

擬似乱数の検定法とSDEの近似解の精度について

山梨大学教育学部 田中 哲 (Satoru Tanaka)

山梨大学教育学部 金川 秀也 (Shuya Kanagawa)

はじめに

確率微分方程式 (SDE)

$$dX(t) = \mu(X(t), t)dt + \sigma(X(t), t)dB(t)$$

の Euler-Maruyama 型の近似解の精度については [5], [6], [9]-[16], [19], [21] などに詳しく調べられている。しかし、Euler-Maruyama 型の近似解が実際にコンピュータ上で応用される場合は確率

$$\begin{cases} dX(t) = X(t)dB(t), 0 \leq t \leq 1 \\ X(0) = 0.1. \end{cases}$$

変数列の代わりにいわゆる擬似乱数とよばれる代数的に構成された数列が用いられる。そこで、擬似乱数の質をどう調べればよいかということが問題になる。本報告では一例として次のような線形な確率微分方程式に注目し、その近似解の精度と擬似乱数の検定法との関係について考察する。

この方程式の真の解は、伊藤の公式によって次のように与えられる。

たいていのプログラミング言語に組み込まれている擬似乱数は、線形合同法のように代数的な方法によって生成されるため必ず有限な周期が存在する。最も頻繁に用いられる線形合同法によって生成される擬似乱数は次元が高くなるにつれて超平面群にのり、一様性について問題があることが知

$$X(t) = 0.1 \exp \left\{ B(t) - \frac{t}{2} \right\}.$$

られている。したがって、擬似乱数が SDE の解にどのような影響を与えるのかを調べるために次の点に注目した。

- i) 統計的検定にはいろいろな種類があるが、どの検定を通った擬似乱数が近似解に適しているのか。
- ii) 擬似乱数列の乱数種 (seed) は、近似解の密度分布に影響を与えるのか。

第 1 章では線形合同法 (3 種類) と M 系列 (2 種類) を用いて次に示す A~E の 5 種類の擬似乱数生成法について述べる。M 系列乱数は、伏見 (1989) を参考にして生成したものである。

- A) $X_i = 1664525X_{i-1} + 1013904223$ の混合型合同法
- B) $X_i = 16807X_{i-1}$ の乗算型合同法
- C) $X_i = 65539X_{i-1}$ の乗算型合同法
- D) $f(x) = 1 + x^{32} + x^{521}$, $\sigma = n = 32$ の M 系列乱数
- E) $f(x) = 1 + x^{273} + x^{607}$, $b = 23, \sigma = 512$ の M 系列乱数

第2章では擬似乱数列の等確率性と無規則性を検証するために一般に良く用いられる次の8種類の統計的検定について述べる。

- 1) 一次元度数検定 (一様分布検定)
- 2) 二次元度数検定 (系列検定)
- 3) ポーカー検定 (poker test)
- 4) 間隔検定 (gap test)
- 5) 連の検定 (run test)
- 6) 順列検定 (permutation test)
- 7) d^2 -検定 (d^2 -test)
- 8) 組み合わせ検定 (combination test)

第3章では第1章で述べた5種類の擬似乱数を用いてコンピュータシミュレーションを行い、近似解の精度を調べる。このシミュレーションには、TurboC++ Ver5.0J を使用してプログラムを作成した。またハードウェアには IBM/AT 互換機を使っている。

擬似乱数の検定には、実際に SDE のシミュレーションを行う時間よりもはるかに長い時間を要するため、多くの検定によって精度の良い擬似乱数を探すことは困難だと言える。そこで、どの検定を通ったものが SDE の近似解に適しているかが分かればたいへん有功である。

1. 一様擬似乱数列の生成法

コンピュータシミュレーションを行うためには大量の擬似乱数が必要とされるために、より簡単な方法で作ることが要求される。しかし真の乱数を得るにはその個数と同数の手順を必要とされることが指摘されていることから (Knuth (1981))、擬似乱数には等確率性と無規則性について本質的な問題が存在する。この節ではいくつかの「擬似乱数生成ルーチン」の作り方、それを変換して正規擬似乱数列を作る方法、およびその統計的検定法について述べる。一様擬似乱数の生成には次のような方法が用いられている。

$\left\{ \begin{array}{l} \text{線形合同法 (乗算型合同法・混合型合同法)} \\ \text{M系列} \\ \text{その他} \end{array} \right.$

一様擬似乱数を生成する方法は、線形合同法とM系列のほぼ二種類に限られている。その他の方法としては、物理現象の中に現れる特性を使って物理乱数と呼ばれる乱数列を作る手段がある。物理乱数には合同法などの代数的な生成法で得られた擬似乱数のような周期がないのが大きな特徴であるが再現性に問題がある。ここでは物理乱数列についてはふれない。

線形合同法

コンピュータのシステムが定義する乱数生成ルーチンの中で、最も広く使われてきたのはレーマーが発表した線形合同法 (linear congruential method) である。これは漸化式

$$X_i = aX_{i-1} + c \pmod{M} \quad (1.1)$$

を用いて0以上 $M-1$ (=RAND_MAX¹) 以下の非負整数列を生成するものである。

¹ ANSI C で規定されているように、rand() の最大の値は RAND_MAX で与えられる。

- $M > 0$ は法 (modulus)
- $a \geq 0$ は乗数 (multiplier)
- c は増分 (increment) と呼ばれ、 $\begin{cases} c = 0 \text{ の場合、乗算型合同法} \\ c \neq 0 \text{ の場合、混合型合同法} \end{cases}$ に分類される。

区間 $[0,1)$ 上の一様擬似乱数列 $\langle u_i \rangle$ は次式によって生成される。

$$u_i = X_i / M \quad (1.2)$$

$\langle X_i \rangle$ の周期やランダムネスは M, a, c の選びかたによって決まる。法 M に関しては、 2^b (b は使用する計算機のビット数)、あるいはこれを超えない最大の素数のとき周期がもっとも長くなるが、 $M = 2^b$ とした場合には、 $\langle X_i \rangle$ の下位の桁があまりランダムにならないという欠点がある。 a, c の選びかたと $\langle X_i \rangle$ の周期に関しては、次のことが知られている (Knuth (1981))。

定理 1.1 線形合同法が最長周期 M を持つための必要十分条件は、

- c が M と互いに素である。
- $a-1$ が、 M を割り切るすべての素数の倍数である。
- M が 4 の倍数であれば、 $a-1$ も 4 の倍数である。

この定理により、 $c = 0$ の乗算型合同法は最長周期 M を達成できない。それでも、 M が十分に大きければ、長い周期を得ることは可能である。

定理 1.2 乗算型合同法数列の周期に関して次のことが成り立つ。

- $M = 2^b$ ($b \geq 4$) の場合、可能な最長周期は $M/4$ であり、これを達成できるのは、 $a \pmod{8} = 3$ または 5 で、 X_0 が奇数のときに限られる。
- M が 2 より大きい素数 p に等しい場合、可能な最長周期は $p-1$ であり、これを達成できるのは、 $X_0 \neq 0$ で、かつ $p-1$ の任意の素因数 q に対して $a^{p-1/q} \not\equiv 1 \pmod{p}$ が成り立つ場合に限られる。

プログラム

擬似乱数 A. 剰乗演算のいらぬ擬似乱数生成法

C 言語では、unsigned long int 型の整数の長さは、32 ビットである。これを使って、32 ビットどうしの掛け算をすれば、結果は真の 64 ビットの積の下位 32 ビットである。したがって、 $M = 2^{32}$ とすれば、剰乗演算 $(\text{mod } M)$ は不必要になり、単に

$$X_i = aX_{i-1} + c \quad (1.3)$$

だけになる。したがって、実行時間がより早くなる。また、ここでは Knuth が見つけた $a = 1664525$ (表 1 の番号 10) と、H.W.Lewis がこの値とともに大規模な検定を行った $c = 1013904223$ ($(\sqrt{5} - 2)M$ に近い素数) を使う ([23]参照)。

擬似乱数 B. Schrage の技法を使った「最低基準」生成法

Park and Miller が提案した「最低基準」生成法は、乗算合同法

$$X_i = aX_{i-1} \pmod{M} \quad (1.4)$$

と次の値を使って擬似乱数列を発生させる方法である。

$$a = 7^5 = 16807 \text{ (表1の番号3)}, \quad M = 2^{31} - 1 = 2147483647$$

この値を使って計算すると、C言語の long int 型の整数の長さ 32 ビットを超えてしまうので、ここで Schrage の技法を使う。これを使えば、必ず「最低基準」生成法が実現できる。もちろん、このプログラムによって生成する擬似乱数列は、0 以上 $2^{31} - 2$ 以下の非負の整数値である。

さて、Schrage の技法は以下のとおりである。これは、 M の近似的素因数分解に基づいている。

$$M = aq + r, \text{ すなわち } q = \lfloor M/a \rfloor, r = M \pmod{a} \quad (1.5)$$

この式より、 r, q の値を決める。この値を使うことで、乗算合同法の擬似乱数列 $aX \pmod{M}$ の値を求めることができる。

定理 1.3

$$aX \pmod{M} = \begin{cases} a\{X \pmod{q}\} - r\lfloor X/q \rfloor, & aX \pmod{M} \geq 0 \\ a\{X \pmod{q}\} - r\lfloor X/q \rfloor + M, & aX \pmod{M} < 0 \end{cases} \quad (1.6)$$

擬似乱数 C. $X_i = 65539 X_{i-1}$ による乗算型合同法 (3次元以上でランダムでない例)

