インターネットと暗号の数学

河東　泰之

東京大学大学院数理科学研究科

2011 年 11 月 16 日

前回の質問について

1. 文字コードはUTFコードなど10進数で表されていると思うが，パソコンの中で2進数に直されているということだろうか．2進数を16進数にすると送れる情報量が増えると思うが，そうはうまくいかないのか．→ コンピュータの中やネットワーク上ではすべて2進数．人間が見やすい表示の問題．
2. 行列式が0になった場合の対策を知りたい．→ もっとデータが来るのを待つ．失敗したというメッセージを送る．
3. 中継点とは、実際のネットワークでは何を指すのか．→ たとえば自宅パソコンから Youtube にアクセスしたとき，データは多くのインターネット上のポイントを経由している．

4. 要は，通信量を減らして回線の負担を減らすためにどうしているか，ということでしょうか．→ そういう工夫の一つ．
5. 送るデータが線形独立でないとき，元のベクトルが復元できないように思うがどうだろうか．→ $ad - bc = 0$ の場合のことか．そうならばその通り．
6. 中継点や転送ルートはどうやって見つけるのか。→ 今ここで考えているのとは違う種類の問題．
7. $(a, b, a\vec{x} + b\vec{y})$ と $(c, d, c\vec{x} + d\vec{y})$ として送らずに、$(1, \vec{x}), (2, \vec{y})$ として送れば，どちらがどちらのベクトルか区別できるし，長さも短いと思うが、これではだめなのか．→ 同じものがダブって着いてしまう危険が高い．

インターネット通信と暗号

インターネット上で，クレジットカード番号などの個人情報を送りたい．インターネットはネットワーク上の様々な経路をデータが流れていくため，盗聴を防ぐことは困難である．そのため暗号化の技術が重要である．

すべてのデジタルデータは数字として送られるので，どうやって数字を他人に傍受されずに送るかが問題である．

例として 8 桁の 2 進数をとる．受信者と送信者で秘密のパスワードとして，たとえば 01001110 を共有しているとしよう．送りたいデータを 10100011 とする．

これらを用いて「暗号化」する．

```
   01001110   元データ
+  10100011   パスワード
----------
   11101101   通信データ ← 各桁ごとに mod 2 で足し算する
```

このとき次のように元データに戻せる．

```
   11101101   通信データ
+  10100011   パスワード
----------
   01001110   元データ ← 同じ足し算で元に戻る
```

通信データを盗聴しても，パスワードがわからなければ解読不可能である．すべての 8 桁の 2 進数が同様にありうる．何桁あってもパスワードが共有されていれば同様に通信できる．

すなわち，あらかじめ巨大な桁数の数字を共有していれば，それをパスワードとして，盗聴されても元のデータを推測できない通信が可能である．わかるのは元のデータの桁数だけであり，しかも無駄なデータをあらかじめつけておくことは可能だから，たとえば「2,000 字以下の文字を送った」ということしかわからない．

実際に高度の機密通信ではこれにあたることをするが，多量のデータを安全，秘密に前もって運ぶことは簡単でない．(パスワード自体を盗まれる，偽造されるという危険もある．)

インターネットサイトで買い物する時など，あらかじめパスワードを受け渡ししておくというのは明らかに不便であるので何かほかの方式が望ましい．

そこで，あらかじめ何の打ち合わせもなく，初めて通信する相手と，どうやって暗号通信をするか，という問題を考える．設定は次のとおりである．

どういう方式の暗号を使うか，どのパスワードを使うかも含めて一から相談して決める．このやり取りのすべてを丸ごと盗聴されていても，通信の秘密が保たれるようにしたい．

そのようなことは原理的に不可能であるように思われるし，実際に不可能である．しかし実際は上記の目標は「ほぼ」達成できる．すなわち，すべてのやり取りを盗聴されていたとしても，暗号の解読が現実的な時間や手間では不可能であるようにできる．

