Deterministic Polynomial-Time Equivalence of Computing the RSA Secret Key and Factoring

解説論文

RSA秘密鍵計算と素因数分解の決定的多項式時間同値性

Deterministic Polynomial-Time Equivalence of Computing the RSA Secret Key and Factoring

神永正博* 深瀬道晴**
Masahiro KAMINAGA Masaharu FUKASE

Abstract: This note presents a lecture on the paper "Deterministic Polynomial-Time Equivalence of Computing the RSA Secret Key and Factoring" by Coron and May. They present the first *deterministic* polynomial-time algorithm that factors N given (d, N), provided that $e, d < \varphi(N)$. We also show several simulation results deriving prime facotrs of N from an RSA secret key using the LLL algorithm.

Keywords: RSA 暗号, 格子基底簡約, LLL アルゴリズム

1 はじめに

本解説論文の目的は, Coron-May[2] の主要部分の解説を与えることである. Coron-May[2] は, RSA 秘密鍵を求めることと公開モジュラスの素因数分解の決定的多項式時間同値性を示した重要な論文である. しかしながら, 専門家を除くとあまり知られていない結果であるため, その解説を工学部研究報告として広く公開することには意味があると思われる.

RSA 暗号系では, 巨大な素数対 p, q に対し, $N = pq$ を公開モジュラス, $ed \equiv 1 \pmod{\varphi(N)}$ となる e, d をそれぞれ公開指数, 秘密指数と呼んでいる. (e, N) を公開鍵とし, (d, N) を秘密鍵として, $C = M^e \bmod N$ で暗号化, $M = C^d \bmod N$ で復号化を行う. 電子署名で用いる場合は, 文書などのデータのハッシュ値に適当なパディング処理を施したデータ M に対し, $S = M^d \bmod N$ を電子署名として用いる. ここで, $\varphi(N) = (p-1)(q-1)$ はオイラー関数である.

RSA 暗号は, 公開モジュラス $N = pq$ の素因数分解の（平均的な）計算量的困難性を安全性の根拠としている. 確かに, N の素因数分解ができれば p, q から, $\varphi(N)$ が求まるので, 拡張ユークリッド互除法を用いて d を求めることは容易である. しかし, この逆は自明ではない. つまり,「RSA 暗号の秘密鍵 (d, N) を知ることと素因数分解を知ることとは計算量的に同値なのか」という問題が残っていることがわかる. 秘密鍵 (d, N) から N の素因数分解を行う効率的なアルゴリズムはあるだろうか. この問題は RSA 暗号誕生時から認識され, RSA 暗号の発明者自身により論文[6] で「確率的」多項式時間アルゴリズムが示されている. しかし, 比較的最近まで決定的多項式時間アルゴリズムは知られていなかった. このアルゴリズムを最初に示したのが, Coron-May[2] である.

ここでアルゴリズムの計算量について簡単に説明しておく. 入力に対して何らかの答を出力するアルゴリズムの計算量は, 計算のステップ数を入力データ a のビット長 $n = \log_2 a$ の関数によって表すことで評価される. 計算のステップ数が, 漸近的に n の多項式で書けるとき多項式時間オーダーの計算量, 漸近的に指数関数で書けるとき指数時間オーダーの計算量という. アルゴリズムの計算量の上限を評価するために使われる記号をオーダー記号（ランダウの記号）といい, $\mathcal{O}$ で表す. 例えば, アルゴリズムのステップ数の上限が漸近的に n の多項式 n^2 で書けるとき, アルゴリズムの計算量を $\mathcal{O}(n^2)$ で表す. 問題 A を確実に解く多項式時間アルゴリズムには, 「常に」多項式時間で答を返すアルゴリズムと, 入力をランダムに与えたときに平均的に多項式時間で答を返すアルゴリズムがある. 前者は, 決定的, 後者は確率的なアルゴリ

*東北学院大学情報基盤工学科, 同大学院電気工学専攻
**東北学院大学情報基盤工学科

ズムと呼ばれる．ここで扱う多項式時間同値性の問題は, $N = pq$ の素因数 p, q から d を求める多項式時間のアルゴリズムが存在し（拡張ユークリッド互除法）, 逆に d から p, q を求める決定的多項式時間のアルゴリズム（本論文で紹介する）が存在するという意味である．

本解説論文では, Coron-May[2] による秘密指数から公開モジュラスの素因数分解を行うアルゴリズムを一つずつ確認する． 加えてアルゴリズムを Victor Shoup[7] による NTL(Number Theory Library) ライブラリを用いて実装しシミュレーションを行って計算時間を調べた．アルゴリズムの中核部分は, 格子基底簡約 (lattice basis reduction) と呼ばれる処理であるが, その代表的なアルゴリズムとして LLL を用いた．なお, 本解説論文は, 情報科学, 応用数学を学ぶ大学生・大学院生を想定しており, 理解するためには, 線形代数, 数論, 不等式, 関数の近似, 集合, 計算量（計算複雑性）についての初歩の知識を必要とする．

2 問題の定式化

RSA 暗号の公開モジュラスを $N = pq$, 公開指数を e, 秘密指数を d とする．これらの間には, ある整数 k が存在して

$$ed = 1 + k\varphi(N) \tag{1}$$

という関係がある．Coron-May[2] で問題にするのは, e, d, N を与えて p, q を求めることである．この問題は一見すると簡単に思える．というのは, 式 (1) は, $(p-1)(q-1) = N = (p+q) + 1$ より,

$$k(N - (p+q) + 1) = ed - 1$$

から $p + q$ を求めることに等しいからである．しかし, 問題は見た目ほど簡単ではない．難しい理由は, この方程式には未知数が $k, (p+q)$ の 2 つあり, 両者の積が現れることにある．

3 $ed \leq N^{3/2}$ の場合

最初に $ed \leq N^{3/2}$ という最も簡単な場合を考える．$ed \leq N^{3/2}$ という条件は, インターネットショッピング等で用いられている SSL/TLS でデフォルトで利用される $e = 2^{16} + 1 = 65537$ の場合に満たされていることに注意しよう．このように小さな e が利用される理由は, 暗号化処理や電子署名検証の処理を高速化するためである．なお, ここでは詳細に触れないが, 秘密指数 d についてはこのように小さく取ることは危険であり, 一般には用いられないことを注意しておく．実際, $d < N^{0.292}$ の場合に公開鍵 (e, N) から効率的に d を計算するアルゴリズムが知られている． 詳細については, Wiener[8], Boneh-Durfee[1] 等を参照されたい．なお, Wiener[8] では連分数展開が, Boneh-Durfee[1] では格子基底簡約が用いられる．これらを含め, 格子基底簡約を用いた RSA 暗号への攻撃についての包括的解説は, Heinek[3] にある．