(1.4)式で、 $a = 65539$ とおいたもの。

スペクトル検定

合同法乱数列を生成する漸化式(1.1)の次数は 1 である。したがって、 k 次元の合同法乱数列の点列 $\{(X_i, X_{i+1}, \dots, X_{i+k-1}); i = 0, 1, 2, \dots\}$ をとると、高次元空間になるほど密度が疎になる。この性質を合同法乱数列の多次元疎結晶構造と呼ばれている。このため k 次元の場合には、たかだか $(k!M)^{1/k}$ 枚の平行で等間隔に並んだ $k-1$ 次元超平面の上ののってしまう。したがって、与えられた乗数 a によって擬似乱数列のでたらめさを判定しようとする考えは Coveyou-MacPherson によって提案され、スペクトル検定 (spectral test) と名付けられた。スペクトル検定では、非零整数ベクトル $\mathbf{s} = (s_1, s_2, \dots, s_k)$ に対して

$$v_k = \min \left\{ \sqrt{s_1^2 + s_2^2 + \dots + s_k^2} \mid s_1 + as_2 + \dots + a^{k-1}s_k = 0 \pmod{M} \right\}, \quad (1.7)$$

の最小値を合同法乱数列のランダムネスの尺度としている。Knuth (1981)は擬似乱数生成法がスペクトル検定に合格するかを決めるための基準について、

$$\log_2 v_k \geq 30/k \quad (2 \leq k \leq 6) \quad (1.8)$$

という条件をあげている。これによると、 $k = 2, 3, 4, 5, 6$ 次元でそれぞれ 15, 10, 7.5, 6.5 ビット程度の精度があればよいことになる。

スペクトル検定の実験結果

ここで選んだ乗数 a は有名なものばかりで、定理 1.1, 1.2 を満たしている乗数である。表 1 のスペクトル検定からわかるように $a = 65539$ (表 1 の番号 1) だけが、3 次元以上で 3.4 という低い値を取っている。ゆえに $a = 65539$ の線形合同法は、3 次元以上である程度の超平面群にのりあまりランダムでないことがわかる。この擬似乱数は実際に以前コンピュータで使用されていたものであり、今回のシミュレーションにも、“他の良いと言われている擬似乱数”と比較するために使用

している。その他の a の値については、 $M=2^{31}-1$ が素数の場合でもそれほど値に差がないことからランダムネス差に影響を与えないものと考えられる。それよりも混合型合同法 ($M=2^{32}$)の方が、乗算型合同法 ($M=2^{30}$)より全体的に高い数値をとっていることより幾分ランダムであることが分かる。またこの検定では単に64ビットのdoubleの変数を使用しても、正しい値が求まるのは2次元までである。したがって、TurboCを使った実験では80ビットのlong doubleを使用した方が良いと考えられる

表1 スペクトル検定の結果

番号	a	M	γ_2	γ_3	γ_4	γ_5	γ_6	$\log_2 \gamma_2$	$\log_2 \gamma_3$	$\log_2 \gamma_4$	$\log_2 \gamma_5$	$\log_2 \gamma_6$
0	基準		32768.0	1024.0	181.0	64.0	32.0	15	10	7.5	6	5
1	65539	2^{31}	23171.9	10.9	10.8	10.8	10.8	14.5	3.4	3.4	3.4	3.4
2	16807	2^{30}	16807.0	655.8	145.8	46.2	16.9	14.0	9.4	7.2	5.5	4.1
3	16807	$2^{31}-1$	16807.0	638.9	147.2	66.6	29.9	14.0	9.3	7.2	6.1	4.9
4	16807	2^{32}	16807.0	1311.5	215.4	55.9	26.1	14.0	10.4	7.8	5.8	4.7
5	69069	2^{30}	16285.0	359.9	98.4	43.6	15.6	14.0	8.5	6.6	5.4	4.0
6	69069	$2^{31}-1$	32570.3	1078.3	163.1	37.6	27.8	15.0	10.1	7.3	5.2	4.8
7	69069	2^{32}	65139.9	1439.6	229.8	83.6	15.6	16.0	10.5	7.8	6.4	4.0
8	1664525	2^{30}	17621.5	642.5	63.1	53.2	21.3	14.1	9.3	6.0	5.7	4.4
9	1664525	$2^{31}-1$	35247.2	1203.1	163.4	31.3	16.1	15.1	10.2	7.4	5.0	4.0
10	1664525	2^{32}	70277.4	1524.0	252.4	64.0	32.2	16.1	10.6	8.0	6.0	5.0
11	39894229	2^{30}	16854.6	816.4	100.3	61.8	19.2	14.0	9.7	6.6	5.9	4.3
12	39894229	$2^{31}-1$	16945.8	1097.5	174.7	61.9	24.9	14.0	10.1	7.4	6.0	4.6
13	39894229	2^{32}	33709.3	1602.5	200.7	73.4	19.2	15.0	10.6	7.6	6.2	4.3
14	48828125	2^{30}	28903.1	451.5	172.9	50.2	30.3	14.8	8.8	7.4	5.7	4.9
15	48828125	$2^{31}-1$	47018.2	1123.5	100.0	63.1	20.6	15.5	10.1	6.6	6.0	4.4
16	48828125	2^{32}	46859.7	825.3	209.6	67.9	35.0	15.5	9.7	7.7	6.1	5.1
17	314159269	2^{30}	29927.9	308.7	142.4	50.6	22.1	14.9	8.3	7.2	5.7	4.5
18	314159269	$2^{31}-1$	37844.9	948.3	192.3	58.5	33.8	15.2	9.9	7.6	5.9	5.1
19	314159269	2^{32}	64615.1	617.4	142.4	56.7	37.8	16.0	9.3	7.2	5.8	5.2
20	397204094	$2^{31}-1$	27705.7	832.4	170.8	69.5	27.6	14.8	9.7	7.4	6.1	4.8
21	1566083941	2^{30}	29514.7	857.2	175.8	48.1	18.3	14.8	9.7	7.5	5.6	4.2
22	1566083941	$2^{31}-1$	26265.9	1299.6	120.4	57.2	27.3	14.7	10.3	6.9	5.8	4.8
23	1566083941	2^{32}	68262.4	1442.1	211.9	68.2	25.7	16.1	10.5	7.7	6.1	4.7
24	1812433253	2^{30}	17424.6	604.7	122.8	51.1	29.3	14.1	9.2	6.9	5.7	4.9
25	1812433253	$2^{31}-1$	24305.4	905.6	166.9	59.4	27.0	14.6	9.8	7.4	5.9	4.8
26	1812433253	2^{32}	65779.4	1209.5	122.8	69.8	30.1	16.0	10.2	6.9	6.1	4.9
27	2100005341	2^{30}	21456.0	744.8	135.2	50.1	28.4	14.4	9.5	7.1	5.6	4.8
28	2100005341	$2^{31}-1$	43487.0	1201.5	205.7	65.2	31.9	15.4	10.2	7.7	6.0	5.0
29	2100005341	2^{32}	50827.6	1342.0	228.9	67.6	34.3	15.6	10.4	7.8	6.1	5.1

M系列乱数

M系列乱数とは、“M系列”を用いて構成された擬似乱数列である。演算には線形合同法と違い排他的論理和を用い、次数が高くても項数が少ない漸化式を用いれば擬似乱数をより速く生成することができる。以下の解説は“伏見 (1989)”に従う。

M系列 (シフトレジスタ系列)

次に理論的性質がよく知られているM系列 (maximum length linearly recurring sequence) について述べる。漸化式

$$a_i = c_1 a_{i-1} + c_2 a_{i-2} + \dots + c_p a_{i-p} \pmod{2} \quad (1.9)$$

を用いて、2つの値 (0と1) の周期列で、これを用いてM系列乱数と呼ばれる擬似乱数列を構成する。初期値 a_0, a_1, \dots, a_{p-1} はすべてが0でなければ任意に選んでよい。また c_1, c_2, \dots, c_p は、ガロア体 GF(2)上の原始多項式

$$f(x) = 1 + c_1 x + c_2 x^2 + \dots + c_p x^p \quad (1.10)$$

の係数である。これによって、M系列の周期は最大の $I = 2^p - 1$ をとる。

M系列の性質

- i) k 次元ベクトル系列 $\langle (a_i, a_{i+1}, \dots, a_{i+k-1}) \rangle$ の1周期分には、零ベクトルを含まないあらゆるベクトルがちょうど1回ずつ現れる。したがって $\langle a_i \rangle$ の1周期中の1の個数は 2^{p-1} 個、0の個数は $2^{p-1} - 1$ 個である。ゆえに、 p が大きければ1の個数と0の個数との比はほぼ1に等しい。
- ii) 1周期中には、長さ l ($1 \leq l \leq p-2$) の1の連および0の連がそれぞれ 2^{p-l-2} 回現れる。

$$\Pr \left\{ \lim_{p \rightarrow \infty} \left(\frac{\text{長さ } l+1 \text{ の連の個数}}{\text{長さ } l \text{ の連の個数}} \right) = 2^{p-l-3} / 2^{p-l-2} = 1/2 \right\} = 1$$

- iii) $\alpha_i = 1$ ($a_i = 0$ のとき)、 $\alpha_i = -1$ ($a_i = 1$ のとき) として得られる系列 $\langle \alpha_i \rangle$ の1周期 $I = 2^p - 1$ にわたる自己相関関数 $R(s)$ は次のようになる。

$$R(s) = \frac{1}{I} \sum_{i=0}^{I-1} \alpha_i \alpha_{i+s} = \begin{cases} 1, & s = 0 \pmod{I} \\ -1/I, & s \neq 0 \pmod{I} \end{cases} \quad (1.11)$$

i)、ii)、iii) よりM系列は、擬似ランダム系列の性質を満たしている。

M系列乱数

M系列乱数は、M系列 $\langle a_i \rangle$ を以下のように b ($b \geq 2$) ビットの2進数に構成したものである。

$$X_i = a_{i+\tau_1} a_{i+\tau_2} \dots a_{i+\tau_b} \quad (2 \text{ 進表現}) \quad (1.12)$$

この位相 $\tau_1, \tau_2, \dots, \tau_b$ の選び方は、擬似乱数列の良さを求めるうえでとても重要である。なぜなら $\tau_1, \tau_2, \dots, \tau_b$ がほぼ等間隔に並んでいるとき、M系列乱数は相関がなく独立である。また、 $\langle X_i \rangle$ の各ビット位置に現れる数列は、同一の漸化式

$$X_i = c_1 X_{i-1} \oplus c_2 X_{i-2} \oplus \cdots \oplus c_p X_{i-p} \quad (1.13)$$

