2次合同法による一様乱数列の生成について

藤野　精一*

Computer Generation of Sequences of Uniform Random Numbers by a Quadratic Congruential Method

Seiichi FUJINO

Abstract

Usually a linear congruential method has been used for computer generation of sequences of uniform random numbers. In this paper we shall propose a quadratic congruential method for generation of such sequences. Some experiments are done showing usefulness of the method.

1．はじめに

コンピュータで多数の乱数を生成する必要が生ずる場合がある。例えば遺伝アルゴリズムを使用しようとするときなどである。このようなとき普通はコンピュータに備え付けられた乱数プログラムを利用するが，場合によってはそれを使用しないで自作の乱数発生プログラムを作成しそれを活用したい場合がある。そういう場合線形合同法が古くから良く用いられたがそれ以外でも実用的な方法はいろいろ考えられる[3]。この論文では2次合同法を利用する方法を示したい。得られた結果は次のとおりである。

(1)　どこから初期値をはじめても不動点に落ち込まないプログラムを作成できた。

(2)　コンピュータでプログラムを作成するので，当然何回か後にはもとのどこかにもどり繰り返しをはじめるが，反復周期の非常に長い反復系を見いだした。

2．一様乱数の発生

いまかなり長い乱数列を生成したいとする。C言語でプログラムする場合によく用いられる方法は静的変数を利用する方法で〝平方採中法〟という，m桁の乱数の2乗（2m桁になる）の中央m桁を次の乱数にとる方法でこの方法はC言語ではシフト演算子があるので非常に好都合である。参考文献(4)で作成されたプログラムは次に

表－1　平方採中法による一様乱数の発生

```
#include <stdio.h>

main()
{
int  i,k;
  for (i=0;i<10;i++)
 {
    k=rans();
    printf("%d,¥n",k);
  }
}
rans()
{
 static long n=24689;
n=((n*n)<<8)>>16;
     return n;
}
```

```
21745,
12042,
-23380,
-27437,
-8539,
22678,
-22665,
-24967,
10133,
7868,
```

＊教養部
平成8年9月13日受理

示す表－１である。

表－１のｒａｎｓ（　）が乱数生成機構の中心で反復関数$f(n)$として

$f(n)=(n^2<<8)>>16 mod 2^{32}$,

すなわち，n^2を32ビットで表現し左に２進８桁シフトし，さらに右に２進16桁シフトして，整数型に変換して返す。表－１で見られる様に負数の形でててくるものもあるが，それは整数の計算でオーバーフローを起こすと，そこで得られた値は2^{32}で割つた余り，即ち，$mod 2^{32}$を意味する。負数になつた場合にはその値に2^{32}を加えた値をとればよい。

表－１で得られた結果を利用して０，１の乱数を生成しようとおもえば例えばｒａｎｓ（　）の絶対値を20000でわれば良い。得られた結果は表－２の通りである。

ちょっと１の個数が多いようであるがこの手続きを行ごとの乱数列作成に書き直し，それを10回繰り返すと表－３のような０，１の乱数列が得られる。

これを見るとたしかにうまく出ているようである。こ

表－２　｛０，１｝の一様乱数列（１）

```
#include <stdio.h>
#include <math.h>

main()
{
int i,k;
  for (i=0;i<10;i++)
  {
     k=abs(rans())/20000;
     printf("%d,¥n",k);
   }
}
rans()
{
 static long n=24689;
 n=((n*n)<<8)>>16;
      return n;
}
```

```
1,
0,
1,
1,
0,
1,
1,
1,
0,
0,
```

表－３　｛０，１｝の一様乱数列（２）

```
1,0,1,1,0,1,1,1,0,0,
1,1,0,0,0,1,0,1,0,0,
0,0,1,0,1,0,1,0,1,0,
1,1,1,0,0,1,1,0,1,0,
0,0,1,0,0,1,1,1,1,0,
0,0,1,0,1,1,0,0,0,0,
0,1,0,0,1,0,1,1,1,0,
0,0,0,1,1,0,1,0,1,1,
1,1,0,0,0,0,0,1,0,1,
0,1,1,1,0,1,1,0,0,0,
```