定理 1. *$N = pq$ とし, 素数 p, q は, 同じビットサイズであると仮定する．e, d が, $ed \equiv 1 \pmod{\varphi(N)}$ を満たすとする． もし, $ed \leq N^{3/2}$ であれば, N, e, d から $\mathcal{O}(\log^2 N)$ オーダーで N の素因数分解を計算することができる．*

証明. 一般性を失わずに $p < q$ と仮定できるから, ビットサイズが同じであることより, $p < \sqrt{N} < q < 2p < 2\sqrt{N}$ がわかる．このとき,

$$p + q < p + 2p < 3\sqrt{N}$$

および,

$$\varphi(N) = N + 1 - (p+q) > N - 3\sqrt{N} > \frac{N}{2}$$

がわかる．但し, $N > 36$ とする．RSA 暗号では N は 1024 ビット程度の大きさがあり, これはもちろん不自然な仮定ではない．仮定より, ある整数 k が存在して,

$$ed = 1 + k\varphi(N) \tag{2}$$

が成り立つ．式 (2) を k について解くと,

$$k = \frac{ed - 1}{\varphi(N)}$$

となるが, ここで, $\varphi(N)$ を N に置き換えて,

$$\tilde{k} = \frac{ed - 1}{N}$$

とする． この $\tilde{k}$ は, k の非常によい近似となる．実際,

$$\begin{aligned} k - \tilde{k} &= \frac{ed-1}{\varphi(N)} - \frac{ed-1}{N} \\ &= \frac{N(ed-1) - (N-p-q+1)(ed-1)}{\varphi(N)N} \\ &= \frac{(p+q-1)(ed-1)}{\varphi(N)N} \\ &< \frac{3\sqrt{N}(ed-1)}{\frac{N}{2} \cdot N} = 6N^{-3/2}(ed-1) \end{aligned}$$

ここで, 仮定より, $ed-1<N^{3/2}$ であるから,

$$k-\tilde{k}<6N^{-3/2}(ed-1)<6$$

となる. つまり, $k-\tilde{k}=0,1,2,3,4,5$ のいずれかである. この 6 つの候補から,

$$N+1+\frac{1-ed}{k}=p+q$$

となるものを選べばよい. これは, $\mathcal{O}(\log^2 N)$ のオーダーの計算量で実行できる. □

4 $ed\le N^2$ の場合

次に $ed\le N^2$ の場合を考える. この条件は, RSA 暗号で一般的に想定される場合に対応している. この場合を解くには, 一変数多項式の「小さな」根を求める必要があるが, そのために格子簡約の技術が必要となる. 以下, 説明する.

4.1 Howgrave-Graham の補題

$ed\le N^2$ の場合を解くには, 一変数の Howgrave-Graham の補題が用いられる. 以下, φ は一般の正の整数であるが, 後に, $\varphi=\varphi(N)$ として用いられるので, この記号になっている.

補題 2. *(Howgrave-Graham)* $h(x)\in\mathbb{Z}[x]$ を高々ω 個の単項式の和であるとする. $|x_0|\le X$ が $h(x_0)\equiv 0 \pmod{\varphi^m}$ を満たし, $\|h(xX)\|<\varphi^m/\sqrt{\varphi}$ であれば, $\mathbb{Z}$ 上で $h(x_0)=0$ が成り立つ.

証明. $|h(x_0)|<\varphi^m$ であることを証明すればよい.

$$\begin{aligned}|h(x_0)| &= |\sum_i a_i x^i| \\ &= \left|\sum_i a_i X^i\left(\frac{x_0}{X}\right)^i\right| \\ &\le \sum_i\left|a_i X^i\left(\frac{x_0}{X}\right)^i\right| \\ &\le \sum_i |a_i X^i|\end{aligned}$$

であることはすぐにわかる. 最後の不等式では, $|x_0|<X$ を用いた. ところで, $h(xX)=\sum_i a_i(xX)^i=\sum_i(a_iX^i)x^i$ であるから, そのノルムは, $\|h(xX)\|^2=\sum_i|a_iX^i|^2$ であることに注意すると, Schwarz の不等式より,

$$\begin{aligned}\sum_i|a_iX^i| &\le \sqrt{\sum_i 1^2}\sqrt{\sum_i|a_iX^i|^2} \\ &= \sqrt{\omega}\|h(xX)\|<\varphi^m\end{aligned}$$

となり補題が証明された. □

4.2 格子基底簡約アルゴリズム

主定理の証明には, 以下の格子基底簡約アルゴリズムが用いられる. ここで格子とは, $\mathbb{Z}^n$（n 個の整数の組全体からなる集合）における一次独立なベクトルの整数係数一次結合のことである.

定理 3. *(LLL)* $\mathcal{L}$ を,

$$\langle u_1,u_2,\cdots,u_\omega\rangle(u_j\in\mathbb{Z}^n, j=1,2,\cdots,\omega)$$

で張られる格子とし, $\max_j\|u_j\|\le B$ とする. このとき, *LLL* は,

$$\|b_1\|\le 2^{(\omega-1)/4}\det(\mathcal{L})^{1/\omega}$$

を満たす $b_1\in\mathcal{L}$ を $\mathcal{O}(\omega^5 n\log^3 B)$ オーダーの計算量で見つけることができる.

注意 1. *Nguyen-Stehlé*[5] はさらに効率的なアルゴリズムを見出している. 次の定理が成り立つ. なお, 本解説論文ではこのアルゴリズムは用いないため詳細は省略する.

定理 4. *(L^2 アルゴリズム) LLL* と同じ上界を持つベクトルを $\mathcal{O}(\omega^4 n(\omega+\log B)\log B)$ オーダーの計算量で見つけることができる.

