暗号の中の数学

青木 和麻呂

maro at isl.ntt.co.jp

NTT

青木って誰?

- 1969年生まれ
- 早大数学修士卒(寺田研)
- パソコン歴25年、インターネット歴16年
- ソフトウェア実装のスピード狂
- 共通鍵暗号Camellia設計者の一人
- 最近は巨大整数の素因数分解をして喜んでいるらしい
- 漫画・アニメ好き

NTTでどんな生活しているの?

- 計算機のおもり
- 論文査読
- 論文書き
- 特許書き

暗号と特許

特許web pageから

http://www.jpo.go.jp/seido/s_tokkyo/chizai04.htm

＜特許法の保護対象＞

特許法第2条に規定される発明、すなわち、**自然法則を利用**した技術的思想の創作のうち高度のものを保護の対象とします。したがって、金融保険制度・課税方法などの人為的な取り決めや**計算**方法・**暗号**など自然法則の利用がないものは**保護の対象とはなりません**。(以下略)

数学の特許化

http://www.jpo.go.jp/shiryou/kijun/kijun2/tokuteigijutu_index.html

【コンピュータにより自然数nから$n+k$までの和を求める装置】

自然数nと$n+k$を入力する手段と、入力されたnを記憶するn記憶手段と、入力された$n+k$を記憶する$n+k$記憶手段と、n記憶手段からnを、$n+k$記憶手段から$n+k$を取得しkを演算する手段と、該kを記憶するk記憶手段と、自然数nから$n+k$までの和sを上記n記憶手段、k記憶手段に記憶されたn、kを用いて$s=(k+1)(2n+k)/2$により求める演算手段と、演算結果を出力する手段とを備えたことを特徴とする、コンピュータにより自然数nから$n+k$までの和を求める装置。

数学と情報の気質

<KUSAKARI.98Jul22003232@is15e0s13.jaist.ac.jp>

数学科の教授が言いました「答はある」

工学部の教授が反論しました「あるってだけじゃ意味がない，実際に答を求めなくては」

情報科の教授が反論しました「求められると言うだけでは意味がない，早く求められなくては」

現代暗号

- 情報理論的安全 (Shannon 49)
  「鍵」は平文の長さ以上必要
- アルゴリズム公開・計算量的安全 (DES 76)
  秘密にするのは「鍵」のみ

証明できている (暗号に限らない) 関数の計算量の下界の最大は、入力長の線形オーダー

やっぱり公開鍵暗号?

$$C = E_{K_P}(M) \quad \text{暗号化}$$
$$M = D_{K_S}(C) \quad \text{復号}$$

M: 平文、C: 暗号文
K_P: 公開鍵、K_S: 秘密鍵

- $\forall M[M = D_{K_S}(E_{K_P}(M))]$
- K_P から K_S を求めるのは計算量的に困難

公開鍵暗号の例: RSA 暗号

公開鍵 (N, e)

秘密鍵 (N, d)

暗号化 $C = M^e \bmod N$

復号 $M = C^d \bmod N$

但し、 $N = pq$**、** p, q**: 素数、** $p \neq q$**、**
$\gcd(\varphi(N), e) = 1$**、** $d \equiv e^{-1} \pmod{\varphi(N)}$

- (N, e) **から** (N, d) **を求めるのは大変そう**
- **攻撃者には群** $(\mathbb{Z}/N\mathbb{Z})^*$ **位数不明**

鍵共有の例: DH

共通パラメータ	$\mathbb{Z}/p\mathbb{Z}$, g ($\in \mathbb{Z}/p\mathbb{Z}$)
Alice 公開鍵、秘密鍵	$y_A = g^{x_A}$, x_A
Bob 公開鍵、秘密鍵	$y_B = g^{x_B}$, x_B
共有鍵	$y_B^{x_A} = y_A^{x_B} = g^{x_A x_B}$

- (y_A, y_B) から $g^{x_A x_B}$ を求めるのは大変そう

ベキ演算 (1/2)

「$M^e \bmod N$」はどう求めるか。

1. M を e 回かける
2. “うまい”方法で M^e 計算
   しかる後に $\bmod N$
3. “うまい”方法で $M^e \bmod N$ を計算

ベキ演算 (2/2)

[binary method]

represent $\sum_{i=0}^{\lfloor \log_2 e \rfloor} e_i 2^i = e \quad (e_i \in \{0, 1\})$

$X \leftarrow 1, Y \leftarrow M$

for $i \leftarrow 0$ **to** $\lfloor \log_2 e \rfloor$

  if $e_i = 1$ **then**

    $X \leftarrow X \cdot Y \bmod N$

  $Y \leftarrow Y^2 \bmod N$

output X

一方向性関数: $y = f(x)$

- x から y を求めるのは簡単
- y から $y = f(x)$ を満たす x を求めるのは難しい

- ぐちゃぐちゃな $f \Rightarrow$ 暗号学的ハッシュ関数
- 落し戸つき $\Rightarrow$ RSA 型 (素因数分解ベース)、DH 型 (離散対数ベース)

その他の**うまい**方法はないのか?