れくらいの長さでこれくらいの回数であればこれでよかろう。ところが，このプログラムを利用して長さ40の乱数列を15組生成しようとする。表－4はその結果である。

表－４の値の列は意外にも途中から０ばかりの列になる。これは，一体何に原因するのであろうか。表－１で使用した反復関数を300回反復した結果を見てみよう。表－５がそれである。

この表でわかるように，$mod 2^{32}$に関する$f(n)$の不動点，すなわち，

$n=(n^2<<8)>>16 \mod 2^{32}$

を満たすnの値は256，すなわち，2^8であり，反復をくりかえすとこの不動点に収束したのである。たしかに

$(2^8)^2=2^{16}$, $2^{16}<<8=2^{24}$,

$2^{24}>>16=2^8$である。

したがって，この反復関数を使用しては長い０，１の乱数列を作成することはできない。ではどうすればよいか。それには，反復関数が極限点をもたないようにする必要がある。しかし，コンピュータで反復関数を繰り返し計算すれば有限回の後に或るサイクルを繰り返すようになるのは避けることができない。出きるかぎり長いサイクルをもつ反復関数を作れないか。長い乱数列を得ようとすれば，そこが問題である。次節で一つの解答を与えよう。

表－４　乱数発生の途中での収束現象（１）

```
1,0,1,1,0,1,1,1,0,0,1,1,0,0,0,1,0,1,0,0,0,0,1,0,1,0,1,0,1,0,1,1,1,0,0,1,1,0,1,0,
0,0,1,0,0,1,1,1,1,0,0,0,1,0,1,1,0,0,0,0,0,1,0,0,1,0,1,1,1,0,0,0,0,1,1,0,1,0,1,1,
1,1,0,0,0,0,0,1,0,1,0,1,1,1,0,1,1,0,0,0,0,0,0,0,0,1,1,0,1,0,0,0,0,0,0,0,1,0,1,0,
0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
```

表－５　乱数発生の途中での収束現象（２）

```
21745,12042,-23380,-27437,-8539,22678,-22665,-24967,10133,7868,-20326,-24548,-53
73,-18302,-2270,20128,9700,-25677,19519,-19081,-19587,-8693,-32492,-4823,25328,1
5521,23517,-2340,21389,17595,29664,29444,-21352,11418,-15028,30222,28904,-13358,
-23880,-668,1743,11867,25812,-18865,13932,-28227,32165,-21876,-31172,-5410,-1674
4,-18950,26488,-11832,22572,24135,-18377,8475,18424,15232,-11200,31248,13121,171
42,-31804,18990,32416,-24092,-26477,-14108,-8949,-14850,9448,21010,20361,-18985,
31674,-13245,29913,21855,30777,30076,-5484,-13595,1072,4489,13179,23101,-12558,2
6204,-4751,22635,-30276,-23870,-2533,25062,28698,5826,1515,8965,-13730,15482,187
94,3487,-18040,26072,-31706,-5323,-20392,-14050,-15329,379,561,1229,5900,4904,28
406,6228,20443,-5915,5596,-8747,-28813,31661,-16462,10007,-2044,16320,-8176,-102
3,4088,-256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,
256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,
256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,
256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,
256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,
256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,
256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,
256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,
256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,256,
256,256,256,256,256,256,256,256,256,256,256,
```

３．或る反復模型の特性

著者は以前に高次合同型反復模型という反復システムを考察した[(1),(2)]。この反復模型の線形で単一系の場合が乱数生成の初期に利用された線形合同法であつた。この模型の２次で単一系の場合を考えてみよう。その中に安定周期がただ一つだけあり，それが勿論最大安定周期であり，かつその周期ができるかぎり長いものを取り出そう。次に掲げるのがその一例である。

