擬似乱数の発生アルゴリズムとその検定

統計数理研究所　逆瀬川　浩　孝

(1980年1月　受付)

Uniform Random Number Generators and Their Statistical Tests

H. Sakasegawa
(The Institute of Statistical Mathematics)

The multiplicative congruential method to generate an uniform random number sequence is said to be bad in statistical nature. For the case with a large modulns, however, this method produces a sufficiently statisfactory random sequence with suitable selections of parameters. The merging method, where several multiplicative congruential sequences are randomly merged, is discussed to generate a pseudo-random number sequence with exceedingly long cycle. It is profitable even for the use in the machine with short word length where one cannot use a large modulus algorithm easily.

1. 乱数を使用する，云わゆるモンテカルロ・シミュレーションを計算機を用いて実行する場合，一様乱数をどのようにして発生させるか，ということが最も基本的な課題となる．これに対しては，現在，少なくとも3つの解決法が用意されていて広く使われているようである．

一つは電子機器の熱雑音のように，本質的にランダムな事象を適当な方法で数量化して乱数を得るという，云わゆる物理乱数によるものである．これは古くから乱数表を作る際に用いられていたもので，現在でも使用される計算機に合せて，いろいろな乱数発生装置が作られている ([4], [10])．この方法の良い点は，周期性 (同じ数の並びが再び現れること) が本質的に存在しないことであるが，一方欠点としては，物理的な装置を必要とする為利用者が限られること，長期的な傾向性が存在する可能性があること，再現性がない為，シミュレーションの結果を再検討する為に，乱数列の検定を行なおうと思っても不可能であること，あるいは，その使用の都度検定しなければならないので時間がかゝること等があげられる．第二の方法は乱数表を使うことである．十分に大きな乱数表を計算機の補助記憶装置に蓄えておき，与えられた規則に従ってそれらの中から順に取り出して使う方法である ([2])．必要に応じた大きさの補助記憶装置さえあれば，理想的な乱数列を使うことが出来るので大変好ましい反面，記憶量に制限があるので特大規模のシミュレーションには不向きである，補助記憶をアクセスするのに時間がかゝる等の欠点がある．しかし，これらの点は，例えば，第一の点については乱数表を定期的に取り替えるとか，取り出し規則を工夫する，第二の点についてはいくつかの乱数を一度にまとめて読むようにする，等の工夫によって，ある程度回避することができる．第三の方法は，云わゆる擬似乱数によるもので，現在最も広く用いられているものである．2つの数の間の四則演算という純粋に規則的な操作の繰り返しの中からランダムのように見える数列を見出し，計算機シミュレーションに利用したのはフォン・ノイマンが最初であろうと言われているが ([11]) その後いろいろな人達が，計算によって作り出される乱数——擬似乱数——について論じている (たとえば [5])．この方法のよい点は，計算機以外の物理的な道具を必要としないこと，発生速度が早いこと，計算規則の選び方によっては，比較的良質の数列が得られること等である．逆に欠点としては，当然のことながら，生成される数列は必ず周期を持つこと，隣り

合う数字の間に，統計的な意味で相関が見出される可能性があること等が言われている．この小論においては，第三の方法，すなわち擬似乱数の発生アルゴリズムと，生成された擬似乱数列の検定について考える．

2. 擬似乱数の発生アルゴリズムの中で最もポピュラーなものは Lehmer [6] によって提唱された乗算合同法である．これは，二つの正の整定数 λ と P，及び，初期値となる正整数 x_0 とから，次式によって次々と $x_1, x_2, \cdots$ を生成する方法である．

