解　説

数論アルゴリズムとその応用

素因数分解と離散対数問題アルゴリズム†

小山謙二††　静谷啓樹†††

1. はじめに

数論の分野で古くから難問とされている代表的な問題に素因数分解問題と離散対数問題がある．素因数分解問題とは，整数n (>1) が与えられたとき，$n=pq$ となるような1より大きい整数 p, q を求める問題である．この問題の効率的な解法が分かれば，大きな整数 n も最終的に素数の積に分解できる．離散対数問題とは，大きな有限体での対数を計算する問題，つまり整数 a, g と素数 p が与えられたとき，$a=g^x \bmod p$ なる整数 x があれば，x を求める問題である．なお，g が原始根（後述）であれば，x が必ず一つだけ存在する．この問題の効率的な解法が分かれば，大きな有限体での対数が求まる．これらの二つの問題は，いくつかの代表的な現代暗号（公開鍵暗号）に用いられ，暗号解読の難しさがこれらの問題を解く難しさに対応している．しかし，素因数分解問題と離散対数問題が「多項式時間」で解けるやさしい問題のクラスに属するか，それよりも時間を要する非常に難しい問題のクラスに属するかは明らかにされていない．現在知られている最良の解法でその計算量が見積もられ，準指数関数的な時間を要することが分かっている．素因数分解と離散対数を解く組織的な解法（アルゴリズム）は1970年代後半から着実に進歩している．この背景には暗号解読や暗号の最適設計の動機があったかもしれないが，計算機科学の真理追究の面も大きかったと思われる．

本稿では，素因数分解と離散対数問題に対する有望なアルゴリズムのいくつかを紹介し，その計算量，数値実験の状況，関連する話題を述べる．

2. 有望な素因数分解法

2.1 概　要

合成数 n を小さな素数 ($2, 3, 5, \cdots$) で順に割っていく方法は，n が大きくなると効率的でない．この試行割算法の計算時間は素因数 p に比例し ($O(p)$ と表す)，最悪の場合 $O(\sqrt{n})$ の計算時間を要する．実際，現在の計算機能力では p が 10^{12} 程度以下の場合にのみ有効である．大きな数 n の素因数分解に有効なアルゴリズムはいずれも，適当な z を求め，n と z の最大公約数 $p=\mathrm{GCD}(n, z)$ を計算して n の因数 $p(\neq 1)$ を求めることが基本となっている．最大公約数の計算は n が大きくてもユークリッドの互除法を用いれば簡単にできる点がポイントである．問題は z の求め方であり，n と互いに素でない z を求めるのにさまざまな手法が提案されている[19]．素因数分解のアルゴリズムは，その計算量が合成数 n のサイズのみによって決まる合成数依存型と，n の（未知の）素因数 p の性質によって決まる素因数依存型に分けられる．素因数依存型のアルゴリズムとしては，試行割算法，Pollard の ρ 法（ロー法），Pollard の $p-1$ 法，Guy の $p+1$ 法，H. W. Lenstra[29] によって発明され，Montgomery[38] らによって改良された楕円曲線法がある．ρ 法は，適当な多項式 f と初期値 x_1 から

$$x_{i+1}=f(x_i) \bmod n$$

を計算して，$\mathrm{GCD}(n, x_{2j}-x_j)$ より素因数 p を求める．この計算量は $O(\sqrt{p})$ なので，試行割算法よりも優れている．$p-1$ 法の計算量は，n の素因数 p に対し，$p-1$ の最大素因数のサイズに比例する．つまり，$p-1$ が小さな素数の積になっている場合に有効である．$p+1$ 法の計算量は，n の

† Algorithms for Factorization and Discrete Logarithm Problems by Kenji KOYAMA (NTT Communication Science Laboratory) and Hiroki SHIZUYA (Tohoku University, Education Center for Information Processing).

†† NTT コミュニケーション科学研究所
††† 東北大学情報処理教育センター

素因数 p に対し，$p+1$ の最大素因数のサイズに比例する．つまり，$p+1$ が小さな素数の積になっている場合に有効である．楕円曲線法は，種数(genus) が1の代数曲線である楕円曲線を用いた素因数分解法であり，素因数 p を求める平均的かつ漸近的な計算量は

$L_p[1/2, \sqrt{2}]$

である．ここで，関数 $L_p[a, b]$ は

$L_p[a, b]=\exp((b+o(1))(\log p)^a(\log\log p)^{1-a})$

と定義され ($o(1)$ は1を超えない定数)，主要なアルゴリズムの計算量評価に用いられている．$n=pq$ かつ p と q のサイズが同じ場合，楕円曲線法の計算量は

$L_n[1/2, 1.020]$

となる．合成数依存型の主要な素因数分解アルゴリズムは，2次合同式をうまく利用している．つまり，2次合同式 $s^2\equiv t^2 \bmod n$ を満たす s と t を求め，$z=s\pm t$ として最大公約数 GCD(n, z) を計算して素因数を求める．この2次合同式法の具体的なアルゴリズムとしては，1975年に Morrison-Brillhart が発明した連分数法，1977年に Schroeppel が発明した線形ふるい法，1983年に Pomerance[48] が発明し，Silverman[52] が改良した2次ふるい法，1990年に Lenstra 兄弟と Manasse, Pollard[30] が発明した数体ふるい法がある．線形ふるい法の計算量は

$L_n[1/2, 1.117]$

であり，2次ふるい法の計算量は

$L_n[1/2, 1.020]$

である．数体ふるい法は，合成数が

$n=c^k\pm s$ (c と s は小さく k は大きい)

の形に限定された場合のみ適用可能とはいえ，計算のオーダを下げる画期的な発明であった．その平均的な計算量は

$L_n[1/3, 2(2/3)^{2/3}]=L_n[1/3, 1.526\cdots]$

である．この数体ふるい法の有効性は数値実験でも実証された (**3.** 参照)．数体ふるい法を一般形の合成数の素因数分解に適用できるような試みも最近盛んに行われている．最初に汎用数体ふるい法を提案したのは，Buhler, Lenstra と Pomerance で，その計算量は

$L_n[1/3, 3^{2/3}]=L_n[1/3, 2.080\cdots]$

である．さらに Adleman[2] と Lenstra は独立に改良アルゴリズムを提案し，その計算量は

$L_n[1/3, 4\cdot 3^{-2/3}]=L_n[1/3, 1.922\cdots]$

である．さらに Coppersmith[9] は

$L_n[1/3, 1.901]$

なる計算量のアルゴリズムを提案している．現在，汎用数体ふるい法による素因数分解は理論解析が先行しており，数値実験での有効性はいまだ実証されていない．

2.2 2次ふるい法

目標とする合成数を n とする．まず，

$Q(x)=(x+m)^2-n,\ m=\lfloor\sqrt{n}\rfloor$

なる関数を導入し，$x=0, \pm1, \pm2, \cdots$ を代入して $Q(x)$ の値を求める．$Q(x)$ は比較的小さい数なので，小さな素因数に分解できる．これらの素因数分解例を集めて掛け合わせ，小さな素因数のべき指数がすべて偶数になるようにする．つまり，

$s^2\equiv t^2 \pmod n$

を満たす s と t を求める．ここで，$(x+m)$ を掛け合わせた部分が s となり，$Q(x)$ の小さな素因数を掛け合わせた部分の平方根が t になる．$z=s\pm t$ として GCD(n, z) を計算すると，n の素因数が求まる．以上の方法が2次ふるい法の基本である．関数 $Q(x)$ を改良した複数多項式2次ふるい法も提案されている[52]．最近は，n-cube 計算機上の超並列処理で2次ふるい法を実現する方法も検討されている[44]．