反復模型$<X, f_k>$,ただし，$X=Z_{2^k}$, f_kは

$$f_k(x)=x(x+1)+1 \ mod \ 2^k$$

で定義されるZ_{2^k}からZ_{2^k}への関数である。この関数の$k=1, 2, \cdots$についてその反復図と反復表を見てみよう。図－１から図－５までが$<Z_{2^k}, f_k>(k=1, 2, 3, 4, 5)$の反復図である。

この反復図から次のようなことがわかる。

(i)$x \in Z_{2^k}$に対して$f_k(x)$はいつでも奇数である。

(ii)サイクルを構成する数はすべて奇数である。

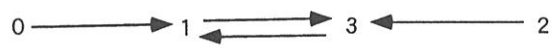
(iii)Z_{2^k}のすべての奇数，その個数は2^{k-1}，でただ一つのサイクルを構成する。

(iv)それぞれの最大安定周期は，1，2，4，8，*16*，すなわち，$2^0, 2^1, 2^2, 2^3, 2^4$である。

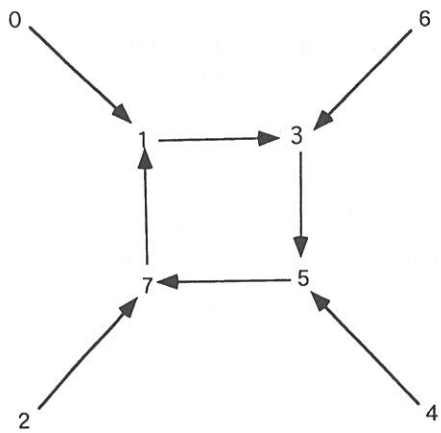
実際，反復模型

$$<Z_{2^k}, f_k>, f_k(x)=x^2+x+1 \ mod \ 2^k$$

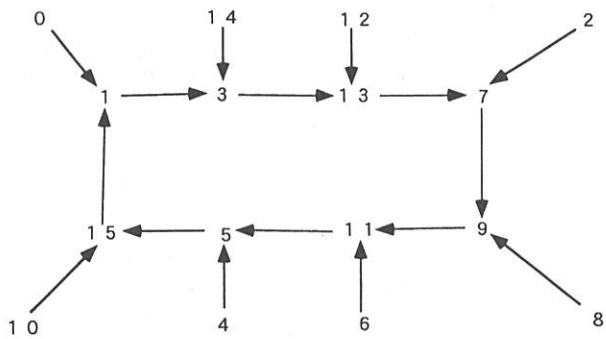
図－1　$f_1(x)=x(x+1)+1 \ mod \ 2$ の反復図



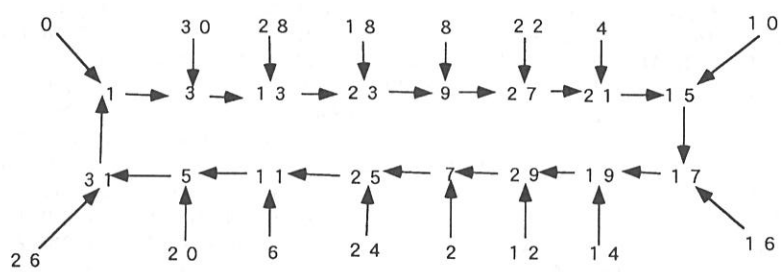
図－2　$f_2(x)=x(x+1)+1 \ mod \ 2^2$ の反復図



図－3　$f_3(x)=x(x+1)+1 \ mod \ 2^3$ の反復図



図－4　$f_4(x)=x(x+1)+1 \ mod \ 2^4$ の反復図



図－5　$f_5(x)=x(x+1)+1 \ mod \ 2^5$ の反復図

の最大安定周期は2^{k-1}である。このことは次の反復表をみればその再帰的な構成からただちに見て取れる（表－6）。

この表からf_kについて次の関係が得られる。

定理1．$Z_{2^{k-1}}$の各xについて

$$f_k(x)=f_k(2^k-1-x)$$

である。

（証明）　各$x \in Z_{2^{k-1}}$について

$$\begin{aligned} f_k(2^k-1-x)&=(2^k-1-x)(2^k-x)+1 \ mod \ 2^k \\ &=(2^k-1-x)2^k-2^kx+x(x+1) \\ &\quad +1 \ mod \ 2^k \\ &=x(x+1)+1 \ mod \ 2^k \\ &=f_k(x) \end{aligned}$$

である。

（証終）

定理2．aを$Z_{2^{k-1}}$の一数とし，$k \geqq 2$とする。

このとき

表－6　$f_k(x)$の反復表$(k=2(1)5)$

x	f_2	f_3	f_4	f_5	x	f_2	f_3	f_4	f_5
0	1	1	1	1	16				17
1	3	3	3	3	17				19
2	3	7	7	7	18				23
3	1	5	13	13	19				29
4		5	5	21	20				5
5		7	15	31	21				15
6		3	11	11	22				27
7		1	9	25	23				9
8			9	9	24				25
9			11	27	25				11
10			15	15	26				31
11			5	5	27				21
12			13	29	28				13
13			7	23	29				7
14			3	19	30				3
15			1	17	31				1

(i)もし$f_{k+1}(a)=f_k(a)$ならば

$$f_{k+1}(2^k-1-a)=f_k(a)+2^k$$

(ii)もし$f_{k+1}(a)=f_k(a)+2^k$ならば

$$f_{k+1}(2^k-1-a)=f_k(a)$$

(iii)$f_{k+1}(a)+f_{k+1}(2^k-1-a)=2f_k(a)+2^k$

(証明) (i)を示す。定理１より

$$\begin{aligned}f_{k+1}(2^k-1-a)&=f_{k+1}(2^{k+1}-1-(2^k-1-a))\\&=f_{k+1}(2^{k+1}-2^k+a)\\&=f_{k+1}(2^k+a)\\&=(2^k+a)(2^k+a+1)+1\\&\quad mod 2^{k+1}\end{aligned}$$

表－７　$f_{16}(x)$の反復表

```
#include <stdio.h>
#include <math.h>

main()
{
int  i,k;

