大きい数が素数かどうか判定するアプローチ

BV24006 田中　久遠

2025 年 5 月 18 日

目次

1　研究動機、なにをするのか

暗号について調べていたら、大きな素数をどのように見つけるかが課題であることが分かった。今回はフェルマーの小定理から、ある大きい数が素数であるかを判定する方法について調べた。

2　フェルマーの小定理

まず今回の主役となるフェルマーの小定理を紹介する。
p を素数、a を p と互いに素な整数とする。このとき

$$a^{p-1} \equiv 1 \pmod{p}$$

が成立する。

3　フェルマーの小定理で合成数を見つけよう

ここで $a^{n-1} \equiv 1 \pmod{n}$ が成り立つと仮定する。フェルマーの小定理は、法と指数に使う数 n が素数の時は必ず成立するので、逆にフェルマーの小定理が成立しないならば、n は合成数であるといえる。このように判定する方法をフェルマーテストという。例えば n=2544623 のような大きな数が素数かどうか確かめるの

は普通にやると非常に困難である。平方根までの可能な約数をすべて試すことは現実的ではない。しかし大きな数の高いべき乗を、大きな数を法として計算するのはコンピュータ上ではそこまで難しくない。

```
def fermat(p,a):
    return a**(p-1)%p
```

```
fermat(5,3)
```

```
1
```

```
fermat(2544623,5)
```

```
1
```

```
fermat(542,59)
```

```
59
```

```
fermat(561,100)
```

```
1
```

```
fermat(561,152)
```

```
1
```

```
fermat(561,42)
```

```
375
```

図 1　実験

色々と a と n の値を実験してみた。偶数である n=542 は当然だが 1 と合同にはならなかった。他には n=561 は一見 1 と合同で素数かもしれないと思うが、a の値を変えてみると 1 と合同にはならず合成数であることが分かった。その数が本当に素数であるか確かめるには、a の値をたくさん変えて実験する必要がある。それでも素数であると言い切ることはできないが。

4　カーマイケル数

n=561 のときについて考える。a=100 のとき 1 と合同であると表示されているが、$561 = 3 \times 11 \times 17$ であり、561 は合成数である。フェルマーの小定理を成立させることが出来る合成数が存在する。これを擬素数という。またこのときの a を嘘つきという。

また 561 は擬素数の中でも特別な数で、561 と互いに素である任意の自然数 a に対して $a^{561} \equiv a$ が成り立つ。このような数ををカーマイケル数という。a と p が互いに素であるとき $a^{561} \equiv a \pmod{p}$ は $a^{p-1} \equiv 1 \pmod{p}$ と変形できる。つまり a がカーマイケル数という合成数と素因数を共有していないならば、カーマイケル数は必ずフェルマーテストを突破する。平たく言うとやばい数である。カーマイケル数と素因数を共有する数を探すのは非常に大変であり、それが大きい数であるほど現実的ではない。さらにカーマイケル数は無限にが存在することが知られている。よってフェルマーテストには欠陥があると言える。

5　ミラーラビン法

フェルマーテストというアルゴリズムを改良したものとして、ミラーラビン法が存在する。ここではその基本的なアイデアを紹介する。n を奇数とすると、$n-1$ は偶数であるから、

$$n - 1 = 2^s t$$

となるような $s \geq 1$ と奇数 t が一つだけ定まる。例えば n=101 のとき、$n-1 = 100 = 2^2 \times 25$ となり、s=2,t=25 となる。ここでフェルマーの小定理について考えると n は素数、gcd(a,n)=1 の条件下で、

$$a^{n-1} = a^{2^s t} \equiv 1 \pmod{n}$$

が成り立つ。

また n を法とすると、$x^2 \equiv 1$ が成り立つときに $(x-1)(x+1) \equiv 1$ となるため、もし n が素数ならば n は x-1 か x+1 を割り切る必要がある。つまり $x \equiv \pm 1$ となり、n が素数ならば 1 の平方根は ± 1 と分かる。よって n が素数であれば

$$a^{2^{s-1} t} \equiv \pm 1 \pmod{n}$$

が成立して、さらにこれが 1 と合同になったとき、平方根をとると

$$a^{2^{s-2} t} \equiv \pm 1 \pmod{n}$$

となる。逆に一度でも 1 か-1 と合同になれば、それを 2 乗すればそれは 1 と合同になる。例えば $a^t \equiv 1(modn)$ か $a^t \equiv -1(modn)$ のとき、$a^{2t} \equiv 1$ となる。a^t から $a^{2^s t}$ までは 2 乗する操作を繰り返すことで到達できるが、このときに 1 度でも n を法にして-1 と合同になるか、全てで 1 と合同になっていることを確認できれば、n は素数であるといえる。

```
def fermat(p,a):
    return a**(p-1)%p
```

```
fermat(561,152)
```

```
1
```

```
fermat(280,152)
```

```
48
```

```
fermat(140,152)
```

```
68
```

図 2　カーマイケル数へのリベンジ

この判定法の嬉しいところはより精密な判定が出来ることと、その精密さを確率であらわすことが出来ることができることである。任意の奇数の合成数 n に対して、1 から n-1 の範囲から一様にランダムに選んだ a が嘘つきでない確率は、少なくとも $\frac{3}{4}$ であることが知られている。つまりテストを何回も繰り返すことで、n という合成数を誤って素数だと判定してしまう確率は指数関数的に減少する。そのためミラーラビン法は現実で広く使われている確率的素数判定アルゴリズムである。

6　今後の課題

今回は有名な素数生成法についてまとめた。基本的な内容は理解したがまだ深めきれていないところがたくさんある。例えばミラーラビンテストについて、合成理由や数を素数と判定してしまう確率が 1 回の操作で少

なくとも 1/4 以下であることの理由は理解していない。他にもこれまでの確率的に素数を判定するアプローチとは違い、確定的な素数判定アルゴリズムとして知られる AKS 素数判定法についてもよくわかっていないので調べようと思った。

参考文献

[1] 神永正博、吉川英機、Python で学ぶ暗号理論、コロナ社
[2] J.H. シルヴァーマン、初めての数論、丸善出版