[例] $n=2201$ の素因数を2次ふるい法で求める．

$Q(x)=(x+46)^2-2201$

なる関数が定義できる．

$Q(1)=47^2-2201=8=2\cdot2\cdot2$

$Q(3)=49^2-2201=200=2\cdot2\cdot2\cdot5\cdot5$

を掛け合わせると，

$47^2\cdot49^2\equiv2^6\cdot5^2 \pmod{2201}$

が成立する．

$s=47\cdot49=2303,\ t=2^3\cdot5=40$

となるので，

GCD$(s-t, n)=(2263, 2201)=31$

として素因数31が求まる．

2.3 楕円曲線法

有限体 $\mathbf{Z}_p$ (p は素数) 上の楕円曲線 E_p

$E_p: y^2\equiv x^3+ax+b \pmod p$

の点の集合 (無限遠点の零点 $\mathcal{O}=(\infty, \infty)$ を含む) は加法群をなす．点と点の加算公式は文献25) の第6章または文献22) を参照のこと．なお，E_p

と等価な楕円曲線

E_p': $By^2 \equiv x^3 + Ax^2 + x \pmod p$

の点を有理整数とみなした加算公式（漸化式）もある[26),38)]．有限群 E_p（または E_p'）の位数（点の個数）を S とすると，すべての点 $\mathbf{P}$ に対し，$S\mathbf{P}=\mathcal{O}$ となる．つまり点 $\mathbf{P}$ を S 回加算した点は零点になる．M が S の倍数のときも $M\mathbf{P}=\mathcal{O}$ となる．

楕円曲線法のキーポイントの第1は，目標の合成数を n としたときに，mod n での楕円曲線 E_n を利用することである．正確に言うと E_n は有限体上の曲線ではなく，有限環 $\mathbf{Z}_n(=\mathbf{Z}/n\mathbf{Z})$ 上の曲線であるが，あえて有限体上の曲線とみなして加算公式を適用する．n の素因数は未知であり，E_n の位数 S の正確な値も未知であるが，S のオーダは大体 n のオーダとほぼ同じである．まず S の倍数と思われる非常に大きな数 M を適当に定める．たとえば，100万以下の素数の積として

$M=2^{33}\cdot 3^{20}\cdot 5^{13}\cdots 499^2\cdot 503\cdots 999983$

とする．ここで，2のような小さな素数は S に多く含まれると予想されるので，S のオーダを勘案してべき指数を定める[26)]．次に E_n 上の初期点 $\mathbf{P}$ を適当に定める．そして点 $M\mathbf{P}$ を計算する．点 $M\mathbf{P}$ が零点ならば，$M\mathbf{P}$ を求める過程での加算公式適用時の分母 d，または零点の分母 d が n と互いに素でなくなる．つまり GCD(d,n) を計算して n の素因数が求まる．点 $M\mathbf{P}$ が零点でないならば，a,b または A,B を変えて別の楕円曲線を選び，同様に計算を進める．すなわち第2のキーポイントは，楕円曲線の係数パラメータを変えるとその位数が一定の範囲で振れることを利用している．

2.4 数体ふるい法

数体ふるい法は，2次ふるい法の概念を代数的整数上で実現したものである．代数的整数論の専門用語や記号の定義と意味については，たとえば文献 20)，40) などを参照されたい．

n を合成数，$f\in\mathbf{Z}_n[X]$ を次数 k の既約モニック多項式とする．m を整数とし，$n=f(m)$ を分解する．f の根の一つを $\alpha\in\mathbf{C}$ とする．代数体 K を $K=\mathbf{Q}(\alpha)$ とし，$\mathcal{O}_K$ を K の整数環とする．$\mathcal{O}_K=\mathbf{Z}[\alpha]$ は素元分解環とする．

数体ふるい法の原理は，φ: $\mathbf{Z}[\alpha]\to\mathbf{Z}_n$ なる環準同型を

$f(\alpha)\to f(m) \bmod n$

として，次の図式に示すように，2通りの分解の違いを利用している．

$c+d\alpha\to\mathcal{O}_K$ での分解

$\varphi\downarrow \qquad \downarrow\varphi$

$c+dm\to\mathbf{Z}$ での分解

なお，有理整数についてスムーズという概念があるように，代数的整数にもスムーズという概念を拡張できる．すなわち，有理整数または代数的整数がスムーズであるとは，その素因子がすべて因子基底 $\mathcal{B}$ に含まれることである．因子基底 $\mathcal{B}$ は，次の二つの集合 $\mathcal{B}_\mathbf{Q}$ と $\mathcal{B}_K$ から構成されている．

$\mathcal{B}_\mathbf{Q}$: ある上限 B より小さな有理素数（代数的整数と区別するために，以下では$\mathbf{Q}$の素数を特に有理素数という）．

$\mathcal{B}_K$: Bより小さいノルムをもつ $\mathcal{O}_K$ の1次の素イデアル（実際には1次の素数を $\mathcal{B}_K$ にとる），及び $\mathcal{O}_K$ の単元（可逆元）の基本集合．

この場合，$c+d\alpha\in\mathcal{O}_K$ のノルム $N(c+d\alpha)$ は

$N(c+d\alpha)=|(-d)^k f(-c/d)|$

によって計算される．

(c_i,d_i) を互いに素な有理整数とする．c_i+d_im は $\mathcal{B}_\mathbf{Q}$ スムーズであり，かつ $c_i+d_i\alpha$ が $\mathcal{B}_K$ スムーズとする．c_i+d_im 及び $c_i+d_i\alpha$ が完全に分解されるようなペア (c_i,d_i) を数多く集め，

$\Pi(c_i+d_im)=s^2$

$\Pi(c_i+d_i\alpha)=t^2$

なる関係式を求める．そしてGCD$(s\pm t,n)$ を計算して n の素因数を求める．

[分解例]

$f(X)=X^3+2$，$\alpha=-2^{1/3}$，$m=11$，

$n=11^3+2=1333$ とする．

いま，$c=2,d=1$ とすると，

$c+\alpha \quad\to\quad (1-\alpha)\alpha(1+\alpha)$

$\varphi\downarrow \qquad\qquad \downarrow\varphi$

$c+11=13\to 13\equiv(-10)\cdot 11\cdot 12 \pmod n$

という違いがでる．これらの分解を集めて s と t を求める．数体ふるい法の全体をとおした数値例は文献 12) に詳しい．

3. 素因数分解の数値実験

3.1 数値実験の現状

有望な素因数分解アルゴリズムを使った数値実験もここ10数年間盛んに行われるようになった．

素因数分解のテストデータは，簡明な式で表されるランダムな数が適当であり，米国数学会のカニンガム・プロジェクトでは

$c^k \pm 1 (c=2, 3, 5, 6, 7, 10, 11, 12)$

の形をした大きな数（カニンガム数）が選ばれている．我が国では，円分数を素因数分解するプロジェクトが森本光生（上智大学）らによって進められ，1980年代後半以降の結果が上智大学数学講究録として出版されている[39]．カニンガム数の素因数分解表は20世紀初頭から更新され続け，1988年に単行本（第2版）として出版された[7]．Sam Wagstaff (Purdue Univ. Computer Science Dept., West Lafayette, Ind. 47907 U.S.A.) がこの表の管理を行っており，更新表を興味のある人々に年に2度送っている．その表には，まだ素因数が分かっていない合成数も列挙され，未知の素因数の発見が世界的に競われている．数多くの素因数を発見している常連は，上記の Wagstaff のほか，Silverman (MITRE 社)，Montgomery (Burroughs 社)，A.K. Lenstra (Bell Core) と Manasse (DEC 社）らの米国勢がおり，日本からは，陶山弘実，木田祐司（立教大学)，小山謙二 (NTT) らが表の更新に貢献している．素因数が未知の最小合成数は最重要指名手配 (Most Wanted Numbers：その時点での世の中の素因数分解の能力を示す，いわば世界記録のようなもの）とされ，その桁数は計算機とアルゴリズムの進歩により年々大きくなっている．たとえば，2^k-1 の形の合成数の場合，1983年の60桁から1992年の103桁へと伸びている．計算時間の理論式から考察すると，100 桁程度の合成数の場合，合成数の桁数が10増えると，素因数分解にかかる時間は約10倍必要となる．

いくつかの実験例を紹介しよう．

（1） 1984 年に米国サンディア研究所の Simmons はスーパコンピュータ CRAY-1 を用いて，2次ふるい法により $2^{251}-1$ を3個の素因数に分解した．要したCPU時間は約32時間であった．