  for (i=0;i<300;i++)
 {
    k=rans();


    printf("%d,",k);
  }
   printf("¥n");

  }
rans()
{

static  long n=1;
n=n*(n+1)+1;
 return n;
}
```

```
3,13,183,-31863,731,10805,-26321,-12335,30835,31773,-26777,17113,-8501,-27707,26
975,31393,21475,20269,7447,21801,-27205,-13227,24719,-3983,595,26941,-32313,-218
95,-26709,-15387,28351,8513,-3133,-17843,-19081,13001,21659,27253,-31761,-751,-2
6573,12893,-22489,-8679,15499,-20475,-28641,29665,24483,-16019,18903,-25495,-165
17,-31595,-31921,30129,-17901,22397,-30073,23993,19819,-10203,19839,-3455,6019,-
7027,23095,4617,22107,-25931,-8529,-9647,-6157,22685,-22297,-22183,19019,-17339,
9951,7457,-25757,-23635,26263,5033,-26309,11477,5647,-21775,-24109,-19011,-31929
,16633,-19669,-9115,-25537,29633,27459,32461,-4361,8521,2075,-17675,-22161,26513
,26547,-4387,-26201,-23399,2059,-18299,11679,30305,1827,-2579,29527,-17687,8955,
-15083,6351,-28623,-18029,-31747,27655,21561,-14101,-12123,23295,-23295,-4349,-3
0451,29623,24713,27099,-13515,-7121,-23343,6003,-2787,31335,-12327,30155,-23355,
-23457,32673,-23837,-18387,28695,-28119,23739,19797,-29809,9585,339,-15811,17607
,-26759,-29013,-19227,-30273,-31167,-23869,845,-6025,-12343,31643,-13451,-28945,
-28143,-2253,27485,19239,11033,-27765,-32507,-29921,14561,28323,-5523,23767,-262
63,18043,-14955,28239,27313,30995,28797,3463,2745,1131,-30427,10367,6017,-31101,
-4723,19767,28425,15707,-17483,-22097,12113,1779,20893,3047,-18855,24907,19781,-
7713,-24031,27235,-29523,14743,-12119,-8133,11733,-16113,24561,8403,-28995,-1477
7,-20999,10795,19813,14143,22721,-26045,18381,-24073,15945,-20709,-25611,13423,3
1889,18099,-22563,-17241,29081,-8437,2437,-22369,-17567,-27101,-24851,1623,14313
,10747,-31211,-31793,1329,-1901,7421,28423,-32455,-21,421,-18945,18945,-8701,462
1,-6473,15753,-12069,27701,12079,31185,-18829,28189,23911,23769,3275,-19003,-835
3,-31583,-3613,8493,-15593,-12503,9147,-12715,-18801,23153,83,6973,1991,-31623,-
31317,-23067,-23361,-5311,20931,19533,7031,27849,-23909,11381,-26129,10001,22067
,-23459,-4569,30745,
```