4.3 $ed\le N^2$ の場合の証明

定理 5. $N=pq$ とし, 素数 p,q は, 同じビットサイズであると仮定する. e,d が, $ed\equiv 1 \pmod{\varphi(N)}$ を満たすとする. もし, $ed\le N^2$ であれば, N,e,d から $\mathcal{O}(\log^9 N)$ オーダーで N の素因数分解を計算することができる.

証明. $U=ed-1$, $s=p+q-1$ としておくと, $\varphi(N)=(p-1)(q-1)=N-s$ となる. X を適当な上界（Howgrave-Graham の補題に出てくる X）として, s を X で割ったときの商を s_0, 余りを x_0 とする. つまり, $s=s_0X+x_0$, $0\le x_0<X$ とする. 以下が成立する.

$$\begin{aligned}U &\equiv 0 \pmod{\varphi} \\ x_0-N+s_0X &\equiv 0 \pmod{\varphi}\end{aligned}$$

第二の式は, φ が φ で割り切れるという当たり前の式である. 従って, 任意の 1 以上の整数 k に対して,

$$\begin{aligned}U^k &\equiv 0 \pmod{\varphi^k} \\ (x_0-N+s_0X)^k &\equiv 0 \pmod{\varphi^k}\end{aligned}$$

が成立する．この性質を考慮して, 次のような多項式を考える．

$$g_{ij}(x) = x^i(x - N + s_0X)^jU^{m-j}$$

すると, $(x - N + s_0X)^j$ は, $x = x_0$ としたとき φ^j の倍数であり, U^{m-j} は, φ^{m-j} の倍数であるから, $g_{ij}(x_0)$ は φ^m の倍数である．つまり,

$$g_{ij}(x_0) \equiv 0 \pmod{\varphi^m}$$

が成り立つ．よって, これらの任意の $\mathbb{Z}$ 係数の一次結合でできる多項式 $h(x)$ は, $h(x_0) \equiv 0 \pmod{\varphi^m}$ を満たす．i, j を以下のように選ぶ．k, m は定められたパラメータである．

$$\begin{aligned} i &= 0 \qquad 0 \le j \le m \\ 1 \le i &\le k \qquad j = m \end{aligned}$$

これは論文にあるものをそっくりそのまま書いたものであるが, 状況が分かりにくい．$j = m$ のときは, $(i, j) = (0, m), (1, m), (2, m), \cdots (m, m)$ のように $i = 0, 1, 2, \cdots m$ が全て現れる．具体的に $m = 3, k = 3$ の場合に上記2つの条件を満たす (i, j) の組み合わせを書き下してみると, $(i, j) = (0, 0), (0, 1), (0, 2), (0, 3), (1, 3), (2, 3), (3, 3)$ となる．

$$g_{ij}(xX) = (xX)^i(xX - N + s_0X)^jU^{m-j}$$

を x の多項式として展開したときの係数を次数が小さい方から並べてベクトルを作り, それらで張られる格子を $\mathcal{L}$ とする．例えば, $2 + 3x + 5x^2 + 4x^3 + x^4 + 3x^5 + 6x^6$ は,

$$(2, 3, 5, 4, 1, 3, 6)$$

というベクトルに対応する．これを行ベクトルとして, $m = 3, k = 3$ のときの格子を行列にすると, 次のような下三角行列になる．$*$ は0ではない何らかのエントリであり, 空欄は0を表す．

	1	x	x^2	x^3	x^4	x^5	x^6
$g_{00}(xX)$	U^3						
$g_{01}(xX)$	$*$	U^2X					
$g_{02}(xX)$	$*$	$*$	UX^2				
$g_{03}(xX)$	$*$	$*$	$*$	X^3			
$g_{13}(xX)$		$*$	$*$	$*$	X^4		
$g_{23}(xX)$			$*$	$*$	$*$	X^5	
$g_{33}(xX)$				$*$	$*$	$*$	X^6

今の場合, 次数は7になり, 一般には, $\omega = m+k+1$ 次の格子ができあがる．$\mathcal{L}$ は, 下三角行列であり, $\det(\mathcal{L})$ は, この対角成分の積であるから, 他のエントリとは関係なく,

$$\det(\mathcal{L}) = U^{3+2+1}X^{0+1+2+3+4+5+6}$$

となる．各多項式 $g_{ij}(xX)$ の最高次の項は,

$$(xX)^i(xX)^jU^{m-j} = X^{i+j}U^{m-j}x^{i+j}$$

となる．格子 $\mathcal{L}$ に出現する項は,

$$\begin{aligned} (i, j) &= (0, 0), (0, 1), (0, 2), \cdots, (0, m) \\ &= (1, m), (2, m), (3, m), \cdots, (k, m) \end{aligned}$$

である．上の段が0次から m 次まで, 下の段が, $m + 1$ 次から $m + k$ 次までの項に対応する．これらに対応する項の係数をかけ合わせると,

$$\begin{aligned} \det(\mathcal{L}) &= \prod_{j=0}^{m} X^jU^{m-j} \prod_{i=1}^{k} X^{i+m} \\ &= X^{1+2+\cdots+(m+k)}U^{m+(m-1)+\cdots+1} \\ &= X^{(m+k)(m+k+1)/2}U^{m(m+1)/2} \end{aligned}$$

LLL を用いれば,

$$\|b\| \le 2^{(\omega-1)/4}(\det(\mathcal{L}))^{1/\omega} \tag{3}$$

となる（整数を成分に持つ）ベクトル b を $\mathcal{O}(\omega^6 \log^3 B)$ オーダーで計算することができる．Howgrave-Graham の補題を用いるには, 条件

$$2^{(\omega-1)/4}(\det(\mathcal{L}))^{1/\omega} < \frac{\varphi^m}{\sqrt{\omega}} \tag{4}$$

が成り立つ必要がある．Coron-May[2] では, 次節で示すようにこの上界を達成する必要条件を求めているが, かえってわかりにくいので, ここでは, 真っ当に評価式を作ることにする．格子の行列式 $\det(\mathcal{L})$ を代入すると, 式 (4) は,

$$2^{(\omega-1)/4}X^{\frac{m+k}{2}}U^{\frac{m(m+1)}{2\omega}} < \frac{\varphi^m}{\sqrt{\omega}} \tag{5}$$

