2003－ＣＳＥＣ－22　(4)
2003／7／17

非線形写像の有限精度における一方向性から構成した擬似乱数生成法

渡辺栄治
メテオーラ・システム株式会社　〒259-1196 神奈川県伊勢原市石田200番地
E-mail: eiji@meteora.co.jp

あらまし：筆者の研究グループ（以下我々）は、非線形写像の有限精度における演算法を調査した結果、二つの写像間に計算困難な関係が存在することを発見した。この計算困難性とは、一方向性が確率論的な事象と関係し特定関数の定義が困難である特徴をいう。ゆえに、任意の一方向関数から擬似乱数生成器を作れるとする理論の適用には無理が在る[1][2]。この事態を打開する他の演算法を研究した。表題の生成法は、この計算困難な関係をパラダイムとする、新たな繰り返し演算法そのものである。この出力乱数から、初期値を求めることも、計算ラウンド単位の種を求めることも、計算量的に困難である：結局、この擬似ランダム性は、計算量的に予測困難、生成スピードは256ビット乱数7800個/second(P2-266Mhz_PC)に達する。パッド算出ソフトとして提供される。

キーワード：擬似乱数生成法、計算量的に予測困難(*Computationally unpredictable*)

Method of the generating pseudorandom numbers composed of one-way functionality of A non-linear mapping in the limit precision

Eiji Watanabe METEORA-SYSTEM Co.Ltd

Abstract: A research team of the author has been exploring an operation method in the limit precision of a non-linear mapping, and discovered a relationship of computational difficulty between mapping functions of two kinds. This computational difficulty is characterized with that one-way functionality is related to probabilistic events but never related with a specific function, to which such a theory [1][2] is also never applicable that a pseudorandom generator is made of any one-way function. Another method of iteration operation has been researched in order to break through it, which leads to the new iteration operation on the paradigm of this computational difficulty, that is our pseudorandom generator to be reported here. It is difficult from those pseudorandom numbers to seek the initial value or a seed of every computational step by means of any computational power. Consequently, the pseudo-randomness features *Computationally unpredictable*. Its generating speed of 256bit random number attains 7800 numbers /second in PC of P2-266Mhz, which software is offered as Pad Calculation Software.

Key words: Pseudorandom numbers generating method, computationally unpredictable

1．まえがき

筆者は、研究者というよりも実務家として、現場の悪戦苦闘や議論の材料を学会へ報告することが精一杯な所と心得ている。擬似ランダム性をターゲットにすることは、本来、デリケートな議論を必要とするが、核心部分を伝えるために必要と思われる基準を妥当な範囲で定義し、議論の方向を絞ることにした。基準とは：①計算量的な予測困難性、②情報エントロピーの場、の二つである。付録にこの概念の定義を載せた。これ以外の基準が無意識の内に議論に入り込まないように留意し、連続量と有限精度との区別にも留意した。

2．関数の定義

2．1．基礎アルゴリズムと有限精度における演算

L(x)，H(x)は、共に効率的に計算可能な関数とする。どちらも2nビットの種が入力されると、2nビットの出力を与える。そのアルゴリズムの基礎は、本来連続量で定義されているロジステイック写像を有限精度における多重精度演算に変換したところのアルゴリズムである。まず、連続量において定義される基礎アルゴリズムは：

Y = 4X (1 − X)，通常、Y ——> X へ再帰演算する， 0 < X, Y < 1 ———— (1)
これを有限精度の演算に移行させる。すなわち、X を有限精度 2n ビットの値とすると、次のように、Y(x)は 4n ビットに拡張される整数値となる：$Y(x) = 4x(2^{2n}-x) = \{0,1\}2^{4n-1} + \{0,1\}2^{4n-2} + \cdots + \{0,1\}2^{3n-1} + \cdots + \{0,1\}2^{2n-1} + \cdots + \{0,1\}2^{n-1} + \cdots + \{0,1\}2^{0}$ ———— (1)' ただし、{0,1}は1ビット列を表す。

これに対して、以下のような、有限精度に対してしか定義できない、二つの異なる演算を導入する：

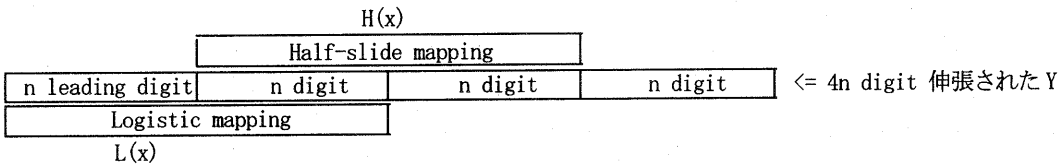

図1：L(x), H(x)の定義
Fig1: Definition of L(x) and H(x)