$$\begin{aligned}&=2^{2k}+a2^k+2^k(a+1)+a(a+1)\\&\quad +1\ mod\ 2^{k+1}\\&=2^{2k}+a2^{k+1}+2^k+a(a+1)+1\\&\quad mod\ 2^{k+1}\\&=2^k+f_{k+1}(a)\\&=2^k+f_k(a)\end{aligned}$$

(仮定より)

(ii)は(i)と同様に定理１と仮定より示される。

(iii)は(i)と(ii)の結果よりでる。

(証終)

定理１と定理２の結果より反復関数の周期は使用する奇数全体の個数に一致し，kが１上がるごとに倍になることがわかる。

$f_1(x)$の周期が1，すなわち2^0であるから，$f_k(x)$の周期は2^{k-1}である。したがつて，$Z_{2^{32}}$でこの反復を実行すると周期2^{31}の反復，すなわち，周期2147483648の反復となり十分実用に供する。仮に$Z_{2^{16}}$における演算でも周期$2^{15}=32768$で元にかえり，これでも実用的には十分である。

４．ふたたび一様乱数の発生について

前節で考察した反復法は確かに乱数列生成の手段として利用できる。これを使用してみよう。表－７は前節で述べた２次の反復法をそのまま適用して300個の乱数をプリントしたものである。

表－８　$f_{16}(x)$に採中法を適用した一様乱数生成

```
/*  乱数０，１の生成  */
#include <stdio.h>

main()
{
int  i,k;
int c;
   c=0;
  for (i=0;i<300;i++)
 {
    k=rans();
   if (k<0)
    {
     k+=32767;
     k+=1;
     }
      k=k/16384;
    printf("%d,",k);
   if (k==1)

     c+=1;

   }
    printf("¥n");
    printf("%d",c);
  }

rans()
{
 int m;
static  long n=1;
n=n*(n+1)+1;
m=n<<8>>16;
 return  m;
}
```

