# テント写像に基づいた擬似乱数生成法

渡辺裕明[†,☆] 金田康正[††,☆☆]

カオスを持つ数列から良い乱数性を持ち周期の長い数列を生成するための手法として，テント写像に基づく擬似一様乱数生成法を提案する．テント写像は，カオスを持つ系列を生成するロジスティック写像と同様のでたらめさを持つ数列を生成することが可能であり，かつ写像の多重度を大きくすることで近隣の数列同士の相関を少なくすることができる．さらにテント写像はロジスティック写像よりも長周期の数列を生成することが可能であり，ロジスティック写像で必須であった一様分布列への変換が不要になる利点がある．この生成法と既存の各種擬似乱数生成法について統計的検定を実施し，検定結果を比較することで，生成された擬似乱数列の乱数性を評価した．その結果，写像の多重度が13以上の場合は，既存の生成法と比較しても遜色のない乱数列を生成できることが分った．

# A Pseudorandom Number Generator Based on Tent Map

HIROAKI WATANABE[†,☆] and YASUMASA KANADA[††,☆☆]

This paper proposes a pseudorandom numbers (PRN) generator based on "tent map" to obtain a PRN sequence which has good randomness feature and long periodness from the chaotic sequence. Tent map can generate the chaotic sequence whose randomness is similar to the sequence generated by logistic map which can also produce chaotic sequence. Correlations among successive numbers generated by tent map can be reduced by the growth of multifold degree. Moreover, if mapping iterations are done by computer, tent map can produce more longer sequence than logistic map. We investigated both proposed generator and current generators with some statistical tests. We compare the results of these tests to evaluate the randomness of proposed generator. According to the investigation, if the multifold degree of mapping iteration is not less than 13, proposed generator can stand comparison with present generators.

## 1. はじめに

計算機が登場して以来，現在までにさまざまな擬似乱数生成法[☆☆☆]が提案されている．これらの生成法のほとんどは整数論に基づくもので，それらは以下のように分類される[1)]．

i) 線形合同法，
ii) 遅れフィボナッチ数列，
iii) 複数の生成法を混合したもの．

それぞれの生成法にはいくつかのパラメーターがあり，それらは生成される数列の性質に多大な影響を与える．そのため性質の良い乱数列を生成できるパラメーターを見つけるための多大な努力が払われている[2)]．

単純なパラメーターで性質の良い乱数列を得るための試みの1つとして，カオスに基づいた生成法が提案されている[3)~8)]．これらの研究では，カオスを持つ数列の生成源としてロジスティック写像が採用されている．この写像により得られる数列は，初期値に依存することなく，非周期的かつほぼランダムに $(0,1)$ の区間を推移し，初期値がわずかに異なる数列同士は写像を繰り返すことによりまったく異なる挙動を示すという，擬似乱数生成法として好ましい性質を持っている．しかし，ロジスティック写像の計算を計算機で行う場合，演算の丸め誤差の影響で，ある初期値から得られる数列は，写像を繰り返すと短い周期に陥ってし

† 東京大学大学院理学系研究科情報科学専攻
Department of Information Science, Graduate School of Science, the University of Tokyo
☆ 現在，トランス・コスモス株式会社
Presently with Trans Cosmos Inc.
†† 東京大学大型計算機センター
Computer Centre, the University of Tokyo
☆☆ 現在，東京大学情報基盤センタースーパーコンピューティング研究部門
Presently with Information Technology Center, Computer Centre Division, the University of Tokyo

☆☆☆ 本論文の議論では，「乱数」とは，「一様乱数」のことを意味する．また以降では，擬似乱数生成法のことを単に「生成法」と呼ぶことにする．

まう☆欠点が指摘されている[7),8)].

本論文では，ロジスティック写像と同じようなカオス性を示すテント写像に基づく生成法を提案する．ロジスティック写像と異なり，テント写像は特別な変換を行うことなく一様分布する数列が得られることが知られている[9)]．本論文において，テント写像から得られる数列の乱数性を評価するために，いくつかの統計的検定とモンテカルロシミュレーションを実施した．さらに既存の数種の生成法についても同様の検定を実施し，その結果を提案する生成法と比較することで提案する生成法の有効性を評価した．

以下，まず2章でロジスティック写像から得られる数列の性質，ロジスティック写像を乱数生成法として利用した関連研究，3章でテント写像の性質について説明し，ロジスティック写像との相違を述べる．4章で本論文で実施する擬似乱数の評価の方法とその基準を示し，既存の5種類の生成法を用い同様の検定を実施することで，提案する生成法との比較を行う．5章において，その検定結果とその考察を行う．6章では，多重化したテント写像による生成法をモンテカルロシミュレーションに適用し，この生成法から得られる乱数列の評価をする．

## 2. ロジスティック写像による擬似乱数生成法

### 2.1 ロジスティック写像

Mayは，式(1)で定義されるロジスティック写像に含まれる乗数 $a$ $(0 < a < 1)$ の変化によって，初期値 $x_0$ $(0 < x_0 < 1)$ の値により得られる数列 $\langle x\rangle \equiv (x_0, x_1, x_2, \ldots)$ の性質が異なることを発見し，特定の例外を除き以下のようになることを示した[10)]．

$$x_{i+1} = ax_i(1 - x_i) \ (i = 0, 1, 2, \ldots). \qquad (1)$$

(1) $x_i = 0$ または $x_i = (a-1)/a$ の場合，$x_{i+1} = x_i$.