を使うことで高速に生成することができる。1 GFSR 法 (generalized feedback shift register algorithm) によると、原始3項式 $f(x) = 1 + x^q + x^p$ ($q < p$) を使えば漸化式が

$$X_i = X_{i-q} \oplus X_{i-p} \quad (1.14)$$

となり、1回の排他的論理和の演算によりさらに速く1個の擬似乱数ができる。よって、プログラムのほとんどは、この原始3項式を使っている。

定理 1.4 数式 $\langle X_i \rangle$ から作られる k 次元ベクトルの系列 $\langle (X_i, X_{i+1}, \dots, X_{i+k-1}) \rangle$ の1周期分 ($1 \leq i \leq 2^p - 1$) が零ベクトルを $2^{p-kb} - 1$ 回、その他のあらゆる b ビット 2進整数ベクトルを 2^{p-kb} 回とるならば、M系列乱数 $\langle X_i \rangle$ は k 次均等分布する。

この定理により、M系列乱数は k 次元の単位超立方体内で近似的に一樣分布する。また p 次の原始多項式によって生成される、 b ビットのM系列乱数の均等分布の最大次数 K は、

$$K = \lfloor p/b \rfloor \quad (1.15)$$

である。

ところで、 τ をM系列 $\langle a_i \rangle$ の周期 I と互いに素な自然数とし、

$$\tau_j = (j-1)\tau, \quad (2 \leq j \leq b) \quad (1.16)$$

とおく。これを使って、(1.12)を変形して、

$$x_i = a_i a_{i+\tau} a_{i+2\tau} \cdots a_{i+(b-1)\tau} \quad (2 \text{ 進表現})$$

$$y_i = a_{\sigma i} a_{\sigma i+1} a_{\sigma i+2} \cdots a_{\sigma i+b-1} \quad (2 \text{ 進表現})$$

とおく。これらはそれぞれ縦型系列 $\langle x_i(f; \tau) \rangle$ と横型系列 $\langle y_i(f; \sigma) \rangle$ と呼ばれる。このとき、次の定理が成り立つ。この定理を用いると、生成するアルゴリズムの設計が容易になる。

定理 1.5 $\sigma \in C_g, \tau \in C_h^2$ ならば、

$$\begin{aligned} \langle x_i(f_0; \tau) \rangle &\cong \langle y_i(f_h; \tau^{-1}) \rangle, \\ \langle y_i(f_0; \sigma) \rangle &\cong \langle x_i(f_g; \sigma^{-1}) \rangle \end{aligned} \quad (1.17)$$

である。ここで σ^{-1}, τ^{-1} は、それぞれ I を法とする乗算に関する σ, τ の逆元を表す。

プログラム

擬似乱数 D. $f(x) = 1 + x^{32} + x^{521}, \sigma = n = 32$ のM系列乱数

ここでは上の漸化式を使うが、初期値の設定には注意が必要である。そのため、初期値の設定には次の算法より求める。

1 \oplus は、たいていの計算機に備わっている排他的論理和 (exclusive OR) の演算。

2 C は剰余類を表し、 f は C に対応する原始多項式である。

算法

- i) 32ビットの2進乱数を X_0, X_1, \dots, X_{15} を乗算合同法によって発生する擬似乱数を使って任意に与える.
- ii) X_{16} は上位9ビットだけあらかじめ i) と同じように任意に与える. 残りは次の式により付け加える.

$$X_{16} = (R^9 X_0 \oplus X_{15}) + X_{16} \text{ の上位9ビット}$$
- iii) $X_{17}, X_{18}, \dots, X_{520}$ は漸化式 $X_i = M^{32} \left((L^{23} X_{i-17} + R^9 X_{i-16}) \oplus X_{i-1} \right)$ により求める.
- iv) 次に求めた初期値 X_0, X_1, \dots, X_{520} を使って、漸化式 $X_i = X_{i-32} \oplus X_{i-521}$ に当てはめるたびにM系列乱数 $\langle X_i \rangle$ が生成される.

擬似乱数E. $f(x) = 1 + x^{273} + x^{607}$, $b = 23, \sigma = 512$ のM系列乱数

この系列は、上位 b' ビット ($1 \leq b' \leq b$) の精度で見ると $[p/b']$ 次均等分布をするという性質を持っている. そのため、高い次元でも近似的に一様分布する.

算法

- i) 整数の $\langle y_i''(f; 32) \rangle$ の初期値 ($0 \leq i \leq 606$) を設定し、それらの下位23ビットを取り出したものを Y_i'' ($0 \leq i \leq 606$) とする.
- ii) 漸化式 $Y_i'' = Y_{i-273}'' \oplus Y_{i-607}''$ を用いて Y_i'' ($607 \leq i \leq 16 \times 606$) を求める.
- iii) $X_i = Y_{16i}''$ ($0 \leq i \leq 606$) とおく.

1.2 正規擬似乱数列の作り方

標準正規分布 $N(0,1)$ に従う正規擬似乱数 z_i を作る方法を説明する.

ボックス・ミュラー法 (Box-Muller method)

この方法は、区間 $[0,1)$ 上の2つの一様乱数列 u_i, u_{i+1} から、2つの独立正規乱数列 z_i, z_{i+1} を作れることが特徴である.

$$z_i = \sqrt{-2 \log u_i} \cos(2\pi u_{i+1}), \quad z_{i+1} = \sqrt{-2 \log u_i} \sin(2\pi u_{i+1}) \quad (1.18)$$

さらに、次の Marsaglia の算法を使えば、三角関数の計算をしないので実行時間が短くなる.

マルサリア法 (Marsaglia method)

$$\cos \theta = v_i / \sqrt{R}, \quad \sin \theta = v_{i+1} / \sqrt{R} \quad (1.19)$$

これは、単位正方形の内側の一様擬似乱数 u_i, u_{i+1} を選ぶ代わりに、原点を中心とする単位円の内側で一様に分布している点 (v_i, v_{i+1}) を選ぶ. そうすれば、 $R = v_i^2 + v_{i+1}^2$ は一様に分布し、また点 (v_i, v_{i+1}) の偏角 θ は $(0, 2\pi)$ 上でランダムな角度をとる. したがって、三角関数の呼び出しが必要なくなる.

プログラム

算法

- i) 一様擬似乱数列 u_i, u_{i+1} を生成し、 $v_i \leftarrow 2u_i - 1, v_{i+1} \leftarrow 2u_{i+1} - 1$ とする.

- ii) $R \leftarrow v_i^2 + v_{i+1}^2$ を求める。
 iii) $R \geq 1$ ならば i) にもどる。
 iv) 正規擬似乱数列を求める。 $z_i = v_i \sqrt{-2 \log R/R}$, $z_{i+1} = v_{i+1} \sqrt{-2 \log R/R}$

2. 一様擬似乱数列の検定法

実際使う擬似乱数列は一周期のうちのごく一部分であることが多いから、これらについては種々の性質に関する理論的保証があれば大変に好ましい。しかしながら、擬似乱数列の局所的な性質を理論的に調べることは、一周期全体にわたる性質を調べるのに比べれば、はるかに難しい。

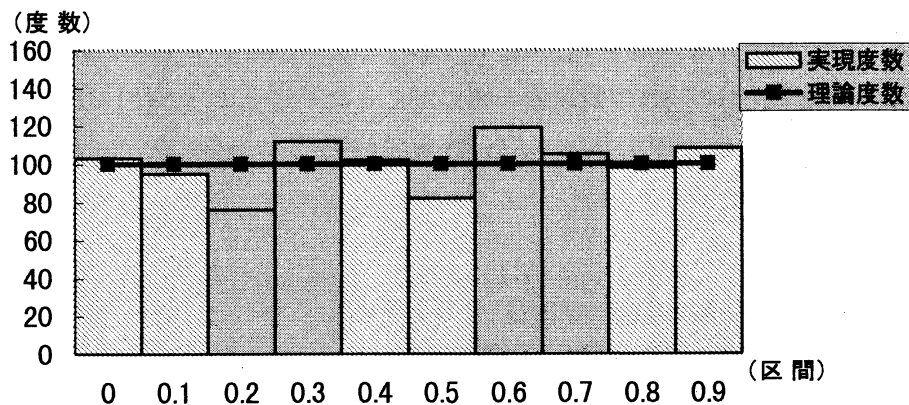
そこで、擬似乱数列の局所的な性質を調べるためには、実際に数列を発生して、統計的な手法を使ってそれを解析するという手段に頼るのが普通である。

また、一般的には、一様擬似乱数に対する検定に比べて、変換された数列（正規擬似乱数）に対する検定が行われるのはずっと少ない。ここでは、一様擬似乱数に対する検定法として、きわめて多数提案されている中から、有名な方法を選んで解説する。なお u_i は区間 $[0,1)$ 上の一様擬似乱数を表わし、 U_i は u_i の 10 進表現での小数第 1 位の数値を表わしている。

一次元度数検定（一様分布検定）

生成された n 個の擬似乱数列 u_0, u_1, \dots, u_n が一様擬似乱数列であるためには、その度数分布が等確率性の性質をもっているかどうかを判定すればよい。そのためには、擬似乱数の領域を l 個（クラスの個数）の部分区間 $[a_0, a_1), [a_1, a_2), \dots, [a_{l-1}, a_l)$ に分割し、各クラスに入っている個数 f_1, f_2, \dots, f_l を数える。この個数をもとに、度数の変化を目で見えてはっきりさせるためには、図 1 のようなヒストグラムであらわす方法がある。

図 1 混合型合同法Aの擬似乱数 1000 個ヒストグラム



このヒストグラムから等確率性がほぼ推測できる。しかし、実際はなかなか目で見るだけでは判断できない場合が多いので、カイ 2 乗検定あるいは Kolmogorov-Smirnov 検定（以下 K-S 検定と略す）を使って等確率性を調べる。