表－９　｛０，１｝の一様乱数列（３）

```
0,0,0,0,0,1,0,0,0,0,1,0,1,1,0,0,0,0,1,1,1,0,0,0,0,0,0,1,0,0,1,1,0,1,1,0,1,1,0,1,
0,1,0,1,0,0,1,0,1,0,1,0,0,0,0,1,1,0,0,1,0,0,0,1,1,0,1,1,1,1,0,1,0,0,0,0,0,1,1,0,0,
1,1,1,1,1,0,1,0,1,1,1,0,1,0,1,1,0,0,1,1,1,0,0,1,0,1,0,1,0,1,0,0,1,1,1,1,1,1,1,0,1,
0,1,1,0,0,1,1,1,0,0,1,0,1,0,0,1,0,0,0,0,0,0,1,1,1,0,1,1,1,1,1,0,1,1,1,1,1,1,1,0,1,0,1,
0,1,1,1,1,1,1,1,0,0,0,1,0,0,0,1,1,0,1,0,0,1,0,1,1,0,0,1,1,0,1,0,1,1,0,1,1,0,0,1,1,
0,1,1,1,0,0,1,0,0,0,0,0,0,0,0,1,1,1,1,1,1,0,0,0,1,1,1,0,0,1,0,1,0,1,1,0,1,0,1,1,0,
0,0,0,1,1,0,1,1,0,1,0,0,1,1,0,0,0,0,0,0,0,1,0,0,0,1,1,0,1,1,0,1,1,0,1,1,1,1,1,1,0,
0,0,0,0,0,0,1,0,1,0,0,0,1,1,0,0,0,0,1,1,
146
```

これに採中法(<<8>>16)を適用したものが表－８である。

さらにこの300個の乱数をそれぞれ16384，すなわち，2^{14}で除した乱数列が表－９である。表－１では目の子で20000を使用して乱数を除したが，上の反復関数の性質からわかるように，この値は2^{14}，すなわち16384，が反復サイクルを構成する奇数全体の個数2^{15}の半数に当たるから，この値が０，１を生成するのに最適の値である。表の最後の値146は出現した１の回数である。

```
0,0,0,131,5634,16938,2969,16079,11896,11900,18071,9538,21470,23187,10857,13690,7
507,2127,21789,26965,24469,15564,11360,16368,12802,9065,5249,30378,1943,15555,31
598,28193,7155,22970,30645,15154,29780,25194,15235,25597,13464,26930,7080,18654,
2108,2480,27280,14707,24415,5057,26441,13724,16319,644,131,23157,27066,16215,111
46,31581,4429,12504,3917,26354,27415,5604,17242,22546,19798,11418,25310,4826,944
7,9560,13224,11689,32330,25020,8486,1565,25243,32675,19302,25107,18585,12332,271
58,15530,22945,23221,17283,16192,19123,3036,30620,26227,2411,12414,24302,21281,3
2008,7354,15785,18279,7015,24302,2713,20388,4616,24248,2605,6006,17159,18421,321
15,27322,22050,16837,9496,25232,7609,29059,22636,15956,4040,18896,28762,26789,48
47,9609,18035,10336,24425,11979,2020,32164,13335,4597,7290,10959,8053,31908,2499
6,27007,7330,26808,31088,19346,17500,13133,28043,18469,20993,32706,26180,29847,1
5502,26292,13705,27014,9122,28931,30184,21455,20603,24267,32142,12690,2807,14955
,25931,15147,659,14209,25483,30776,2926,24042,15196,153,28486,197,32622,26986,47
29,368,26893,24074,11524,24201,808,25879,22406,3821,26957,20847,11325,10427,2320
9,24367,15366,21585,24843,31926,5985,2893,19425,16290,11114,9100,16185,1488,1302
4,9773,5825,16991,31520,29326,23750,30381,15658,11597,5431,22360,19098,31559,732
9,7486,20143,2203,23348,636,25414,23463,3004,27505,14815,29705,22184,15291,6294,
13726,262,22327,27433,8070,28547,20229,10232,20252,15727,14977,18687,20225,10165
,15434,6366,12818,3046,13117,5072,31340,15151,6009,7350,19566,24925,860,17932,28
085,16351,26756,28401,4129,31939,28879,29731,26830,17846,2906,13568,4891,1799,96
04,11397,16293,30372,4075,24145,12108,13083,3948,22434,31276,13977,6695,1110,886
```