不等式 (5) を X について解くと

$$\begin{aligned} X &< \left(\frac{\varphi^m}{2^{\frac{\omega-1}{4}}\sqrt{\omega}U^{\frac{m(m+1)}{2\omega}}} \right)^{\frac{2}{m+k}} \\ &= \frac{\varphi^{\frac{2m}{m+k}}}{\sqrt{2}\omega^{\frac{1}{m+k}}U^{\frac{m(m+1)}{(m+k)(m+k+1)}}} \end{aligned} \tag{6}$$

不等式 (6) の右辺を $\Gamma(m, k)$ とおく．その下限が問題となる．仮定より $U \le N^2$ であり, さらに,

$\omega^{\frac{1}{m+k}} = (m+k+1)^{\frac{1}{m+k}} < e$ がすぐにわかる. $N \geq N_0$ のとき $\varphi = N - s > c_{N_0} N$ となる定数 $0 < c_{N_0} < 1$ が存在するので, 分母を大きく, 分子を小さくすることによって,

$$\frac{c_{n_0}^{\frac{2m}{m+k}}}{\sqrt{2}e} N^{\frac{2mk}{(m+k)(m+k+1)}} < \Gamma(m,k)$$

を得る. ここで, $\frac{2m}{m+k} \leq 2$ であるから,

$$\frac{c_{n_0}^2}{\sqrt{2}e} N^{\frac{2mk}{(m+k)(m+k+1)}} < \Gamma(m,k)$$

となる. よって, x_0 の上界 X が,

$$X < \frac{c_{n_0}^2}{\sqrt{2}e} N^{\frac{2mk}{(m+k)(m+k+1)}}$$

を満たすことは, Howgrave-Graham の補題の十分条件である. この段階で, この上界をできるだけ拡げることを考える. 以下の補題から, 右辺は, $k = m$ で最大となる.

補題 6. $m > 0$ を固定したとき, k の関数 $\gamma(m,k) = \frac{2mk}{(k+m)(k+m+1)} (k > 0)$ は, $k = m$ 上で最大値 $\frac{m}{2m+1}$ を取る.

証明はやさしいので省略する. 代わりに $m = 5$ の場合のグラフ (図 1) を示す.

よって, $k = m$ として,

$$X < \frac{c_{n_0}^2}{\sqrt{2}e} N^{\frac{1}{2} - \frac{1}{4m+2}} \tag{7}$$

という条件で $\omega = m + k + 1 = 2m + 1$ 次の格子を LLL にかければ, $0 < x_0 < X$, $s = s_0 X + x_0$ を満たす x_0 が効率的に計算できる. X をこの上界ぎりぎりに取れば, $s = p + q - 1 \leq 3\sqrt{N}$ より,

$$s_0 \leq \frac{s}{X} \approx \frac{3\sqrt{N}}{\frac{c_{n_0}^2}{\sqrt{2}e} N^{\frac{1}{2} - \frac{1}{4m+2}}} = \frac{3\sqrt{2}e}{c_{n_0}^2} N^{\frac{1}{4m+2}} \tag{8}$$

が得られる. $m = \lfloor \log N \rfloor$ とすれば, s_0 は上から定数でおさえることができる. そこで, この範囲にある s_0 の候補に対して上記アルゴリズムを実行すればいい. 計算量は, LLL が $\mathcal{O}(\omega^6 \log^3 B)$ のオーダーで, $\omega = 2m + 1 (m = \lfloor \log N \rfloor)$ かつ, $B = \mathcal{O}(N^{2m})$ であるから, $\log B = \mathcal{O}(\log^2 N)$ であり, 結局 LLL のルーチンは, $\mathcal{O}(\omega^6 \log^3 B) = \mathcal{O}(\log^{12} N)$ のオーダーの計算量で短いベクトルを計算する. L^2 アルゴリズムを用いれば, 計算量は, $\mathcal{O}(\omega^4 n(\omega + \log B) \log B) = \mathcal{O}(\omega^5 n \log B) + \mathcal{O}(\omega^4 n \log^2 B) = \mathcal{O}((\log^5 N \times \log N \times \log^2 N) + \mathcal{O}(\log^4 N \times \log N \times ((\log N)^2)^2) = \mathcal{O}(\log^9 N)$ オーダーとなる. □

4.4 アルゴリズムのまとめ

アルゴリズムをまとめるとつぎのようになる. まず, 不等式 (7) を満たす最大の整数 X を選ぶ. 次に, $m = \lfloor \log N \rfloor$ として, 不等式 (8) を満たす s_0 の候補に対して, LLL または L^2 アルゴリズムを用いて $h(x) = \sum_{i,j} b_{ij} g_{ij}(x)$ の係数ベクトル b を決め, Howgrave-Graham の補題を用いて $x_0 (|x_0| < X)$ を計算し, $s = p + q - 1 = s_0 X + x_0$ を求める. $N = pq$ はわかっているので, p, q を解に持つ二次方程式 $z^2 - (s+1)z + N = 0$ を解き, 素因数を求める.

5 LLL アルゴリズムによるシミュレーション

4.4 節のアルゴリズムのシミュレーションが Coron-May[2] において示されているが, そこでは NTL ライブラリ [7] において実装されている LLL アルゴリズム [4] が用いられた. ここでは, Coron-May[2] と同様に, NTL ライブラリの LLL アルゴリズムによるシミュレーションを示す. 使用したライブラリはバージョン 9.6.2 であり, 計算機については, OS は macOS, プロセッサは 2.5GHz Intel Core i5 である.

格子の次元を決定する重要なパラメータが m であるが, このパラメータは Coron-May[2] によると, N が 512 ビットや 1024 ビットの問題に対して, $m = 10, 14, 16$ のような値を選択することができる. m を大きめに取ると, s_0 のブルートフォース探索の負担が軽減するが, LLL の負担が増える. 逆に, m を小さめに取ると, LLL の負担が軽減するが, s_0 のブルートフォース探索の負担が増える. 本稿におけるシミュレーションでは, s_0 のブルートフォース探索は実施せず, 正解の s_0 のみを使用する.