（2） Silverman は24台の Sun 3 ワークステーションを結び付け，2次ふるい法により 90 桁前後の合成数を数多く素因数分解している．計算機の空き時間を利用すると約6週間で素因数分解できるという．

（3） 1988 年に小山はスーパコンピュータ CRAY-2 を用いて，改良楕円曲線法により 215 桁の合成数 $2^{273}-1$ を分解し，24桁の素因数を発見した[27]．

（4） 1990年6月に A.K. Lenstra と Manasse は米国を中心とした数カ国の約 1000 台の計算機を E-mail 通信で接続し，数体ふるい法により約3カ月をかけて9番目のフェルマ数 $F_9=2^{512}+1$ を素因数分解した[31),32)]．F_9 は 10 進 155 桁の数であり，そのうちの7桁の素因数は昔から知られていたが，残りの148桁の数を49桁と99桁の素因数に分解した．この合成数はフェルマ数ということで特別な関心がもたれていたものである．なお，同じ計算機環境と時間を費やして2次ふるい法を適用すると110桁の合成数が分解できるだけである．

（5） 1992年6月時点での各アルゴリズムのチャンピオン・データは以下のとおりである．2次ふるい法で素因数分解された最大サイズの合成数は，$10^{142}+1$ の因数である116桁の合成数であり，Lenstra と Manasse が数多くの計算機を用いて発見した．楕円曲線法で素因数分解された合成数[12]のうち，最大サイズの素因数は，$10^{201}-1$ の 42 桁の素因数であり，Rusin が1台の SparcStation を用いて発見した．なお，合成数のサイズと素因数のサイズがともに大きな場合は素因数分解が非常に難しく，2番目の素因数が最大の発見数 (Largest penultimate prime factor) として更新表にまとめられている．そのチャンピオンは，$2^{467}-1$ の因数である 133 桁の合成数に対して，58 桁と 76 桁の二つの素因数に分解したものであり，Silverman が数台の計算機を用いて数体ふるい法により発見した．

3.2 指名手配の数

RSA 暗号が発明された1977年に発明者の一人である Rivest は 129 桁の数 n を素因数分解する問題を懸賞問題として Scientific American 誌上で提出している．賞金は100ドルであるが，1992年10月現在いまだ解かれていない．この n は二つの素数の積である．

$n=$ 114381625757888867669235779976146612
010218296721242362562561842935706935
245733897830597123563958705058989075
147599290026879543541

129 桁の数を素因数分解するには 100 桁の数を素因数分解するのに比べ，約 400 倍の計算能力が必

要である．したがって，この懸賞問題はあと数年以内に解かれるかもしれない．

ところで，懸賞金は付いていない指名手配のカニンガム数の捜査状況をみてみよう．1991年3月時点での指名手配のカニンガム数は16個であった[28]．その後1992年6月時点でそのうちの11個が素因数分解されている．たとえば，$2^{445}-1$ の因数である103桁の合成数（[2, 445-, c103] と表す）が指名手配であったが，49桁と55桁の素因数に分解された．残りの未発見の5個の数は

[7, 163+, c117], [10, 149+, c123],
[11, 131−, c134], [11, 127+, c120],
[12, 137−, c123]

である．これらのほかに現時点での未発見の小さな合成数の例を以下に示す．

$6^{169}+1=7\cdot53\cdot937\cdot4057\cdot37571$
$\cdot1167200703423487 21\cdot c101$

$6^{172}+1=1297\cdot1721\cdot1904729\cdot5993513377$
$\cdot3449220534377\cdot c99$

$12^{127}+1=13\cdot7621\cdot1373887\cdot2567179$
$\cdot25804859677522145597 59\cdot c99$

$12^{136}+1=17\cdot17\cdot97\cdot6529\cdot260753\cdot277169$
$\cdot3145763540507811991784369\cdot c104$

ただし，$6^{169}+1$ は132桁であり，すでに7や53などの31桁分の素因数は求まっていることに注意されたい．c101などの合成数は大きな素因数を含んでいる可能性が非常に高いが，安価で高速なワークステーションなどを使って，読者の方々も挑戦してみましょう．

4. 離散対数の概念

4.1 離散対数問題の定義

離散対数 (discrete logarithm) は初等整数論の分野では指数 (index) として古くから知られていた概念である．まずこの指数を定義しよう．

有限素体 $\mathbf{Z}_p=\mathbf{Z}/p\mathbf{Z}=\{0,1,\cdots,p-1\}$（以下では代表元のみを考える）において，$\mathbf{Z}_p^*=\mathbf{Z}_p\setminus\{0\}$ は位数 $p-1$ の巡回群となり，その生成元 $g\in\mathbf{Z}_p^*$ を mod p の原始根と呼ぶ．たとえば $p=7$ のとき，$g=3$ を選ぶと，$\langle g\rangle=\{g^i \bmod p \mid 0\le i<p-1\}$ という巡回群は $\{1,3,2,6,4,5\}$ となり，集合として $\mathbf{Z}_p^*$ と一致するので，$g=3$ は mod 7 の原始根の（一つ）であることが分かる．一般に，ある $g\in\mathbf{Z}_p^*$ が mod p の原始根であるためには，$p-1$ のすべての素因数 $p_1,\cdots,p_l$ に対して $g^{(p-1)/p_i}\not\equiv1 \pmod p$ $(1\le i\le l)$ であることが必要十分である．そこで，mod p の原始根 g を一つ固定すれば，任意の $a\in\mathbf{Z}_p^*$ に対して，$a\equiv g^r \pmod p$ となる $r\in\mathbf{Z}_{p-1}$ が一意的に定まる．この r を，底 g に対する a の指数と呼び，

$$r=\mathrm{ind}_g a$$

と書く．r がなぜ $\mathbf{Z}_{p-1}$ の元であるかは，$\mathbf{Z}_p^*$ の任意の元 a に対して $a^{p-1}\equiv1 \pmod p$ であることによる．すなわち，$p-1$ 乗は0乗と同じ効果であり，したがってべきの部分の代数構造は $p-1$ を法とする剰余環 $\mathbf{Z}_{p-1}$ とみなしてよい．

ここで，ind_g と対数との類似性は明らかであろう．たとえば，

$$\mathrm{ind}_g ab\equiv\mathrm{ind}_g a+\mathrm{ind}_g b \pmod{p-1},$$
$$\mathrm{ind}_g a^k\equiv k\cdot\mathrm{ind}_g a \pmod{p-1},$$

などが成り立つ．詳しく言えば，ind_g は $\mathbf{Z}_p^*$ から $\mathbf{Z}_{p-1}$ への同型を与えており，これはたとえば自然対数 $\log_e$ が，乗法群としての正実数の集合 $\mathbf{R}_{>0}$ から加法群としての実数の集合 $\mathbf{R}^+$ への同型を与えるのと類似のことである．そして，指数の概念は $\mathbf{Z}_p^*$ の上に限らず，一般の巡回群の上に容易に拡張される．すなわち，群 G が定まっているとき，$g\in G$ に対して $H=\langle g\rangle$（g で生成される G の巡回部分群）とすれば，任意の $a\in H$ について $\mathrm{ind}_g a$ が定義できる．

要するに，$\mathrm{ind}_g a$ の具体的な値を計算すること，そのことこそが離散対数問題なのである．すなわちわれわれは今日，理論計算機科学の分野において指数を離散対数と言い直しているに過ぎない．ここで，あらためて離散対数問題を定義する．

離散対数問題：群 G と $g\in G$ が固定されたとき，与えられた $a\in H=\langle g\rangle\subseteq G$ に対して，$g^x=a$ となる整数 x を求めよ．

ついでに記号も言い直して，$x=\log_g a$ という記法を採用することにする．

4.2 暗号理論との関係

十分小さな位数の群では，簡単に離散対数問題が解ける．なぜなら，群のすべての元の対数をまえもって計算して表にしておけばよいからである．重要なことは，大きな位数の群では一般に離散対数の計算が難しいという点である．すなわち，離散対数問題を解く効率的なアルゴリズムが発見されていないことが，離散対数問題の研究を