表－10　｛0，1｝の一様乱数列（4）

0,0,0,0,0,1,0,0,0,0,1,0,1,1,0,0,0,0,1,1,1,0,0,0,0,0,0,1,0,0,1,1,0,1,1,0,1,1,0,1,
0,1,0,1,0,0,1,0,1,0,1,0,0,0,0,1,1,0,0,1,0,0,0,1,1,0,1,1,1,0,1,0,0,0,0,0,1,1,0,0,
1,1,1,1,1,0,1,0,1,1,1,0,1,0,1,1,0,0,1,1,1,0,0,1,0,1,0,1,0,1,0,0,1,1,1,1,1,1,1,0,1,
0,1,1,0,0,1,1,1,0,0,1,0,1,0,0,1,0,0,0,0,0,0,1,1,1,0,1,1,1,1,1,0,1,1,1,1,1,1,0,1,0,1,
0,1,1,1,1,1,1,0,0,0,1,0,0,0,1,1,0,1,0,0,1,0,1,1,1,0,0,1,1,0,1,0,1,1,0,1,0,1,1,0,0,1,1,
0,0,0,1,1,0,1,1,0,1,0,0,1,1,0,0,0,0,0,0,0,1,0,0,0,1,1,0,1,1,0,1,1,0,1,1,1,1,1,1,0,
0,0,0,0,0,0,1,0,1,0,0,0,1,1,0,0,0,0,1,1,0,1,1,1,1,0,0,1,0,1,1,1,0,0,0,0,1,1,0,1,
0,1,0,0,1,0,0,0,0,1,1,0,0,0,0,1,1,0,0,1,1,1,1,1,0,1,1,1,1,1,1,1,0,1,0,1,0,0,1,0,1,1,
1,1,0,0,1,0,0,1,0,0,0,1,0,1,0,0,0,0,0,0,1,1,0,1,0,0,1,0,0,1,0,0,0,0,0,1,1,0,0,1,0,1,0,
0,1,1,1,1,1,0,0,1,1,1,0,1,1,0,1,1,0,0,1,0,0,1,0,0,0,0,0,1,1,1,0,0,0,0,1,0,1,1,0,1,
1,1,0,1,1,0,0,0,0,0,0,1,0,0,1,0,0,0,0,0,1,1,0,0,0,1,1,0,1,1,0,1,0,1,1,1,1,1,1,1,0,
1,0,1,0,1,0,1,1,1,0,0,1,1,1,0,0,1,0,1,0,0,1,0,0,1,1,0,0,0,0,1,0,1,0,0,1,0,0,1,1,
0,1,0,0,0,0,1,1,0,0,0,0,0,0,0,1,1,1,0,0,0,1,0,1,1,0,1,1,1,0,0,0,1,0,1,0,0,0,0,1,
0,1,1,1,0,1,0,1,0,0,1,0,1,0,0,1,1,1,1,1,0,1,1,0,1,1,1,0,1,0,1,0,0,1,0,1,0,0,0,0,0,
1,0,0,1,1,0,0,0,1,1,1,1,1,1,0,1,1,0,1,0,0,1,0,0,1,0,0,1,0,1,0,1,1,1,0,1,1,1,1,1,1,0,
1,1,1,0,0,0,0,0,0,0,0,1,1,1,0,1,0,1,0,1,1,1,0,1,0,1,0,1,1,1,1,1,1,0,1,1,1,0,1,0,0,1,
1,1,1,1,1,1,0,1,1,1,1,1,1,1,0,1,1,1,1,0,1,1,1,1,0,0,0,1,1,0,1,1,1,0,1,1,1,1,0,1,1,0,
0,1,1,0,1,1,0,0,0,1,1,1,1,1,0,1,1,1,0,0,0,1,1,1,1,1,0,1,0,1,0,1,1,0,1,0,1,1,0,1,1,
1,1,1,0,1,0,1,0,0,1,0,0,0,1,0,0,0,0,1,0,1,1,1,0,0,0,0,1,1,0,1,0,1,1,1,1,0,0,0,1,1,1,
1,0,0,1,0,0,1,1,1,0,0,0,1,1,1,1,0,0,0,0,0,0,1,0,1,0,1,1,0,1,1,1,0,0,1,0,1,1,0,1,0,
1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,1,1,0,0,1,0,1,0,1,0,0,0,0,0,0,0,1,1,1,0,1,0,1,1,0,0,0,1,0,
1,1,1,0,1,1,1,0,0,0,0,1,0,1,1,0,1,1,1,0,
455

