the GRH)

R=31780254623174755539291764915494863617-
2763163478260.945231457

This required 87 days on a Beowulf cluster of 16
55 MHz Pentium III computers.

Jacobson, Scheidler, W. 2001

Remarks

If this algorithm were to fail to find a correct value for R'_d it would, nevertheless, find a close rational approximation of an integral multiple of R_d. It will also find a divisor of $h(d)$.

The subexponential algorithm can also be generalized to solve the DLP in O. That is, if we are given a principal ideal, we can find the logarithm of its generator in expected subexponential time.

Where Do We Need the GRH

Correctness

1. Factor base (prime ideals of small norm) must generate the entire class group.
   - without GRH, factor base size is exponential.

2. Find H such that $H < h(d)R_d < 2H$
   - accurate error estimate in approximation of $L(1, \chi_d)$ requires GRH.

Problems

The exponential methods are slow.

The subexponential method is dependent on the truth of the GRH.

Notation change:

We let

$$\mathcal{R}_d \quad \text{denote} \quad \log_2 \varepsilon_d\,.$$

Questions

1. Can we verify the value of $\mathcal{R}_d$ (actually $\mathcal{R}'_d$, an approximation of $\mathcal{R}_d$) computed by Buchmann's algorithm more quickly than the current exponential techniques?

2. Can we be sure that our value of $\mathcal{R}'_d$, satisfies

$$|\mathcal{R}_d - \mathcal{R}'_d| < 1?$$

Error Sources

Any fast algorithm for computing $\mathcal{R}_d$ involves many approximations to transcendental numbers.

Possible sources of error

1. Round-off, truncation

2. Software

3. Hardware

4. Accidents

Verification of R'_d

Given R'_d from the index calculus algorithm, we want to verify unconditionally that R'_d is within 1 of R_d.

Recently, de Haan, Jacobson and W. (2007) showed how this can be done, but unfortunately the process still takes exponential time. The good news is that it is less expensive than any other method.

Reduced Ideals

If $\mathfrak{a}$ is a primitive ideal (has no rational integer divisors) of O, then $\mathfrak{a}$ is said to be reduced if there does not exist any non-zero α in $\mathfrak{a}$ such that both

$$|\alpha| < N(\mathfrak{a}), \quad |\alpha'| < N(\mathfrak{a})$$

hold. Here α' denotes the conjugate of α.

Theorem If $\mathfrak{a}$ is reduced, then $N(\mathfrak{a}) < \sqrt{d}$

The Cycle of Reduced ideals

Let $\mathfrak{a}(=\mathfrak{a}_1)$ be a given reduced ideal. The simple continued fraction algorithm produces a sequence of all the reduced ideals equivalent to $\mathfrak{a}$:

$$\mathfrak{a}_1, \mathfrak{a}_2, \mathfrak{a}_3, \ldots, \mathfrak{a}_n, \ldots$$

such that

$\mathfrak{a}_{n+1}= \mathfrak{a}_1$, $\mathfrak{a}_i=(\theta_i)\mathfrak{a}_1$, $\theta_{i+1} > \theta_i$, for $i =1, 2, 3\ldots$

(f,p)-Representations

Definition: Let $p \in \mathbb{N}$, $f \in \mathbb{R}$, $1 \leq f < 2^p$ and $\mathfrak{a}$ be any primitive ideal of $\mathbb{K}$. An (f, p)-representation of $\mathfrak{a}$ is a triple $(\mathfrak{b}, m, k)$, where $k \in \mathbb{Z}$, $m \in \mathbb{N}$, $2^p < m \leq 2^{p+1}$ and $\mathfrak{b}$ is a primitive ideal of O equivalent to $\mathfrak{a}$.

Furthermore $\mathfrak{b} = (\theta)\,\mathfrak{a}$ with $\theta \in \mathbb{K}$ and

$$|2^{p-k}\,\theta/m - 1| < f/2^p\,.$$

If $\mathfrak{b}$ is a reduced ideal, we say that $(\mathfrak{b}, m, k)$ is a reduced (f, p)-representation of $\mathfrak{a}$.