カイ 2 乗検定

カイ 2 乗検定では、先ほどの個数 f_1, f_2, \dots, f_l を実現度数と呼び、このときの帰無仮説を

H_0 : 生成された擬似乱数の頻度分布が、1つの確率分布(分布関数 $F(U)$ で示す)に適合するとする。この帰無仮説 H_0 のもとで理論度数は、

$$F_i = n \times (F(a_i) - F(a_{i-1})) \quad (i = 1, 2, \dots, l) \quad (2.1)$$

である。これらの理論度数と実現度数を使って、各クラスごとに計算したものが、次のカイ2乗統計量である。

$$\chi_0^2 = \sum_{i=1}^l \frac{(f_i - F_i)^2}{F_i} \quad (2.2)$$

これは、 f_i と F_i との差が大きければ χ_0^2 の実現値も大きくなる。ゆえに、 χ_0^2 の実現値の大きさの程度によって、帰無仮説 H_0 を受け入れるかを決めることが妥当である。このためには F_i がある程度(目安として $F_i \geq 100$)大きいとき、 χ_{l-1}^2 が自由度 $\gamma = (l-1)$ のカイ2乗分布に従うことを利用する。

Kolmogorov-Smirnov 検定

K-S 検定は、カイ2乗検定が有限個のクラスの1つである場合に適用できたのに対して、連続分布のように確率的な変数が無限に多くの値をとる場合に用いられる。ここで n 個のデータ x_1, x_2, \dots, x_n の経験分布関数(empirical distribution) $F_n(x)$ とする。

$$F_n(x) = \frac{1}{n} \{x_i \leq x \text{ を満たすデータの個数}\} \quad (2.3)$$

K-S 検定では $F_n(x)$ と $F(x)$ の差を次の統計量を用いて評価する。

$$K_n^+ = \sqrt{n} \max(F_n(x) - F(x)), \quad K_n^- = \sqrt{n} \max(F(x) - F_n(x)) \quad (2.4)$$

K_n^+ は $F(x)$ より大きい $F_n(x)$ の最大偏倚値、 K_n^- は $F(x)$ よりも小さい $F_n(x)$ の最大偏倚値である。ところで経験分布関数 $F_n(x)$ は $1/n$ の整数倍の値しかとらないことから $F(b) - F(a) = 1/n$ を満たす任意の区間 $[a, b]$ に属する x_i の中で $\{i/n - F(x_i)\}$ 及び $\{F(x_i) - i/n\}$ を最大にするものを探せば良いことが分かる。

算法

- i) 初期値の設定 $i = 0, k = 0, D^- = 0, D^+ = 0$ とする。
- ii) $k \leftarrow k + 1$ もし $k = n$ ならば vii) に進む。
- iii) $n_k = 0$ ならば ii) に戻る。(n_k は k 番目の区間の度数)
- iv) $D^- \leftarrow \max(D^-, (F(x_k) \text{の最小値} - i/n))$
- v) $i \leftarrow i + n_k$
- vi) $D^+ \leftarrow \max(D^+, (i/n - F(x_k) \text{の最大値}))$ の後、ii) に戻る。
- vii) $K_n^+ \leftarrow \sqrt{n}D^+, K_n^- \leftarrow \sqrt{n}D^-$ とする。

2次元度数検定 (系列検定 (serial test))

区間 $(0,1]$ ではなくて1辺の長さが1である正方形の中で点列 $(u_{2i}, u_{2i+1}), 0 \leq i < n$ が等確率で現れるかどうかを調べる検定である。よってカイ2乗分布を用いた1次元度数検定の2次元への拡張である。そのためには、正方形を $d \times d$ 個の等しい面積の小正方形に区切り、各メッシュに入った点の個数 $n_i (1 \leq i \leq d^2)$ を数える。カイ2乗検定をこれら $l = d^2$ 個の種類について等確率 $1/d^2$ として行う。また、この時の自由度は $d^2 - 1$ である。 d は少なくとも $n > 5d^2$ にすべきである。一般に、生成された擬似乱数列の区切り方を3個ずつ、4個ずつと増すことによって、3次元以上についても同様な検定が行える。

ポーカー検定 (poker test) (分割検定 (partition test))

ポーカー検定は、相続く5つの整数の組 $(U_{5i}, U_{5i+1}, \dots, U_{5i+4}), 0 \leq i < n$ に区切り、各組を次の7つの型に分類し、各種類の5個組みの数についてカイ2乗検定を行う。

- a b c d e (すべて異種)
- a a b c d (対1個)
- a a b b c (対2個)
- a a a b c (3個同種)
- a a a b b (3個同種と対)
- a a a a b (4個同種)
- a a a a a (5個同種)

このクラス分けを少し簡単にしたのが、J.C. Butcher が提案した分割検定である。これは、5個組の中の異なる種類の数について分類し検定する。

- 5種類 (すべて異種)
- 4種類 (対1個)
- 3種類 (対2個、3個同種)
- 2種類 (3個同種と対、4個同種)
- 1種類 (5個同種)

これはプログラムを作る上で組織的に計算することができ便利である。さらに検定の性質は、ポーカー検定とほとんど変わらないのである。一般には k 個のあい続く整数 $((0,1, \dots, d-1)$ のどれかの値)の組を n 個観測し、 r 種類の数からなる k 個組の数を数える。カイ2乗検定には r 種類の数を含む確率

$$P_r = \frac{d(d-1)(d-2) \cdots (d-r+1)}{d^k} \left\{ \begin{matrix} k \\ r \end{matrix} \right\} \quad (2.5)$$

を用いる。1たとえば、5種類の数からなる5個組みの確率 P_5 と、5個組みの数 n が1000個ある場合の理論度数 F_r について行う。具体的な数値は次のとおりである。ところで $r=1,2$ に対する P_r, F_r の値は小さいので、この二つのクラスはまとめてある。したがって自由度は3でカイ2乗検定を行う。

¹ {} はスターリング数である。

表2 ポーカー検定

r	1,2	3	4	5
P_r	0.0136	0.1800	0.5040	0.3024
F_r	13.6	180	504	302.4

間隔検定 (gap test)

この検定は、 u_i がある範囲 $[\alpha, \beta)$ $0 \leq \alpha < \beta < 1$ に入るまでの間隔 (gap) を調べて、間隔の長さ l になる個数についてカイ 2 乗検定を行うものである。このとき $\alpha \leq u_i < \beta$ となる確率は $P = \beta - \alpha$ であるから、間隔の長さ l が起こる確率は

$$\Pr\{g = l\} = P(1-P)^l \quad (l = 0, 1, 2, \dots) \quad (2.6)$$

である。

算法

- i) 初期値の設定 $count[l] \leftarrow 0 (0 \leq l < 30), d \leftarrow U_0, i = -1, s = 0$ とする。 s は間隔の総数である。
- ii) l の初期値の設定 $l \leftarrow 0$
- iii) $i \leftarrow i + 1, 0.1d \leq u_i < 0.1d + 0.1$ または $i = 4999$ ならば v) へ。
- iv) $l \leftarrow l + 1$ の後 iii) にもどる。
- v) $s \leftarrow s + 1$ とした後、 $l \geq 30$ ならば $count[30] \leftarrow count[30] + 1$ そうでなければ $count[l] \leftarrow count[l] + 1$ とおく。
- vi) $i < 4999$ ならば ii) にもどる。
- vii) 下の表 3 を参考に自由度 14 でカイ 2 乗検定を行う。

表3 間隔検定

l	0	1	2	3	4	5	6	7
Pr	0.1	0.09	0.081	0.0729	0.06561	0.059049	0.0531441	0.04782969

8	9	10~14	15~19	20~24	25~29	30以上
0.0430461	0.0387420	0.1427873	0.0843145	0.0497869	0.0293986	0.0423912

連の検定 (run test)

この検定は、擬似乱数列の数字の並び方の無規則性の検証を行う検定法の 1 つである。ある数列

$$| 2 \ 5 \ 8 \ | \ 3 \ 6 \ | \ 1 \ | \ 7 \ 9 \ | \quad (2.7)$$

において、隣り合った数が上昇している部分ごとに区切られている場合を“上昇の連 (runs up)”と呼ぶ。また下降している部分ごとに区切られている場合を“下降の連 (runs down)”と呼ぶ。区切られた個数を連の長さといい、(2.7)の数列は左から長さ 3 の連、2 の連、1 の連、最後に 2 の連がある。検定ではこれらの連の長さを調べることによって検定を行う。擬似乱数の個数を例えば n

個 (n は少なくとも 1 万以上) とすると、このとき長さ l の連の総個数を r_l とするとその期待値 $E(r_l)$ は

$$E(r_l) = \frac{(l^2 + l - 1)n}{(l + 2)} - \frac{l^3 - 4l - 1}{(l + 2)}, \quad 1 \leq l \leq n - 1, \quad E(r_n) = \frac{1}{n!} \quad (2.8)$$

で示されることが知られている (伏見(1989)).