$$x_{n+1} \equiv \lambda x_n \pmod{P} \quad (n = 0, 1, 2, \cdots) \tag{1}$$

これを $M(x_n; \lambda, P, x_0)$ と書くことにしよう．数列 $\{x_n\}$ は (λ, P, x_0) の3つ組によって完全に規定されるが，これら3つ組の選び方によっては，得られた数列は，あたかも1から $P-1$ 迄の整数値をランダムにとる離散確率変数の実現値であるかのようにふるまう．もし，P が十分に大きければ $x_n' = x_n/P$ は，0と1の間の連続な一様分布からのランダムサンプルと見做せる．$M(x_n; \lambda, P, x_0)$ の性質とか，λ, P の選び方についてはいろいろな人によって論じられているが，例えば局所的な性質等解明されるべき問題が残されている．$M(x_n; \lambda, P, x_0)$ のいろいろな性質の中でも，Marsaglia ([8]) によって指摘された多次元配置の規則性に関する次の性質は，かなり衝撃的である．すなわち，$M(x_n; \lambda, P, x_0)$ の隣り合う k ケの数字の組 $(x_n, x_{n+1}, \cdots, x_{n+k-1})$ を k 次元空間の1つの点とみなせば，$\{(x_n, x_{n+1}, \cdots, x_{n+k-1})\}_{n=0,1,2,\cdots}$ は非常に少数の k 次元空間超平面の上にのっている，というものである．これが実際のシミュレーション結果にどれだけ影響を与えているかという点については必ずしも明確でないが，このような規則性を予想して，乗算合同法を改良する為，彼は MacLaren と共に，次のような発生アルゴリズムを提案している ([7])．

MacLaren と Marsaglia の Shaffling アルゴリズム (M & M 法)

1. $M(x_n; \lambda, P, x_0)$, $M(y_n; \mu, Q, y_0)$, 正定数 m 及び，$(0, Q)$ から $(1, m)$ への変換 f を用意する．
2. $\omega_l \leftarrow x_l$ $(l = 1, \cdots, m)$, $n \leftarrow 0$
3. $n \leftarrow n+1$, $z_n \leftarrow \omega_{f(y_n)}$, $\omega_{f(y_n)} \leftarrow x_{n+m}$
4. 3に戻る．

$\{z_n\}_{n=1,2,\cdots}$ が目的の数列である．これを $M^2(z_n; \lambda, P, x_0, \mu, Q, y_0, f)$ と書くことにする．原著論文では，$\lambda = 2^7+3$, $\mu = 2^7+1$, $P = Q = 2^{35}$, $m = 128$, $f(x) = [x/2^{28}]+1$ ([ ] はガウス記号) であった．彼等が m の値を $128 = 2^7$ としたのは，割り算をすることなく，それより高速で出来るシフト演算によって先頭の7ビットを取り出すことが出来ると考えてのことであろう．λ, μ の選び方についても同様である．しかし，現在の計算機においては，このような制約は余り意味を持たない．又，$P=Q$ とし，$m|P$ であることによって新しい乱数列の周期は，$\{x_n\}$ のそれと同じであり，この点でも改良の余地がある．ここでは，次のようなパラメータの選択を提案しておこう．すなわち，m, P, Q を互いに素である正の整数，$S = [(Q+m-1)/m]$ を使って $f(x) = [x/S]+1$ を定義し，上の手順に従って $\{z_n\}$ を生成する．実際の計算では，P は2の巾乗，Q は大きな素数，m は1桁の奇数とすれば良い．

3. M & M 法ではテーブルを使い，その要素がなくなると共通の数列 $\{x_n\}$ から補充するという方法をとっていたが，乱数源を1つの $\{x_n\}$ に限定する必然性は余りないようである．そこで，テーブルの各欄に一つの数列を対応させ，ある要素が使われるとその欄に固有の数列から補充することにしよう．こうすればテーブルがいらなくなる．云いかえれば，何本かの数

列を用意して，それらをランダムに「織り交ぜる」(merge する) 方法である．このような方法でも，乗算合同法による擬似乱数列の順序を乱すことが出来る．後述するように，このような方法は P が大きい時には，統計的な意味で，乱数列を「改善」するのに余り役立っていないように思われるが，P が小さい時，その効果は大である．$M\,(x_n;\ \lambda,\ P,\ x_0)$ は，必ず長さ P 以下の周期を持ち，また，P が 2 の巾乗の場合は，後述するような準周期が存在するので，P が小さい場合には，$\{x_n\}$ のうち，使える範囲が非常に狭いものになってしまう．また，P が小さくなるにつれて，系列相関が強くなる傾向がある．このため P が小さい時は，周期を延ばし，順序を乱すことが，重要な意味を持ってくるのである．