(2) $0 \le a \le 3$ の場合，数列は収束する．

(3) $3 < a < 4$ の場合，数列は周期を持つ．

(4) $a = 4$ で数列は区間 $(0, 1)$ の間を非周期的かつランダムな挙動を示す（カオス状態）．

擬似乱数列の生成については，非周期的かつランダムに振る舞うカオス的状態である $a = 4$ の写像関数を用いることがふさわしい[11),12)]．したがって，以降の「ロジスティック写像」では，$a = 4$ の場合だけを考える．

☆ 以下の議論では，この状態を「退化」(degenerate) と呼ぶことにする．

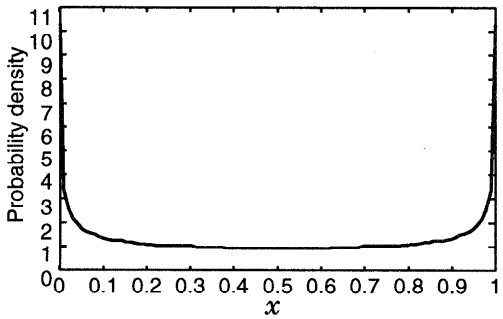


図1 ロジスティック写像から得られる数列の確率密度関数
Fig. 1 Probability density function of the sequence from the logistic map.

### 2.2 ロジスティック写像の性質

式(2)で，1/4, 1/2, 3/4 を除く区間 (0, 1) 内のほとんどすべての実数を初期値 $x_0$ として与えると，カオスを持つ数列 $\langle x\rangle$ が得られる．本節では，$\langle x\rangle$ の性質として，確率密度，初期値に関する敏感さ，多重化された写像関数から得られる数列の性質について述べる．

$$x_{i+1} = 4x_i(1 - x_i) \qquad (2)$$

Ulamら[3),13)]は，式(2)により得られる数列 $\langle x\rangle$ の確率密度関数が，$f(x) = 1/\pi\sqrt{x(1-x)}$ となることを示した．これは，この数列の性質は $x = 0$ と $x = 1$ の近くの値が最も出現頻度が高く，$x = 1/2$ が最低の出現頻度であることを示している（**図1** 参照）．したがって，一様乱数の生成源としてこの数列を利用するためには，何らかの変換手法が必要になる．

初期値が異なる数列同士 $\langle x\rangle$, $\langle y\rangle$ $(x_0 \neq y_0)$ はお互いにまったく独立であり，非常に近接した初期値から生成された数列も，写像を繰り返すことでまったく異なる挙動を示す．このことは乱数を生成する際に非常に好ましい性質である[5),7)]．

しかし，式(2)より得られる数列 $\langle x\rangle$ は隣り合う数同士に強い相関関係があり，数列を擬似乱数として適用する場合に大きな問題となる．たとえば，ある写像回数 $i$ における写像の値 $x_i$ が1に非常に近い値の場合，次に得られる $x_{i+1}$ は必ず0に非常に近い値になる．これを回避するための一般的な方法として，得られた数列を「かき混ぜる」方法があるが，かき混ぜにより得られた数列について理論的な解析を行うことが難しくなる欠点がある．かき混ぜ以外の方法として，「写像の多重化」があげられる．$x_i$ と $x_{i+\tau}$ 同士について $\tau$ を1から増やしていくと，お互いの相関関係は次第に弱くなっていく．この $\tau$ を本論文では「多重度」(multifold degree) と呼ぶことにする．

### 2.3 Lyapunov 指数

Lyapunov 指数とは，カオスを持つ数列の局所不安定性を表す 1 つの指標である．具体的には，初期状態で $\delta x_0$ $(>0)$ の差がある 2 つの数列 $\langle x\rangle$, $\langle x'\rangle$ $(|x_0 - x'_0| = \delta x_0)$ が与えられた場合，それぞれ $i$ 回の写像を行って得られた 2 つの値の差 $\delta x_i$ $(= |x_i - x'_i|)$ が $|\delta x_i| \approx |\delta x_0| e^{\lambda i}$ と近似されるときに，$\lambda$ を「Lyapunov 指数」と呼んでいる．ロジスティック写像の Lyapunov 指数は $\ln 2$ であることが知られており[9]，式 (3) となるように多重度 $\tau$ を選べば隣り合う数同士の相関関係は完全に失われ[7]，ロジスティック写像から得られる数列を乱数列として利用することが可能になる．

$$\tau \geq -\ln|\delta x_0| / \ln 2 \qquad (3)$$

**図 2** のグラフは $\delta x_0$ として計算機イプシロン $\varepsilon$ を与えた場合，どの程度の多重度でロジスティック写像から得られる数列の隣り合う数同士の相関関係が失われるかを示している．図 2 は 1000 個☆の初期値 $x_{0,j}$ $(j = 0, 1, \ldots, 999)$ を区間 $(0, 1)$ の範囲から等間隔に 1000 個選び，初期値 $x_{0,j}$ として $(j+1)/1001 - \varepsilon$, $x'_{0,j} = (j+1)/1001 - 2\varepsilon$ とそれぞれ定めた（$\varepsilon$：計算機イプシロン）☆☆．次に，各写像回数 $i$ における $|\delta x_i|$ の平均値 $(\sum_j |x_{i,j} - x'_{i,j}|/1000)$ を求めプロットした．なお，図中の "Single (Double) Precision" は IEEE 754 標準の単（倍）精度実数で計算した結果である．

