数理解析研究所講究録
1032巻 1998年 118-124

擬似乱数の目指すもの —— 計算量理論の視点から

九州大学大学院数理学研究科　　杉田　洋 (Hiroshi SUGITA)

乱数の定義が「計算量」を基に与えられていることはよく知られている([9]). ところが, 擬似乱数も計算量を用いて定義する試みがあることは, 確率数値解析の研究者の間ではそれほど知られていないようだ. 実は, 現在のところ, その試みは少なくとも確率数値解析では現実レベルまで達していないので, 仕方ないのかも知れない. しかし, 工学的にはほぼ成功を収めつつある今こそ, 次に目指すべき擬似乱数像を求めて計算量の視点からこれを見直すことは意義深いことと思う.

本稿の目的は, 計算量と擬似乱数の問題を説明することである. その際, M-系列([2, 12]) と無理数回転による方法([10, 11]) を引き合いに出す. ただし, 計算量の話を手短にかつ正確に紹介するのは困難なので, ここでは解説風に述べるに留める. 詳しくは[3, 5, 6, 7, 8] などを参照されたい.

なお, 本稿で扱う乱数および擬似乱数のモデルは「公平な硬貨投げの確率過程」, すなわち, $\{0,1\}$-値 i.i.d. で平均 $1/2$ とする.

二つの例

後の節ために, 擬似乱数生成法を二つだけ紹介する. いずれも力学系の枠組みを用いている. 力学系による擬似乱数生成を一般的に述べると, 次の通り. 集合 Ω 上の変換

$$F : \Omega \to \Omega \tag{1}$$

と写像

$$G : \Omega \to \{0,1\} \tag{2}$$

を用意する. 初期値 (「種 (seed)」と呼ぶ) $\omega \in \Omega$ をユーザーが選択し, $\{0,1\}$-値擬似乱数 $\{Y_n\}_{n=0}^{\infty}$ を次で定義する.

$$\omega_n := F(\omega_{n-1}) \tag{3}$$

$$Y_n(\omega) := G(\omega_n) \tag{4}$$

と定義する. ただし, $\omega_0 := \omega$ とする.

第一の例は「M-系列」と呼ばれる代数的手法によるものである. M-系列は非常に長い周期を持つ擬似乱数生成法として一世を風靡した. ここで紹介するのは初期の「三項間漸化式による M-系列」である([2, 12]).

定義 1. 三項間漸化式によるM-系列は，$p > q > 0$ として漸化式

$$x_n := x_{n-q} + x_{n-p} \pmod 2 \tag{5}$$

を用いて $\{0,1\}$-列 $\{x_n\}$ を生成する．ただし，多項式 $x^p + x^q + 1$ は GF(2) 上で原始的であるとする．初期値 $x_0, \ldots, x_{p-1} \in \{0,1\}$ はすべてが 0 ではないように与える．

$\{x_n\}$ が擬似乱数として用いられる最大の根拠は次の定理にある．

定理 2. $\{x_n\}$ は周期が $2^p - 1$ で，相続く p 項は $(0, \ldots, 0)$ を除くすべてのパターンが一周期中に一回だけ現われる (p 次均等分布性)．

第二の例は筆者が [10, 11] で発表した「無理数回転による擬似乱数生成法」と呼ばれる確率論に基づく手法である．

定義 3. (無理数回転による擬似乱数生成)
無理数 $\alpha \in [0,1)$ と自然数 m に対して，$\{0,1\}$-擬似乱数を次のように定義する．$n = 1, 2 \ldots$ として，

$$Y_n^{(m)}(\omega) := \sum_{i=1}^{m} d_i\left(\{\omega + (n-1)\alpha\}\right) \pmod 2 \tag{6}$$

ここに，$\omega \in [0,1)$ は初期値．ただし，$A \pmod 2$ というのは A を 2 で割った余り，すなわち，A が偶数のときは 0，奇数のときは 1 とすることを表す．$d_i(\omega)$ は ω の2進小数展開における小数点以下第 i 桁の数 (0 または 1) である．また $\{t\}$ は実数 t の小数部分を表す記号．