（1）L(x)演算の定義

x に初期値 s を与え(x = s)、4n ビットの Y(s)に伸張し、その値から、先頭 2n ビットの値を抜き出す操作を関数 L(s)と記号化する。L(s)は、ロジステイック写像の有限精度における演算である。今、2n ビットを 256 ビットにすれば、先頭の 256 ビットが L(s)の値になる。ロジステイック演算法という。

（2）H(x)演算の定義

x に初期値 s を与え(x = s)、4n ビットの Y(s)に拡張し、先頭と末尾の n ビットを捨てる、この操作を H(s)と記号化する。初期値 s を 256 ビットにすれば、やはり、256 ビットが関数値となる。H(s)をハーフスライド演算法という。

（3）L(x)と H(x)の比較

どちらの演算も、所謂「開いて閉じる」カオスのメカニズムを有限精度に持ち込む操作である：開いて閉じる演算(1)式は、連続量にはこれしか無いが、有限精度演算では複数存在するということが、私達の発見であった[3]。図1を注意して見ると、L(x)には、H(x)の後半 n ビットの値が欠落する一方、H(x)では、L(x)の先頭 n ビットの値が欠落する、という対等な関係に在る。しかし、L(x)と H(x)をそれぞれプロットしてみると、図2に示すように、歴然と違いが現れてくる。

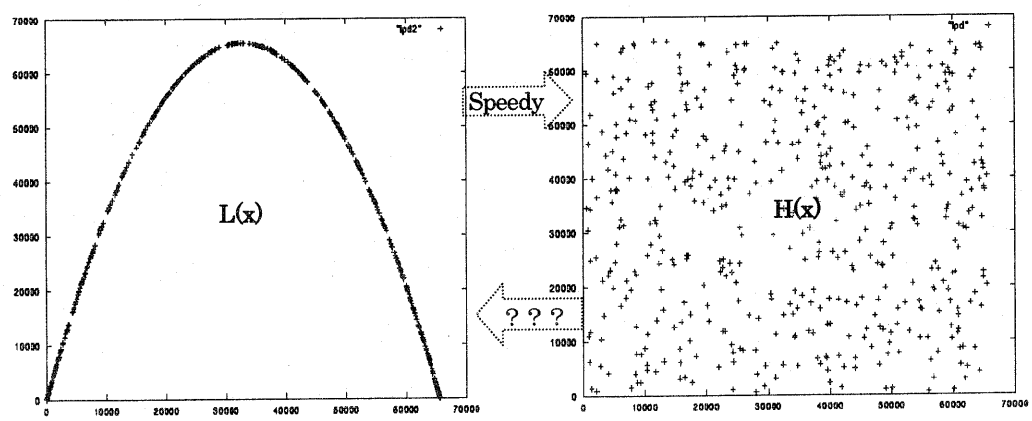

図2：L(x)と H(x)の繰り返し演算軌跡（j-1, j）
Fig 2: The trajectory of iteration operation of L(x) and H(x) in (j-1, j) pairs

図２の左図は、L(x)演算を $L^j(s) \equiv L((L^{j-1}(s)))$ のように繰り返し、その(j-1, j)の対をプロットしたものである。同右図は、同じく、H(x)演算を $H(L^j(s)) \equiv H(L^{j-1}(L(s)))$ のように繰り返し、その対をプロットしたものである。これは、直感的な理解を助ける以外に他愛はないが、見通しを良くする。すなわち、図２の左図では、この中の任意のL(x)値が与えられた場合を想定すると、その前後の値 $L^{j-1}(s)$ と $L^{j+1}(s)$ を決定する計算式が存在しているように見える。一方、右側のH(x)には、任意の一個からその前後の値 $H(L^{j-1}(s))$ $H(L^{j+1}(s))$ を計算量で決められるだろうか、という疑問が出てくる。

実際、次のような違いを確認している：図中、任意の値が与えられた場合、ロジステイック演算法 L(s)では、一つ前の値 $L^{j-1}(s)$＝計算ラウンドの種 を計算量で決定できる：s⟶L(s)，L(s)⟶ L(L(s))，及び、この逆、s⟵L(s)，L(s)⟵ L(L(s))が計算可能である。一方、ハーフスライド演算法 H(s)では、任意の値が与えられても、そのH(s)から一つ前の値 $H(L^{j-1}(s))$ あるいは初期値 s を決定する解法を発見していない。これがH(s)演算とL(s)演算との決定的な違いである。以下、詳細に議論する。