表4 連の検定

実現度数	r_1	r_2	r_3	r_4	r_5	r_6
理論度数	$\frac{1}{6}n + \frac{2}{3}$	$\frac{5}{24}n + \frac{1}{24}$	$\frac{11}{120}n + \frac{7}{60}$	$\frac{19}{720}n + \frac{47}{720}$	$\frac{29}{5040}n + \frac{13}{630}$	$\frac{1}{840}n + \frac{29}{5040}$

しかし、このままでは独立性が満たされないためカイ 2 乗検定を行うことができない。ゆえに、次のようにしてカイ 2 乗統計量を求める。

$$\chi_0^2 = \frac{1}{n} \sum_{j=1}^6 \sum_{i=1}^6 a_{ij} (R_i - E(R_i))(R_j - E(R_j))$$

$$a_{ij} = \begin{bmatrix} 4529.354 & & & & & & \\ 9044.902 & 18097.03 & & & & & \\ 13567.95 & 27139.46 & 40721.33 & & & & \\ 18091.27 & 36186.65 & 54281.27 & 72413.61 & & & \\ 22614.71 & 45233.82 & 67852.04 & 90470.08 & 113261.8 & & \\ 27892.16 & 55788.83 & 83684.57 & 111580.1 & 139475.6 & 172860.2 & \end{bmatrix} \quad (2.9)$$

ここで注意すべきは、このカイ 2 乗分布の自由度が 5 ではなく 6 に従うことを利用して検定する。

順列検定 (permutation test)

この検定は、数列を k 個組 $(u_{ki+1}, u_{ki+2}, \dots, u_{ki+k}), 0 \leq i < n$ に分割する。各組の要素の大きさの順番は $k!$ 通りある。例えば、 $k = 4$ とすると

$u_{4i+1} < u_{4i+2} < u_{4i+3} < u_{4i+4}, u_{4i+1} < u_{4i+2} < u_{4i+4} < u_{4i+3}, \dots, u_{4i+4} < u_{4i+3} < u_{4i+2} < u_{4i+1}$ の 24 通りが作れる。ゆえに、24 種類数えて自由度 23 でカイ 2 乗検定を行う。このとき、順列の確率はすべて $1/k!$ である。

d^2 -検定 (d^2 -test)

2次元平面上の 2 点 $(u_{4i}, u_{4i+1}), (u_{4i+2}, u_{4i+3}), 0 \leq i < n$ をとり、2 点間の距離 d の平方 d^2 が次のような確率分布をもつことが知られている (宮武 (1878)).

$$\Pr\{d^2 \leq \alpha^2\} = \begin{cases} \pi\alpha^2 - \frac{8}{3}\alpha^3 + \frac{1}{2}\alpha^4 & (\alpha^2 \leq 1) \\ \frac{1}{3} + (\pi - 2)\alpha^2 + 4(\alpha^2 - 1)^{1/2} + \frac{8}{3}(\alpha^2 - 1)^{2/3} - \frac{\alpha^4}{2} - 4\alpha^2 \sec^{-1} \alpha & (1 < \alpha^2 \leq 2) \end{cases} \quad (2.10)$$

実際の検定では、この値を計算して α^2 の値を 11 区間に分けてカイ 2 乗検定を行っている。

表5 d^2 -検定

α^2	[0,0.1]	(0.1,0.2]	(0.2,0.3]	(0.3,0.4]	(0.4,0.5]	(0.5,0.6]
Pr	0.234832	0.174973	0.139495	0.112718	0.090969	0.072614

(0.6,0.7]	(0.7,0.8]	(0.8,0.9]	(0.9,1.0]	(1.0,2.0]
0.056748	0.042814	0.030430	0.019333	0.025074

組み合わせ検定 (combination test)

k 個組 $(u_{ki+1}, u_{ki+2}, \dots, u_{ki+k}), 0 \leq i < n$ のそれぞれの要素が $0 \leq u_i < 0.5$ に含まれる個数が l 個現われる確率は

$$\Pr\{X = l\} = {}_k C_l \left(\frac{1}{2}\right)^k, \quad (l = 0, 1, 2, \dots, k) \quad (2.11)$$

となる.

算法

- i) 初期値の設定 $k = 0$ とする.
- ii) $l \leftarrow 0, k \leftarrow k + 1$
- iii) $0 \leq u_i < 0.5$ ならば $l \leftarrow l + 1$ を 10 回繰り返す.
- iv) その後、 $l > 0$ ならば $l \leftarrow l - 1$
- v) $l = 9$ ならば $l = 8$ にする.
- vi) $count[l] \leftarrow count[l] + 1$
- vii) $k < 1000$ ならば ii) に戻る.
- viii) 下の表を参考に自由度 8 でカイ 2 乗検定を行う.

表6 組み合わせ検定

l	0, 1	2	3	4	5	6	7	8	9, 10
Pr	0.011	0.044	0.117	0.205	0.246	0.205	0.117	0.044	0.011

3. 確率微分方程式の解のシミュレーション

1 章で示されたいくつかの生成法による擬似乱数列を用いて確率微分方程式 (SDE) の近似解を構成しその精度を調べる. これらの擬似乱数列の中で統計的検定によって棄却されたものと近似解の精度が良くなかったと判断された擬似乱数列の数を比較する.

確率微分方程式の近似解

Maruyama (1955) は SDE

$$\begin{cases} dX(t) = \mu(t, X(t))dt + \sigma(t, X(t))dB(t), & 0 \leq t \leq T \\ X(0) = X_0, \end{cases} \quad (3.1)$$

の唯一の解の存在を次式によって定義された Euler 型近似解 $\{Z_n(t), 0 \leq t \leq 1\}$ によって示した.

$$Z_n(t) = X_0 + \int_0^t \mu_n(u)du + \int_0^t \sigma_n(u)dB(u), \quad 0 \leq t \leq 1, \quad (3.2)$$

ただし、

$$\begin{aligned} \mu_n(u) &= \mu\left(\frac{k}{n}, x_k\right), \quad \frac{k}{n} \leq t < \frac{k+1}{n}, \quad k = 0, 1, \dots, n-1, \\ \sigma_n(u) &= \sigma\left(\frac{k}{n}, x_k\right), \quad \frac{k}{n} \leq t < \frac{k+1}{n}, \quad k = 0, 1, \dots, n-1, \end{aligned}$$

$$\begin{cases} x_0 = X_0, \\ x_k = X_0 + \sum_{j=1}^k \mu\left(\frac{j-1}{n}, x_{j-1}\right) / n + \sum_{j=1}^k \sigma\left(\frac{j-1}{n}, x_{j-1}\right) \eta_j, \quad k = 1, 2, \dots, n, \end{cases}$$

$$\eta_k = B\left(\frac{k}{n}\right) - B\left(\frac{k-1}{n}\right), \quad k = 1, 2, \dots, n.$$

X と Z_n の誤差の評価について Gihman-Skorohod (1979)、Shimizu (1984)、Kanagawa (1988) では、真の解と近似解の $Z_n(t)$ との差の P 次モーメントについて次の評価を得た.

定理 3.1 任意な $0 \leq s, t \leq 1$, $x, y \in \mathbf{R}$ について

$$|\mu(t, x) - \mu(s, y)|^2 + |\sigma(t, x) - \sigma(s, x)|^2 \leq L_1(|x - y|^2 + |t - s|^2), \quad (3.3)$$

$$|\mu(t, x)|^2 + |\sigma(s, x)|^2 \leq L_2, \quad (3.4)$$

ただし L_1 と L_2 は、 s, t, x, y の無関係なある正の定数である. そのとき任意の $p \geq 2$ について

$$E\left(\max_{0 \leq t \leq 1} |X(t) - Z_n(t)|^p\right) = O\left(n^{-\frac{p}{2}}\right), \quad (n \rightarrow \infty). \quad (3.5)$$

Faure (1990) は有界性の条件(3.4)をゆるめて上の定理を拡張した. Kohatsu-Higa (1996) は境界条件をもつ SDE の近似解について調べた. Ogawa (1992) はある種の SDE であらわされる非線形の拡散過程のオイラー丸山の近似解の誤差を評価した. また, Mackevicius (1994) は軌道空間上の汎関数 h のある広いクラスに対して、 $E[h(Z_n)] - E[h(X)]$ を調べることによって弱収束の速さを与えた. さらに、他のタイプの近似解やそれらの数値解析については Kloeden-Platen (1992) と Kloeden-Platen-Schurz (1994) を参照されたい.

厳密にはコンピュータ上でブラウン運動のような連続過程をそのまま取り扱うことはできないので、実際には $Z_n(t)$ を離散化した近似解 $X_n = \{X_n(t), 0 \leq t < 1\}$ を用いる。

$$\begin{cases} X_n(t) = x_k, & k/n \leq t < (k+1)/n, \quad k = 0, 1, \dots, n-1, \\ X_n(1) = x_n. \end{cases} \quad (3.6)$$

離散化された近似解と真の解との誤差について次の評価が知られている (Kanagawa (1988)).

定理 3.2 定理 3.1 の条件の下で、任意の $p \geq 2$ および $\varepsilon > p/2$ に対して

$$E\left(\max_{0 \leq t \leq 1} |X(t) - X_n(t)|^p\right) = O\left(n^{-\frac{p}{2}} (\log n)^\varepsilon\right), \quad (n \rightarrow \infty). \quad (3.7)$$

さらに詳しい評価については [10] ~ [13], [15], [19], [21] を参照されたい。

Euler-Maruyama 型近似解によるシミュレーション

これから次の線形な確率微分方程式について、数値実験を行う。

$$\begin{cases} dX(t) = X(t)dB(t), & 0 \leq t \leq 1 \\ X(0) = 0.1 \end{cases} \quad (3.8)$$

ただし伊藤の公式を使えば、この SDE の真の解は

$$X(t) = 0.1 \exp\left\{B(t) - \frac{t}{2}\right\} \quad (3.9)$$

と与えられる。この SDE の Euler-Maruyama 型近似解 $X_N = \{X_N(t), 0 \leq t \leq 1\}$ は次のように与えられる。ステップ数 $N = 1600$ 、サンプル数 1000、初期値 $x_0 = 0.1$ とする。

$$\begin{cases} X_{1600}(t) = x_k, & k/1600 \leq t < (k+1)/1600 \\ X_{1600}(1) = x_{1600}, \end{cases} \quad (3.10)$$

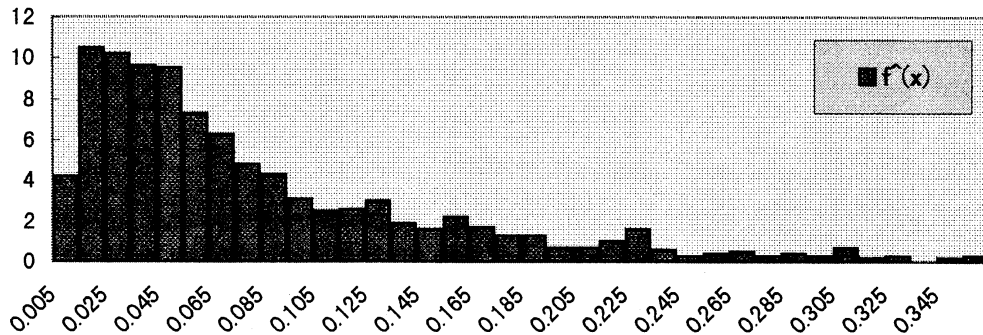
$$x_k = x_0 + \sum_{i=1}^k x_{i-1} Z_i / 40,$$