本稿のシミュレーションにおいては, まず, m を 10 などの値に決定し, X を $X \leq \frac{1}{16} N^{1/2 - 1/(4m+2)}$ を満たす最大の整数とし, X と正解の s_0 とを用いて, 格子の基底を生成した. そして, 得られた基底を LLL によって簡約し, 基底の第一ベクトル b_1 を $h(x)$ の係数として, $h(x) = 0$ の根を求め, 得られた実数根のうちの少なくとも一つ x_0 について, $s = s_0 X + x_0$ であることを確認した. このシミュレーションを, Coron-May[2] と同様に, N が 512 ビットと 1024 ビットの問題に対して実施した. 実施したシミュレーションにおいては, 全ての場合において, 目的の根 x_0 の求解に成功した. 以下に,

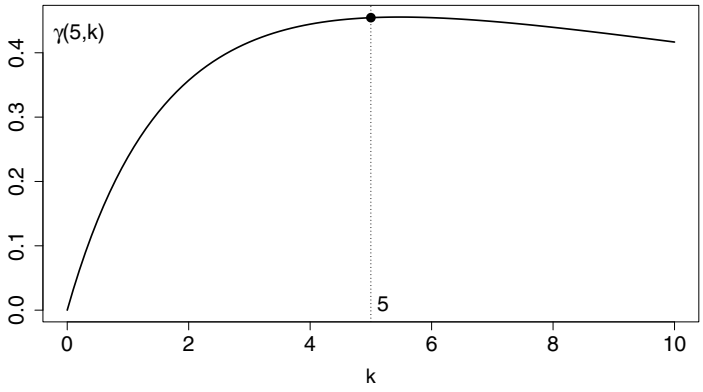

図 1: $\gamma(5,k)$ のグラフ

シミュレーション結果を表 1 に示す．表が示すように，Coron-May[2] のシミュレーションに対して，LLL の計算時間が概ね 10 倍以上短縮されているが，この要因としては，計算機性能の違いと NTL ライブラリのバージョンや用いた LLL の実装のタイプの違いなどが考えられる．

表 1: 格子の次元と計算時間

N(ビット)	次元	計算時間 (LLL)	Coron 等の結果
512	21	5s	70s
512	29	37s	7min
512	33	81s	16min
1024	21	29s	7min
1024	29	204s	40min
1024	33	438s	90min

また，格子の次元をより大きくしたときの挙動を調べるために，512 ビットの問題に対して，$m = 18, 20, 22, 24, 26$ に設定してシミュレーションを実施した結果，それぞれ，167，314，568，973,1613[sec] の時間を要した．512 ビットの問題に対して，次元と計算時間の関係を両対数でプロットした結果を図 2 に示す．回帰分析の結果，LLL の計算時間は格子の次元の 6.1715 乗に比例していることが分かった．これは，定理 3 において，$n = \omega$ とおいた場合に LLL の計算量が格子の次元の 6 乗に比例することと整合的である．

6 p, q のビット長がアンバランスな場合

N の秘密素因数が同じビット長を持つことを仮定していた．ビット長が異なる場合，(d, N) から N の素因数分解を行う方法を説明する．前節までは，p, q が同じビット数を持つ場合 (balanced primes case) を扱っている．この場合は，$s = p + q - 1$ が高々 $N^{1/2}$ のオーダーであるが，アンバランスな場合には，$s \gg N^{1/2}$ となり，証明はそのままでは通用しない．

6.1 2 変数の Howgrave-Graham の補題

p, q のビット長がアンバランスな場合の証明には，2 変数の Howgrave-Graham の補題が用いられる．

補題 7. *(Howgrave-Graham)* $T(x, y) \in \mathbb{Z}[x, y]$ を高々 ω 個の単項式の和として表される多項式とする．ここで，$T(x_0, y_0) \equiv 0 \pmod{e}$，$|x_0| < X$，$|y_0| < Y$ かつ $\|T(xX, yY)\| < \varphi^m/\sqrt{\omega}$ が成り立てば，$\mathbb{Z}$ 上で，$T(x_0, y_0) = 0$ が成り立つ．

証明. $T(x, y) = \sum_{i,j} t_{ij} x^i y^j$ とする．

$$
\begin{aligned}
|T(x_0, y_0)| &= \left| \sum_{i,j} t_{ij} x^i y^j \right| \\
&= \left| \sum_{i,j} t_{ij} X^i Y^j \left(\frac{x_0}{X}\right)^i \left(\frac{y_0}{Y}\right)^j \right| \\
&\leq \sum_{i,j} \left| t_{ij} X^i Y^j \left(\frac{x_0}{X}\right)^i \left(\frac{y_0}{Y}\right)^j \right| \\
&\leq \sum_{i,j} \left| t_{ij} X^i Y^j \right| \\
&\leq \sqrt{\sum_{i,j} 1} \sqrt{\sum_{i,j} |t_{ij} X^i Y^j|^2} \\
&\leq \sqrt{\omega} \|T(xX, yY)\| < \varphi^m
\end{aligned}
$$

となる．下から 2 行目で，Schwarz の不等式を用いた．□

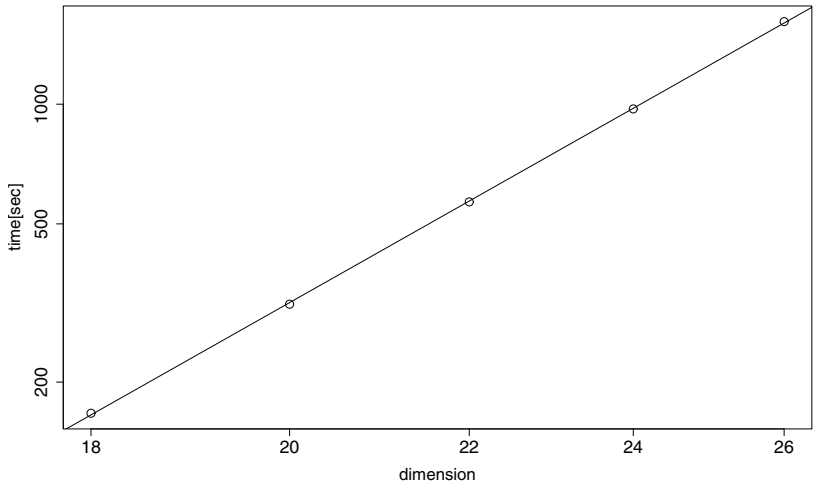

図 2: 格子の次元と LLL の計算時間の関係 (両対数グラフ)

6.2 主定理と証明

定理 8. $N = pq$ とし, 素数 p, q は, $0 < \delta \leq 1/2$ となる δ に対し, $p < N^{\delta}$, $q < 2N^{1-\delta}$ を満たすものとする. e, d は, $ed \equiv 1 \pmod{\varphi(N)}$ を満たし, かつ, $2\beta\delta(1-\delta) \leq 1$ を満たす β に対し, $1 < ed \leq N^{\beta}$ を満たすとする. このとき, N, e, d から $\mathcal{O}(\log^9 N)$ オーダーで N の素因数分解を計算することができる.