２．２．二つの演算間に現れる計算困難な関係

s からH(s)を、及びL(s)からH(L(s))を求める計算は大変効率良く、逆に、H(s)から初期値 s、及びH(L(s))から種L(s)を求める計算には、その関数と解法が発見されていない。二つの演算の間に一方向関係が現れる。

（１）H(s)演算系列の中にも在る計算困難な関係

ハーフスライド演算法H(s)の系列の間にも一方向関係が在る、例えば、H(L(s))からH(s)を求めるには計算量的に困難がある。この理由は上記2.2節から派生する。何故なら、H(s)とH(L(s))の関係を初期値 s から見ると、L(s)が求まれば初期値も求まる：s⟵L(s)。しかし、H(L(s))から⟶ラウンドの種L(s)を求める計算が困難であるから、s⟵L(s)のパスから決まる s が求められない。初期値 s⟶L(s)，s⟶One-Way⟶ H(s)のパスが成立しないから、H(L(s))から一つ前の H(s)を計算量で決められない、となる。しかし、複数の H(s)、H(L(s))が与えられた場合、どうか？　これには、次のような指摘が有った。

（２）H(s)の軌道値が２個連続して公開された場合

H(s)の値が２個連続して公開された場合、その種を現実的な計算量で決められる、という指摘があり[脚注1]、それを確認した。この数列を乱数として使う場合、公開しない環境下に置くか、ハッシュ関数を導入して、情報エントロピーの落差を設け、256bit H(s)乱数を128bitにして利用するなどの対策が要る。

３．計算量的に予測困難な擬似ランダム性の構成法

前章2.2.節に紹介した一方向関係に着目して、計算量的に識別困難な擬似乱数生成法を研究した。当初、二つの関数の一方向関係に対して「繰り返しパラダイム[1][2]」を適用することを考えたが、2.2 節のような問題もあり、このパラダイム[1][2]の適用には無理があった。そこで、別途、前章2,2節をパラダイムとする繰り返し演算を調査した結果、計算量的な予測困難性を議論できる構成法を得た。そこにはロジステイック演算とハーフスライド演算が繰り返し出てくるので、その演算の種類を識別する記号、[UC]、[HC]、[RC]、[SC]を導入して説明する。前章2.2節パラダイムが[UC]と[HC]で表現されている。まず、初期値 s に始まり：

J ＝ 0		1		2		3	…	j	
[UC]: s	⟶	L(s)	⟶	L(L(s))	⟶	L(L(L(s)))	… ⟶	$L(L^{j-1}(s))$	——— (2)
[HC]: s	⟶	H(s)	⟶	H(L(s))	⟶	H(L(L(s)))	… ⟶	$H(L^{j-1}(s))$	——— (3)
[RC]: H(s)	⟶	L(H(s))	⟶	L(H(L(s)))	⟶	L(H(L(L(s))))	… ⟶	$L(H(L^{j-1}(s)))$	——— (4)
[SC]: H(s)	⟶	H(H(s))	⟶	H(H(L(s)))	⟶	H(H(L(L(s))))	… ⟶	$H(H(L^{j-1}(s)))$	——— (5)

ただし、Jは計算ステップ数を表す、J = 1, 2, 3, , , n、及び、$L^j(s) \equiv L(L^{j-1}(s))$，$L^0(s) = s$

さらに、式(5)の出力を、ステップＪの奇数と偶数ごとに分離し、偶数の関数を[SC]_1、奇数の関数を[SC]_2 とする。式[SC]_2 出力の一般式は次のようになる：

[脚注1]：平成12年10月、IPA擬似乱数部門公募におけるスクリーニング結果、「全数検索以外の攻撃が考えられないことを理由とした予測不可能性」の主張に対して下された評価：ご担当された方に大変感謝しています。

$$[SC]\_2 \quad = \quad \underset{J=1}{H(H(s))} \cdot \underset{3}{H(H(L^2(s)))} \cdot \underset{5}{H(H(L^4(s)))} \cdots \underset{j\ (奇数)}{H(H(L^{j-1}(s)))} \quad \text{-------- (5)'}$$