「原理的に可能」であることと「現実的な時間や手間で可能」なこととの間にはきわめて大きな差がある．この違いを生かすことによって，上述の，一見不可能な要請が実現できるのである．

上のパスワードに当たるものを一般に「鍵」という．普通はこれは秘密にしておくものだが，鍵 (の一部) を公開しても，暗号通信が可能である．これを「公開鍵暗号」という．

実際にブラウザで `https` で始まるアドレスの場合は，このような手法が使われている．(この方式はかなり手間がかかるので，実際はこの手法を用いて秘密のパスワードを暗号として送り，そのあとはそれを用いて通常の暗号化を行う．)

その説明のため数学的な準備を行う．

素数 p を考える．先週と同様 「p で割った余り」を考えるが 0 以外の $1, 2, \ldots, p-1$ をとり，掛け算も普通に掛けて p で割った余りを考えることにする．

たとえば $p = 11$ としてみる．11 で割った世界で，2 から出発し，次々2 倍していくと，

$$2, 4, 8, 5, 10, 9, 7, 3, 6, 1$$

となり，これで 0 以外のすべての数，10 通りが出たことになる．

$p = 7$ のときは，2 から始めると，$2, 4, 1$ となり，6 通りの余りのうち 3 つしか出てこないが，3 から始めれば，次々3 倍して，$3, 2, 6, 4, 5, 1$ となり，6 通りすべてが得られる．

実は一般に，どんな素数 p に対しても，p で割った余りの世界で考えれば，うまく x を選ぶと $x, x^2, x^3, \ldots, x^{p-1} = 1$ がちょうど $1, 2, 3, \ldots, p-1$ 全体になるようにできることがわかっている．(最後に 1 になる．この x の選び方は一通りではない．)

このとき，どの元も x^k の形なのでそれを $p-1$ 乗すると $x^{k(p-1)} = (x^{p-1})^k = 1$ となることがわかる．別の書き方をすれば，p が素数で a が p の倍数でないとき，$a^{p-1} \equiv 1 \pmod{p}$ ということである．これは Fermat の小定理と呼ばれる．

$p = 6, a = 5$ としてみると $5^5 \equiv 5 \pmod{6}$ なので上の等式を満たしていない．よって $p = 6$ は素数ではないという (あたりまえの) ことが確認できる．

さらに素数とは限らない自然数 n をとろう．$1, 2, 3, \ldots, n$ のうち，n と互いに素なものの数を $\varphi(n)$ と書く．たとえば，$n = 20$ であれば，互いに素なものは $1, 3, 7, 9, 11, 13, 17, 19$ なので $\varphi(20) = 8$ である．また n が素数であれば，$1, 2, \ldots, n-1$ までが n と互いに素なので $\varphi(n) = n - 1$ である．

p が素数の時，a が p の倍数でなければ，$a^{p-1} \equiv 1 \pmod{p}$ となるのであった．この一般化で，素数とは限らない n について，a が n と互いに素であれば，$a^{\varphi(n)} \equiv 1 \pmod{n}$ である．

たとえば，上の例で $n = 20$ のとき，$a = 3$ を取れば，$3^8 \equiv 1 \pmod{20}$ である．

Euclid 互除法

自然数 n, m が与えられたとして，$n < m$ とする．n, m の**最大公約数**を求めたい．

1. $n = 0$ ならば m が答えである．
2. m を n で割った余りを m' とする．m, n をそれぞれ n, m' で置き換えて (1) に戻る．

これによって，m, n の最大公約数 d が求まり，さらに次ページに示すように，この方法で $mx + ny = d$ となる整数 x, y を求めることもできる．