4. 与えられた数列を乱数列と見做してよいか否かを判定する為の確固たる手順は，現在のところ存在しないと言ってよい．これは，有限列のランダムネスの定義が十分に整っていないところに由来している．現在，普通に行われている方法は，その数列が，一様母集団からのランダムサンプルである，という仮説を，いくつかの方法で検定し，その総合結果から判断する，というものである．検定の種類とか，実行手順，パラメータの選択等については，検定を行なう者の好みにまかされており，標準的な検定方法さえ確立していない．我々の採用した検定は次のようなものである．

(1) 一次元・二次元・三次元の頻度検定 (T_{11}, T_{12}, T_{13})

(2) マルコフ性の検定（与えられた数列を，状態間の推移が等確率であるようなマルコフ過程の一つの実現値とみて，標本推移度数の等確率性を検定する）(T_{21})

(3) モーメント（平均・分散）の検定 (T_{31}, T_{32})

(4) 遅れが 1, 2, ･･･ の系列相関係数の検定 (T_{41}, T_{42}, ･･･)

(5) 二次元・三次元のランダム距離の検定 (T_{51}, T_{52})

(6) 連の検定（上昇連・下降連の長さの分布，個数の平均，符号連）(T_{61}, T_{62}, T_{63})

詳しい検定手順については補遺にまとめてある．各発生アルゴリズムごとに生成された擬似乱数列の中から長さ 20000, 40000, 60000, 100000 の連続した部分列を（出発点を変えて）取り出し，それぞれを 20 等分して各々に対して上述の 6 通りの検定を行なった．例えば，長さ 20000 の数列について，先頭から 1000 ずつ 20 ケの部分列に分け，部分列ごとに，各テストにおける統計量の値を計算し，有意水準を 10%, 5%, 1% とした場合の仮説の棄却された回数をカウントする．更に，20 ケの統計量の値それぞれに対してその上側確率を求め，それらが (0, 1) の間で一様に分布しているか否かを Kolmogorov-Smirnov の検定によって調べた．

5. 先ず，$P \leqq 2^{16}$ の場合，及び，それを組み合せた M & M 法，Merge 法についての検定結果について述べる．使われたパラメータの値は次のようなものである．

(a) $M\,(x_n;\ 3989,\ 2^{16},\ x_0)$

(b) $M\,(x_n;\ 2^6+3,\ 2^{16},\ x_0)$

(c) $M^2\,(z_n;\ 3989,\ 2^{16},\ x_0;\ 293,\ 2^{16},\ y_0,\ f)$
但し，$f(x) = [x/2^9]+1$ $(m = 128)$

(d) $M^2\,(z_n;\ 3989,\ 2^{16},\ x_0;\ 512,\ 63299,\ y_0;\ g)$
但し，$g(x) = [x/9199]+1$ $(m = 7)$

(e) $M\,(x_n^{(1)};\ 3989,\ 2^{16},\ x_0^{(1)})$, $M\,(x_n^{(2)};\ 109,\ 2^{16},\ x_0^{(2)})$
$M\,(x_n^{(3)};\ 512,\ 54563,\ x_0^{(3)})$ を $M\,(y_n;\ 512,\ 63299,\ y_0)$ と $h(x) = [x/13577] \pmod 3 + 1$ によって Merge する方法 ($x_n^{(i)}$ はその法で割って 0 と 1 の間の数に揃えておく必要がある．)