単精度の場合では計算機イプシロンが $2^{-23}$，倍精度の場合では $2^{-52}$ である．式 (3) から，それぞれ $\tau \geq 23$，$\tau \geq 52$ において，近隣の数列同士の依存関係が失われるはずである．図 2 より，単精度の場合は，およそ $i \geq 23$ 以降，倍精度ではおよそ $i \geq 52$ 以降で $|\delta x_i|$ が $i$ に依存しなくなる☆☆☆ことが分かる．

### 2.4 ロジスティック写像による生成法

ロジスティック写像を一様乱数列の生成源とするための方法として，1) 写像により得られた数列を一様乱数列に直接変換する方法，2) ある閾値を用いて 1 ビットのベルヌイ試行列を得る方法，3) 得られた実数列の仮数部の一部のビット列を一様乱数列と見なす方法が提案されている．

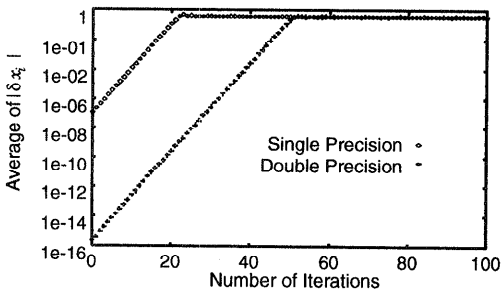


**図 2** 写像回数 $i$ と $|\delta x_i|$ の平均値との関係
Fig. 2 Relation between the number of iterations $i$ and the average of $|\delta x_i|$.

#### 2.4.1 直接変換法

ロジスティック写像により得られる数を直接変換して，区間 $(0, 1)$ に一様分布する数列を作り出す方法である．この方式の生成法として Ulam ら[3]による方法と，Phatak ら[7]による方法がある．

Phatak らは，ロジスティック写像の多重度を変化させ，それに対応する数列の乱数性についての複数の統計的検定を実施した．その結果，多重度が 10 以上であれば良質な乱数列が得られると結論づけた．

#### 2.4.2 0, 1 のビット列に変換する方法

図 1 から明らかなように，式 (2) から得られる数列 $\langle x\rangle$ は $x = 1/2$ を軸に対称な分布を示している．したがって，$x_i < 1/2$ ならば 0 を返し $x_i > 1/2$ ならば 1 を返す関数を定義することで，0 と 1 が等確率に生起する 1 ビットの系列 $\langle b\rangle \equiv (b_0, b_1, b_2, \ldots)$ を生成することができる．

香田ら[4)~6)]はビット列 $\langle b\rangle$ について統計的検定を実施した結果，写像の多重化を行うことで $\langle b\rangle$ が良質のベルヌイ試行列となることを示した．

しかしこの生成法では 1 ビットの系列しか生成できず，1 つの要素の生成に多くのビット列を生成しなければならない．したがって，実際のアプリケーションに適用する場合は非常に多くの演算回数が必要になるという問題点が指摘されている[8]．

#### 2.4.3 写像で得られる値の仮数部を利用する方法

2.4.1 項，2.4.2 項であげた生成法は，隣り合う数同士の相関をなくすために写像の多重化が必須である．このことは，数列の生成速度が低下するだけでなく，得られる数列の周期を短くしてしまう．

ところでロジスティック写像では，$4x_i$ と $1 - x_i$ と

☆ Phatak ら[7]が，単精度実数を用いて，同様な実験を 1000 個の初期値を与えて行っているので，これに準拠した．

☆☆ 初期値をこのように設定した理由は 3 つある．1) まず区間 $(0, 1)$ の中から等間隔にサンプリングしたのは，得られる数列の性質は初期値の大きさに依存しないことを示すためであり，2) ロジスティック写像では，$x_0$ に 1/4, 1/2, 3/4 を与えると，数列はすぐに 0 に収束してしまうので，これを避けるため分母を奇数にした．3) さらに，異なる 2 つの初期値 $x_0, y_0$ により生成される 2 つの数列 $\langle x\rangle$, $\langle y\rangle$ は $x_0 + y_0 = 1$ の関係にあると，$x_i = y_i$ $(i \geq 1)$，と初期値以外はすべて同一の要素を持つ数列が得られてしまう．そのため，各初期値の与え方として $1/1001 - \varepsilon$ のように $\varepsilon$ を引く演算を行った．

☆☆☆ 図 2 のグラフが横軸と水平になっている部分は一見すると一定の値をとっているように思われるが，グラフの縦軸は対数目盛であり実際はランダム（一様ではない）な振舞いをしている．