特に m, n が互いに素な時は $d = 1$ であり，上の方法により，$mx + ny = 1$ となる整数 x, y を求めることができる．

たとえば，$n = 39$, $m = 171$ とする．

$$
\begin{aligned}
171 \div 39 &= 4 \text{ 余り } 15, \\
39 \div 15 &= 2 \text{ 余り } 9, \\
15 \div 9 &= 1 \text{ 余り } 6, \\
9 \div 6 &= 1 \text{ 余り } 3, \\
6 \div 3 &= 2 \text{ 余り } 0
\end{aligned}
$$

なので，最後の式の割る数 3 が最大公約数である．

余りは当然いつも割る数より小さいので，m, n が非常に大きくてもこの計算は高速に行える．

前ページの式を，最後のものを除いて，掛け算と引き算を使って書くと次のようになる．

$$\begin{aligned} 171 - 39 \times 4 &= 15, \\ 39 - 15 \times 2 &= 9, \\ 15 - 9 \times 1 &= 6, \\ 9 - 6 \times 1 &= 3, \end{aligned}$$

最後の式の 6 のところに一つ上の式を代入し，その 9 のところに一つ上の式を代入し，さらに 15 のところにもう一つ上の式を代入し，とすると次の式を得る．

$$39 \times 22 - 171 \times 5 = 3.$$

RSA 暗号

Rivest, Shamir, Adleman の 3 人が 1977 年に開発した.

1. (数百桁の) 大きな素数 p, q を取る.
2. $n = pq$, $L = (p-1)(q-1)$ とおく.
3. L より小さく L と互いに素な自然数 e を取る.
4. $de \equiv 1 \pmod{L}$ となる d を求める.

(3) の互いに素であることの判定と，(4) には Euclid の互除法が使える.

大きな素数を見つけることは簡単ではないが，その問題は後回しにする.

(e, n) を公開し，d を秘密にしておく.

暗号通信の方法

$n = pq$, $L = (p-1)(q-1)$, $de \equiv 1 \pmod{L}$ であった．

n 未満の自然数 M を取る．これが送るべき秘密のデータである．

$M^e \pmod{n}$ を送る．

すると，$n = pq$ については，$1, 2, \ldots, n$ のうち，n と互いに素でないものは，p の倍数が q 個，q の倍数が p 個で，重なり 1 個を引いて，$p + q - 1$ 個である．よって，

$$\varphi(n) = pq - p - q + 1 = (p-1)(q-1) = L$$

である．よって，a が $n = pq$ と互いに素ならば，

$a^L \equiv 1 \pmod{n}$ である．

$n = pq$, $L = (p-1)(q-1)$, $de \equiv 1 \pmod{L}$ であった．

復元には，受け取ったデータを mod n で，d 乗する．この方法で元の秘密データ M が復元できることは次のようにわかる．

もし M が $n = pq$ と互いに素であれば，$M^L \equiv 1 \pmod{n}$ であるので，$de = Lt + 1$ (t は自然数) と書けることより，

$$M^{de} \equiv (M^L)^t M \equiv M \pmod{n}$$

となるからである．

これによって，受け取ったデータ M^e を，秘密にしておいた d を使って，d 乗することにより，秘密の送信データを復活できた．

$n = pq$, $L = (p-1)(q-1)$, $de \equiv 1 \pmod{L}$ であった．

M が，$n = pq$ と互いに素でなければ，M は p, q のいずれかの倍数である．どちらでも同じことなので，p の倍数としよう．このとき M は q とは互いに素なので，$M^{q-1} \equiv 1 \pmod{q}$ である．よって，$de = Lt + 1$ として，

$$M^{de} \equiv (M^{q-1})^{(p-1)t} M \equiv M \pmod{q}$$

である．このとき，$M^{de} - M$ は，p, q の両方で割れるので $M^{de} \equiv M \pmod{n}$ となってデータが復活する．

この場合もやはり，秘密にしておいた d を使って，d 乗することにより，秘密の送信データを復活できた．

なお，これらの計算で d 乗，e 乗しているが，これらの計算は実際に d 回，e 回掛けなくてももっと早く計算できることに注意する．

たとえば，$e = 1000$ としよう．

$$e = 512 + 256 + 128 + 64 + 32 + 8$$

なので，これを 2 進数表示すると，$e = 1111101000$ である．M に対し，$M^2, M^4, M^8, M^{16}, \ldots$ はすぐ計算できるので，