ここで、 $\{Z_k, k \geq 1\}$ は独立で同じ分布を持ち $E\{Z_1\} = 0, E\{Z_1^2\} = 1$ である。

実験 1

まず、それぞれの擬似乱数の種が、近似解の密度分布にどのような影響を与えるのか調べる。近似解の精度を調べるために $X_n(1)$ のヒストグラム $\bar{f}(x)$ を用いた。

図2 ヒストグラム $\bar{f}(x)$ の例 (擬似乱数A, 乱数種 33)



ヒストグラム $\bar{f}(x)$ は、Euler-Maruyama 型近似解のサンプルパスを 1 つの種から 1000 本の擬似乱数列を発生させ、それらのサンプルパスの $t=1$ での値が各区間 $[0,0.01), [0.01,0.02), \dots, [0.99,1)$ に入る度数を表したものである。(サンプルパスを 1000 本発生させるには、擬似乱数が 1600000 個必要である。) またこの実験には、1~999 までの奇数を種として次の A から E までのそれぞれの乱数生成法によって構成された計 2500 の擬似乱数列が用いられる。

- A) $X_i = 1664525X_{i-1} + 1013904223$ の混合型合同法
- B) $X_i = 16807X_{i-1}$ の乗算型合同法
- C) 3次元以上でランダムでない $X_i = 65539X_{i-1}$ の乗算型合同法
- D) $f(x) = 1 + x^{32} + x^{521}$, $\sigma = n = 32$ の M 系列乱数
- E) $f(x) = 1 + x^{273} + x^{607}$, $b = 23, \sigma = 512$ の M 系列乱数

$X(1)$ の密度関数 $f(x)$ は次式で与えられる。

$$f(x) = \frac{1}{\sqrt{2\pi x}} \exp\left[-\frac{1}{2}\{0.5 + \log(10x)\}^2\right], 0 \leq x < \infty \quad (3.11)$$

近似解のヒストグラム $\bar{f}(x)$ と密度関数 $f(x)$ の差を、次の 4 種類の方法で検定した。

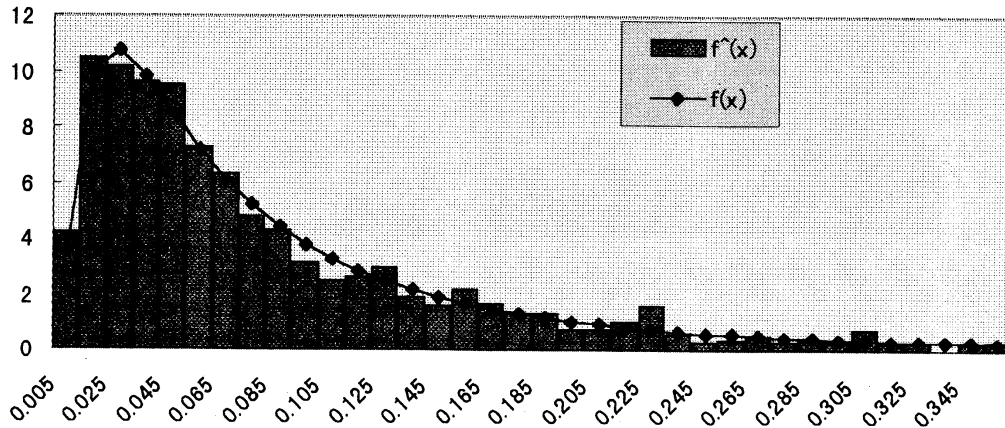
- i) 階級値における $\bar{f}(x)$ と $f(x)$ の 2 乗誤差の和
- ii) 階級値における $\bar{f}(x)$ と $f(x)$ の絶対誤差の最大値
- iii) 階級幅における $\bar{f}(x)$ と $f(x)$ の面積の 2 乗誤差の和
- iv) 階級幅における $\bar{f}(x)$ と $f(x)$ の面積の絶対誤差の最大値

具体的には以下のとおりである。

i) は図 3 のように階級値 0.005, 0.015, 0.025, \dots , 0.995 での $\bar{f}(x)$ と $f(x)$ の差を、それぞれ 2 乗してからすべて足した値を求めた。したがって、求めた数値が大きいくほど $X(1)$ の密度関数 $f(x)$ から離れることがわかる。

ii) は i) と同様に階級値 0.005, 0.015, 0.025, \dots , 0.995 での $\bar{f}(x)$ と $f(x)$ の差を調べ、その中で最も大きな数値を求めた。例えば図 4 で求めると、階級値 0.225 の差が最大値 0.85 である。

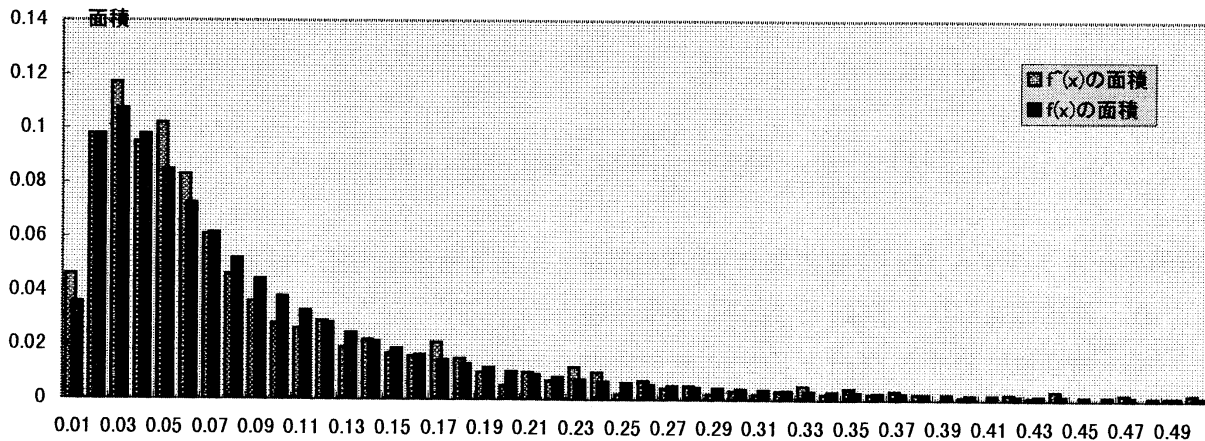
図3 階級値における $\bar{f}(x)$ と $f(x)$ の比較 (擬似乱数A、乱数種33)



iii) は、各区間 $[0,0.01), [0.01,0.02), \dots, [0.49,0.5)$ における $\bar{f}(x)$ と $f(x)$ の面積を求め、それぞれの差を2乗してからすべて足した数値を求めた。

iv) は、各区間 $[0,0.01), [0.01,0.02), \dots, [0.49,0.5)$ における $\bar{f}(x)$ と $f(x)$ の面積の差を調べ、その中で最も大きな数値を求めた。例えば図5で求めると、 $[0.09,0.10)$ のときが最大値0.01である。

図4 階級幅における $\bar{f}(x)$ と $f(x)$ の面積の比較 (擬似乱数A、乱数種27)

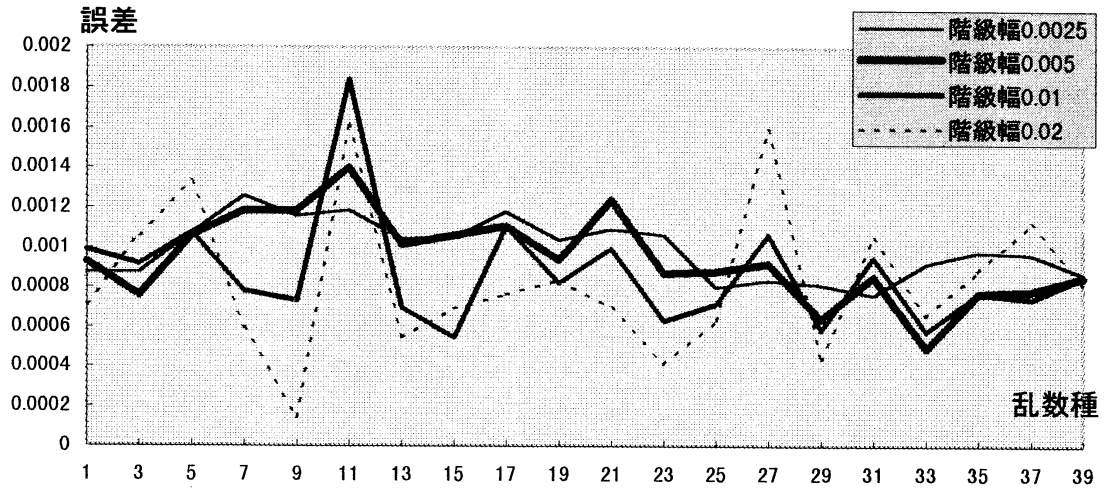


さらに、それぞれの階級幅を0.0025、0.005、0.01、0.02と変えて実験した。

実験1の結果

図5は、擬似乱数の生成法Aを使用した実験iii)にあたる面積の2乗誤差の和を、乱数種を変えて表したものである。横軸に乱数種を1~39まで表示してあるが、実際の実験で乱数種は1~999までの奇数の値を使用した。縦軸には、 $\bar{f}(x)$ と $f(x)$ の誤差を表わした。

図5 面積の2乗誤差



4本の折れ線は、ヒストグラムの階級幅を細かくした場合と広くした場合4種類を示している。例えば、このグラフから乱数Aの乱数種11を用いてシミュレーションを行うと、誤差が非常に大きいことを表わしている。

次の図6、8は、この実験において誤差が非常に大きかった乱数種が11の場合と、誤差が小さかった乱数種が9の場合の密度関数を表わしている。

図6 乱数種が9の場合の $\bar{f}(x)$ と $f(x)$ (擬似乱数A、乱数種9)