素因数分解問題

入力: 合成数 N

出力: 非自明な因子 p **(**$1 < p < N, p \mid N$**)**

- **一般の場合効率の良いアルゴリズム(**N **のビット数に関する多項式時間) は知られていない**

素因数分解アルゴリズムの計算量

方法	計算量	経験的有効領域
TD	$L_p[1, 1]$	$p \leq 2^{28}$
ECM	$L_p[1/2, 1.414]$	$p \leq 2^{130}$
MPQS	$L_N[1/2, 1.020]$	$N \leq 2^{320}$
SNFS	$L_N[1/3, 1.526]$	$N > 2^{320}$
GNFS	$L_N[1/3, 1.923]$	$N > 2^{320}$
MPGNFS	$L_N[1/3, 1.902]$	$N > 2^{2000}$(?)

$$L_x[s, c] = \exp((c + o(1))(\log x)^s(\log\log x)^{1-s})$$

GNFSの記録

合成数	ビット数	発表日	分解者
RSA-200	663	05/05	Bonn大ら
RSA-640	640	05/11	Bonn大ら
c176 in $11^{281}+1$	582	05/05	NTTら
RSA-576	576	03/12	Bonn大ら
c164 in $2^{1826}+1$	545	03/12	NTTら
RSA-160	530	03/04	Bonn大

出典 http://www.crypto-world.com/FactorAnnouncements.html等

SNFSの記録

合成数	ビット数	発表日	分解者
c274 in $6^{353}-1$	911(913)	06/01	NTTら
c248 in $2^{1642}+1$	822	04/03	NTTら
$2^{809}-1$	809	03/01	Bonn大
c244 in $5^{349}-1$	809(811)	06/04	Kruppa+Bonn大
c239 in $2^{811}-1$	793(811)	04/06	NFSNET
c234 in $3^{491}+1$	777(779)	04/09	NFSNET+CWI
c227 in $2^{773}+1$	774(753)	00/11	CWIら

出典 http://www.crypto-world.com/FactorAnnouncements.html等

ECMの記録

合成数	$\log_2 p$	発表日	分解者
c180 in $3^{466}+1$	219	05/04	Dodson
c311 in $10^{311}-1$	212	05/09	Aoki+Shimoyama
c175 in $3^{533}+1$	209	05/11	Kruppa
c187 in $2^{2034}+1$	205	05/04	Dodson
c242 in $2^{1099}+1$	197	05/10	Dodson
c162 in $10^{233}-1$	194	05/02	Dodson

出典 http://www.loria.fr/~zimmerma/records/top100.html

数体篩法の流れ

find $x, y \in \mathbb{Z}$ **s.t.** $x^2 \equiv y^2 \pmod{N}$

	処理	分散計算	GNFS176 での利用
1	多項式選択	容易	52 台
2	篩	容易	400 台
3	filtering	比較的容易	2 台
4	線形代数	密結合	36 台
5	平方根	比較的容易	36 台

プログラムの流れ

	GNFS176 での時間	
多項式選択	20d	pol51m0b → pol51opt
	2h	mkprime
篩	27d	ltsieve
filtering	4h	classifyRel → uniqRel, 32to64
	3h	getLP → countLP → lptxt2bin
	2h	sfctr
	8h	scmpi
	1h	compff → mkprematrixbin
	2d	splitpm + smerge
線形代数	1h	shufflematrix → mkmatrixbin
	1h	cut224mat → splitmatrix
	5d	planczos256
	1h	solve224mat → rff → gaussext
平方根	1h	anneal
	1h	papprox
	1h	pcouveignes, rsqrt

プログラム行数

処理	行数	割合
多項式選択	5626	10%
篩	16943	30%
filtering	17607	32%
線形代数	7352	13%
平方根	8150	15%
合計	55678	100%

2005年10月時点

数体篩法の原理

$$f_1(x), f_2(x) \in \mathbb{Z}[x] \quad f_1(\theta_1)=f_2(\theta_2)=0 \text{ in } \mathbb{C}$$

1. f_1, f_2 は$\mathbb{Z}$上既約、$\gcd(f_1, f_2) = 1$
2. $\exists M \in \mathbb{Z}[f_1(M) \equiv f_2(M) \equiv 0 \pmod{N}]$

$$\begin{array}{ccc} a + b\theta_1 : \mathbb{Z}[\theta_1] \text{ での分解} & \overset{\theta_1 \leftarrow M}{\longrightarrow} & \mathbb{Z} \text{ での分解} \\ \uparrow\downarrow & & \| \\ a + b\theta_2 : \mathbb{Z}[\theta_2] \text{ での分解} & \overset{\theta_2 \leftarrow M}{\longrightarrow} & \mathbb{Z} \text{ での分解} \end{array}$$