$$M^{1000} = M^{512} M^{256} M^{128} M^{64} M^{32} M^8$$

とすればよい．

最初から $e = 65537 = 2^{16} + 1$ とすることもよく行われている．

例を計算してみる．実際には大きな素数 p, q を取るが，それではここで計算できないので，$p = 11, q = 13$ としてみる．

$n = 11 \times 13 = 143$, $L = 10 \times 12 = 120$ である．

$e = 7$ としてみる．$7d \equiv 1 \pmod{120}$ を解くと $d = 103$ である．

$M = 9$ としよう．

$$M^7 \equiv 4782969 \equiv 48 \pmod{143}$$

である．これに対し，

$$48^{103} \equiv 9 \pmod{143}$$

と M が求まる．

盗聴者は，n と M^e を知ることができる．もし $n = pq$ という素因数分解ができれば，$L = (p-1)(q-1)$ として $de \equiv 1 \pmod{L}$ となる d は簡単に求めることができる．

$n = pq$ と分解することは原理的には何の困難もない．しかし，n が数百桁 (あるいはそれ以上) のときに，現実的な時間内に p, q を求めることはできないと考えられている．(n が特別な形をしていてたまたまわかると言うことはありうる．)

また，可能性としては $n = pq$ という素因数分解を経由しないで M^e から M を復元することもできるかもしれないが，そのような方法は (多くの人の努力にもかかわらず) 知られていない．

以上で，この暗号は，原理的には「簡単」に解読できるにもかかわらず，実際には普通の時間内には (たとえば我々の生きている間には) できないと考えられている．

この方法を実際に使うには，巨大な素数 p, q を見つける必要があるが，これはそれほど簡単ではない．

与えられた数 p が素数であるかどうかを判定することは古くから研究されており，さまざまな必要十分条件が知られている．2002 年にも新しい方法が，Agrawal, Kayal, Saxena の 3 人によって発表され話題になった．

しかしどれもかなりの時間と手間がかかる．

別の素数判定法として**確率的素数判定法**というものがある．

Fermat の小定理のように，素数であれば満たしているはずの条件がたくさんある．ある数 p についてこれらの条件をたくさん試してみて，どれか一つでも失敗すれば，p は素数ではないことが証明できる．

これらの条件を「たくさん」クリアすすれば，素数であることが論理的には証明できなくても，「**素数である確率が高い**」と考える．

p が素数であるかどうかは確定していることなので，「確率」という考え方は変だが，実用上このような方式は有効である．ほかのさまざまな理由により通信が失敗する可能性は 0 ではないので，小さい失敗確率は「**許容**」できるからである．

次に本当に相手が自分の思っている正しい相手かどうか，データが改ざんされていないかを判定する「デジタル署名」について考える．通常の署名にあたる手続きをデジタルデータを用いて行うのでこのような名前がついている．

銀行サイトを装ったにせサイトに誘導して，パスワードを打ち込ませると言った事件は現実に起きている．

この対策として，公開鍵暗号が有効である．公開鍵暗号を使って，自分が本物であり，データも改ざんされていないことを相手に対して主張する方法を考える．ここでは，相手の公開鍵は正しく入手できるものとする．(これについても実際はさまざまな偽装がありうるので，別の対策が必要であるが．)

ハッシュ関数

その準備としてハッシュ関数というものを考える．これはデータ (数字列) x に対してある関数 f をほどこして，(もっと短い) 数字列 $f(x)$ を返すようなものである．たとえば，2 進数 1024 桁ごとに区切って，それらを全部足す (桁あふれは捨てる) といったものだが，これでは単純すぎて問題がある．次のような性質を期待したい．具体的な形はさまざま関数が研究されている．