いう異なる仮数部同士の積によって得られるため，新しく得られた数 $x_{i+1}$ の仮数部は2つの異なるビット列により混ぜ合わせられている．写像の多重化を行わずに乱数列を生成する方法として，この仮数部のビット列に渡辺ら[8]は注目した．

すなわち，ロジスティック写像により得られた値をIEEE 754標準の倍精度実数で表現し，仮数部の一部であるビット列を乱数列として取り出し，このビット列の一様性と乱雑さについて統計的検定を実施し，高速に発生可能な一様乱数発生法としての検討を行った．その結果，仮数部の上位ビットは混ぜ合わされるビット数が少ないために偏りがあることが分かり，仮数部の下位32ビットから得られる系列は頻度検定と最大値検定に合格したことが示された[8]．

## 3. テント写像による生成法

式(2)において $x_i = (1-\cos\theta_i)/2$ $(0 < \theta_i < \pi)$ を代入すると，$x_{i+1} = (1-\cos 2\theta_i)/2$ が得られる．したがって，ロジスティック写像の $\theta_i$ についての写像は式(4)のように定義することができる．

$$\theta_{i+1} = \begin{cases} 2\theta_i & (\theta_i < \pi/2) \\ 2\pi - 2\theta_i & (\theta_i > \pi/2) \end{cases} \qquad (4)$$

この写像はテント写像と呼ばれ，ロジスティック写像と共型変換の関係にあることが知られている[7),9)]．式(4)では，初期値 $\theta_0$ が $\pi$ の有理数倍であるなら，数列 $\langle\theta\rangle$ は周期を持つが，そうでなければ数列は周期を持たない．したがって，$\pi$ の無理数倍を初期値として選べば，区間 $(0, \pi)$ の範囲で一様分布する非周期的な数列 $\langle\theta\rangle$ が得られる[5),14)]．

ここで数列 $\langle\theta\rangle$ を $z_i = \theta_i/\pi$ と変換し，区間 $(0, 1)$ に正規化した数列 $\langle z\rangle$ を考える．$\langle z\rangle$ はPhatakが提案した生成法による数列 $\langle y\rangle$ と実数の場合では同じであるが，計算機上で同じ初期値を与えて得られる系列は同じものではない．**図3**は $y_0$ と $z_0$ に同じ初期値 $1/\pi\cdot\cos^{-1}(2/3)$ を与え，各写像回数 $i$ $(1 \le i \le 17)$ における数列同士の隔たり $|y_i - z_i|$ を示したものである．

この図から分かるように，$|y_i - z_i|$ ははじめ非常に小さな量であったが，$i$ の増加に従って増加している．したがって，これらの2つの数列は計算機上では異なる数列であることが示された．

次にそれぞれの数列を生成する計算過程を考えると，式(2)により得られる数列 $\langle x\rangle$ は0と1付近での頻度が最も高いため（図1参照），相対誤差の大きい計算が頻発する．これに対し，数列 $\langle z\rangle$ は区間 $(0,1)$ に一様分布しており，$\langle x\rangle$ を生成する場合と比べて相対誤差の大きな計算の起きる頻度は少なくなっている．

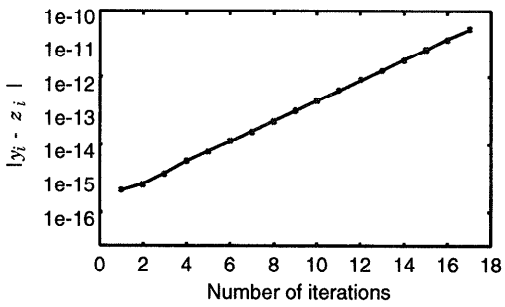


**図3** 各写像回数における $y_i$ と $z_i$ の隔たり
Fig. 3 Difference between $y_i$ and $z_i$ in each iteration.

**表1** ロジスティック写像から得られる数列が退化するまでの写像回数とその頻度
Table 1 Frequency of the number of iterations until logistic map's sequences degenerate.

| 写像回数 $i$ | 頻度 |
|---|---|
| $i < 10^6$ | 4 |
| $10^6 \le i < 10^7$ | 114 |
| $10^7 \le i < 2\times10^7$ | 250 |
| $2\times10^7 \le i < 4\times10^7$ | 122 |
| $4\times10^7 \le i < 7\times10^7$ | 292 |
| $i \ge 10^8$ | 218 |

このためテント写像はロジスティック写像の欠点である数列の退化が起きにくく，ロジスティック写像に比べてより長い数列が生成可能であると期待できる．我々はこの仮説を検証するために，1000個の初期値 $x_0$ を用いて，ロジスティック写像から得られる数列とテント写像から得られる数列が，それぞれ退化するまでの写像回数を求める実験を行った．

まず区間 $(0,1)$ の中から1000個の初期値 $x_{0,j}$ $(j = 0,1,\ldots,999)$ を与える．具体的な初期値の設定方法は，2.3節の実験と同様に $x_{0,j} = (j+1)/1001 - \varepsilon$ とした．これらの初期値 $x_{0,j}$ から写像を繰り返し，数列が退化するまでの写像回数 $i$ を求める☆．