数体篩法の概要

find many $(a, b) \in \mathbb{Z}^2$ **s.t.**

$$\begin{cases} \left|(-b)^{\deg f_1} f_1(-\frac{a}{b})\right| = \prod_{p<B_1} p^{e_p^{(a,b)}} \\ \left|(-b)^{\deg f_2} f_2(-\frac{a}{b})\right| = \prod_{q<B_2} q^{e_q^{(a,b)}} \end{cases}$$

find dependency in $\mathbb{F}_2$

$$\{[e_p^{(a,b)}, \ldots, e_q^{(a,b)}, \ldots]\}_{(a,b)}$$

$$\Rightarrow x^2 \equiv y^2 \pmod{N}$$

数体篩法: 多項式選択

given N, find $f_1, f_2 \in \mathbb{Z}[x], M \in \mathbb{Z}$ s.t.

- $f_1(M) \equiv f_2(M) \equiv 0 \pmod{N}$
- f_1, f_2 は$\mathbb{Z}$上既約、$\gcd(f_1, f_2) = 1$
- $i = 1, 2$ のどちらでも $(-b)^{\deg f_i} f_i(-a/b)$ が smooth になる (a, b) を効率よく見つけられる

「x がy-smooth」$\overset{\text{def}}{\Leftrightarrow}$「$p \mid x, p$: 素数$\Rightarrow p \leq y$」

多項式選択の例: M 進展開

d を fix (N が世界記録付近なら $5 \sim 6$)

1. $M \leftarrow \lceil N^{1/(d+1)} \rceil$
2. $\sum_{i=0}^{d} c_i M^i = N$ $(0 \leq c_i < M)$ と書く
3. $f_1(x) = x - M$, $f_2(x) = \sum_{i=0}^{d} c_i x^i$ と書く

M 進展開の問題点

- |多項式の係数| $\leq M \approx N^{1/(d+1)}$
- 多項式の係数の積(多項式の表現法の種類) $\approx N^{(d+2)/(d+1)}$

|多項式の係数| $\approx N^{1/(d+2)}$
となる f_1, f_2 はありそう。

どう求める???

数体篩法: 平方根処理

入力 $(a_i, b_i) \in \mathbb{Z}^2$, θ, M
($f(\theta) = 0$ in $\mathbb{C}$, $f(M) \equiv 0 \pmod{N}$)

出力 $\sqrt{\prod(a_i + b_i\theta)}\Big|_{\theta \leftarrow M} \bmod N$

- $\prod(a_i + b_i\theta)$ **が平方数であることは保証**
- $(-b_i)^{\deg f} f(-a_i/b_i)$ **の素因数分解は既知**

難しい数論を使わずに求める方法はない???

数体篩法: filtering

$x^2 \equiv y^2 \bmod N$ となる (x, y) 対がいくつか保存される範囲で、行列の基本変形により疎なまま行列を縮小する

例(GNFS176):

456M×329M (w: 9G?) $\longrightarrow$ 8.5M×8.5M (w: 1.7G)

- もっと効率のよい方法はないの?
- どれくらい小さくなるの?
- 篩と線形代数との trade-off はどれくらい?

AES

- 平文・暗号文128ビット、鍵長128, 192, 256ビットの共通鍵ブロック暗号
- 2001年FIPS制定
- 3DESから徐々に移行し、共通鍵暗号の事実上の標準の地位を確立しつつあり

BES

AES の解読は次の問題が解ければ十分:
$C = f_{k_{10}}(f_{k_9}(\cdots f_{k_1}(P + k_0)\cdots))$ **を満たす** (P, C) **を多数入手し、** $k_0, k_1, \ldots, k_{10}$ **を求める。**

$f_k : ((\mathbb{F}_{2^8})^{128})^2 \longrightarrow \mathbb{F}_{2^8}^{128}$

$f_k(x) = L(S(x)) + k$

- $S(x) = (x_0^{-1}, x_1^{-1}, \ldots, x_{127}^{-1})$ **(**$0^{-1} = 0$**)**
- $L(x)$ **は** $\mathbb{F}_{2^8}$ **のアフィン変換**

むすび

- 暗号業界には数学者には自明な問題が転がっているかも
- 純粋さはないが、数学の専門知識が生かせる業界
- 優秀な学生を送り込んで下さい　:-)
- 意見歓迎

Email: maro at isl.ntt.co.jp