900個まで続けると表－10の様になる。455は1の出現回数である。

さて，このプログラムを利用して乱数発生回数を種々変化させて1の出現回数を列記し，それぞれの場合を表の形に表したのが表－11である。

これに対してカイ自乗検定を実行してみよう[5]。次の仮説H_0をおく。

H_0：0，1の出る確率はともに1／2である，
すなわち，Xを確率変数として確率$P(X=i)=1/2(i=0,1)$である。

表－11の最後を見ると10000個の乱数にたいして，1の出現回数が5026であるので，0のでる相対度数は$\frac{4974}{10000}$，1のでる相対度数は$\frac{5026}{10000}$である。よつて，χ^2に対応する確率変数Z^2の値z^2は

$$z^2=\frac{(-26)^2}{5000}+\frac{(26)^2}{5000}=0.2704$$

である。自由度1のχ^2-自乗分布の表から$Z^2\approx\chi_1^2$として

$$P(Z^2>\chi_0^2)=0.05$$

より

$$\chi_0^2=3.841$$

である。したがつて棄却領域Wは

$$W=(3.841,\infty)$$

となる。これより

$$z^2\notin W$$

となり，有意水準5％で仮説H_0は採択される。

表－11　1の出現回数実験

乱数発生回数	1の出現回数
50	19
100	45
150	72
200	102
250	126
300	146
350	173
400	192
450	217
500	243
・・・	・・・
900	455
1000	503
2000	1013
3000	1504
4000	1970
5000	2502
・・・	・・・
10000	5026

このことを情報量規準を用いてもう一度検定してみよう。0，1の出現確率がともに1／2である想定モデル$P(1/2,1/2)$に対して最大対数尤度MLLと情報量規準AICとは，それぞれ

$$MLL=10000\log\frac{1}{2}=-6931.5,$$

$$AIC=(-2)\times MLL+2\times 0=13863.0$$

である。

また，0,1の相対度数$\frac{4974}{10000}$，$\frac{5026}{10000}$をそれぞれの出現確率とする最尤推定モデル$P\left(\frac{4974}{10000},\frac{5026}{10000}\right)$に対する最大対数尤度$MLL$と情報量規準$AIC$とは，それぞれ

$$MLL=10000\times\{0.4974\log 0.4974+0.5026\log 0.5026\}$$
$$=-3473.646-3457.690$$
$$=-6931.336,$$
$$AIC=(-2)\times MLL+2\times 1=13864.673$$

である。よつて，この場合情報量規準の最小なモデルである想定モデル$P(1/2,1/2)$をこの乱数生成系の確率モデルとして選択することになる。

参考文献

1）北川敏男，藤野精一：n元連立1次合同型反復模型とその挙動解析，RMC64-09J（1989），pp.171。
2）藤野精一：多変数合同型高次反復模型の反復図とその特性について，久工大研究報告№19（1995），47-51。
3）伏見正則：乱数，（東大出版会，1994），pp.160.
4）戸川隼人：ザC_{++}，（サイエンス社，1993），pp.223。
5）猪野富秋，伊藤正義：数理統計入門，（森北出版，1987），pp.206。