初期値は，いずれも適当に選ばれた奇数とした．法が 2^{16} でないものはいずれも素数で，この場合の乗算定数は原始根である．

有意水準1%で仮説が棄却された回数が20回中2回あるいはそれ以上だったものは，(a)では長さ4万の場合の T_{31}(平均)（これを $T_{31}(4)$ と書く），(b)では $T_{62}(10)$（連の数の平均），(c)も(b)と同じ $T_{62}(10)$，(d)は(a)同様 $T_{31}(4)$，(e)では，そのようなことは起らなかった．危険率5%，10%での棄却回数も考慮すると，(a)では $T_{63}(2)$（符号検定），(b)では T_{41}（遅れ1の系列相関係数）T_{52}（三次元のランダム距離），T_{62}，(c)では T_{13}（三次元の頻度検定），T_{31}，(d)では $T_{13}(4)$，T_{31}，(e)では T_{31}，$T_{35}(10)$，$T_{51}(4)$（二次元のランダム距離），$T_{62}(4)$ が余り好ましくない成績であった．総合的に考えると，棄却回数で見る限り(b)を除けば，どれも乱数列と見做してさしつかえない，という結論を導びくことができる．しかし，上側確率に関するK-S検定によると，その偏りが顕在化する．(a)，(b)については T_{11}，T_{12}，T_{13}，T_{21}，T_{51}，(c)，(d)については T_{11} がいずれも大きな値が出すぎるのである．これらは，いずれも χ^2- 検定であるので，言いかえるとどの部分列も，理論分布に近いふるまいをしているということになる．これは一見良いようであるが，乱数表から10ケの数字を取り出した時0から9迄のすべての数字が揃うというのはまれであるということからも，乱数としての好ましい性質ではない，ということが判る．(a)，(b)において頻度検定が殆んど棄却されないということは，$M(x_n;\ \lambda,\ P,\ x_0)$ に存在する準周期の存在によって説明できる．乗算合同法で $P=2^b$ とした場合，その周期は 2^{b-2} になることから，$M(x_n;\ \lambda,\ 2^b,\ x_0)$ から長さ 2^{b-2} の数列を取り出してこれを mod 2^{b-1} で見れば，全く同じ数列が2度繰り返し現れることになる．あるいは一般に，これを mod 2^c $(c<b)$ で見れば，2^{b-c} 回の同じ数列の繰り返しとなっていることが判る．これを乗算合同法における準周期と呼ぶが，b が小さい時は，この存在が，数列の統計的性質に大きな影響を及ぼすことになる．

テストされた(a)～(e)の5つの1語16ビット用の乱数発生アルゴリズムの中では，ここで提案されたMerge法が最も良好な結果を与えているようである．パラメータの選び方に関する理論的な基準はなく，経験的に定めるだけであるが，小さい計算機でも長いシミュレーションに使用可能な乱数列を発生できるという点で，この方法は一考に値すると思われる．

6. 一語32ビットの計算機を想定し，次のような擬似乱数列に対して行なった検定結果を検討する．

(a) $M(x_n;\ 39894229,\ 2^{32},\ x_0)$

(b) $M(x_n,\ 65539,\ 2^{32},\ x_0)$

(c) $M(x_n;\ 1542272173,\ 2^{32},\ x_0)$

(d) $M^2(z_n;\ 39894229,\ 2^{32},\ x_0,\ 2718285,\ 2^{32},\ y_0,\ f)$
但し，$f(x)=[x/2^{25}]+1$

(e) $M^2(z_n;\ 39894229,\ 2^{32},\ x_0;\ 512,\ 1999307,\ y_0,\ g)$
但し，$g(x)=[x/285629]+1$

(f) $M(x_n^{(1)};\ 39894229,\ 2^{32},\ x_0^{(1)})$，$M(x_n^{(2)};\ 2718285,\ 2^{32},\ x_0^{(2)})$
$M(x_n^{(3)};\ 512,\ 1991027,\ x_0^{(3)})$ を $M(y_n;\ 512,\ 1999307,\ y_0)$ と $h(x)=[x/432121]\pmod 3)+1$ を使って Merge する方法 $(x_n^{(i)}$ はいずれも取り出される前にその法で割っておく)

パラメータの値について説明しよう．(b)は乗算よりシフト演算の方がスピードが速いといわれていた時代に愛用されていたもので $(65536=2^{16}+3)$，多くの人によってその性質の悪さを指摘されていながら，現在もなお広く使われているものである．(c)の定数は原田[3]によっ

て提案されたもので，Coveyon & MacPherson [1] のスペクトル解析によるパラメータ選択法を $P=2^{32}$ の場合に適用したものである．(d) でのテーブルの大きさは 128，(e) では 7 になる．(e)，(f) で，法が 2^{32} でないものはいずれも素数で，乗算定数はそれらの原始根になっている．また，乗算によって 2^{31} を越えないようにとってあるので，FORTRAN でプログラムした時は，整数計算だけで済むようになっている．

