数理解析研究所講究録
1032巻 1998年 148-151

Cellular Automaton 擬似乱数に対する Random Walk 検定

高嶋　惠三　　(大阪教育大学)

1 Cellular automaton による擬似乱数生成法

1次元 cellular automaton による擬似乱数生成法は, [10], [9] などににによって研究されているが, 一般に以下のように定式化される.

数列 $\{a_n, -\infty < n < \infty, 0 \leq a_n < k\}$ をある法則 ϕ によって並列に, 同時に次世代の数列 $\{a'_n\}$ に更新する :

$$a'_n = \phi(a_{i-r}, a_{i-r+1}, \ldots, a_{i+r}).$$

ここで, $k, r(>0)$ は整数とする. 以下では, $k = 2, r = 1$ の場合のみ考える. 実際の計算機上に実現するには, $\{a_n\}$ は有限列でなければならないので, S を正の整数として以下のような周期条件を付加する :

$$a'_1 = \phi(a_S, a_1, a_2),$$

$$a'_S = \phi(a_{S-1}, a_S, a_1).$$

さらに, 興味ある場合として [10], [1] 等で良い生成法として研究, 紹介されている2つの生成法 :

Type (I) $\quad a'_n = a_{n-1} + a_n + a_{n+1} + a_n a_{n+1} \pmod 2,$

或いは同等な,

$$a'_n = a_{n-1}\mathbf{XOR}(a_n \mathbf{OR} a_{n+1}),$$

Type (II) $\quad a'_n = 1 + a_{n-1} + a_{n+1} + a_n a_{n+1} \pmod 2,$

或いは同等な,

$$a'_n = a_{n-1}\mathbf{XOR}(a_n \mathbf{OR}(\mathbf{NOT} a_{n+1})),$$

について考察する. 但し, ここで **OR**, **XOR**, **NOT** はそれぞれ, 論理和, 排他論理和, 論理否定, である. これらは非線型な生成法であり, S が充分大きい場合 (例えば $S \geq 100$), 充分長い周期をもつ (cf. [10]). なお, [10] では **Type (II)** は **Type (I)** に比較してやや望ましくない生成法とされているが, 本報告では比較のために取り上げることにする.

2 Random walk 検定

前節の擬似乱数生成法に対して, [4] ~ [8] と同様に以下のような random walk の見本関数の汎関数に基づく統計的検定を試みる. 例えば, 見本関数の最大値による検定の方法は以下の通りである : L を正の整数とし. y_n は検定の対象の 0 , 1 の擬似乱数列とし,

$$x_n = 2y_n - 1,$$

とおく.

Step 1. (初期化) 対象となる, 擬似乱数列の配列を線型合同法を利用して初期化する.

Step 2.(見本関数の構成) x_n より, 長さ $2L$ の random walk の見本関数 $\{s_n, 0 \leq n \leq 2L\}$ を以下のように構成する :

$$s_n = \sum_{i=1}^{n} x_{i+2jL+2kNL}, \ s_0 = 0.$$

この見本関数の最大値

$$mx_{2L}^{j} = \max\{s_n : 0 \leq n \leq 2L\},$$

を求める.

Step 3. (χ^2 検定) Step 2 を $j = 0, \ldots, N-1$, に対して繰り返し mx_{2L}^{j} の経験分布を求める :

$$f_m = \#\{j : mx_{2L}^{j} = m\}, \ m = 0, 1, \ldots, 2L.$$

この経験分布に対して χ^2 検定を行いその値 χ_k^2 を計算する :

$$\chi_k^2 = \sum_{m=0}^{K} \frac{(f'_m - N\mu_{2L,m})^2}{N\mu_{2L,m}},$$

但し, $K = \max\{k : \Pr(MX_{2L} = k) > 10.0\}$, $\mu_{2L,k} = \Pr(MX_{2L} = k)$, $f'_k = f_k$, for $k < K$, and $\mu_{2L,K} = \sum_{k=K}^{2L} \Pr(MX_{2L} = k)$, $f'_K = \sum_{k=K}^{2L} f_k$. であり, MX_{2L} は見本関数の最大値である. ここで, χ^2 検定の自由度は K である.