このとき，十分大きな m に対して $\{Y_n^{(m)}\}$ は擬似乱数として用いられる．その根拠を与えるのが次の定理である．

定理 4. ほとんどすべての無理数 α に対して，初期値 ω をルベーグ確率測度に従って選べば (6) によって定義された確率過程 (擬似乱数) $\{Y_n^{(m)}(\omega)\}_{n=1}^{\infty}$ は $m \to \infty$ のとき，硬貨投げの確率過程に有限次元分布の意味で収束する．

アルゴリズムのランダム化と擬似乱数

擬似乱数の検定の実施を見ると，従来から次のような方法が一般的であることに気づく．: 種をランダムに選んで擬似乱数を生成し，その検定を行う．採択される擬似乱数を生成する種の比率がその検定の危険率程度ならば，全体として「採択」とするのである．この検定方法は「擬似乱数は，初期値をランダムと見なす確率過程であり，検定は，確率過程として分布が硬貨投げの確率過程に近いことを検証する作業」であることを示している．

すでに Martin-Löf の理論 [9] によって決定論的アルゴリズムからは決して真の乱数は得られないことが分かっているし，実際，種を固定すると検定で棄却される擬似乱数をたびたび生成してしまう．そこで，アルゴリズムを「ランダム化」して次善の策を練ろうとい

うわけだ．アルゴリズムのランダム化は計算量の理論からすれば非常に大きな関心がある．たとえば，現在，計算量理論の最大の問題の一つとなっている「$\mathbf{P} \neq \mathbf{NP}$ 予想」は，まさに，こうしたランダム化が決定論的アルゴリズムの世界よりも真に豊かな内容を有するであろうことを予想している．

一方，個々の擬似乱数生成プログラムでは選ぶことのできる種の総数が限られているから，生成される擬似乱数のランダム性 (=どれだけ硬貨投げの確率過程に分布が近いか) にも限界がある．だから，様々な問題に対応するためには，理論上,「擬似乱数生成プログラムの列」を考える必要がある．

従来から行われている検定のやり方と,「擬似乱数生成法の列」という考え方を合わせてみると，擬似乱数は少なくとも「ランダムな入力 (種のこと) を持つコンピュータアルゴリズムで生成された確率過程の列で，硬貨投げの確率過程に分布収束するもの」でなければならない．

無理数回転による擬似乱数生成は，これらの要請を踏まえて設計されている．もっとも，M-系列の方法でも，GF(2) 上の原始多項式の列でどんどん次数の大きくなるようなもの (具体的に構成するのは大変かもしれないが存在することは明らか) を考えれば，定理 4 と同様の確率過程としての擬似乱数の収束定理を示すことができるだろう．

ところが，この二つの方法は，確率過程の収束という点では同じだけれども，計算量の観点から見ると，大きく趣きが異なっているのである．

計算量理論による擬似乱数の定義

ところで,「アルゴリズムをランダム化すれば，ランダムなものが得られるのは自明で，これではトートロジーに過ぎないのではないか」といぶかる人もいるだろう．もちろん，n ビットのランダムな種から n ビットのランダムな出力データを得るだけだとしたら，まさにトートロジーである．しかし，擬似乱数生成の目的は，n ビットのランダムな種から n より (ずっと) 大きい $\ell(n)$ ビットの「ランダムに見える」出力を得ることなのである．

もちろん，n ビットのランダムな種から n より大きい $\ell(n)$ ビットの「ランダムな」出力を得ることは不可能である．場合の数を考えてみればすぐ分かる．真にランダムな $\ell(n)$ ビットの異なるデータは全部で $2^{\ell(n)}$ 個あるべきなのに，n ビットのランダムな種からは高々 2^n 個しか出力が得られない．

「ランダムに見える」とは検定に採択されるの意である．ランダムでないものは，必ず何らかの検定で棄却される．だから「すべての検定」を考えたのではもとより望みがない．そこで，検定のクラスを制限して考えよう．このとき，各々の検定のクラスに応じて，そのクラスに属する検定には採択されるがそうでない検定には棄却されることを許されるような「擬似乱数」が定義される．

それでは検定のクラスは何を基準に定めればよいか．その最も信頼できる基準が「計算量」と考えられるのである．

例を挙げて説明しよう．いま，k 次元分布についての検定を考える．長さ k の $\{0,1\}$-列の総数は 2^k である．従って，長さ k の $\{0,1\}$-列の集合の総数は 2^{2^k} 個あることになる．k