検定の結果，仮説の棄却回数という点からみて，不満足な結果を得たのは，(a) では T_{12} (10)（二次元頻度検定）T_{21} (4)（マルコフ性検定），(b) では T_{12} (6)，T_{52} (6)，T_{51} (10)，T_{61} (10)，T_{62} (10)，(c) では T_{45} (2)，T_{12} (10)，(d) では T_{62} (10)，(e) では T_{44} (10) で，(f) には特に悪い性質は現れなかった．又，全体的な傾向として，(b) が他に比べて棄却回数が多かった．これで見るように，(b) を除けば，どのアルゴリズムも，どのような部分列をとってもある検定に対して常に同じ傾向を示すということがない．一方，上側確率に関する K-S 検定では，特に目立った傾向を検出することは出来なかった．従って，ここで取り上げた性質に関する限り，(b) を除いては，どのアルゴリズムから生成される数列も，一様乱数列として使用し得る，といってよい．

7. 検定の種類は上で取り上げた以外にもまだ沢山あって，種々の検定手順を集めた乱数列検定用のプログラムパッケージもいくつか作られているようであるが，検定の種類を増やすことには余り意味がないように思われる．むしろ，使用目的に合せて，どのような検定を通したらよいのか，ということとか，得られた結果をどう解釈するのか，ということの方が重要である．後者について言えば，例えば，M (x_n; 39894229, 2^{32}, x_0) のある初期値から出発した 10 万個の数列の検定では，T_{12} が不合格であったが，同じ数列の最初の 2 万個，4 万個，6 万個の数字について検定してみると，T_{12} で不合格になるものはなく，新たに，T_{11}（一次元の頻度検定），T_{42}（遅れ 2 の相関係数の検定），T_{61}, T_{62}（上昇連・下降連に関する検定）等で不合格になるものが出てきた．このような検定結果の不規則性は，乱数列の検定というものの困難さを想像させる．乱数列の検定では $P=2^{16}$ の場合に見られた局所的な均質性とでも呼べるような，系統的に現れる好ましくない性質を検出したり，多数回の検定の結果，棄却回数が期待されるそれに比べて著しく多かった，というような総合的判断が重要であって，ある一回の検定で，有意水準 1% で棄却された，というようなことは，余り重要ではないと言ってよい．

一方，第一の点については，用途に応じてそれぞれ異なった検定法が考えられ，こうでなければならないと言うような一般的法則は存在しない．只，検定の冗長性を避ける為，検定の間の相関を調べておく必要がある．各 T_{ij} ごとに得られる 20 ケの上側確率の値から，相関行列を作り，T_{ij} 間の相関の強さを推測することができる．ここで取り上げた 6 種類の検定の中で相関が高かったのは遅れ 1 の系列相関検定と，符号連の検定で，この両者の間には負の相関が見られるが，他のどの二つについても系統的な相関は検出されなかった．

8. 以上の数値実験の結果から，乗算合同法のアルゴリズムによって生成される数列は，パラメータの選び方によっては統計的に劣悪でない乱数列と見做し得ることが出来る．また，Marsaglia によって指摘された多次元配置の規則性については，結果の判っているいくつかの問題について，パラメータをいろいろに変えた乱数列によってシミュレーションをした場合，推定値が偏りを示さなかった，ということから，乗算合同法アルゴリズムを放棄する理由にならない，と考えられる．元の数列の統計的性質を「改良」する為に考えられた方法は，(1) 式における P が十分大きい時，ここで調べられた種類の検定に関する限り，目立った改善になっていない．一方，P が小さく，2^{16} 程度の場合には，準周期の影響で，使用可能な範囲が狭め

られるので，その周期を延す工夫が必要で，この場合，一本の数列の順序を乱すだけの M & M 法では不十分で，何本かの数列を一本にまとめて使う Merge 法が良いようである．P が大きい場合でも，一回の計算に P の何倍もの乱数を必要とする場合は Merge 法を使わねばならないことは言うまでもない．