$\mathfrak{a}(x)$

Definition

Let $(\mathfrak{a}_i, m_i, k_i)$ $(i=1,2,3,\ldots)$ be reduced (f,p)-representations of $\mathfrak{a}_1$. For a positive integer x, we define $\mathfrak{a}(x)$ to be an ideal $\mathfrak{a}_j$ such that $k_j < x$ and $k_{j+1} \geq x$.

Here $\mathfrak{a}(x) = \mathfrak{a}_j = (\theta_j)\,\mathfrak{a}_1$ and

$$x-4+\log 15 - (1/2)\log d < \log \theta_j < x-3+\log 17$$

when $f < 2^{p-4}$. That is, the value of $\log \theta_j$ is close to that of x.

Algorithms

1. Given $(\alpha(x'), m', k')$ an (f', p) - representation of α and $(\alpha(x''), m'', k'')$ an (f'', p) representation of α, we can compute $(\alpha(x' + x''), m, k)$, an (f, p) - representation of α, where

$$f \; = f^* + 11/4 + 2^{-(p+1)} f^*,$$

$$f^* = f' + f'' + 2^{-p} f' f''$$

in $O(d^{\varepsilon})$ operations.

2. Given α and x, we can compute a certain (f,p) – representation of α

$$(\alpha(x), m, k)$$

in $O((\log x)\, d^{\varepsilon})$ operations.

If

$2^p > 20.2 x \max\{16, \lceil \log x \rceil\}$, then $f < 2^{p-4}$.

Two theorems

Put
$\mathfrak{a}_1 = (1), S = \{\bar{\mathfrak{a}}_4, \bar{\mathfrak{a}}_3, \bar{\mathfrak{a}}_2, \mathfrak{a}_1, \mathfrak{a}_2, \mathfrak{a}_3, \mathfrak{a}_4\}$

Theorem If $c \in \mathbb{Z}^+$ and $|\mathcal{R}'_d - \mathrm{c}\mathcal{R}_d| < 1$

then

$$\mathfrak{a}\left(\lceil \mathcal{R}'_d \rceil\right) \in S.$$

If $\mathfrak{a}\left(\lceil \mathcal{R}'_d \rceil\right) \in S$., we can easily modify the value of $\mathcal{R}'_d$ to ensure that $|\mathcal{R}'_d - c\mathcal{R}_d| < 1$.

Theorem If $q \geq 2$ and

$x = \lceil \mathcal{R}'_d / q \rceil, \mathcal{R}_d > q\left(2\log d + \log 17/4\right),$

then $q|c$ if and only if $\mathfrak{a}(x) \in S$.

Strategy

Given $\mathcal{R}'_d$ from the subexponential algorighm we must show that

1. $\mathcal{R}_d > K$ (cost $O(K^{1/2}\, d^{\varepsilon})$)

2. $|\mathcal{R}'_d - c\mathcal{R}_d| < 1$ (cost $O(d^{\varepsilon})$)

If (2) holds, then

$$c < 1 + \mathcal{R}'_d / \mathcal{R}_d < 1 + \mathcal{R}'_d / K$$

3. $c = 1$ (cost $O(d^{1/2+\varepsilon}/K)$)

Optimal value of K is such that

$$K^{1/2} = d^{1/2} / K \Rightarrow K = d^{1/3}$$

$$\Rightarrow \text{overall cost is } O\left(d^{1/6+\varepsilon}\right)$$

Timings

2 Intel P4 Xeon 2.4 GHz processors, 1 GB of RAM

$d \approx 10^{\cdots}$	Subexponential	$d^{1/6 + \varepsilon}$	$d^{1/5 + \varepsilon}$
15	0.29 sec	0.42 sec	0.25 sec
20	0.45 sec	0.93 sec	3.65 sec
25	0.68 sec	3.20 sec	2 min, 20 sec
30	1.44 sec	14.60 sec	44 min, 26 sec
35	2.57 sec	1 min, 27 sec	2 days, 13hr
40	6.06 sec	6min, 12 sec	N/A
45	26.27 sec	1hr, 10min	N/A
50	1min, 27 sec	1 day 9hr	N/A

Further Timings

$d \approx 10^{\cdots}$	$R_d \approx \ldots \times 10^{30}$	Processors	Time
60	1.1	180	4 d, 9 h
62	3.4	240	6 d, 3 h
63	5.2	240	8 d, 2 h
64	8.1	240	10 d, 13h
65	195.7	240	102 d, 7h

Example:

d=3928637573454259474975805083515165509211884853083339874356156848l

(65D)

$R_d =$

195696290466865524253842387693615.276328149

Ideal Principality

Let $\mathfrak{b}$ be a reduced ideal of $\mathbb{K}$.