証明. $U = ed - 1$, $s = p + q - 1$ とおく. $\varphi = \varphi(N) = N - p - q + 1$ であるから, 以下が成立する.

$$\begin{aligned} U &\equiv 0 \pmod{\varphi} \\ p + q - (N+1) &\equiv 0 \pmod{\varphi} \end{aligned}$$

第二の式は, φ が φ で割り切れるという当たり前の式である. 従って, 任意の 1 以上の整数 k に対して,

$$\begin{aligned} U^k &\equiv 0 \pmod{\varphi^k} \\ (p + q - (N+1))^k &\equiv 0 \pmod{\varphi^k} \end{aligned}$$

が成立する. この性質を考慮して, 次のような多項式を考える. $m \geq 1, a \geq 1, b \geq 0$ を整数として,

$$g_{ijk}(x, y) = x^i y^j U^{m-k} (x + y - (N+1))^k$$

$$\begin{cases} i \in \{0, 1\}, & j = 0, & k = 0, 1, \cdots m, \\ 1 < i \leq a, & j = 0, & k = m \\ i = 0, & 1 \leq j \leq b, & k = m \end{cases}$$

と定める. ここで, x と y の積 xy が現れたらこれを N に置き換えることにすれば, $g_{ijk}(x, y)$ は, x, y それぞれの冪となる単項式の和となっている. (p, q) は, $g_{ijk}(x, y)$ の法 φ^m における根になっている. つまり, 上の条件を満たす全ての (i, j, k) に対し,

$$g_{ijk}(p, q) \equiv 0 \pmod{\varphi^m}$$

が成り立つ. ここで, p を X で割った商を p_0, 余りを x_0, q を Y で割った商を q_0, 余りを y_0 とする. つまり, $p = p_0 X + x_0 (0 \leq x_0 < X)$, $q = q_0 Y + y_0 (0 \leq y_0 < Y)$ とする. (x_0, y_0) を Howgrave-Graham の補題を用いて計算し, (p_0, q_0) は, ブルートフォースで探索する. ここで,

$$t_{ijk}(x, y) = g_{ijk}(p_0 X + x, q_0 Y + y)$$

とすれば, (x_0, y_0) は, $t_{ijk}(x, y)$ の法 φ^m における根になっている. すなわち,

$$t_{ijk}(x_0, y_0) \equiv 0 \pmod{\varphi^m}$$

が成り立つ. 小さなノルムを持つ係数ベクトルを見つけ, 対応する $h(x, y)$ を作れば, $h(x_0, y_0) \equiv 0 \pmod{\varphi^m}$ から, Howgrave-Graham の補題により, $\mathbb{Z}$ 上で $h(x_0, y_0) = 0$ が成り立つ.

注意 2. 一般に 2 変数の多項式の根を計算するには, 2 つの多項式が必要になるが, x, y (p, q に対応) は $xy = N$ という関係があるので, このような $h(x, y)$ に対し, x_0 を次のような 1 変数の多項式 $h'(x)$ の根として求めることができる.

$$h'(x) = (p_0 X + x)^{m+b} h(x, N/(p_0 X + x) - q_0 Y)$$

一見すると多項式に見えないが, $t_{ijk}(x,y)$ の y に関する次数が $j+k$ であり, j の最大値は b, k の最大値は m であるから, $h(x,y)$ の y に関する次数は $m+b$ となる. よって, $h(x,N/(p_0X+x)-q_0Y)$ の分母に現れる p_0X+x の冪が $(p_0X+x)^{m+b}$ で約分されることがわかる.

$t_{ijk}(xX,yY)$ の係数を並べたベクトルからなる格子 $\mathcal{L}$ を構成する. 先にも述べたように, $xy=N$ と置き換えられるため, x, y それぞれの冪のみで構成されることに注意する.

$$1, x, y, x^2, y^2, x^3, y^3, x^4, x^5, y^4, \cdots$$

のように並べることによって下三角行列ができる. $m=3$, $a=2$, $b=1$ の場合の格子を表 6.2 に示す. $*$ は 0 ではない何らかのエントリであり, 空欄は 0 を表す.

□

補題 9. 格子のサイズ $\omega=\dim\mathcal{L}$ は, $\omega=2m+a+b+1$ となる.

証明.

$$\begin{cases} i\in\{0,1\}, & j=0, & k=0,1,\cdots m,-(1) \\ 1<i\le a, & j=0, & k=m-(2) \\ i=0, & 1\le j\le b, & k=m-(3) \end{cases}$$

であるから, (1) の場合が, $2(m+1)$ 通り, (2) の場合が, $a-1$ 通り, (3) の場合が, b 通りであるから,

$$2(m+1)+(a-1)+b=2m+a+b+1$$

となる. □

次に, $t_{ijk}(xX,yY)$ の x^l, y^l の係数を求める.

$$\begin{aligned} & t_{ijk}(xX,yY) \\ = \ & g_{ijk}(p_0X+xX,q_0Y+yY) \\ = \ & (xX)^i(yY)^jU^{m-k}(p_0X+xX \\ & +q_0Y+yY-(N+1))^k \end{aligned}$$

である. まず, (1) に対応する部分の係数を計算する. $i=1,j=0$ とおけば, $t_{10k}(xX,yY)$ の x^{k+1}（x に関して最高次）の項は,

$$(xX)U^{m-k}(xX)^k=U^{m-k}X^{k+1}x^{k+1}$$

となる. 次に $i=0,j=0$ とおけば,

$$\begin{aligned} & t_{00k}(xX,yY) \\ = \ & g_{00k}(p_0X+xX,q_0Y+yY) \\ = \ & U^{m-k}(p_0X+xX \\ & +q_0Y+yY-(N+1))^k \end{aligned}$$

となるが, この y^k（y に関して最高次）の項は,

$$U^{m-k}(yY)^k=U^{m-k}Y^ky^k$$

となる. (2) に対応する部分は,

$$\begin{aligned} & t_{i0m}(xX,yY) \\ = \ & g_{i0m}(p_0X+xX,q_0Y+yY) \\ = \ & (xX)^i(p_0X+xX \\ & +q_0Y+yY-(N+1))^m \end{aligned}$$

となる. この x^{i+m} の項は,

$$(xX)^i(xX)^m=X^{i+m}x^{i+m}(1<i\le a)$$

となる.