Step 4. (Kolmogorov-Smirnov 検定) Step 3 を $k = 0, 1, \ldots, 29$, に対して繰り返し χ_k^2 の経験分布 F_{30} を求める. 自由度 K の χ^2 分布関数 $F(x)$ に基づき, F_{30} に対して Kolmogorov-Smirnov 検定を行い, その Kolmogorov-Smirnov 統計量 K_{30}^{+} , K_{30}^{-} を求める :

$$\begin{aligned} K_{30}^{+} &= \sqrt{30} \max_{-\infty<x<\infty}(F_{30}(x) - F(x)), \\ K_{30}^{-} &= \sqrt{30} \max_{-\infty<x<\infty}(F(x) - F_{30}(x)). \end{aligned}$$

Step 5. Steps 2-4 を 100 回繰り返し, K_{30}^{+}, K_{30}^{-} の値が 95 % ~ 99 % , に入る個数, 及び 99 % 以上となる個数を求める.

3 Maximum 検定の結果について

結果の一部を表 (1) に示す. これらの結果からすぐに分かることは, random walk の見本関数の長さ $2L$ がcellular automaton のセルの数 S より短い場合は比較的良い検定結果が得られるが, $2L \geq 1.2S$ の場合には極端に悪い結果が得られることである.

M系列擬似乱数や加算生成法などの場合も, random walk の見本関数の長さが特性多項式の次数より長い場合, 統計的に偏りが見られるが (cf. [4] ~ [8]), 表 (1) の結果はそれらよりさらに偏っており, cellular automaton 擬似乱数はrandom walk などの離散確率過程のシミュレーションには不適当であることが見て取れる.

最後に, cellular automaton 擬似乱数の研究者の一人S. Wolfram はすでによく知られているとおり, 著名な数式処理ソフト *Mathematica* の制作者である. また, *Mathematica* の発売元である Wolfram Research Inc. は *Mathematica* に組み込まれている擬似乱数生成関数のアルゴリズムを公表していないが, cellular automaton のアルゴリズムに基づく擬似乱数であるようだ. そこで実際に *Mathematica* の擬似乱数生成関数を用いてmaximum 検定を行ってみたが, 表 (1) のような検定結果は得られなかった ($2L$ は 20000 程度まで行ってみた). そこで *Mathematica* の擬似乱数は前節で述べた **Type (I)**, **Type (II)** とは別のアルゴリズムによるものと想像されるが, 正確なところは不明である.

参考文献

[1] Engel : Exploring Mathematics with your Computer, Mathematical Association of America, 1993.

[2] W. Feller : *An Introduction to Probability Theory and Its Applications*, **Vol. 1**, 3rd. ed. Wiley, (1968).

[3] D.E. Knuth : *The Art of Computer Programming*, **Vol. 2**, *Semi-numerical Algorithms*, 2nd ed. Addison-Wesley, (1981).

[4] K. Takashima : Sojourn time test for maximum-length linearly recurring sequences with characteristic primitive trinomials, J. Japanese Soc. Comp. Stat. **7**, (1994) 77-87.

[5] K. Takashima : Sojourn time test of m-sequences with characteristic pentanomials, J. Japanese Soc. Comp. Stat. **8**, (1995) 37-46.

[6] K. Takashima : Last visit time tests for pseudorandom numbers, *Journal of Japanese Society of Computational Statistics*, **9**, 1996, 1 - 14.

[7] K. Takashima : Random walk tests of additive number generators, "*Proceedings of the Workshop on Turbulent Diffusion and Related Problems in Stochastic Numerics*" (eds. S. Ogawa and K. Sabelfeld), Inst. Stat. Math., 1997, 55 - 65.

表 1: K-S Maxmum test, 100 samples.

Path length	Path number	K_{30}^{+}		K_{30}^{-}	
		95~99%	99% ~	95~99%	99% ~
Type (I), with 1000 sites					
400	50,000	2	2	8	0
500	50,000	7	2	4	1
600	50,000	8	2	2	0
1200	50,000	0	0	0	100
Type (II), with 1000 sites					
400	50,000	4	0	10	0
500	50,000	2	2	4	0
600	50,000	1	0	8	6
1200	50,000	0	0	0	100
Type (I), with 100 sites					
120	50,000	0	0	0	100
Type (I), with 127 sites					
160	50,000	0	0	0	100
Type (I), with 200 sites					
240	50,000	0	0	0	100

[8] K. Takashima : Random walk tests of reciprocal m-sequences, *Monte Carlo Methods and Simulations*, **3**, 1997, 155 - 166.

[9] S. Tezuka and M. Fushimi : A Method of Designing Cellular Automata as Pseudorandom Number Generators for Built-in Self-test for VLSI, *Contemporary Math.*, **168**, 363 - 367.

[10] S. Wolfram : Random Sequence Generation by Cellular Automata, *Adv. Appl. Math.*, **7**, 123 - 169, 1986.