が少し大きいと，これはとてつもなく大きい数だ．$\{0,1\}$-列の集合の一つひとつをできるだけ短いプログラムを書いて指定することを考えよう．これらはすべて異なっていなければならないから，全部で 2^{2^k} 個のプログラムが必要になる．すると，そうしたプログラムのうち長いものは 2^k ビット程度の長さを持たなければならない．実は大多数のプログラムは 2^k ビット程度の長さを持つことがすぐ分かる．(たとえば，長さが半分の 2^{k-1} ビット以下のプログラムの総数は高々 $2^{2^{k-1}}$ 個で，これは全体 2^{2^k} 個の $2^{2^{k-1}}$ 分の 1 に過ぎない！)

さて，そんな長いプログラムでないと記述できないような $\{0,1\}$-列の集合を考えよう[1]．与えられた擬似乱数 ($\{0,1\}$-列) がその集合に入るかどうかについて検定を行うことを考えると，少し大きい k に対して，このような検定は計算量があまりに大きくて，まったく実施不可能であることが分かる．だから，擬似乱数がこのような検定に採択される必要はない，と考えよう，というアイデアが生まれる．

一方，計算量の小さい検定には採択される必要がある．もっとも，「計算量の小さい検定」のクラスには，現実に行われているようなすべての検定が属するから，もちろん，これは簡単な話ではない．

以上，述べたことから，現実に実施可能な検定というのは検定の全体から考えると，ほんのわずかに過ぎないことが分かる．実施可能な検定に採択されるのことが「ランダムに見える」の真意であり，すべての検定に採択されなければならない「真にランダム」とは大変な違いがある．ここに，擬似乱数生成の可能性が開けて来る[2]．

結局，「種の小さいランダム性を，計算量の小さい分布に重点的に配分して，計算量の大きい分布は放っておく」というのが擬似乱数生成法の極意なのだ．計算量理論の分野では，このような性質を持つ数列を「擬似乱数」と定義している．

具体的に Luby の本 [6] に即して述べてみよう[3]．各 n に対して $f_n : \{0,1\}^n \to \{0,1\}^{\ell(n)}$ なる関数の列 $f = \{f_n\}_{n\in\mathbf{N}}$ を擬似乱数生成法 (pseudorandom number generator) という．ここで $\ell(n) > n$ は増大する自然数列であり，関数 f_n によって n ビットのランダムな種から，$\ell(n) > n$ の長さの使用可能な擬似乱数列が生成できると期待されていることを表わす．Z_n を $\{0,1\}^n$ 上の一様測度 P_n に従う確率変数とするとき，$f_n(Z_n)$ は $\{0,1\}^{\ell(n)}$-値の確率変数であるが，前述のとおり，それは一様には分布しない．ここで，Z_n はユーザーがランダムに選ぶ擬似乱数の種であり，$f_n(Z_n)$ が擬似乱数である．

次に，検定のための関数を考えよう．$A = \{A_n\}_{n\in\mathbf{N}}$ を $A_n : \{0,1\}^{\ell(n)} \to \{0,1\}$ なる関数 ($\{0,1\}^{\ell(n)}$ の部分集合の定義関数に他ならない) の列とし，

$$\delta_A(n) := \left| P_{\ell(n)}\left(A_n(Z_{\ell(n)}) = 1\right) - P_n\left(A_n(f_n(Z_n)) = 1\right) \right| \tag{7}$$

とおく．擬似乱数生成法としては，$\ell(n)$ が n よりずっと大きいこと，関数 f_n の計算が素早くできること，なおかつ，多くの A に対して $\delta_A(n)$ が十分小さいこと，が望ましい．

---

[1] 実は Martin-Löf の理論 ([9]) によって，k が大きいとき，その集合の確率はほぼ 1/2 である．

[2] もちろん，決定論的なアルゴリズムで生成される $\{0,1\}$-列は，そのアルゴリズム自体を用いた検定 — これは当然，実施可能な検定である — で棄却されるから，擬似乱数生成のアルゴリズムはランダム化されている必要がある．このとき，「$\mathbf{P} \neq \mathbf{NP}$ 予想」のようにランダム化されたアルゴリズムの世界が決定論的アルゴリズムの世界より真に豊かであれば，擬似乱数生成が可能，というわけだ．