興味深いものにしている主要因であり，また暗号への応用という実際的な領域への進出をも可能にしている．

たとえば，離散対数問題の難しさを応用した代表的なプロトコルとして，Diffie-Hellman[11] のプロトコルがある．これは，A，Bという二人が安全でない（盗聴される可能性のある）通信路を介して情報を交換し，最終的には安全に秘密情報を共有するための方式である．

いま，$f(x)=g^x \bmod p$ と定義すると，f は実際的な意味での一方向性関数となっている．そもそも，一方向性関数が最初に暗号系に応用されたのは 1960 年代のことと言われており，関数の構成にはすでに離散対数問題が用いられていたという[35]．また離散対数問題は暗号化関数を具体的に構成する際の重要な構成要素として，暗号方式にしばしば応用される．離散対数問題の暗号系への応用に関する現在までの成果は，池野・小山[19]，辻井・笠原[51]，及び黒澤・藤岡・宮地[21]に詳しく述べられている．

4.3 さまざまな群と離散対数問題

さて，一般の群Gについて離散対数問題が定義される（**4.1**）が，応用面から言うと，群 G の元同士の演算（たとえば $a\cdot b \bmod p$）が容易な群が興味の対象となる．つまり，G の二項演算（$G\times G\to G$）が効率的に計算できること，G は有限群であること，などがGの自然な要件となる．そのような群の族は多数存在するのだが，具体的には，有限体の乗法群 $\mathbf{Z}_p^*$, $\mathbf{F}_q^*(=\mathbf{F}_q\setminus\{0\}, q=p^n)$，あるいは剰余環の乗法群 $\mathbf{Z}_n^*$ を G としてとるのが，歴史的にみても代表例と言える．一方，Koblitz[23] と Miller[36] は独立に，楕円曲線上の離散対数問題を考え出した．これは有限体 $\mathbf{F}_q$ の上の楕円曲線 E の点（と無限遠点）のなすアーベル群 $E(\mathbf{F}_q)$ を G としてとる離散対数問題である．Koblitz[24] はこれをさらに一般化し，有限体 $\mathbf{F}_q$ で定義された超楕円曲線 C（種数 $g>2$）のヤコビ多様体 $\mathbf{J}(C;\mathbf{F}_{q^n})$ をGとしてとる離散対数問題を定義した（ヤコビ多様体はアーベル群の一種）．これを楕円曲線上の離散対数問題と呼ぶ．

このように，一見すると初等整数論の範囲と思われた離散対数問題は，現在では代数幾何の概念をも巻き込んでおり，それを解くアルゴリズムには代数的整数論の手法が駆使されるまでに拡がりをみせている．さまざまなGに関する個別のアルゴリズムをみることも大切であるが，ここでは，最も基本的と思われる $G=\mathbf{Z}_p^*$，特に $G=H=\mathbf{Z}_p^*$（すなわち底 g が $\bmod p$ の原始根）の場合にできるだけ限定して，この離散対数問題に対するいくつかのアルゴリズムとその評価を述べる．さらに，関連する研究状況も説明する．

5. 離散対数問題を解くアルゴリズム

5.1 概　要

離散対数問題のアルゴリズムは，その基本的な手法によって次の2種類に大別することができる．

1. $H=\langle g\rangle\subseteq G$ としたとき，H の位数 $\sharp H$ に依存して $\log_g a$ を求めるアルゴリズムで，$\sharp H$ の最大素因数のサイズの指数時間オーダの実行時間となるもの．

2. $G=\mathbf{F}_q^*(q=p^k)$ として，指数計算法 (index calculus method) と呼ばれる手法で $\log_g a$ を求めるアルゴリズムで，有限体 $\mathbf{F}_q$ のサイズの準指数時間オーダの実行時間となるもの．

前者に属するものとしては，Shanks[49]，Pohlig-Hellman[45]，Pollard[46] のアルゴリズムが知られている．

後者に属するものとしては，Adleman[1]，Coppersmith[8]，ElGamal[13]，Pomerance[47]，Coppersmith-Odlyzko-Schroeppel（ガウスの整数法）[10]，Gordon[14),15] のアルゴリズムが知られている．これらのアルゴリズムが特に有効に働く有限体 $\mathbf{F}_{p^k}$ とその実行時間の関係の大略を述べると，$k=1$ のときはガウスの整数法が $L_p[1/2, c]$ で実行でき，$k>1$ で標数 p が比較的大きいときは ElGamal のアルゴリズムが $L_{p^k}[1/2, c]$ で，p が小さいとき（たとえば $p=2$ の場合）は Coppersmith のアルゴリズムが $L_{p^k}[1/2, c]$ で実行できる．ここに L_{p^k} は **2.1** で定義した関数で，c はそれぞれ小さな定数である．

ここでは前者の代表例として，Shanks, Pohlig-Hellman のアルゴリズムを紹介し，後者に関しては，指数計算法の基本的な手法と Adleman のアルゴリズムを述べた後に，特に最近になって開発された Gordon のアルゴリズムを重点的に紹介する．

5.2 Shanks のアルゴリズム

Shanks[49] は，有限群の元の位数を計算するためのアルゴリズムを示したが，これは離散対数問題を解くアルゴリズムとしても働くことが分かった (Knuth の指摘による)．このアルゴリズムは一般の有限群 G を対象としているので，$\mathbf{Z}_p^*$ はもちろん，$E(\mathbf{F}_q)$ や $\mathbf{J}(C;\mathbf{F}_{q^n})$ の上の離散対数問題にも適用できる．以下，一般の有限群を扱う形でアルゴリズムを記述するが，適宜 $\mathbf{Z}_p^*$ の場合を注意することにする．なお，Shanks のアルゴリズムは baby-step-giant-step アルゴリズムとも呼ばれている．

アルゴリズム (Shanks):

$g\in G$ とし，$H=\langle g\rangle\subseteq G$ の位数 $\sharp H$ を n とする．H に対して $f:H\to\{1,\cdots,n\}$ なる単射が存在し，$\log n$ の低次多項式時間 (たとえば $O(\log n)$) で計算できるとする ($G=H=\mathbf{Z}_p^*$ ならば $n=p-1$ で，f は自明)．各 $a\in H$ について，$\log_g a$ は区間$[1,n]$ に入るので，$m=[n^{1/2}]$，$0\le r\le m$，$0\le q\le m$ とすると，$\log_g a=mq-r$ なる表現が可能である．$\log_g a$ を求めるには r,q を求めれば十分である．

Step 1: 次の集合 S,T を計算する ($G=H=\mathbf{Z}_p^*$ ならば f を考えなくてよい)．

$$S=\{(i,f(ag^i))\mid i=0,\cdots,m\},$$
$$T=\{(i,f(g^{mi}))\mid i=0,\cdots,m\}$$

Step 2: S の各元を第2成分についてソートする．T についても同様にソートする．

Step 3: ソートされた S,T について，S の元の第2成分と T の元の第2成分が一致しているものを探索する．これを $s=(r,f(ag^r))\in S$，$t=(q,f(g^{mq}))\in T$ とする．

Step 4: s,t を発見したら，このとき $ag^r=g^{mq}$ が成立し (f は単射)，$a=g^{mq-r}$，したがって $\log_g a=mq-r$ が得られた．■

このアルゴリズムでは，S と T のソートに $O(m\log m)$ 回の大小比較，S と T の生成に $O(m\log n)$ 回の群の二項演算を要するので (f の計算時間は $O(\log n)$ としている)，結局全体の計算時間は $O(n^{1/2}\log n)$ となる．($G=H=\mathbf{Z}_p^*$ の場合は $O(p^{1/2}\log p)$.) すなわち，$\log n$ の指数関数になる．さらに，S,T の $O(n^{1/2})$ 個の元を格納する記憶領域も必要である．Pollard[46] も Shanks のアルゴリズムと同等の実行時間の確率的アルゴリズムを示したが，記憶領域は Shanks のものよりもずっと小さくできる．しかし，いずれの場合でも実行時間は指数関数になる．