(3) に対応する部分は,

$$\begin{aligned} & t_{0jm}(xX,yY) \\ = \ & g_{0jm}(p_0X+xX,q_0Y+yY) \\ = \ & (yY)^j(p_0X+xX \\ & +q_0Y+yY-(N+1))^m \end{aligned}$$

となる. この y^{j+m} の項は,

$$(yY)^j(yY)^m=Y^{j+m}y^{j+m}(1\le j\le b)$$

となる.

よって, $\det\mathcal{L}$ は,

$$\det\mathcal{L}=X^{\frac{(m+a)(m+a+1)}{2}}Y^{\frac{(m+b)(m+b+1)}{2}}U^{m(m+1)} \tag{9}$$

Howgrave-Graham の補題を適用するためには,

$$2^{\frac{\omega-1}{4}}(\det\mathcal{L})^{1/\omega}<\varphi^m/\sqrt{\omega}$$

であればよい. これは,

$$\det\mathcal{L}<2^{-\frac{\omega(\omega-1)}{4}}\omega^{-\frac{\omega}{2}}\varphi^{m\omega}$$

と書き換えることができるが, $\sqrt{\omega}\le 2^{\frac{\omega-1}{2}}$, $\varphi>N/2$, $\omega-1\ge m$ であるから,

$$N^{m\omega}2^{-2\omega(\omega-1)}<2^{-\frac{\omega(\omega-1)}{4}}\omega^{-\frac{\omega}{2}}\varphi^{m\omega} \tag{10}$$

が成り立つ.

ここで, 適当な実数 u,v を用いて

$$\begin{aligned} a &= \lfloor(u-1)m-1\rfloor \\ b &= \lfloor(v-1)m-1\rfloor \end{aligned}$$

と表現しておくと, $a\le(u-1)m-1$, $b\le(v-1)m-1$ が成り立つから, $(m+a)(m+a+1)\le$

表 2: $\mathcal{L}$, $m=3$, $a=2$, $b=1$

	1	x	y	x^2	y^2	x^3	y^3	x^4	x^5	y^4
$t_{000}(xX,yY)$	U^3									
$t_{100}(xX,yY)$	*	U^3X								
$t_{001}(xX,yY)$	*	*	U^2Y							
$t_{101}(xX,yY)$	*	*	*	U^2X^2						
$t_{002}(xX,yY)$	*	*	*	*	UY^2					
$t_{102}(xX,yY)$	*	*	*	*	*	UX^3				
$t_{003}(xX,yY)$	*	*	*	*	*	*	Y^3			
$t_{103}(xX,yY)$	*	*	*	*	*	*	*	X^4		
$t_{203}(xX,yY)$	*	*	*	*	*	*	*	*	X^5	
$t_{013}(xX,yY)$	*	*	*	*	*	*	*	*		Y^4

$(um-1)um \le u^2m^2$, $(m+b)(m+b+1) \le (vm-1)vm \le v^2m^2$ となる. $X = N^{\delta_x}$, $Y = N^{\delta_y}$ とおくと式 (9) と $U \le N^{\beta}$ から,

$$\begin{aligned}
& \frac{\log_2(\det \mathcal{L})}{\log_2 N} \\
= & \frac{\log_2(X^{\frac{(m+a)(m+a+1)}{2}} Y^{\frac{(m+b)(m+b+1)}{2}} U^{m(m+1)})}{\log_2 N} \\
= & \frac{\log_2(N^{\delta_x \frac{(m+a)(m+a+1)}{2} + \delta_y \frac{(m+b)(m+b+1)}{2}} U^{m(m+1)})}{\log_2 N} \\
\le & \frac{\log_2(N^{\delta_x \frac{(m+a)(m+a+1)}{2} + \delta_y \frac{(m+b)(m+b+1)}{2} + \beta m(m+1)})}{\log_2 N} \\
= & \delta_x \frac{(m+a)(m+a+1)}{2} + \delta_y \frac{(m+b)(m+b+1)}{2} \\
& + \beta m(m+1) \\
\le & m^2 \left(\delta_x \frac{u^2}{2} + \delta_y \frac{v^2}{2} + \beta \right) + \beta m
\end{aligned}$$

$m(u+v) - 3 < \omega = 2m+a+b+1 \le m(u+v)$ であるから,

$$\begin{aligned}
& \log_2(N^{m\omega} 2^{-2\omega(\omega-1)}) \\
= & m\omega \log_2 N - 2\omega(\omega-1) \\
\ge & m(m(u+v)-3)\log_2 N - 2m^2(u+v)^2
\end{aligned}$$

となる. 以上の不等式 3 つを合わせると, Howgrave-Graham の補題が成り立つ十分条件として,

$$\begin{aligned}
& m(m(u+v)-3)\log_2 N - 2m^2(u+v)^2 \\
\le & m^2 \left(\delta_x \frac{u^2}{2} + \delta_y \frac{v^2}{2} + \beta \right) + \beta m
\end{aligned}$$

が得られる. この不等式の両辺を $m^2 \log_2 N$ で割って整理すると, 以下の不等式となる.

$$u+v-\delta_x \frac{u^2}{2} - \delta_y \frac{v^2}{2} - \beta \ge \frac{\beta+3}{m} + \frac{2}{\log_2 N}(u+v)^2 \tag{11}$$

不等式 (11) の左辺は, u, v に関する二次関数であるので, 平方完成して最大化することができる. 左辺は, $u = 1/\delta_x$, $v = 1/\delta_y$ のときに最大となる. u, v は右辺にもあるので, 同時に動くのであるが, 不等式 (11) は勝手な u, v に関して成立するので, $u = 1/\delta_x$, $v = 1/\delta_y$ とおくことができる. このとき, 不等式 (11) は,

$$\frac{1}{2\delta_x} + \frac{1}{2\delta_y} - \beta \ge \frac{\beta+3}{m} + \frac{2}{\log_2 N}\left(\frac{1}{\delta_x} + \frac{1}{\delta_y}\right)^2 \tag{12}$$

ここで, アルゴリズムをまとめておく. まず, $X = N^{\delta_x}$, $Y = N^{\delta_y}$ を (12) を満たすように取る. 次に, 与えられた p_0, q_0 に対し, $p = p_0X + x_0$, $q = q_0X + y_0$ に対応する (x_0, y_0) を既に述べたアルゴリズムで求める. よって, 問題となるのは, 最初に与える p_0, q_0 の上限である.

