ショアの素因数分解アルゴリズム

2016 年 3 月 2 日

1　この章の目的

この章ではまず，素数を利用した暗号である RSA 暗号を解説する．その次に，1994 年にショア（Peter W. Shor）が発表した，素因数分解アルゴリズムを解説する．

2　RSA 暗号

2.1　鍵生成・暗号化・復号化

2 つの大きな素数 p, q を生成し，それらの積 $n = pq$ を求める．また，$(p-1)(q-1)$ 未満の適当な正の整数 e を選び，その逆数 d を求める．ただし，e は $(p-1)(q-1)$ と互いに素でなければならない．e, d の関係は以下の式で表される．

$$de \equiv 1 \pmod{(p-1)(q-1)} \tag{1}$$

mod はモジュロ（法）を表し，上記の式では de を $(p-1)(q-1)$ で割った余りと，1 を $(p-1)(q-1)$ で割った余りが等しいということを示している．d を秘密鍵とし，n, e を公開鍵とする．当然，p, q がわかれば d も計算で求まるため，これらは安全に破棄する．

0 以上 n 未満の整数の集合を $\mathbb{Z}_n$ とする．平文 $a \in \mathbb{Z}_n$ を暗号化するには以下のようにする．

$$b = a^e \bmod n \tag{2}$$

b が暗号文であり，a^e を n で割った余りにあたる．

復号化は以下のようにする．

$$a = b^d \bmod n \tag{3}$$

試しにこの両辺を e 乗してみれば，式が正しいことがわかる．

各パラメータを以下のように決めて実際に鍵の生成，暗号化，復号化を行ってみる．

$$p = 11, \qquad q = 13 \tag{4}$$
$$n = pq = 11 \times 13 = 143 \tag{5}$$
$$(p-1)(q-1) = 10 \times 12 = 120 \tag{6}$$
$$e = 77 \tag{7}$$
$$a = 97 \tag{8}$$

d を求めるためには，例えば図 1 のような掛け算表を作成する．掛け算の答えが 120 以上となるような場合は，120 を 0，121 を 1，という具合に置き換えて書いていく．この表の 77 の行は図 2 のようになっている．このとき，積が 1 となっている列の数を見ると 53 となっている．これが $e = 77$ の逆数である．

$$d = 53 \tag{9}$$

×	0	1	2	3	4	5	6	7	8	9	10
0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8	9	10
2	0	2	4	6	8	10	12	14	16	18	20
3	0	3	6	9	12	15	18	21	24	27	30
4	0	4	8	12	16	20	24	28	32	36	40
5	0	5	10	15	20	25	30	35	40	45	50
6	0	6	12	18	24	30	36	42	48	54	60
7	0	7	14	21	28	35	42	49	56	63	70
8	0	8	16	24	32	40	48	56	64	72	80
9	0	9	18	27	36	45	54	63	72	81	90
10	0	10	20	30	40	50	60	70	80	90	100

図 1　掛け算表

×	0	1	2	3	4	5	49	50	51	52	53	54	55	56	57
77	0	77	34	111	68	25	53	10	87	44	1	78	35	112	69

図 2　掛け算表（77 の行）

と求まったところで，平文 $a = 97$ を暗号化する．[*1]暗号文 b は

$$b = a^e \bmod n = 97^{77} \bmod 143 = 15 \tag{10}$$

これを復号化すると，

$$b^d \bmod n = 15^{53} \bmod 143 = 97 = a \tag{11}$$

となり，正常に復号化できていることがわかる．

2.2　RSA 暗号の安全性

前述のとおり，公開鍵の n の因数 p, q を求めることさえできれば，上記の鍵生成の方法で秘密鍵 d が求まり，復号化することができてしまう．つまり，この暗号は大きな整数の素因数分解が困難であることが安全性の根拠となっている．現在のところ，768 ビット（10 進数で 232 桁）の整数が素因数分解できている．しかし，それ以上の桁数も遠くない未来において素因数分解できるようになると見られている．

[*1] 実は，97 というのはアスキーコードにおいて文字 'a' を表す．

3 ショアの素因数分解アルゴリズム

ショアの素因数分解アルゴリズムは量子コンピュータ上で素因数分解を実行するアルゴリズムである．最新鋭のスーパーコンピュータで 1 万桁の整数を素因数分解すると 1000 億年以上[*2]かかるところ，量子コンピュータを使えば数時間で終了する[*3]．