パラメータの決め方に一定の規則はなく，ここで実験に使われたのも，ある種の定数表から抜き出してきたものである．経験的に次のような決め方が推奨できる．現在の計算機は殆んど二進法であり，桁あふれを無視することによって自動的に mod 2^b (b は一語を構成するビットの数）の計算が行われるので，P は 2^b に取るのが良く，λ は，2^{b-7} と 2^{b-3} の間にあって $\lambda \equiv 5 \pmod 8$ となるような数で，二進展開した時，0と1が適当に混ざり合っているようなものを選ぶ．このようにすれば，任意の奇数 x_0 を初期値として，周期が 2^{b-2} であるような擬似乱数列を得ることが出来る．このような単純な乗算合同法を使えば，実際のコーディングは一行で済ませるから，シミュレーションプログラムの中に組込みで（サブルーチンにすることなく）使うことが出来る．このことは，計算効率を上げ，結果的に，推定の精度を上げるのに役立つ．HITACHI M-180 計算機におけるコーディングの例を補遺に載せる．このコーディングによる乱数1つの生成速度は約 4μ sec であり，サブルーチン形式をとる他の改良法がいずれも約 20μ sec の時間を要することから，この組込み型の乱数発生法の有利さが判る．

なお，乗算合同法アルゴリズムの変形としては，乗算定数 λ を各回ごとに変える方法とか，Westlake [12] のように2つの数列をビットごとの繰り上がりなし足し算（排他的論理和）により，一つの数列に合成する方法等があり，乗算合同法以外のアルゴリズムとしては，云わゆる M-系列を利用した Tausworthe [9] の方法がある．後者の2つの方法はいずれも注目に値するが，ビット演算を必要とする為，余り一般的になりにくいので，ここでは考察の対象からはずした．

謝　　辞

有益な助言をいたゞいた査読者に感謝します．

補　　遺

A. ここで使われた乱数の検定方式について若干説明する．検定すべき仮説は，数列 $x_1, x_2, \cdots, x_N$ が0と1の間で一様分布している確率変数の独立サンプルである，というものである．

A1. **k 次元頻度検定**，$(x_{lk-k+1}, x_{lk-k+2}, \cdots, x_{lk})_{l=1,2,\cdots}$ を k 次元単位立方体の点とみて，これらが空間内に一様に分布しているか否かを検定する．k 次元単位立方体を適当な小区間に分割して標本度数分布を作り，適合度検定を行なう．

A2. **マルコフ性の検定**，$f(x) = [kx]$ とする．仮説の下では，$f(x_1), f(x_2), \cdots$ は，状態数 k，推移行列のすべての要素が $1/k$ であるようなマルコフ連鎖の1つの実現値とみなせるから，その尤度比検定を考えることが出来る．$n_{ij} = \#\{l; f(x_l) = i,\ f(x_{l+1}) = j\}$ とすれば，

$$\frac{k^2}{N}\sum_{i,j}\left(n_{ij} - \frac{N}{k^2}\right)^2 - \frac{2k}{N}\sum_i\left(\sum_j n_{ij} - \frac{N}{k}\right)^2$$

は近似的に自由度 $(k-1)^2$ の χ^2-分布に従う．

A3. **モーメントの検定**，標本平均・標本分散は近似的にそれぞれ $N\left(\frac{1}{2}, \frac{1}{12n}\right)$，$N\left(\frac{1}{12}, \frac{1}{180n}\right)$ に従うことを使う．

A4. **遅れ k の系別相関検定**，

$r_k = \left\{\frac{1}{N-k}\sum x_l x_{l+k} - \left(\frac{1}{N}\sum x_l\right)^2\right\} \Big/ \left\{\frac{1}{N}\sum x_l^2 - \left(\frac{1}{N}\sum x_l\right)^2\right\}$ が近似的に $N\left(-\frac{1}{N-1}, \frac{n(n-3)}{(n-1)^2(n+1)}\right)$ に従うことを使う．