補題 10. $0 < \epsilon \le \delta/2$ のとき, 次の不等式が成り立つ.

$$\frac{1}{\delta-\epsilon} + \frac{1}{1-\delta-\epsilon} - 2\beta \ge \epsilon \left(\frac{1}{\delta^2} + \frac{1}{(1-\delta)^2} \right)$$

証明.

$$\begin{aligned}
\frac{1}{\delta-\epsilon} &= \frac{1}{\delta(1-\epsilon/\delta)} \\
&= \frac{1}{\delta}(1 + \epsilon/\delta + \epsilon^2/\delta^2 + \cdots) \\
&\ge \frac{1}{\delta}\left(1 + \frac{\epsilon}{\delta}\right)
\end{aligned} \tag{13}$$

同様にして,

$$\frac{1}{1-\delta-\epsilon} \ge \frac{1}{1-\delta}\left(1 + \frac{\epsilon}{1-\delta}\right) \tag{14}$$

一方, $2\beta\delta(1-\delta) \le 1$ であるから,

$$2\beta \le \frac{1}{\delta(1-\delta)} = \frac{1}{\delta} + \frac{1}{1-\delta} \tag{15}$$

(13), (14), (15) より,

$$\begin{aligned}
&\frac{1}{\delta-\epsilon}+\frac{1}{1-\delta-\epsilon}-2\beta \\
\geq\ & \frac{1}{\delta}\left(1+\frac{\epsilon}{\delta}\right)+\frac{1}{1-\delta}\left(1+\frac{\epsilon}{1-\delta}\right) \\
& -2\left(\frac{1}{\delta}+\frac{1}{1-\delta}\right) \\
\geq\ & \epsilon\left(\frac{1}{\delta^2}+\frac{1}{(1-\delta)^2}\right)
\end{aligned}$$

□

$\delta_x=\delta-\epsilon$, $\delta_y=1-\delta-\epsilon$ とおいて, 不等式 (10) を (12) に適用すると,

$$\begin{aligned}
\frac{\delta}{2} &\geq \epsilon \\
&\geq 2\left(\frac{\beta+3}{m}+\frac{2}{\log_2 N}\left(\frac{1}{\delta-\epsilon}+\frac{1}{1-\delta-\epsilon}\right)^2\right) \\
&\quad \left(\frac{1}{\delta^2}+\frac{1}{(1-\delta)^2}\right)^{-1}
\end{aligned}$$

さらに, $0<\epsilon\leq\delta/2$, $\delta<1/2$ より,

$$\frac{1}{\delta-\epsilon}\leq\frac{2}{\delta},\quad \frac{1}{1-\delta-\epsilon}\leq 4$$

であるから, 次の十分条件が得られる.

$$\begin{aligned}
\frac{\delta}{2} &\geq \epsilon \\
&\geq 2\left(\frac{\beta+3}{m}+\frac{2}{\log_2 N}\left(\frac{2}{\delta}+4\right)^2\right) \\
&\quad \left(\frac{1}{\delta^2}+\frac{1}{(1-\delta)^2}\right)^{-1} \qquad (16)
\end{aligned}$$

$m=\lfloor\log_2 N\rfloor$ とすれば, 十分大きな N に対して条件 (16) が常に成立する. また, $\epsilon=\mathcal{O}(1/\log_2 N)$ となる. このとき, p_0, q_0 の上界は, 次のようになる.

$$\begin{aligned}
p_0 &\leq \frac{p}{X}\leq N^{\delta-\delta_x}=N^{\epsilon}\approx 2^{(\log_2 N)\epsilon}\leq C \\
q_0 &\leq \frac{q}{Y}\leq 2N^{1-\delta-\delta_y}=2N^{\epsilon}\approx 2\cdot 2^{(\log_2 N)\epsilon}\leq 2C
\end{aligned}$$

$C>0$ は適当な N に依存しない定数である. この範囲の p_0, q_0 を探索すればよいことになる.

7 おわりに

本解説論文では, Coron-May[2] における, RSA 秘密鍵計算と素因数分解の決定的多項式時間同値性の証明を解説するとともに, アルゴリズムの正当性を NTL ライブラリを利用して確認した. また, その処理時間と次元の関係を実際に計測して調べた. Coron-May[2] は, RSA 暗号における基本的な問題を解いているという意味で重要な論文であるが, 格子理論の典型的かつ本質的な応用例を与えているという意味でも意義深いものである. 本稿を通じて広く知られることを望む.

参考文献

1) D. Boneh, G. Durfee, Cryptanalysis of RSA with Private Key d Less Than $N^{0.292}$, IEEE Transactions on Information Theory, 46, No.4(2000), pp. 1339-1349, Extended abstract in proceedings of Eurocrypt 1998.

2) J. S. Coron, A. May, Deterministic Polynomial-Time Equivalence of Computing the RSA Secret Key and Factoring, Journal of Cryptology 20(2007), pp.39-50.

3) M. J. Hinek, *Cryptanalysis of RSA and Its Variants*, CRC press(2009).

4) A.K. Lenstra, H.W. Lenstra, L. Lovász, Factoring Polynomials with Rational Coefficients, Mathematische Ann., Vol. 261(1982), pp. 513-534.

5) P. Nguyen, D. Stehlé, Floating-point LLL revisited, Proceedings of EUROCRYPT 2005, pp. 215-233, LNCS 3494(2005)

6) R. Rivest, A. Shamir and Adleman, A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, Communications of the ACM, Vol. 21, No.2(1978), pp. 120-126.

7) V. Shoup, NTL - A Library for Doing Number Theory, Available at `http://www.shoup.net/ntl/index.html`.

8) M. Wiener, Cryptanalysis of short RSA secret exponents, IEEE Trans. Inform. Theory 36(1990), pp.553-558.