ここでは，RSA 暗号の場合を想定して，n ビット長の自然数 N が 2 つの素数 p, q の積で成り立つ（すなわち $N = pq$）ときに，p, q を求めることを考える．これをしらみつぶしに探す場合，最高で $2^n - 1$ 回割り算する必要がある．桁数の増加に対して指数関数的に計算時間が増加する．量子コンピュータを用いれば，これが多項式時間で解けるようになる．

3.1 素因数分解の手順

以下に素因数分解の手順を示す．ただし，整数 a, b の最大公約数を $\gcd(a, b)$ と書き，0 から $N - 1$ の整数の集合を $\mathbb{Z}_N \equiv \{0, \cdots, N-1\}$ と定義する．

1. x を $\mathbb{Z}_N$ の中からランダムに 1 つ選ぶ．
2. $\gcd(x, N)$ を求め，1 でなければそれが 1 つの因数なので，素因数分解完了である．以下では $\gcd(x, N) \neq 1$ とする．
3. $x^r \equiv 1 \pmod N$ となるような整数 $r > 0$ を探す．
4. r が奇数ならば最初に戻る．
5. r が偶数ならば $x^r - 1 = (x^{\frac{r}{2}} + 1)(x^{\frac{r}{2}} - 1) \equiv 0 \pmod N$ が成立している．
6. $x^{\frac{r}{2}} + 1 \not\equiv 0 \pmod N$ ならば，$\gcd(x^{\frac{r}{2}} + 1, N), \gcd(x^{\frac{r}{2}} - 1, N)$ のいずれかが自明でない因数となっている．

まず最大公約数についてだが，これは**ユークリッドの互除法**を用いることにより，古典コンピュータでも高速に求めることができる．例として $\gcd(1152, 200)$ を解いてみる．

$$1152 \bmod 200 = 152 \tag{12}$$
$$200 \bmod 152 = 48 \tag{13}$$
$$152 \bmod 48 = 8 \tag{14}$$
$$48 \bmod 8 = 0 \tag{15}$$
$$\therefore \gcd(1152, 200) = 8 \tag{16}$$

要するに，2 つの整数の割り算の余りを求め，その余りと 2 つ整数のうち小さい方の割り算の余りを求め，ということを繰り返して 0 となった演算の前の答えが最大公約数である．2 つの整数のうち，小さい方の 2 進数での桁数が n ビットのとき，計算量は $O(n)$ となる．

したがって，量子コンピュータを用いて高速化する部分は手順 3 の「$x^r \equiv 1 \pmod N$ となるような整数 r を探す」という部分である．ここで，関数 $f(a)$ を

$$f(a) = x^a \bmod N \tag{17}$$

と定義する．このとき，この関数はある周期 s があり，$f(a) = f(a + s) = f(a + 2s) = \cdots$ を満たしている．例として $N = 5, x = 2$ としたとき，

$$f(0) = f(4) = f(8) = \cdots = 1 \tag{18}$$
$$f(1) = f(5) = f(9) = \cdots = 2 \tag{19}$$
$$f(2) = f(6) = f(10) = \cdots = 4 \tag{20}$$

[*2] 太陽の寿命が残り約 50 億年であることからも，明らかに計算不可能であることがわかる．
[*3] [6] p154

という具合に周期性を持つ．素因数分解アルゴリズムでは，

$$f(0) = f(s) = f(2s) = \cdots = 1 \tag{21}$$

の s が求めたい r にあたる．つまり，ショアのアルゴリズムの核となる部分は，周期発見のためのアルゴリズムなのである．

3.2 周期発見問題を解く量子アルゴリズム

関数 $f(a) = x^a \bmod N$ の周期 s を求める．上の例と同じく $N = 5, x = 2$ として具体的に求めてみる．求めるべき答えは $s = 4$ である．

このアルゴリズムでは 2 つのレジスタを用いる．レジスタのビット数は $t = 4$ ビットとする．以下ではレジスタの状態を，例えば 12 ならば $|12\rangle$ と表すが，これは $|1100\rangle$ という意味である．

1. 2 つのレジスタを初期化する．

$$|0\rangle\,|0\rangle \tag{22}$$

2. レジスタ 1（左側）に同じ振幅の重ね合わせ状態を生成する．

$$\frac{1}{\sqrt{2^t}} \sum_{a=0}^{2^t-1} |a\rangle\,|0\rangle = \frac{1}{\sqrt{16}}(|0\rangle\,|0\rangle + |1\rangle\,|0\rangle + |2\rangle\,|0\rangle + \cdots + |15\rangle\,|0\rangle) \tag{23}$$