A5. **k 次元ランダム距離の検定**，$(x_{2kl-2k+1}, x_{2kl-2k+3}, \cdots, x_{2kl-1})$，$(x_{2kl-2k+2}, x_{2kl-2k+4}, \cdots, x_{2kl})$ を k 次元の点とみて，これらの間の距離の自乗 d_k^2 を計算する．

$$d_k^2 = \sum_j (x_{2kl-2k+2j-1} - x_{2kl-2k+2j})^2$$

仮説の下での d_k^2 の分布函数 $F_k(x)$ から，$a_j = F_k^{-1}\left(\frac{j}{10}\right)$ $(j = 0, 1, \cdots, 10)$ を計算し，その分点にもとずくヒストグラムによって適合度検定を行なう（[13]）．

A6. **連の検定**，$x_l > x_{l+1} < x_{l+2} < \cdots < x_{l+k} > x_{l+k+1}$ となる数の並びを，長さ k の上昇連，不等号が逆向きの場合は下降連と呼び，両方合せたものを単に連と呼ぶ．長さ 1, 2, 3, 4 及び 5 以上の連の数をカウントし，理論度数との乖離を適合度検定によって調べる．又，全体の連の数は近似的に $N\left(\frac{2n-1}{3}, \frac{16n-29}{90}\right)$ に従うから，正規分布表を使って検定できる．一方，$x_n - \frac{1}{2}$ の符号の並びを考え，両側を異符号によってはさまれた同一符号の並びを符号連と呼ぶことにすると，符号連の総数は，近似的に $N\left(\frac{2nm}{n+m}+1, \frac{2nm(2nm-n-m)}{(n+m)^2(n+m-1)}\right)$ に従う．（n, m はそれぞれ，+，− の数である）．

B. FORTRAN でのコーディングの例として次のようなものがある（HITACHI M-180（一語 32 bit）用である）

1. 乱数を必要とする各プログラム単位に次の宣言文を置く．

```
DATA  LAMBDA/39894229/, R32/.23283064E-9/
```

ここで R32 は 2^{-32} である．

2. 初期値を定義するか入力する．

```
KR = 1234567
```

又は

```
Read (5, ××) KR
```

3. 乱数を必要とする都度，次の文を入れる．

```
KR = KR*LAMBDA
```

これによって KR には "次の" 乱数がセットされる．計算は mod 2^{32} で行われるが M-180 FORTRAN では，先頭のビットを符号と見なすので，KR は -2^{31} と $2^{31}-1$ の値の数になっている．U (0, 1) のサンプルに変換するには

```
R = KR*R32+0.5
```

とすればよい．

参 考 文 献

[1] Coveyou, R.R. and MacPherson, R.D. (1967) Fourier analysis of uniform random number generators, *J. Ass. Comput. Mach.*, **14**, 100–119.

[2] Goto, M., Inoue, T. and Asano, C. (1978) Confirmative studies of NISAN random numbers, *Res. Rep.* 91, RIFIS Kyushu Univ.

[3] 原田紀夫 (1974) スペクトル検定に対する乱数列発生法の最適係数，情報処理 15 巻．

[4] Ishida, M. and Ikeda, H. (1956) Random number generators, *Ann. Inst. Statist. Math.* **8**, 119-126.

[5] Knuth, D.E. (1969) *The Art of Computer Programming*, Vol 2, Addison Wesley.

[6] Lehmer, D.H. (1948). Mathematical method in large-scale computing units, *Proc. 2nd Symp. Large Scale Calculating Machinery*, 141–146.

[7] MacLaren, M.D. and Marsaglia, G. (1965) Uniform random number generators, *J. Ass. Comput. Mach.*, **12**, 83–89.

[8] Marsaglia, G. (1968) Random numbers fall mainly in the planes, *Proc. Nat. Acad. Sci.* **61**, 25–28.

[9] Tausworthe, R.C. (1965) Random numbers generated by linear recurrence modulo two, *Math. Comp.*. **19**, 201–209.
[10] Tocher, K.D. (1963) *The Art of Simulation*, English Univ. Press.
[11] Ulams, S. (1976) *Adventures of A Mathematician*, Charles Scribner's Sons.
[12] Westlake, W.J. (1967) A uniform random number generator based on the combination of two congruential generators, *J. Ass. Comput. Mach.*, **14**, 337–340.
[13] 山本英二, 菅野長武 (1979) 3次元の random distance にもとづく一様乱数の検定, 第47回日本統計学会予稿集 152-153.