実験の結果，ロジスティック写像から得られる数列が退化するまでの写像回数は**表1**のようになった．最低の写像回数は約 $10^5$，最大は約 $10^8$，平均が約 $4\times10^7$ であった．写像回数が少ない数列 $(i < 10^7)$ のほとんどは0に収束したものであった．これに対し，テント写像から得られる数列はすべての初期値☆☆について写像回数が $2^{32}$ を超えても退化した数列が現れなかった．

☆ なお，写像を行う計算はすべてIEEE754標準の倍精度実数で実施した．以降の計算でも同様である．

☆☆ $z_{0,i} = 1/\pi\cdot\cos^{-1}(1-2x_{0,i})$ の変換を行って初期値を与えた．

したがって，テント写像はロジスティック写像の場合に比べ数列の退化が起こりにくいので，より長い数列が生成可能であることが実験的に示された．またテント写像はロジスティック写像と共系な数列であり，写像のLyapunov指数も $\ln 2$ である[9]．このため多重化による数列の乱数性への影響も同様であり，IEEE754標準の倍精度実数を用いた場合，多重度が52以上になれば隣り合う数列の依存関係はまったく失われる．

ここで，多重度 $\tau$ のテント写像を区間 $(0,1)$ に正規化した数列 $\langle z\rangle_\tau$ を，式(5)のように定義する．ただし，式中の $\theta_i$ は，式(4)から得られる値である．これまでの考察から，多重度 $\tau$ を大きくした場合の数列 $\langle z\rangle_\tau$ は性質の良い乱数列である可能性が高いと思われる．

$$z_i = \frac{1}{\pi}\theta_i,$$
$$\langle z\rangle_\tau = (z_0, z_\tau, z_{2\tau}, \ldots). \tag{5}$$

## 4. 乱数性の評価

本章では提案する生成法の評価のために，擬似乱数を評価する場合に用いられる一般的な統計的検定法とモンテカルロシミュレーションとをそれぞれ実施する．また現在までに提案されている擬似乱数生成法の中から5種類の生成法を選び同様の評価を行い，その結果から提案する生成法が乱数性の面で実用的であるかどうかを検討する．

### 4.1 統計的検定による評価

以下のような統計的検定手法[2),15)]を用いて，提案する擬似乱数生成法の評価を行う．

- 一次元頻度検定（10分割，100分割）
- 二次元頻度検定（10分割/次元）
- 分割検定（0から9に正規化された数の5個組）
- 札集め検定（0から9に正規化された数）
- 連の検定（上り，下り）
- 5個の数の最大値検定
- 衝突検定（10次元，4分割/次元）
- 間隔検定（0から9に正規化された数）

本論文で実施した擬似乱数列の検定方法の概略を以下に示す．

1ブロックとして1つの初期値から出発して得られる $10^7$ 個の乱数列を生成し，それぞれの検定手法を用いて，1つのカイ2乗統計値 $\chi^2$（5個の数の最大値検定のみKolmogorov-Smirnov（K-S）統計値[2)] $(K^+, K^-)$）を計算する．

この計算を初期値を変えた数列1000ブロックについて実施し，1000個の $\chi^2$ もしくは $(K^+, K^-)$ を得る．得られた $\langle\chi^2\rangle \equiv (\chi_0, \chi_1, \ldots, \chi_{999})$ または $\langle K^+\rangle \equiv (K_0^+, K_1^+, \ldots, K_{999}^+)$，$\langle K^-\rangle \equiv (K_0^-, K_1^-, \ldots, K_{999}^-)$ が理想的な分布に近いどうかを再度K-S検定[2)]により検定する．

### 4.2 提案する生成法の生成条件

Phatakらの提案した生成法ではロジスティック写像の多重度が約10以上で良好な結果を示しているため，我々は多重度が1～16の場合とさらに大きな多重度での結果を調べるために多重度を32と53とした場合の数列について調査を行った．

次に数列を生成する初期値は図2で与えたものと同様に $1/1001-\varepsilon, 2/1001-\varepsilon, \ldots, 1000/1001-\varepsilon$ を与えて数列を生成させた．これらの初期値から生成された数列が退化するまでの写像回数は，3章の実験結果のとおりすべての初期値で $2^{32}$ を超えているので，多重度が53の場合でもすべての初期値から $10^7$ 個の数列を退化させることなく生成できる．

### 4.3 既存の主な生成法の生成条件

以下の5種類の生成法から得られる擬似乱数についても，4.1節であげた統計的検定を実施☆し，提案する生成法との結果を比較することで，提案する生成法の有効性を検証する．

**線形合同法[16)]（LCG）**

$$X_n = 16807X_{n-1} \pmod{2^{31}-1}$$

**最大周期列[15)]（MSQ）**

$$X_n = X_{n-521} \oplus X_{n-32}$$

**加算型遅れフィボナッチ数列[17)]（LFA）**

$$X_n = X_{n-55} + X_{n-24} \pmod{2^{32}}$$

**乗算型遅れフィボナッチ数列[17)]（LFM）**

$$X_n = X_{n-55} \times X_{n-24} \pmod{2^{32}}$$

**複合型生成法[18)]（RN2）**

文献18)に記述されているプログラム `ran2()` による数列

### 4.4 評価基準

以下に我々が検定を行う生成法の評価基準を述べる．

まず各検定手法における有意水準は1/1000とした．すなわち乱数列が理想的なものであれば，$K > 1.945$ である確率は，1/1000未満であることから，K-S統計値 $K = \max(K^+, K^-)$ が1.945を超えるものを棄却する．ただしLCGについては得られる数列の周期が $2^{31}-2 \approx 2\times 10^9$ であるので，提案する手法と同様のブロック数 $(=10^3)$ で検定を行うと必要な乱