[3] 計算量による擬似乱数の定義のアイデアを最初に出したのは M.Blum と S.Micali [1] であった．Luby の本では A.Yao [13] による定式化を紹介している．

さらに，A_n の時間計算量 (time comlexity) を $T_A(n)$ とするとき，

$$S_A(n) := T_A(n)/\delta_A(n) \tag{8}$$

とおく．擬似乱数生成法のクラスは時間計算量 $T_f(n), \ell(n), S_A(n)$ の増大度ごとに考えることができる．たとえば，$T_f(n), \ell(n)$ が多項式増大度で，すべての A に対して $S_A(n)$ が多項式増大度を超えるとき，f を「多項式時間の擬似乱数生成法」という[4]．

既存の擬似乱数生成法と計算量

三項間漸化式によるM-系列(5)の場合，「p 次均等分布性」(定理 2)によって，次元 p までの分布は事実上まったく完全で，非常に計算量の大きい検定にさえ採択されるはずだが，$(p+1)$ 次元では，わずか三項間の相関を調べる検定 — 系列を定義している漸化式(5)自身を利用するもので，きわめて小さな計算量の検定 — で棄却されてしまう．これでは，先に述べた意味では，三項間漸化式によるM-系列はよい擬似乱数とは呼べない．M-系列で項数を増やすなどして，定義式の相関が実用的な計算に影響を及ぼしにくいように，工夫することがよく行われているが，定義式自身を検定に利用して，やはり棄却できるから，先に述べた意味では相変わらず，よい擬似乱数と呼ぶことはできない．

確率数値解析の研究者からは，「系列の定義式自身を用いる検定にさえ耐えなければいけない，とは擬似乱数に対する要請が厳しすぎる」，という声が聞かれるかも知れない．計算量理論では「どんな強力な敵によっても破られない暗号」の開発を目指して擬似乱数を研究しているため，この分野での擬似乱数に対する要請は極めて厳しいのである．暗号を破るために敵は，当然，擬似乱数の定義式自体をも標的にするであろう．これを防ぐには，擬似乱数の種だけではなく定義式自体を秘密にしてしまうことも考えられようが，すると，その擬似乱数生成法を普及させることができなくなるので一般的な暗号生成システムとしては使用できなくなる．すなわち，「暗号化アルゴリズム (=擬似乱数生成法) は公開し，秘密の鍵 (=擬似乱数の種) でもって誰でも暗号は作れるが，鍵を知らない者が暗号文を解読することは事実上不可能である」ようにしたいのである．

応用上の問題において意見の分かれるところかも知れない．しかし，理論上，定義式を利用する検定を他の検定と区別して扱う合理的な理由はない．

それでは，擬似乱数がその定義式自身を利用した検定で棄却されないようにすることなど，可能だろうか．可能であると考えられる．力学系を利用した擬似乱数生成の場合，出力される数列から力学系の状態を再現するための手続きが複雑，すなわち大きな計算量が必要，であればよい (一方通行関数)[5]．M-系列の場合，この手続きの計算量が小さすぎるのである．これに比べて，無理数回転による方法では，逆手続きには相当大きな計算量が必要なので，定義式を利用した実施可能な検定は，現時点では，知られていない．

---

[4] 多項式時間の擬似乱数生成法の定義では $T_A(n)$ が多項式オーダーの関数 A のみを考えればよいことが分かる．この意味で検定のクラスを多項式時間のクラスに制限している．

[5] 一方通行関数が直ちに擬似乱数生成法に結び付くわけではないが，原理的には一方通行関数から擬似乱数生成法を導く一般的方法が知られている ([6])．

力学系による擬似乱数生成において具体的にいうと，(2) の写像 G をうまく (十分複雑に) 取って，生成される列 $\{Y_n\}$ から力学系の状態 $\{\omega_n\}$ を求める手続きが大きな計算量であるようにする必要がある．力学系 (1) の複雑性は恐らくあまり重要でない．