1. Use the subexponential algorithm to find $\mathcal{R}'_d$ and h – the class number

2. Verify $\mathcal{R}'_d$

3. Find $(\mathfrak{c},\mathrm{m},\mathrm{k})$ a reduced (f,p) – representation of $\mathfrak{b}^h$ $(\mathfrak{c} = (\phi)\mathfrak{b}^h)$

4. Use the subexponential algorithm to solve the DLP for $\mathfrak{c}$ to obtain $g \in \mathbb{Q}$ such that $| g - \log_2 \alpha | < 1$, where $\mathfrak{c} = (\alpha)$ (Jacobson, 2000).

5. Put $\mathfrak{a}_1 = (1)$ and find $i \in \{\pm 3, \pm 2, \pm 1, 0\}$ such that

$$\rho^i\left(\mathfrak{a}\left(\lceil g \rceil\right)\right) = \mathfrak{c}$$

6. Compute m', k' such that $(\mathfrak{c}, m', k')$ is a reduced (f,p) – representation of $\mathfrak{a}_1$.

If $\mathfrak{b}$ is principal, then

$$\mathfrak{b} = (\beta) \quad (1 \le \beta < \varepsilon_d)$$

$$\Rightarrow$$

$$\beta^h = \alpha\phi^{-1}\lambda \quad \left(\lambda = {\varepsilon_d}^r\right)$$

<u>Theorem</u>

$$-2 \le r \le h$$

<u>Theorem</u> If

$$b(r) = \left\lceil \frac{r\mathcal{R}'_d + k' - k}{h} \right\rceil,$$

then

$$|\log\beta - b(r)| < 2 + 3/h.$$

Put

$$S' = \left\{ \rho^{i}(\mathfrak{b}) :\ |i| \leq 9 \right\}$$

$\mathfrak{b}$ is principal if and only if

$$\mathfrak{a}(b(r)) \in S'$$

for some

$$r \in \{-2, -1, 0, \ldots, h\}.$$

Complexity

By infrastructure techniques we can determine unconditionally whether or not $\mathfrak{b}$ is principal in $O\left(R_d^{1/2} d^{\varepsilon}\right)$ operations.

By our technique we can do this in

$$O\left(d^{1/6+\varepsilon}\right) + O\left(hd^{\varepsilon}\right) \quad \text{operations.}$$

Since

$$hR_d = O\left(d^{1/2+\varepsilon}\right),$$

we have a Las Vegas algorithm for solving this problem of complexity

$$O\left(d^{1/6+\varepsilon}\right).$$

Example (Jacobson and Williams, 2002)

$d_1 = 187060083$

$d_2 = 148946762383055129$

$d_3 =$
$131194254072438972350592900266788017500 5208$

$j_1 = 2$

$j_2 = 2104044625155634711504852164 5334887$

We need to verify that the equation

$$d_1X^2 - d_3Y^2 = (d_3\, j_1 - d_1\, j_2)\,/\,d_2 := c$$
$$= 880813063496060911643645$$

has no solutions.

We do this by verifying that there are no principal ideals of norm $c\, d_1$ in the quadratic field $\mathbf{Q}(\sqrt{(d_1 d_3)})$. This yields the field discriminant

$d = d_1 d_3 =$
$2454120805591352218033661302311608869705287$
33912264 (51D)

From the subexponential algorithm
$h(d) = 1024$

$R_d = 6851106675369184895740.2467$

Verification

We now need to verify that all of the ideals of norm $cd_1 = 3 \cdot 5 \cdot 7 \cdot 769 \cdot 33809 \cdot 8907623 \cdot 6775714175075849$ are not principal in O by using the new method. Note that 3, 7, 8907623 each divide the discriminant d, so we have a total of 16 ideals of norm cd_1.

Using the new method, we were able to show in 5 hours that none of these is principal. 87% of the time needed to do this was required to verify the value of R'_d.