実は $\mathbf{Z}_p^*$ 上の離散対数問題に関する限り，これを解くアルゴリズムで，数論的に証明されていない命題 (つまり予想) を仮定せずに厳密な実行時間が明らかになっているのは，この Shanks のアルゴリズムと，次に述べる Pohlig-Hellman のアルゴリズム，それに Pomerance のアルゴリズム (Gordon のアルゴリズムの項で触れる) だけである．

5.3 Pohlig-Hellman のアルゴリズム

このアルゴリズムを Pohlig-Hellman[45] と呼ぶのは慣例に従ったものであるが，Silver によっても独立に発見されている (実際，McCurley[35] や Koblitz[25] は Silver-Pohlig-Hellman のアルゴリズムと呼んでいる)．このアルゴリズムは一般の有限群 G の上の離散対数問題を解くものであるが，$H=\langle g\rangle\subseteq G$ の位数 $\sharp H$ が特別な性質をもったときに特に強力に働き，実行時間は $\log\sharp H$ の多項式オーダとなる．ただし，任意の位数の離散対数問題を多項式時間で解くものではない．以下，一般的な形で記述するが，例によって $G=H=\mathbf{Z}_p^*$ の場合を時々注意する．

アルゴリズム (Pohlig-Hellman):

$n=\sharp H=\sharp\langle g\rangle$ とし，n は y-スムーズであるとする．すなわち n の素因数はどれも y 以下である ($G=H=\mathbf{Z}_p^*$ ならば $p-1$ が y-スムーズである必要がある)．そこで，

$$n=\prod_{i=1}^{k} q_i^{e_i},\ q_1<\cdots<q_k\le y$$

という素因数分解が既知であるとする．離散対数 $x=\log_g a$ は明らかに $\mathbf{Z}_n$ の元で，n の素因数分解が分かっているのだから，剰余環の直積環への分解 (中国剰余定理)：

$$\mathbf{Z}_n\cong\mathbf{Z}_{q_1^{e_1}}\otimes\cdots\otimes\mathbf{Z}_{q_k^{e_k}}$$

によって，各 $\mathbf{Z}_{q_i^{e_i}}$ における $\log_g a$ の値が求まれば，$\log_g a\in\mathbf{Z}_n$ が求まる．

以下では，各 $\mathbf{Z}_{q_i^{e_i}}(1\le i\le k)$ について同じアルゴリズムを繰り返すので，添字 i を除き，$q^e\mid n$，$q^{e+1}\nmid n$ とし，$x=\log_g a \bmod q^e$ とする．いま，

$$x=\sum_{j=0}^{e-1} b_j q^j,\ b_j\in\mathbf{Z}_q$$

とおくと，$b_j(0\le j\le e-1)$ が求まればよい．そこ

で次を計算する．

• $a^{n/q}$ と $\gamma=g^{n/q}$ を計算する．

• $\gamma^i=a^{n/q}$ となる i を探す．これは，$i=0,1,\cdots$ と次々に試せばよい．発見したら，$b_0=i$ とする（$a^{n/q}=g^{nb_0/q}$ に注意）．$e>1$ ならば，さらに以下を続ける．

• $a_1=ag^{-b_0}$ として，$\gamma^i=a_1^{n/q^2}$ となる i を探し，$b_1=i$ とする．

• $a_2=a_1g^{-b_1q}$ として，$\gamma^i=a_2^{n/q^3}$ となる i を探し，$b_2=i$ とする．

…（以下 b_{e-1} まで実行）

以上から $b_0,\cdots,b_{e-1}$ が求まり，$x=\log_g a \bmod q^e$ が定まる．これらを $\mathbf{Z}q_i^{e_i}(1\leq i\leq k)$ について実行すれば，各 $x=\log_g a \bmod q^e$ から $x=\log_g a \bmod n$ が求まる．■

このアルゴリズムの実行時間は，群の二項演算の回数で表現すると $O(\sum_{i=1}^{k} e_i(\log n+q_i))$ であり，n の最大素因数 $\max\{q_1,\cdots,q_k\}$ のサイズの指数関数オーダとなる．しかし，n が $O(\log n)$-スムーズであるとすれば，ただちに $O((\log n)^2)$ を得る．したがって，群 $H\subseteq G$ の位数 n が $O(\log n)$-スムーズとなるような離散対数問題は多項式時間で解けることになる．これは $G=H=\mathbf{Z}_p^*$ の場合，$p-1$ が $O(\log p)$ 程度の素因数のみに分解される場合に相当する．逆に言えば，$p-1$ が大きな素因数をもてば，Pohlig-Hellman のアルゴリズムは効率的に働かないことになる．特に $p-1=2q$ で q が素数のときが最も非効率的で，実行時間は指数関数時間になってしまう．このように，$2q+1$ が素数となるような素数 q は Sophie Germain の素数と呼ばれている．

5.4 Gordon のアルゴリズム

このアルゴリズムは，指数計算法（index calculus method）として分類されるアルゴリズムの一種であり，実行時間の評価の際に若干の未証明事項を仮定しなければならないが，高速である．ただし，高速とは言っても多項式時間までには及ばず，準指数関数時間を要する．

一般に，指数計算法のアルゴリズムは2部構成となっており，第1部（Stage 1）では，$H=\langle g\rangle\subseteq G$ から適切に選んだ部分集合（因子基底）の離散対数を求めてデータベースとして蓄積し，第2部（Stage 2）では，このデータベースを利用して実際に $\log_g a$ を求めている．すなわち，

指数計算法の一般的アルゴリズム：

Stage 1：$H=\langle g\rangle\subseteq G$，$\#H=n$ とする．因子基底として $\mathcal{B}=\{q_1,\cdots,q_m\}\subseteq H$ を定め，

$$g^{b_i}=\prod_{j=1}^{m} q_j^{a_{ij}}$$

と分解される b_i を探す．両辺の対数をとれば

$$b_i\equiv\sum_{j=1}^{m} a_{ij}\log_g q_j \pmod n$$

となり，これは $\log_g q_i$ を未知数とする方程式とみなせる．このような b_i が多数集まり，$\#\{b_i\}\geq\#\mathcal{B}$ となったとする．このとき，$\mathbf{Z}_n$ 上の行列 $A=(a_{ij})$ の階数が $m(=\#\mathcal{B})$ となれば，$\log_g q_j(1\leq j\leq m)$ に関する線形方程式は一意に解けて，$\mathcal{B}$ の各元の離散対数が分かる．

Stage 2：ランダムに r を選び，ag^r が次のように分解されるかどうかを検査する．

$$ag^r=\prod_{j=1}^{m} q_j^{e_j}$$

失敗したら，上式を満たすまで別の乱数を選ぶ．成功したら，

$$\log_g a=\sum_{j=1}^{m} e_j\log_g q_j-r$$

となり，$\log_g a$ が求まる（$\log_g q_j$ は Stage 1 で既知であることに注意）．■

Adleman[1] は $G=H=\mathbf{Z}_p^*$ の離散対数問題に対して，因子基底 $\mathcal{B}$ として $u=L_p[1/2,\ 0]$ 以下の素数の集合を設定するアルゴリズムを示した．すなわち，Stage 1 では b_i をランダムに選び，g^{b_i} がスムーズになるような b_i だけをふるいにかけることになる（スムーズかどうかの検査自体は，あらかじめ定めた $\mathcal{B}$ の各元で次々と割り算を試行するだけなので高速に実行可能である）．このように $\mathcal{B}$ を選ぶと，g^{b_i} がスムーズになる確率の評価に若干の仮定が入るが，Stage 1 と Stage 2 を合わせた総合的な実行時間は $L_p[1/2,\ c](c\approx 1)$ となる．一方 Pomerance[47] は，Adleman のアルゴリズムを改良して，仮定なしの厳密な実行時間が評価できるアルゴリズムを与えた．その速度は，オーダ的には Adleman のアルゴリズムとほぼ同等で，$L_p[1/2,\ \sqrt{2}]$ である．

$G=H=\mathbf{Z}_p^*$ の離散対数問題に対する指数計算法に関しては，Adleman のアルゴリズム以後は画期的な進展はなかったが，最近になって Gordon[14),15)] は素因数分解のために開発された数体ふるい法[30)]の概念を応用し，従来のものより高