☆ なお，`ran2()` 以外の生成法の初期値を生成するにあたり，UNIX Cの標準ライブラリー関数である `random()` を利用した．`ran2()` については0を初期値とし，`random()` については，`srandom(1)` で初期化した．

**表2** 各多重度のテント写像による生成法と既存の5種類の生成法の各検定手法における Kolmogorov-Smirnov 統計値（下線で示された数値は有意水準 1/1000 の検定で棄却されることを示している）

Table 2 Kolmogorov-Smirnov statistics from each empirical tests for both the generator based on tent map with each multifold degree and five currently used generators (The part of underlined number indicates that the generator is rejected in significant level of 1/1000).

| 検定手法 | 写像の多重度：$\tau$ | | | | 既存の生成法 | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 8 | 12 | 13 | 14 | LCG | MSQ | LFA | LFM | RN2 |
| 頻度 ($1 \times 10$) | 0.947 | 0.763 | 0.526 | 0.784 | 1.389 | 0.786 | 0.732 | 0.977 | 1.050 |
| 頻度 ($1 \times 100$) | 0.982 | 0.826 | 0.708 | 0.727 | 0.513 | 1.760 | 0.960 | 0.736 | 0.724 |
| 二次元頻度 | 31.62 | 8.550 | 3.364 | 1.206 | 0.842 | 1.426 | 0.575 | 1.034 | 0.933 |
| 分割検定 | 28.99 | 0.889 | 1.282 | 1.133 | 0.520 | 1.593 | 0.616 | 0.865 | 0.638 |
| 札集め検定 | 5.800 | 0.435 | 0.658 | 1.497 | 0.751 | 0.943 | 0.812 | 0.678 | 1.172 |
| 上りの連 | 0.515 | 0.612 | 0.846 | 1.403 | 1.474 | 1.059 | 0.688 | 0.901 | 2.015 |
| 下りの連 | 1.172 | 1.396 | 0.886 | 0.592 | 1.187 | 1.080 | 0.508 | 1.105 | 0.941 |
| 最大値 ($K^+$) | 29.86 | 2.652 | 1.455 | 0.445 | 1.100 | 0.699 | 0.800 | 0.702 | 0.456 |
| 最大値 ($K^-$) | 29.92 | 3.069 | 1.259 | 1.857 | 0.889 | 1.263 | 0.608 | 1.028 | 0.838 |
| 衝突検定 | 0.963 | 0.825 | 0.607 | 1.034 | 0.685 | 1.198 | 0.485 | 1.083 | 0.950 |
| 間隔 ($D = 0$) | 31.59 | 1.733 | 0.652 | 0.790 | 0.672 | 1.029 | 0.533 | 0.464 | 0.852 |
| 間隔 ($D = 1$) | 29.65 | 0.588 | 1.248 | 1.311 | 1.969 | 0.817 | 0.430 | 0.868 | 1.286 |
| 間隔 ($D = 2$) | 29.67 | 0.746 | 1.067 | 0.854 | 0.592 | 0.699 | 1.257 | 0.753 | 0.693 |
| 間隔 ($D = 3$) | 29.74 | 0.576 | 0.844 | 0.495 | 0.952 | 1.116 | 0.903 | 0.571 | 0.545 |
| 間隔 ($D = 4$) | 29.24 | 0.767 | 0.562 | 1.014 | 0.629 | 0.877 | 1.325 | 1.267 | 1.301 |
| 間隔 ($D = 5$) | 30.00 | 0.722 | 0.581 | 0.671 | 0.643 | 0.682 | 0.761 | 1.001 | 0.841 |
| 間隔 ($D = 6$) | 31.61 | 0.612 | 0.835 | 0.605 | 0.660 | 1.164 | 0.672 | 0.764 | 0.951 |
| 間隔 ($D = 7$) | 29.16 | 1.166 | 1.032 | 1.083 | 0.547 | 1.045 | 1.589 | 0.835 | 0.986 |
| 間隔 ($D = 8$) | 28.90 | 0.721 | 0.876 | 1.064 | 1.043 | 0.406 | 1.162 | 0.680 | 1.183 |
| 間隔 ($D = 9$) | 29.83 | 1.032 | 0.689 | 1.125 | 0.930 | 0.726 | 0.907 | 0.549 | 0.495 |

数の個数は全周期を超えてしまう．したがって，1ブロックは $10^7$ として，そのブロック数を200として検定を実施した．すなわち，ある検定手法を適用した結果，K-S統計値が1.938を超えた生成法は，有意水準 1/1000 で棄却する．