1. $f(x)$ は簡単に計算できる．
2. y が与えられたとき $f(x) = y$ となる x を一つでも見つけるのは難しい．
3. 異なる x_1, x_2 に対し $f(x_1) = f(x_2)$ となることが少ない．

RSA 暗号によるデジタル署名

RSA 暗号では，M^e を送り，秘密の d を使って，
$M^{ed} \equiv M \pmod{n}$ と計算するのであった．

逆に，「私が○○です」といった意味の M というデータを作り，秘密の d を使って M^d を先に作って相手に送る．これを公開の e を作れば M^{de} から M が作れる．これで送信者が秘密の d を持っていることが確認できる．

さらにハッシュ関数を使って，メッセージ M に，そのハッシュ関数値 $m = f(M)$ について m^e をつけて送れば，M を解読して求めた $f(M)$ と m^{de} を比べることにより，改ざんも防げる．

Diffie-Hellman 鍵共有方式

別の公開鍵暗号方式を取りあげる．素数 p を取る．p で割った余りを考えたとき，集合として

$\{1, 2, \ldots, p-1\} = \{x, x^2, \ldots, x^{p-1}\}$ となる x がいつでもあると前に述べた．このような x を一つとる．

ランダムに $p-2$ 以下の自然数 k を取り，これを自分の秘密鍵とする．$y = x^k$ を作りこれを公開する．

通信したい相手は，ランダムに $p-2$ 以下の自然数 r を取り，x^r を相手に送る．また，公開されている y を使い $y^r = x^{kr}$ を計算する．受け取った側は，x^r と秘密の k を使って，x^{rk} が計算できるので双方で同じ数が得られた．これをパスワードとする．

今度は盗聴者は，素数 p と，x, x^k，x^r を知ることができる．ここから原理的には「簡単に」 k が求まるので，それを使うとパスワード x^{kr} が求まってしまう．

これは $\bmod p$ の世界で，x を知っているときに y が与えられて $y = x^k$ となる k を求める問題である．これは通常の対数と同じ設定だが，x, y が実数ではなく，離散的な値 $1, 2, \ldots, p-1$ を動くので**離散対数問題**という．

これについても，現実的な時間内に解くことは困難と信じられている．(素数 p が与えられたときに，上の性質を持つ x を早く見つけることはできる．)

例を取り上げよう．実際には大きな素数を取らなくてはいけないが簡単のためごく小さい例を取る．

$p = 11$ とする．$x = 2$ とすれば，最初の方で見たように $\{x, x^2, x^3, \ldots, x^{10}\}$ で $\{1, 2, \ldots, 10\}$ が作れるのであった．$k = 4$ を秘密に選ぶ．$2^4 \equiv 5 \pmod{11}$ なのでこれを公開する．

さて通信者は $r = 6$ を選び，$2^6 \equiv 9 \pmod{11}$ なのでこれを送る．受信者は $9^4 \equiv 5 \pmod{11}$ とパスワードを求められる．

一方送信者は，$5^6 \equiv 5 \pmod{11}$ と共通のパスワードを求められる．あとはこれを使って交信する．(この例では 5 が小さすぎて意味がないが，実際の例ではずっと大きな数字である．)

参考文献 (学術的文献ではなく読み物である．)

サイモン・シン 「暗号解読〈上/下〉」(新潮文庫)

サイモン・シンは「フェルマーの最終定理」,「宇宙創成〈上/下〉」(いずれも新潮文庫) の著者でもある．前者は数学についての本である．

古代から現在の「量子暗号」までさまざまな暗号の歴史が語られる．公開鍵暗号についても説明がある．

今回の話題の暗号とは直接関係ないが，第 2 次世界大戦のドイツの暗号エニグマが，完全に数学的な原理によって破られる描写が詳しい．