速な準指数関数時間アルゴリズムを2種類与えた．一つは一般数体ふるい法に基づくもので，他方は特殊数体ふるい法によるものである．「一般」と「特殊」は，アルゴリズム自体はほとんど同じであるが，その違いは $\mathbf{Z}_p$ の標数 p の性質による．すなわち，「一般」の場合は文字どおり一般の p に対して適用可能であり，「特殊」の場合は，ある性質をもつ p に対して適用可能である．「一般」の実行時間は

$$L_p[1/3,\ 3^{2/3}]=L_p[1/3,\ 2.08008]$$

であるのに対して，「特殊」のそれは

$$L_p[2/5,\ 1.00475]$$

である．つまり，漸近的には「一般」のほうが「特殊」のほうより高速である．しかし実際には，p が 10 進数で約 320,000 桁以下ならば「特殊」のほうが速い．したがって離散対数問題に基づく暗号系に用いられる現実的な素数のサイズを高速領域としてカバーしている点で「特殊」のほうが影響が大きい．そこでここでは「特殊」版を紹介する．特殊性の名の由来となる p の特別な性質については，以下で述べる．

さて，特殊版では **2.2**（3）で定義した整係数既約モニック多項式 f（次数は k）を次のように選ぶ．実は，逆にこのような f の選択が可能な p を，このアルゴリズムでは特殊な素数と言っているのである．

1. $f\in\mathbf{Z}_p[X]$ の係数は適当に小さい．
2. ある整数 x, y が存在し，その大きさはどちらも $p^{1/k}$ 程度で，$y^k f(x/y)\equiv 0 \pmod p$ を満たす．
3. $\mathcal{O}_K=\mathbf{Z}[\alpha]$ は一意分解整域．

たとえば $p=5$, $f(X)=X^2+1\in\mathbf{Z}_p[X]$ とすると，$x=2$, $y=1$ がとれて，$y^2f(x/y)\equiv 0 \pmod p$, $\alpha=\sqrt{-1}$ であり，これは $K=\mathbf{Q}(\alpha)$ として虚2次体をとったことになり，その整数環 $\mathcal{O}_K$ はガウスの整数環 $\mathbf{Z}[\sqrt{-1}]$ となる．このとき，

$$N(c+d\alpha)=(-d)^2f(-c/d)=c^2+d^2.$$

Gordon のアルゴリズムで用いる因子基底 $\mathcal{B}$ は，素因数分解で用いる数体ふるい法と同様に定義する．ここでは離散対数の底 g はスムーズであると仮定する．もしそうでないならば，スムーズな別の底 g' を選び，後に変換

$$\log_g a=\log_{g'} a/\log_{g'} g \bmod p-1$$

を使えばよい．

アルゴリズム (Gordon)：

Stage 1：互いに素な有理整数のペア (c, d) で，$cy+dx$ 及び $c+d\alpha$ がともにスムーズなものを集める．すなわち，たとえば c を固定し，d を広い範囲で動かしてふるいにかける（数体ふるい法の名の由来）．ここでスムーズな $c+d\alpha$（因子がすべて $\mathcal{B}_K$ の元）は，そのノルム $N(c+d\alpha)$ がスムーズ（素因子がすべて $\mathcal{B}_\mathbf{Q}$ の元）であればよいことが証明されるので，

$$cy+dx=\prod_{s\in\mathcal{B}_\mathbf{Q}} s^{w_s(c,d)},$$

$$|N(c+d\alpha)|=\prod_{s\in\mathcal{B}_\mathbf{Q}} s^{v_s(c,d)}$$

となるような (c, d) のペアを集めることになる．ここで，$w_s(c,d)$, $v_s(c,d)\geq 0$ であり，(c, d) を付けて書いたのは，w_s, v_s が (c, d) に依存していることを表現する識別子である．このとき，イデアル $(c+d\alpha)$ について

$$(c+d\alpha)=\prod_{\mathfrak{p}\in\mathcal{B}_K} \mathfrak{p}^{v_\mathfrak{p}(c,d)}$$

が成り立つが，この場合は $(c+d\alpha)$ は整数 $c+d\alpha$ で，また各 $\mathfrak{p}\in\mathcal{B}_K$ はその生成元で置き換えてよいので，準同型 $\varphi: \mathbf{Z}[\alpha]\to\mathbf{Z}_p$ を用いれば，結局

$$(cy+dx)\varphi(y(c+d\alpha))^{-1}\equiv\prod_{s\in\mathcal{B}} s^{u_s(c,d)}\equiv 1 \pmod p$$

が得られる．これにより，因子基底の離散対数 $\log_g s\ (s\in\mathcal{B})$ を求めるのに必要な方程式の一つ

$$\sum_{s\in\mathcal{B}} u_s(c,d)\log_g s\equiv 0 \pmod{p-1}$$

が得られる．このような方程式を $\#\mathcal{B}$ 以上集めて，その係数行列の階数が $\mathbf{Z}_{p-1}$ 上で $\#\mathcal{B}$ であれば，$\log_g s\ (s\in\mathcal{B})$ について解ける．

Stage 2：因子基底 $\mathcal{B}$ の各元の離散対数が求まったので，具体的に $x=\log_g a$ を求める．そのために，問題を有理素数 q_i（大きさは $O(p^{1/k})$ あるいは x, y よりずっと小さい）の離散対数を求める問題に帰着させる．

Step 1　まず $r\in\mathbf{Z}_{p-1}\setminus\{0\}$ をランダムに選び，$g^r a \bmod p$ が v-スムーズかどうかを検査する．この検査は $g^r a \bmod p$ を楕円曲線法[29]により素因数分解することで行われる．（スムーズでないなら別の r を選び，スムーズになるまでこれを繰り返す．）$g^r a \bmod p=q_1^{t_1}\cdots q_\ell^{t_\ell}\ (q_\ell\leq v)$ とする．

Step 2　次に，各有理素数 q_i について，$(cy+dx)/q_i$ 及び $N(c+d\alpha)$ がともに $\mathcal{B}$-スムーズになるような (c, d) のペアを一つ発見する（たとえば

d を固定し，$c=c_0+eq_i$ としてふるいにかける)．

Step 3 $\mathcal{B}$ の各元の離散対数は Stage 1 で既知であるから，これより各 q_i について $\log_g q_i$ が求まるので，これより

$$\log_g a=\sum t_i \log q_i - r \bmod p-1 \quad \blacksquare$$

このアルゴリズムで，$B=L_p[2/5, \delta]$ として $\delta(>0)$ と f の次数 k を注意深く選び，また $v=L_p[3/5, (1/100)^{1/5}]$ とすれば，スムーズになる確率の評価に若干の仮定が入るが，その実行時間は $L_p[2/5, 1.00475]$ となる．

実は Gordon のアルゴリズム(「特殊」版)は，Coppersmith-Odlyzko-Schroeppel[10] による「ガウスの整数法」(Gaussian integer method) の拡張になっている．実際，ガウスの整数法は Gordon のアルゴリズムにおいて，$k=2$ で K が虚2次体の場合に相当する．ただし，ガウスの整数法の実行時間は $L_p[1/2, 1/(2c)](c\geq 1)$ と評価されている．

5.5 暗号方式への影響

Gordon のアルゴリズムは，$\mathbf{Z}_p$ の標数 p が特殊な性質をもつ素数 (**5.4** で述べた3条件を満たす f が選べるような素数) であるときに有効に働く．ある p が与えられたとき，それが「特殊」であるかどうかはただちには判定できないが，ひとたびそれが特殊であることを知れば (具体的に都合の良い f を入手できれば)，Gordon のアルゴリズムが強力に働き，従来のアルゴリズムでは困難と思われた離散対数問題が，特殊性を知っている者にだけは簡単に解けてしまう可能性がある．すなわち，素数 p に対するこの特殊性の情報が落し戸 (trapdoor) となり，悪用される恐れがある．