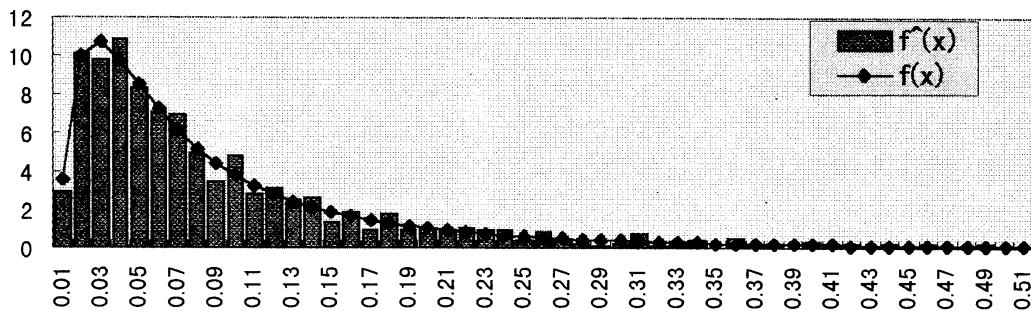
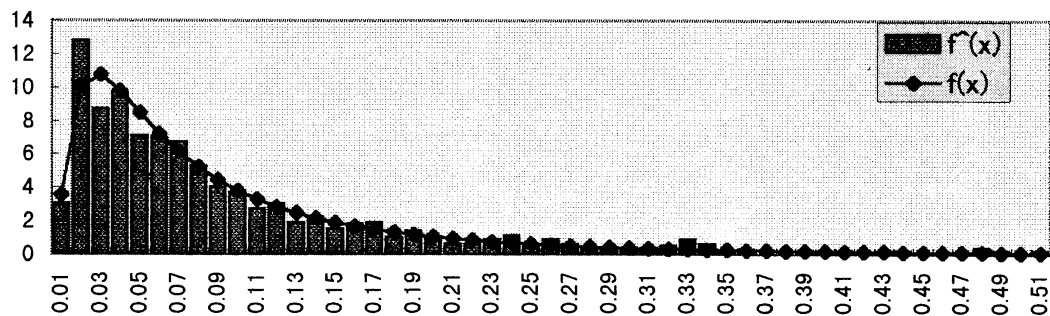


図7 乱数種が11の場合の $\bar{f}(x)$ と $f(x)$ (擬似乱数A、乱数種11)



考察

図5の4本の折れ線のグラフの変動から、乱数種による近似の良し悪しについては階級幅には無関係であると推測される。したがってそれぞれの階級幅について相関係数を求め、ある程度の相関があることを確認した。ゆえに、どの階級幅を使用しても、ほとんど同じ結果を得ることがわかった。このため、階級幅0.01における $\bar{f}(x)$ と $f(x)$ の面積の差の2乗誤差の和を使用して調べた。今回使用したA~Eまでの5種類の乱数生成法については、それぞれ乱数の種500個における誤差の平均をとった場合、大きな開きは見られなかった。したがって、実験した5種類の乱数生成法とシミュレーションの結果には、はっきりした関係を見出すことはできなかった。

擬似乱数列の統計的検定

統計的検定にはいろいろな種類があるが、どの検定を通った擬似乱数が、実験1で求めた近似解に適しているのか調べている。このため、まず次の8種類の統計的検定を使って、乱数の良さを判断している。

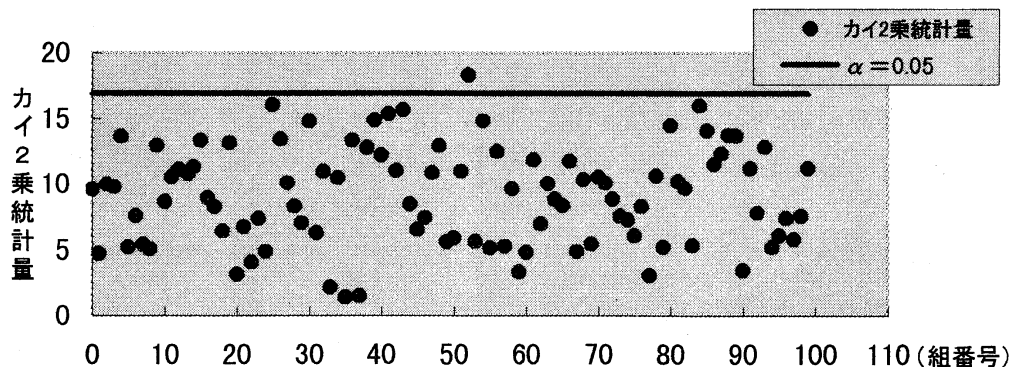
- 1) 一次元度数検定 (一様分布検定)
- 2) 二次元度数検定 (系列検定)
- 3) ポーカー検定
- 4) 間隔検定
- 5) 連の検定
- 6) 順列検定
- 7) d^2 -検定
- 8) 組み合わせ検定

実験2 (カイ2乗検定を用いた場合)

A~Eまでの5種類の乱数生成法を用いて、500個の乱数の種からそれぞれ1600000個の一様擬似乱数について、8種類の検定を行った。例えば、1次元度数検定の場合は、はじめから1000個ずつ組にし、組の番号を1番から1600番までつけて組ごとに順次カイ2乗検定を行った。カイ2乗統計量を求めるのには、 $[0,1)$ 上を10個の等しい区間に分割し式(1.3.2)を使った。また、自由度 $\gamma=9$ とし有意水準 $\alpha=0.05, 0.1, 0.2$ の3つの場合で行った。

図8は、1番から1600番までの最初の1部分をとってきた図である。この図のように有意水準5%を超えている点が100組のうち5組以内であると、一様擬似乱数であるとみなすことができる。

図8 一次元度数検定 (混合型合同法による擬似乱数10万個)



実験3 (Kolmogorov-Smirnov)

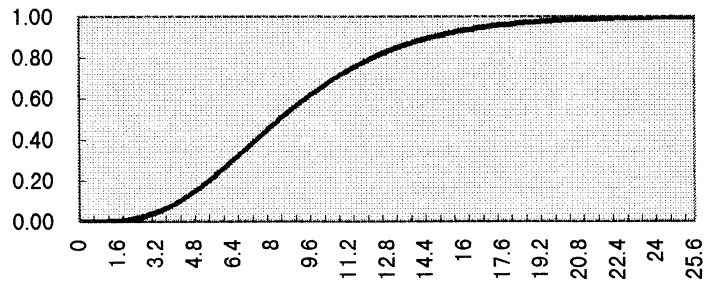
実験2で求めたカイ2乗統計量を、さらにカイ2乗の理論分布に近いかどうかをK-S検定によって判定する。このとき、自由度 γ のカイ2乗分布の分布関数の近似式(Wilson-Hifertyの近似式)を用いるとコンピュータでの計算速度が速くなる(伏見(1989))。

$$z_\gamma = \sqrt{\frac{9\gamma}{2}} \left\{ \left(\frac{\chi^2}{\gamma} \right)^{1/3} - 1 + \frac{2}{9\gamma} \right\} \quad (3.12)$$

また、正規近似はHastingsと戸田秀雄氏らによる次の近似式を使っている。図8が実際にコンピュータにより求めたものだが、ほとんど正確にカイ2乗分布の分布関数を表わすことができる。

$$\begin{aligned} \phi(z) &= 1 - \frac{1}{2} \left(1 + d_1 z + d_2 z^2 + d_3 z^3 + d_4 z^4 + d_5 z^5 + d_6 z^6 \right)^{-16}, z \geq 0 \\ d_1 &= 0.0498673470, d_4 = 0.0000380036, d_2 = 0.0211410061 \\ d_5 &= 0.0000488906, d_3 = 0.0032776263, d_6 = 0.0000053830 \end{aligned} \quad (3.13)$$

図9 近似式によるカイ2乗分布関数(自由度9)



実験2・3の結果

表7, 8は、5種類の乱数生成法の検定結果である。これは、1つの乱数生成法に対して乱数の種500個を用いて実験した。さらに、その500個を使って、カイ2乗検定またはK-S検定の棄却域に入る組の数を平均化したものである。したがって理想の数値は、それぞれの表の上を示した数値である。したがって、この値よりかなり離れているものにはマークをつけた。

理想の数値は、次のように求めた。

$$\text{理想の数値} = \text{有意水準} (\alpha) \times \text{組数} \quad (3.14)$$

例えば表7の1次元度数検定の場合、有意水準が0.05で乱数1600000個を1600組に分けたので理想の数値は80である。

表7 乱数の精度の検定(カイ2乗分布の場合)

$\alpha=0.05$	80	40	16	16	8	8	20	20	8
検定	1次元	2次元	ホーカ	間隔検定	連の上	連の下	順列	D2	組み合わせ
乱数A	81.24	40.69	16.11	16.11	13.21	13.25	24.2	20	7.906
乱数B	80.23	40.22	16.01	15.852	13.43	13.12	24.2	20	7.802
乱数C	79.98	39.34	16.02	16.292	17.36	17.41	23.6	20	7.598
乱数D	81.32	40.64	16.26	16.2	13.2	13.57	24	20	7.932
乱数E	80.32	39.98	16.08	16.14	13.31	13.47	23.8	20	7.936

表8 乱数の精度の検定 (K-S 検定の場合)

$\alpha=0.05$	20	20	8	8	8	8	10	10	8
検定	1次元	2次元	ホーカ	間隔検定	連の上	連の下	順列	D2	組み合わせ
乱数A	20.28	19.7	8.2	8.114	8.354	8.758	10.7	10	7.942
乱数B	19.63	20.11	8.004	7.976	8.49	8.644	10.4	10	7.712
乱数C	19.86	21.24	8.092	8.112	9.518	9.602	10.5	9.8	8.168
乱数D	20.45	20.51	7.986	7.96	8.798	8.834	10.5	10	8.116
乱数E	19.98	20.08	8.272	8.05	8.71	8.738	10.6	9.9	8.256