我々は20種類の検定手法を適用して，そのそれぞれについてK-S統計値を求め有意水準 1/1000 の検定を行った．仮に理想的な乱数を使い，すべての検定結果が互いに独立（ある検定結果が他の検定結果と相関関係がない）と見なせば，検定結果は試行回数が20のベルヌイ試行列となる．この仮定のもとで，20個のK-S統計値について有意水準 1/1000 の検定を行った場合，1つ以上の統計値が棄却される確率はほぼ 1/100 になる．

しかし，本論文で用いたそれぞれの検定法同士が完全に独立ということを示したわけではない．したがって，有意水準が 1/1000 未満となるように，20種類の検定手法を適用した結果，2種類以上棄却された乱数生成法については妥当でないと判断する☆．

☆ なお，各検定法が互いに独立である場合，理想的な乱数列が2種類以上のK-S統計値で棄却される確率は約 $4.5 \times 10^{-5}$ である．

## 5. 検定結果

**表2**は各多重度 $\tau$ のテント写像による生成法と，既存の5種類の生成法の統計的検定の結果である．表中の下線がつけられた数値の部分の生成法は，それぞれの検定法において有意水準 1/1000 の検定で棄却されることを示している．

既存の5種類の生成法については，LCGでは間隔検定で1つだけ棄却されたものがあり，RN2では上りの連の検定で棄却される結果を得た．これら以外の検定にはそれぞれの生成法が合格しており，全体的に見ればこれらの生成法も良好な結果を示している．

多重化されたテント写像による生成法は，多重度が高くなるにつれて性質が向上する傾向が見られる．そして多重度が13の数列は1つだけ棄却され，それ以上の多重度の数列 ($\tau = 14, 15, 16, 32, 53$) はまったく棄却されないという結果を得ており，これらの生成法は既存の生成法と比較しても遜色のない乱数性を持つ数列を生成できることが明らかになった．

さらにこのような結果は，検定に使用した数列の各ブロック同士が互いに独立でなければ得られない．したがって，この生成法は単純な方法により初期値を選んだにもかかわらず，初期値が異なる数列同士が互い

に独立した数列であることが示されている．これは乱数生成を並列処理で行う場合には非常に重要な意味を持っている．

## 6. モンテカルロシミュレーションによる評価

テント写像に基づく生成法が実際のシミュレーションに適用できるどうかを評価するため，解析的に計算できる問題を検定される乱数列を利用してモンテカルロシミュレーションを用いて解き，真の値からどれだけかけ離れているかで，生成された乱数列を評価する．

### 6.1 高次積分の計算

次式で与えられる高次積分 $S_n$ に対し，モンテカルロシミュレーションにより得られた結果と真の値との誤差が妥当なものであるかどうかを検討する．

$$S_n = \int_0^1 \cdots \int_0^1 e^{-(x_1+\cdots+x_n)}\, dx_1 \cdots dx_n \quad (6)$$

$S_n$ の値は解析的に求められ，$S_n = \left(\int_0^1 e^{-x}\, dx\right)^n = (1-e^{-1})^n$ となる[19]．なお，シミュレーションにより計算された積分値は，真の値 $S_n$ と区別するために $S'_n$ とする．

乱数列の評価をするために，どのくらいの次元数のシミュレーションを行うのが適当であるかは，適用する問題に応じて決まるはずであり一概に決めることはできない．本論文では1次元から10次元までのシミュレーションを行い，その結果について考察を行うことにする．

### 6.2 シミュレーション結果

**図4**は，提案する生成法を用いた10次元のシミュレーション結果である．図中の横軸は10次元空間にプロットした点の個数であり，縦軸は積分値を示している．グラフの中央に水平に伸びる直線は，理論値 $S_{10}$ であり破線で示された2本の滑らかな曲線は95%の誤差限界を与える曲線である．それぞれの折れ線がこれらの曲線の間を外れた場合，生成した乱数列は有意水準5%で棄却されることになる．

図4より多重度 $\tau$ が13から53までの範囲で生成されたすべての数列は，すべての試行回数で誤差限界内に入る結果を得た．最終結果の偏りもみられていない．さらにシミュレーションの次元数 $n$ を1から9まで変化させた場合でも良好な結果が得られている．

## 7. おわりに

カオスと持つ数列から性質の良い一様乱数列を生成するための方法として，ロジスティック写像と共系な数列を生成するテント写像による擬似乱数生成法を提案し，その乱数性について評価を行った．

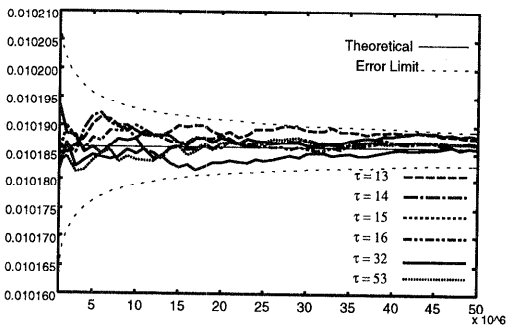


**図4** 多重化したテント写像によるモンテカルロシミュレーションの結果

Fig. 4 Result of Monte Carlo simulation for multifolded tent map.