3. レジスタ 1 を入力とした f の出力をレジスタ 2 に書き込む．

$$\frac{1}{\sqrt{2^t}} \sum_{a=0}^{2^t-1} |a\rangle\,|f(a)\rangle = \frac{1}{4}(|0\rangle\,|1\rangle + |1\rangle\,|2\rangle + |2\rangle\,|4\rangle + \cdots + |15\rangle\,|3\rangle) \tag{24}$$

4. レジスタ 2 について測定を行う．
   ここでは，4 が観測されたとする．

$$\sqrt{\frac{s}{2^t}} \sum_{a:f(a)=4} |a\rangle\,|4\rangle = \sqrt{\frac{s}{2^t}}(|2\rangle + |6\rangle + |10\rangle + |14\rangle)\,|4\rangle \tag{25}$$

5. レジスタ 1 の部分について量子フーリエ変換を行う．
   量子フーリエ変換 F とは，t ビットの量子ビット $|a\rangle$ に対して以下のような関係を満たす変換である．[*4]

$$F\,|a\rangle = \frac{1}{\sqrt{2^t}} \sum_{k=0}^{2^t-1} e^{i\frac{2\pi ak}{2^t}}\,|k\rangle \tag{26}$$

   量子計算なので，F も当然ユニタリ変換となっている．レジスタ 1 の部分について量子フーリエ変換を行う．

$$F\sqrt{\frac{s}{2^t}}(|2\rangle + |6\rangle + |10\rangle + |14\rangle) = \sqrt{\frac{s}{2^{2t}}} \sum_{k=0}^{15} \left(e^{i\frac{1}{4}\pi k} + e^{i\frac{3}{4}\pi k} + e^{i\frac{5}{4}\pi k} + e^{i\frac{7}{4}\pi k}\right)|k\rangle \tag{27}$$

$$= \sqrt{\frac{s}{2^{2t}}} \sum_{k=0}^{15} e^{i\frac{1}{4}\pi k} \sum_{\ell=0}^{3} e^{i\frac{2}{4}\pi \ell k}\,|k\rangle \tag{28}$$

   ここで，

$$S(k) = \sum_{\ell=0}^{3} e^{i\frac{2}{4}\pi \ell k} \tag{29}$$

[*4] $|a\rangle$ を時間領域信号 $f(a)$，$|k\rangle$ を周波数領域信号 $F(k)$ と見れば，離散フーリエ変換と同じ形である．（ディジタル信号処理 - 「離散フーリエ変換対」）

とおく．これを計算すると，$S(0) = S(4) = S(8) = S(12) = 4$ となり，それ以外の k では 0 となる．

$$\therefore F\sqrt{\frac{s}{2^t}}\left(|2\rangle + |6\rangle + |10\rangle + |14\rangle\right) = 4\sqrt{\frac{s}{2^{2t}}}\left(|0\rangle - |4\rangle + |8\rangle - |12\rangle\right) \tag{30}$$

6. 上記 (30) 式を測定する．0,4,8,12 のいずれかが等確率で得られる．
7. 手順 1 から 6 を適当な回数繰り返し，測定で得られた 0 を除く整数の最大公約数を求めると，周期 s が求められる．

手順 7 を見るとわかるように，このアルゴリズムは確率的に周期を求めている．しかし，ある程度の回数実行すれば高い確率で周期が求められる上，求めた周期の正誤は簡単に確かめられるので問題はない．また，レジスタのビット数を増やせばより高い確率で正しい周期が求められるようになる．

参考文献

[1] Michael A. Nielsen and Isaac L. Chuang. **量子コンピュータと量子通信** *I* -**量子力学とコンピュータ科学**-. Trans. by 木村達也. オーム社, 2004.

[2] Michael A. Nielsen and Isaac L. Chuang. **量子コンピュータと量子通信** *II* -**量子コンピュータとアルゴリズム**-. Trans. by 木村達也. オーム社, 2004.

[3] Michael A. Nielsen and Isaac L. Chuang. **量子コンピュータと量子通信** *III* -**量子通信・情報処理と誤り訂正**-. Trans. by 木村達也. オーム社, 2004.

[4] 石坂 智 et al. **量子情報科学入門**. 共立出版, 2012.

[5] *The RSA Challenge Numbers*. `http://japan.emc.com/emc-plus/rsa-labs/historical/the-rsa-challenge-numbers.htm`. (2016-02-28 参照).

[6] 竹内 繁樹. **量子コンピュータ　超並列計算のからくり**. 講談社, 2005.