実験1, 2, 3の結果

表9については、実験1のシミュレーションで誤差の大きかった乱数の種と、統計的検定により良くないと考えられる乱数の種を比較し、重なる乱数の種の数を表示したものである。詳しく述べてみると、1つの乱数生成法について500個の乱数の種に注目している。実験1で近似解の誤差が大きい乱数の種50個とそれぞれの統計的検定により良くない（棄却域に入る組の数が多い）乱数の種50個を比較し、重なる乱数の種の数を表示したものである。

また図10は、表9をグラフ化したものである。これより間隔検定が他の検定に比べて高いことがはっきりする。

表9 擬似乱数列の統計的検定と近似解の関係 (K-S 検定の場合)

検定	1次元	2次元	ホーカ	間隔検定	連の上	連の下	順列	D2	組み合わせ
乱数A	5	10	5	15	4	8	6	6	6
乱数B	5	6	9	8	10	4	8	7	5
乱数C	5	8	7	9	3	7	8	10	8
乱数D	10	6	8	10	9	9	7	3	5
乱数E	4	7	2	15	8	6	7	9	8

図10 表9のグラフ化

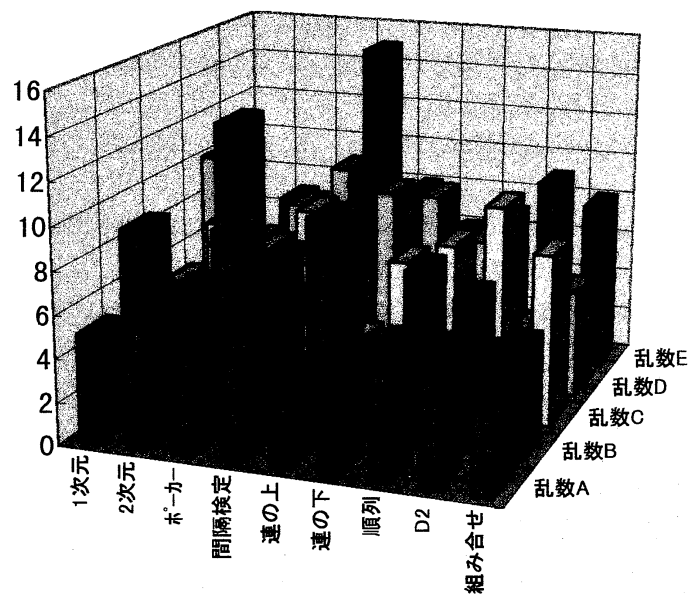


表 10 は、乱数生成法 A について実験 1 で近似解の精度が良かった乱数の種と統計的検定との関係を調べたものである。実験 1 で求めた誤差の小さかった乱数の種 25 個の中で、統計的検定の結果との関係に注目した。それぞれの統計的検定によって良くないと考えられる乱数の種 50 個をマークした。

表 10 近似解の精度が良かった乱数種(擬似乱数 A)と統計的検定との関係

乱数種	1次元	2次元	ホーカ-	間隔検定	連の上	連の下	順列	D2	組み合わせ
69	13	15	8	3	2	8	4	9	4
85	14	21	14	8	8	7	6	11	9
103	16	14	9	7	9	8	9	12	5
107	21	18	8	9	5	5	7	8	8
127	20	18	6	7	10	10	6	8	8
131	16	16	9	6	6	10	11	10	4
137	23	20	5	9	11	6	8	9	6
145	25	20	12	10	11	12	7	8	6
203	23	20	10	3	11	11	8	4	8
207	13	19	8	7	7	9	11	7	6
223	16	19	6	8	6	9	10	7	5
253	27	13	12	10	7	5	12	11	9
353	20	19	11	10	10	8	13	10	7
477	24	18	4	7	10	11	11	9	9
491	21	18	8	6	7	10	7	12	10
513	18	22	12	9	10	8	12	10	4
525	23	23	14	10	10	9	12	12	9
541	28	21	13	6	12	15	12	9	6
587	17	22	11	8	4	7	12	11	9
699	24	22	14	5	5	12	11	7	9
779	18	20	10	10	7	10	12	9	11
801	20	18	9	10	9	9	10	7	5
921	21	16	9	3	10	4	13	8	8
965	19	22	8	13	11	12	13	11	10
979	20	27	6	5	7	6	6	13	11

全体のまとめ

図 5 から乱数種によって、近似解の精度が大きく変化することがわかった。また表 7、8 から乱数生成法 C は、連の検定において非常に高い数値をとることが分かった。したがって乱数生成法 C は、スペクトル検定(表 1)からも分かるようにランダム性に問題があると考えられる。その他の乱数生成法においても連の検定では棄却域に入る乱数種が多い。

表 9 については、間隔検定だけが他の検定に比べて、近似解の誤差が大きい乱数種と重なる数が多いので、近似解の精度が悪くなかった乱数種との相関が考えられる。さらに、表 10 から分かるように間隔検定と連の上の検定では棄却された乱数種は 1 つしかない。したがって近似解の精度が良かった乱数種は、間隔検定と連の上の検定によって棄却されにくい。

以上の二つの点から間隔検定が、SDE の近似解の精度と関係が深いと考えられる。ゆえにコンピュータシミュレーションにおいて近似解を構成する擬似乱数の乱数種は、間隔検定をパスした乱数種を用いれば良いと考えられる。

補足

SDE の解の近似解の精度を評価する場合、

- i) 階級値における $\bar{f}(x)$ と $f(x)$ の差の 2 乗誤差の和
- ii) 階級値における $\bar{f}(x)$ と $f(x)$ の差の絶対誤差の最大値
- iii) 階級幅における $\bar{f}(x)$ と $f(x)$ の面積の差の 2 乗誤差の和

iv) 階級幅における $\bar{f}(x)$ と $f(x)$ の面積の絶対誤差の最大値

及びそれぞれの階級幅を 0.0025、0.005、0.01、0.02、を変えて 16 のパターンで比較した。この結果、図 5 から階級幅による差は少ないといえる。

実験 2、3 から、スペクトル検定において 3 次元以上でランダムでないと予想される擬似乱数 $X_i = 65539 X_{i-1}$ の乗算型合同法は、連の検定により質の良い擬似乱数ではないことがはっきりした。表 7、8 より近似解の密度分布に対して擬似乱数の乱数種によって大きな差が生じることが分かったが、M 系列と線形合同法とのはっきりした違いは確認できなかった。

参考文献

1. 伏見正則、乱数 (東京大学出版会、1989) .
2. 伏見正則、確率的方法とシミュレーション (岩波書店、1994) .
3. 宮武修、脇本和昌、乱数とモンテカルロ法 (森北出版株式会社、1978)
4. Birger, J., *Random Number Generators*, Victor Pettersons Bookindustri Aktiebolag, Stockholm, 1966.
5. Faure, O., Numerical pathwise approximate of stochastic differential equations, preprint (1990).
6. Gihman, I. I. and Skorohod, A. V., *The Theory of Stochastic Processes III*, Springer-Verlag, Berlin, 1979.
7. Hopkins, T.R. (1983) Algorithm AS193. A Revised Algorithm for the Spectral Test, 17-21
8. Ikeda, N. and Watanabe, S., *Stochastic Differential Equations and Diffusion Processes*, North-Holland Publ. Co., Amsterdam, Oxford, New York; Kodansha Ltd., Tokyo 1981.
9. Kanagawa, S. (1988), On the rate of convergence for Maruyama's approximate solutions of stochastic differential equations, *Yokohama Math. J.*, 36, 79-85.
10. Kanagawa, S. (1989), The rate of convergence for approximate solutions of stochastic differential equations, *Tokyo J. Math.*, 12, 33-48.
11. Kanagawa, S., (1995) Error estimation for the Euler-Maruyama approximate solutions of stochastic differential equations, *Monte Carlo Methods Appl.*, 1, 165 - 171.
12. Kanagawa, S., Convergence rates for the Euler-Maruyama type approximate solutions of stochastic differential equations, in *Probability Theory and Mathematical Statistics, Proceedings of the Seventh Japan-Russia Symposium*, pp. 183-192, World Scientific Publ., Singapore, 1996.
13. Kanagawa, S., Confidence intervals of discretized Euler-Maruyama approximate solutions of SDE's, to appear in *Proceedings of the Second World Congress of Nonlinear Analysts*, Athens, Greece, 1996.
14. Karlin, S., *A First Course in Stochastic Processes*, Academic Press, New York, 1971 (佐藤健一、佐藤由美子訳、産業図書、1974) .
15. Kloeden, P. E. and Platen, E., *Numerical Solutions of Stochastic Differential Equations*, Springer-Verlag, Berlin, 1992.
16. Kloeden, P. E., Platen, E. and Schurz, H., *Numerical Solutions of SDE Through Computer Experiments*, Springer-Verlag, Berlin, 1994.
17. Kohatsu-Higa, A., Weak approximations for stochastic differential equations with boundary conditions, Preprint (1996).
18. Knuth, D. E., *Semi-numerical Algorithms, The Art of Computer Programming, vol. 2*, Addison-Wesley, Reading, Mass., 1981, (準数値算法/乱数、渋谷正昭訳、サイエンス社) .

19. Mackevicius, V., Convergence rate of Euler scheme for stochastic differential equations: Functionals of solutions, preprint (1994).
20. Maruyama, G. (1955), Continuous Markov processes and stochastic equations, *Rend Circ. Mat. Palermo*, 4, 48-90.
21. Milshtein, G. N. (1978), A method of second-order accuracy integration of stochastic differential equations, *Theor. Prob. Appl.*, 23, 396-401.
22. Ogawa, S. (1992), Monte Carlo Simulation of Nonlinear Diffusion processes, *Japan J. of Industrial and Appl. Math.*, 9, 22-33.
23. Press, W. H., Teukolsky, S. A., Vetterling, W. T. and Flannery, B. P., *NUMERICAL RECIPES in C*, Cambridge University Press, 1988 (丹慶勝一・奥村晴彦・佐藤俊郎・小林誠訳、技術評論社刊、1993)
24. Shimizu, A. (1984), Approximate solutions for stochastic differential equations, *Proc. Symp. on Stochastic Differential Equations for Population Genetics*, 201-210 (in Japanese).