その結果，写像の多重度が13以上の数列については，既存の擬似乱数生成法による数列と比較しても遜色のない数列をロジスティック写像を用いるよりも長い周期で生成することが可能であることが分かった．

今後の課題として，テント写像により生成される数列が退化するまでの写像回数に関する数値実験，より多くのモンテカルロシミュレーションを行って，本生成法の有効性の確認がある．さらに，初期値を変えることで独立な数列を生成することができるテント写像を利用する並列擬似乱数生成法の評価も必要である．

なお本研究の遂行にあたって，一部科学研究費補助金（特定領域研究（A）「発見科学」，課題番号10143103，計画研究班代表者：金田康正）の適応を受けた．

## 参考文献

1) James, F.: A Review of Pseudorandom Number Generators, *Comput. Phys. Commun.*, Vol.60, pp.329–344 (1990).
2) Knuth, D.E.: *The Art of Computer Programming*, Vol.2, Seminumerical Algorithms, Addison-Wesley (1981).
3) Ulam, S.M. and von Neumann, J.: On Combination of Stocastic Determistic Processes, *Bull. AMS.*, Vol.53, p.1120 (1947).
4) 香田 徹，緒方栄次：ベルヌイ試行とカオス，電気通信学会論文誌，Vol.J68-A, No.2, pp.146–152 (1985).
5) 香田 徹，柿本厚志：擬似乱数の歩行長，電気通信学会論文誌，Vol.J68-A, No.10, pp.1016–1023 (1985).
6) 香田 徹，柿本厚志：擬似乱数とカオス，情報処理学会論文誌，Vol.27, No.3, pp.289–296 (1986).
7) Phatak, S.C. and Rao, S.S.: Logistic map: A Possible Random-number enerator, *Phys. Rev. E*, Vol.51, No.4, pp.3670–3678 (1995).

8) 渡辺裕明，金田康正：ロジスティック写像による擬似乱数発生法，第53回情報処理学会全国大会論文集，Vol.1, pp.65–66 (1996).
9) McCauley, J.L.: *Chaos, Dynamics and Fractals*, Chapter 2, Cambridge Nonliner Science Series 2 (1993).
10) May, R.M.: Simple Mathematical Models with Very Complicated Dynamics, *Nature*, Vol.261, pp.459–467 (1976).
11) Parakash, S. and Peng, C.-K.: Determistic Diffusion Generated by a Chaotic Map, *Phys. Rev. A*, Vol.43, No.12, pp.6564–6571 (1991).
12) Peng, C.-K. and Parakash, S.: Randomness Versus Determistic Chaos: Effect on Invasion Percolation Clusters, *Phys. Rev. A*, Vol.42, No.8, pp.4537–4542 (1990).
13) von Neumann, J.: Various Techniques Used in Connection With Random Digits, *John von Neumann Collected Works*, Vol.V, pp.768–770 (1963).
14) Grossman, S. and Thomae, S.: Invariant Distributions of One-Dimentional Discrete Processes, *Z. Natureforsh*, Vol.32a, pp.1353–1363 (1977).
15) 伏見正則：乱数，東京大学出版会 (1989).
16) Part, S.K. and Miller, K.W.: Random Number Generators: Good Ones are Hard to Find, *Comm. ACM.*, Vol.31, No.10, pp.1192–1201 (1988).
17) Codddington, P.D.: Random Number Generators for Parallel Computers, *NHSE Review* (1996). http://nhse.cs.rice.edu/NHSEreview/RNG/.
18) Press, W.H., Flannery, B.P., Teukolsky, S.A. and Vetterling, W.T.: *Numerical Recipes in C*, Chapter 9, Cambridge University Press, Reading (1988).
19) 宮武 修，脇本和昌：乱数とモンテカルロ法，数学ライブラリー47，森北出版 (1978).

(平成10年8月11日受付)
(平成11年5月 7 日採録)

**渡辺 裕明**

1971年生．1995年工学院大学工学部電子工学科卒業．1998年東京大学大学院理学系研究科情報科学専攻修士課程修了，同研究生を経て1999年トランス・コスモス（株）入社，現在に至る．在学中はカオス的な挙動を示す数列から擬似乱数を生成する手法の研究に従事．第53回情報処理学会全国大会奨励賞受賞．

**金田 康正**（正会員）

1949年生．1973年東北大学理学部物理第二学科卒業．1978年東京大学大学院理学系研究科博士課程修了．理学博士．1978年名古屋大学プラズマ研究所助手，1981年東京大学大型計算機センター助教授，同教授を経て現在東京大学情報基盤センター教授．その間英国ケンブリッジ大学計算機研究所客員研究員，名古屋大学プラズマ研究所客員助教授，核融合科学研究所客員助教授．昭和58年度（欧文）および平成10年度（邦文）情報処理学会論文賞受賞．平成6年度情報処理学会Best Author賞受賞．著書「$\pi$のはなし」（東京図書），共著「アドバンスト・コンピューティング—21世紀の科学技術基盤」（培風館），編著「Trends in Supercomputing」（World Scientific）．日本応用数理学会，プラズマ・核融合学会，ACM，SIAM各会員．研究テーマは「大規模数値計算」および「研究の研究」．円周率計算桁数および乱数発生速度に関する世界記録を保持．