1991年，米国国立標準技術局 (NIST)[55] は，離散対数問題の困難さに安全性の根拠をおくディジタル署名方式の標準化案 (DSS) を発表した．DSS で用いられる 512 ビットの素数 p は，$p-1$ が 160 ビットの素数 q を因数にもつ条件が付けられている．p と q はランダムに選んでよいが，他人から与えられた p と q には Gordon 流の落し戸が仕掛けられているかもしれない[56]．しかし，落し戸の可能性に関して言えば，512 ビットの長さの素数のうち，Gordon の「特殊」版アルゴリズムによって危険になるような素数の割合は 2^{-100} 未満であることが判明し[15]，人間の恣意性を排除するプロトコルで素数を決定するようにすれば，落し戸の可能性は問題にならないとされている．

6. 離散対数問題の数値実験

どれほど大きな位数の有限群の離散対数問題が現実の計算機で実際に解けるのかについては，さまざまな数値実験が試みられている[16),18),33)]．素体 $\mathbf{Z}_p$ 上の離散対数問題に関しては，LaMacchia-Odlyzko[33] が 192 ビットの素数のもとで解いた離散対数問題が現在のところは最高記録である．アルゴリズムとしては，ガウスの整数法が用いられた．標数2の拡大体の場合，Gordon-McCurley[17] が $\mathbf{F}_{2^{401}}$ のもとで解いた離散対数問題が最高記録である．また，McCurley[35] は，Diffie-Hellman プロトコル (**4.2** 参照) の通信履歴から，最終的に共有された秘密情報を言い当てる問題を 100 ドルの懸賞問題としている．問題は次のとおり．

懸賞問題： 素数 p を $p=2\cdot 739\cdot q+1$ (ただし $q=(7^{149}-1)/6$ で，q は素数) とし，mod p の原始根 g として $g=7$ をとる．A と B の二人は Diffie-Hellman のプロトコルを忠実に実行し，A から B へは $y_A=g^a \bmod p$ が，B から A へは $y_B=g^b \bmod p$ が送られた．ここに，

$y_A=$ 127402180119973946824269244334322849
749382042586931621654557735290322914
679095998681860978813046595166455458
144280588076766033781

$y_B=$ 180162285287453102444782834836799895
015967046695346697313025121734059953
772058475958176910625380692101651848
662362137934026803049

である．このとき $y=g^{ab}=y_A{}^b=y_B{}^a \bmod p$ を求めよ．

1992年10月1日現在，この問題はまだ解かれていない．離散対数 $a=\log_g y_A$ または $b=\log_g y_B$ が求まれば y が求まる．ただし，Diffie-Hellman のプロトコルを破ることと離散対数問題を解くことが等価であるかどうかは未解決である[3]．したがって，離散対数 a や b を求めずに y が求まってしまう可能性を否定できない．

7. 離散対数問題と計算量理論

離散対数問題と他の問題の帰着関係を調べることにより，離散対数問題の相対的な難しさが分かる．このような帰着関係は未解決の帰着関係を含めて，Adleman-McCurlty[3], Woll[54] によって網羅

的に調べられている．たとえば，$\mathbf{Z}_n^*$ (n は素数とは限らない) の上の離散対数問題と n の素因数分解問題を比較すると，後者は前者に確率的に Turing 帰着することが分かっている．すなわち，$\mathbf{Z}_n^*$ 上の離散対数問題を解く多項式時間アルゴリズムの存在を仮定すれば，n の素因数分解問題は確率的多項式時間で解ける．また，有限体 $\mathbf{F}$ で定義された楕円曲線の点のなすアーベル群上の離散対数問題は，その有限体の乗法群上の離散対数問題よりも難しいであろうという予想があったが，Menezes-岡本-Vanstone[37),42)] は，超特異 (supersingular) と呼ばれる楕円曲線の場合には，Weil-pairing という写像を応用して，$\mathbf{F}$ の拡大体の乗法群上の離散対数問題に帰着できることを証明した．

離散対数問題を構造的計算量理論の枠組のなかで最初に検討したのは Brassard[6)] である．まず，離散対数問題は計算問題であるから，その計算問題と，言語 L へのインスタンスの所属の判定問題が等価になるように L を定める．そして，L が所属する計算量のクラスを解明すれば，計算量のクラスによる離散対数問題の特徴づけが可能になる．その結果，$\mathbf{Z}_p^*$ 上の離散対数問題は NP∩co-NP で特徴づけられることが示された．一方，静谷-伊東-桜井[50)]は，楕円曲線上及び超楕円曲線上の離散対数問題について同様の特徴づけを行い，前者は NP∩co-NP，後者は NP∩co-AM となることを示した．AM とは Arthur-Merlin ゲームと呼ばれる対話証明系をもつ言語の1クラスで，NP の確率的な拡張になっており，また BPNP とも呼ばれる．さらに，岡本-桜井[41)]は，特別な場合を除いてほとんどの超楕円曲線では，その上の離散対数問題は NP∩co-NP となることを示した．なお，一般の有限群の上の離散対数問題に対する同様の特徴づけは岡本-桜井-静谷[43)]によって検討されている．

このほか，離散対数問題は計算量理論のさまざまなトピックスに関連している．たとえば，暗号学的に安全な擬似乱数の生成[4),21)]，離散対数のハードビットの長さ[34)]，離散対数問題の困難さの仮定による計算量クラスの一致の証明[5)]，ランダム自己帰着性と完全零知識証明[53)]などである．これらの詳細については，各論文を参照されたい．

8. むすび

素因数分解と離散対数問題のアルゴリズムは数論の中心的なテーマの一つであり，過去 20 年間に着実に進歩してきた．今後，汎用の高速アルゴリズムが発明されるだろうか．特殊な高速アルゴリズムが開発された場合，どのような条件が合成数や素数に課せられるのだろうか．将来が楽しみな分野である．

参考文献

1) Adleman, L. M.: A Subexponential Algorithm for the Discrete Logarithm Problem with Applications to Cryptography, Proc. 20th FOCS, pp. 55-60 (1979).
2) Adleman, L. M.: Factoring Numbers Using Singular Integers, Proc. of STOC (1991).
3) Adleman, L. M. and McCurley, K. S.: Open Problems in Number Theoretic Complexity, Discrete Algorithms and Complexity: Proc. of the Japan-U. S. Joint Seminar, Academic Press, pp. 237-262 (1987).
4) Blum, M. and Micali, S.: How to Generate Cryptographically Strong Sequences of Pseudo-random Bits, SIAM J. of Comput., Vol. 13, pp. 850-864 (1984).
5) Bellare, M., Micali, S. and Ostrovsky, R.: The (true) Complexity of Statistical Zero-Knowledge, Proc. 22nd STOC, pp. 494-502 (1990).
6) Brassard, G.: A Note on the Complexity of Cryptography, IEEE Trans. Inf. Theory, Vol. IT-25, No. 2, pp. 232-233 (1979).
7) Brillhart, J., Lehmer, D. H., Selfridge, J. L., Tuckerman, B. and Wagstaff, S. S. Jr.: Factorizations of $b^n \pm 1$, $b=2, 3, 5, 6, 7, 10, 11, 12$ up to High Powers, Contemporary Mathematics, 22, Second Edition (1988).
8) Coppersmith, D.: Fast Evaluation of Logarithms in Fields of Characteristic Two, IEEE Trans. Inform. Theory, Vol. IT-30, pp. 587-594 (1984).
9) Coppersmith, D.: Modifications to the Number Field Sieve, to appear in J. Cryptology.
10) Coppersmith, D., Odlyzko, A. M. and Schroeppel, R.: Discrete Logarithms in $GF(p)$, Algorithmica, Vol. 1, pp. 1-15 (1986).
11) Diffie, W. and Hellman, M.: New Directions in Cryptography, IEEE Trans. Information Theory, Vol. 22, pp. 472-492 (1976).
12) Dixon, B. and Lenstra, A. K.: Massively Parallel Elliptic Curve Factoring, Proc. of Eurocrypt '92 (1992).
13) ElGamal, T.: A Subexponential-Time Algorithm for Computing Discrete Logarithms Over $GF(p^2)$, IEEE Trans. Inform. Theory, Vol. IT-31, pp. 473-481 (1985).