実は，そもそも M-系列の最大の特長である「p 次均等分布性」(定理 2) は，皮肉なことに，本来，目指すべきことではなかった．高次均等分布性は擬似乱数の持つランダム性を計算量の大きい分布に手厚く配分してしまうから，計算量の小さい分布がおろそかになる．すなわち，高次均等分布性を求める以上，計算量の小さい検定で棄却されてしまうのは止むを得ない．

なお，[5] では，線形合同法などのいくつかの擬似乱数生成法についての計算量の考察が行われている．

計算量による擬似乱数理論の問題

重大な問題は，「多項式時間の擬似乱数生成法が存在するか」という問が，実は，未解決であることだ．もし存在すれば，計算量理論における重要な予想「$\mathbf{P} \neq \mathbf{NP}$」を証明することができる．証明には至らないものの，多項式時間の擬似乱数生成法ではないかと予想されているアルゴリズムが幾種類も発表されている．しかしながら，それらのアルゴリズムをコンピュータ上で実現し確率数値解析の目的のために利用するのは，少なくとも現時点では，きわめて困難である．詳しくは [6] を見よ．

実際的な問題として，多項式時間の擬似乱数生成法が発見されても，それが実用に耐えるかどうかは直ちには分からない．多項式時間のアルゴリズム全体は理論上取扱やすいクラスではあるものの，実用的かどうかは個々のアルゴリズムに依るからである[6]．

このような事情から，擬似乱数の定義を何らかの方法で緩めて，その存在証明や性質についてある程度，把握できるようなものを探究すべきである，と考えられなくもない．

筆者の目下の興味は，力学系による擬似乱数生成における (2) の写像 G の複雑さと生成される数列の擬似乱数性の関連を知ることにある．具体的には，定理 4 のような確率過程の分布収束が，実際にはどのような有限次元分布から収束しているのか (計算量の小さい分布ほど早く収束しているかなど)，を調べたいと思っているのだが....

参考文献

[1] M.Blum and S.Macali, How to generate cryptographically strong sequences of pseudo-random bits, *SIAM J. on Computing*, vol. **13**, 850–864 (1984). A preliminary version appears in *Proceedings of the IEEE Foundations of Comput. Sci.* 1982, 112–117.

[2] 伏見正則，乱数，(東京大学出版会)，(1989)

[6] たとえば，$O(n)$ と $O(n^{100})$ はともに多項式増大度であるが，現実にはまったく異なったレベルの増大度である．

[3] 笠井琢美，計算量の理論，近代科学社，1987

[4] D.E.Knuth, *The Art of Computer Programming*, 2nd ed., Addison-Wesley, (1981), (邦訳)『準数値算法/乱数』(渋谷政昭訳)，サイエンス社，(1983).

[5] E.Kranakis, Primality and cryptography, *Wiley-Teubner Series in Computer Science*, John Wiley & Sons, Ltd., Chichester; B. G. Teubner, Stuttgart, 1986

[6] M.Luby, Pseudorandmness and Cryptographic Applications, Princeton Computer Science Notes, Princeton University Press, 1996

[7] J.H. Lutz, Category and measure in complexity classes. *SIAM J. Comput.* vol.**19-6**, 1100–1131 (1990)

[8] J.H. Lutz, Pseudorandom Sources for BPP, *J. Comput. System Sci.*, vol.**41**, 307-320 (1990)

[9] P.Martin-Löf, The definition of random sequences, *Inform. Control* **7**, 602-619 (1966)

[10] H. Sugita, Psuedo-random number generator by means of irrational rotation, *Monte-Carlo Methods and Applications*, vol.**1-1**, 35-57 (1995)

[11] 杉田 洋，無理数回転による擬似乱数生成，数理解析研究所講究録 915，数値計算アルゴリズムの現状と展望 II，(1995)

[12] R.C. Tausworthe, Random numbers generated by linear recurrence modulo two, *Math. of Comput.*, vol.**19**, 201-209 (1965)

[13] A.Yao, Theory and applications of trapdoor functions, *Proceedings of the IEEE Foundations of Comput. Sci.* 1982, 80–91.

杉田 洋
九州大学大学院数理学研究科 (工学部分室)
812-81 福岡市東区箱崎 6–10–1
E-mail: sugita@math.kyushu-u.ac.jp