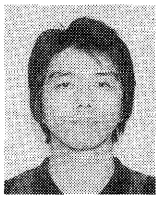
14) Gordon, D. M.: Discrete Logarithms in $GF(p)$ Using the Number Field Sieve, to appear in SIAM Journal on Discrete Math.
15) Gordon, D. M.: Designing and Detecting Trapdoors for Discrete log Cryptosystems, to appear in Advances in Cryptology: Proc. Crypto '92, LNCS, Springer-Verlag.
16) Gordon, D. M. and McCurley, K. S.: Massively Parallel Computation of Discrete Logarithms: Proc. Crypto '92, LNCS, Springer-Verlag.
17) Gordon, D. M. and McCurley, K. S.: Computation of Discrete Logarithms in $GF(2^n)$, presentation at Crypto '91, Santa Barbara (1991).
18) Harper, G., Menezes, A. and Vanstone, S. A.: Public-Key Cryptosystems with Very Small Key Length, to appear in Advances in Cryptology: Proc. Eurocrypt '92, LNCS, Springer-Verlag (1992).
19) 池野信一, 小山謙二: 現代暗号理論, 電子情報通信学会 (1986).
20) 石田 信: 代数的整数論, 森北出版 (1974).
21) 黒澤 馨, 藤岡 淳, 宮地充子: 暗号理論への応用, 本特集号, pp. 195-206.
22) 木田祐司, 牧野潔夫: 素数判定アルゴリズム, 本特集号, pp. 150-156.
23) Koblitz, N.: Elliptic Curve Cryptosystems, Math. Comp., Vol. 48, No. 177, pp. 203-209 (1987).
24) Koblitz, N.: Hyperelliptic Cryptosystems, J. Cryptology, Vol. 1, No. 3, pp. 139-150 (1989).
25) Koblitz, N.: A Course in Number Theory and Cryptography, GTM 114, Springer-Verlag, New York (1987).
26) 小山謙二: 高速楕円曲線法による素因数分解, 信学論(D), J 70-D, 12, pp. 2730-2738 (1987).
27) 小山謙二: 楕円曲線法の高速化とその素因数分解実験, 信学技報, ISEC 88-19 (1988).
28) 小山謙二: 暗号と素数, 数学セミナー, 特集・素数っておもしろい, 1991年10月号, pp. 48-52.
29) Lenstra, H. W. Jr.: Factoring Integers with Elliptic Curves, Annals of Mathematics 126, pp. 649-673 (1987).
30) Lenstra, A. K., Lenstra, H. W. Jr., Manasse, M. S. and Pollard, J. M.: The Number Field Sieve, Proc. 22nd STOC, pp. 564-572 (1990).
31) Lenstra, A. K. and Manasse, M. S.: Factoring by Electric Mail, Lecture Notes in Computer Science 434 (1990).
32) Lenstra, A. K. and Manasse, M. S.: Factoring with Two Large Primes, Lecture Notes in Computer Science 473, pp. 72-82 (1991).
33) LaMacchia, B. and Odlyzko, A. M.: Computation of Discrete Logarithms in Prime Fields, Designs, Codes and Cryptography, Vol. 1, pp. 47-62 (1991).
34) Long, D. L. and Wigderson, A.: The Discrete Log Hides $O(\log n)$ Bits, SIAM J. Comput., Vol. 17, pp. 363-372 (1988).
35) McCurley, K. S.: The Discrete Logarithm Problem, Cryptology and Computational Number Theory, Proc. Symposia in Applied Mathematics, Vol. 42, AMS, pp. 49-74 (1990).
36) Miller, V. S.: Use of Elliptic Curves in Cryptography, Advances in Cryptology: Proc. Crypto '85, LNCS 218, Springer-Verlag, pp. 417-426 (1985).
37) Menezes, A., Okamoto, T. and Vanstone, S. A.: Reducing Elliptic Logarithms to Logarithms in a Finite Field, Proc. 23rd STOC, pp. 80-89 (1991).
38) Montgomery, P. L.: Speeding the Pollard and Elliptic Curve Methods of Fractorization, Math. Comp., pp. 243-264 (1987).
39) 森本光生, 木田祐司, 小林美千代: 円分数の素因数分解 (その3), 上智大学数学講究録, No. 35, ISSN 0914-3378 (1992).
40) 小野 孝: 数論序説, 裳華房 (1987).
41) Okamoto, T. and Sakurai, K.: Efficient Algorithms for the Construction of Hyperelliptic Cryptosystems, Advances in Cryptology: Proc. Crypto '91, LNCS 576, Springer-Verlag, pp. 267-278 (1991).
42) 岡本龍明, 桜井幸一: 代数幾何学的アルゴリズム, 本特集号, pp. 180-188.
43) Okamoto, T., Sakurai, K. and Shizuya, H.: How Intractable is the Discrete Logarithm Problem for a General Finite group?, to appear in Advances in Cryptology: Proc. Eurocrypt '92, LNCS, Springer-Verlag (1992).
44) Peralta, R.: Quadratic Sieve on the n-Dimensional Cube, Proc. of Crypto '92 (1992).
45) Pohlig, S. and Hellman, M.: An Improved Algorithm for Computing Logarithms over $GF(p)$ and Its Cryptographic Significance, IEEE Trans. Information Theory, Vol. 24, pp. 106-110 (1978).
46) Pollard, J. M.: Monte Carlo Methods for Index Computation mod p, Math. Comp., Vol. 32, pp. 918-924 (1978).
47) Pomerance, C.: Fast, Rigorous Factorization and Discrete Logarithm Algorithm, Discrete Algorithms and Complexity: Proc. of the Japan-U. S. Joint Seminar, Academic Press, pp. 119-143 (1987).
48) Pomerance, C.: The Quadratic Sieve Algorithm, Lecture Notes in Computer Science 209, pp. 169-182 (1985).
49) Shanks, D.: Class Number, a Theory of Factorization, and Genera, Proc. Symposium Pure Mathematics, AMS (1972).
50) Shizuya, H., Itoh, T. and Sakurai, K.: On the Complexity of Hyperelliptic Discrete Logarithm Problem, Advances in Cryptology: Proc. Eurocrypt '91, LNCS 547, Springer-Verlag, pp. 337-351 (1991). (Final version in Trans. of the IEICE, Vol. E-74, No. 8, pp. 2129-2135 (1991).)
51) 辻井重男, 笠原正雄 (編著): 暗号と情報セキュリティ, 昭晃堂 (1990).

52) Silverman, R. D.: The Multiple Polynomial Quadratic Sieve, Math. Comp., 48, pp. 243-246 (1987).
53) Tompa, M. and Woll, H.: Random Self-Reducibility and Zero Knowledge Interactive Proofs for Possession of Information, Proc. 28th FOCS, pp. 472-482 (1987).
54) Woll, H.: Reductions among Number Theoretic Problems, Information and Computation, Vol. 72, pp. 167-179 (1987).
55) Specifications for a Digital Signature Standard, National Institute for Standards and Technology, Federal Information Processing Standard Publication XX, draft (1991).
56) Debating Encryption Standards, CACM, Vol. 35, No. 7, pp. 32-54 (1992).

(平成4年10月28日受付)

小山 謙二 (正会員)

1949年生. 1972年京都大学工学部電気工学科卒業. 1974年同大学院修士課程修了. 同年NTT入社. 1985～86年ウォータルー大学客員教授. 現在, コミュニケーション科学研究所主幹研究員. 工学博士. 電子情報通信学会論文賞, 著述賞, 科学技術庁長官賞, 本会ベストオーサ賞各受賞. 暗号理論と情報セキュリティに興味をもつ. 著書「現代暗号理論」など. 電子情報通信学会, IACR 各会員.



静谷 啓樹

1981年東北大学工学部通信工学科卒業. 1987年同大学院工学研究科博士課程修了. 工学博士. 同年東北大学情報処理教育センター助手. 1990年同工学部通信工学科助手. 1991～92年モントリオール大学理学部情報科学科招聘教授. 1992年東北大学情報処理教育センター助教授, 現在に至る. 暗号理論, 構造的計算量理論に興味をもつ. ACM, IACR, IEEE, IEICE, SITA 各会員.