表1：計算量的に予測困難な乱数の算出システム

Table1: Calculation system of computationally unpredictable random numbers

一方向性関係／計算ラウンド			Iteration paradigm		
J	L (x) [UC]	H (x) [HC]	L (x) [RC]	H (x) *Computationally unpredictable*	
				[SC]_1	[SC]_2
0	**s**	H(s)	H(s)	H(s)	
	⇓				
	4s(1-s)				
	⇓				
1	L(**s**)	H(L(s))	L(H(s))		H(H(s))
	⇓				
	4L(s)(1-L(s))				
	⇓				
2	L(**L(s)**)	H(L(L(s)))	L(H(L(s)))	H(H(L(s)))	
	⇓				
	4 L(L(s))(1- L (L(s)))				
	⇓				
3	L(**L(L(s))**)	H(L(L(L(s))))	L(H(L(L(s))))		H(H(L(L(s))))

一様に選択される初期値 s に対して、ハーフスライド演算[SC]_2 が一方向関数である。何故なら、[RC]から[UC]を求める過程には、まず[HC]から[UC]に至る計算困難が在る、例えば、L(s) <── difficulty <── H(L(s))のようにH(L(s))からL(s)への計算困難が存在するから、[RC]のL(H(L(s)))を推定できたとしても、そこから──> H(L(s))──> L(**s**)に至るパスが成立しない。また、ハーフスライド演算[SC]_2 には連続した軌道値が無いから、ロジステイック演算[RC]を求める計算も困難である。ゆえに、演算[SC]_2 出力から、その種を求めるパス、H(H(L(s)))──> L(H(L(s)))──> H(L(s))──> L(**s**) は計算困難であり、[SC]の種、すなわち[UC]を求める計算そのものが成立しない。ゆえに、この[SC]_2 の出力は、計算量的に予測困難な乱数となる。

4．計算困難性の背景

L(s)からH(L(s))を求める計算は効率良いが、H(L(s))から種L(s)を決める解法を発見していない。この背景について報告する。連続量ではロジステイック演算もハーフスライド演算の区別も無い。故に、連続量でのカオスの性質が有限精度でも成立していると主張するのはナンセンスである。したがって、計算の困難性を考察するに際し、研究対象が全く違うものとして扱わねばならない。ただ、連続量における一方向性について、後の議論のために、以下、紹介する必要がある。

$X_n = 4\ X_{n-1}\ (1 - X_{n-1})$, $0 < X_n < 1$, 初期値= s: を連続量の計算式とする。X_nからX_{n-1}を求めると、解が二つ現れる：$X_{n-1} = \{1 \pm \sqrt{(1-X_n)}\}/2$. したがって、逆方向には1ビットの情報エントロピーの落差が有る [3]：(+) or (-)の1ビット。今、初期値=X_0からn回演算した後の軌道値X_nを固定し、そこから初期値を推定する問題を考える。これには逆演算をn回行なう必要がある。逆演算過程にはnビットの情報エントロピーの落差が生じ、それ

は２"個の発散解が得られる。このような情報エントロピーを供給できる原因は、演算の場が連続量であるからである。そして、明らかに、情報エントロピーが計算困難性の原因である。

このような議論が有限精度で許されるか研究を要するが、少なくもハーフスライド演算のヒントを上記のような事情から得たという経緯がある[3]。すなわち、初期値ｓを秘密とするシステムを考えると、初期値ｓを暗号鍵にしないで、ロジステイック写像の初期値ｓからｎ回演算した後の軌道値X_nを暗号鍵に使う。それにはｎ回の演算時間がかかるが、この時間を１回の演算時間に短縮しようと意図した、それがH(x)演算である。図１にて定義したように、ロジステイック演算の先頭ｎビットの値を落としてH(x) 演算を作った、そうした理由は、関数値H(x) から初期値に向かって（逆演算方向に）情報エントロピーがｎビット現れることを期待したのである。もちろん、こういう議論が出来るのは連続量の場合だけであることは言うまでもない。演算対象が整数のみ、整数が有限の個数しか供給されない計算機においては、別の観点（軌道解析）からも考える必要があるが、少なくも、ハーフスライド演算の種はロジステイック演算の逆演算プロセスの中に存在する、という絞り込みができる。

４．１．軌道解析

線形合同法に、軌道（繰り返し演算値の系列）の種類という概念を導入すると、全ての整数群が結晶格子点を構成し、単純に１種類である。なぜなら任意の１点が与えられると、ぞの前後の整数値を一意に決定できる。任意の一点における情報エントロピーはゼロである：情報エントロピー = zero 場と考える。図３における問題は、何でも、効率よく、確率１で推測可能になる。

ロジステイック演算の軌道はこんな単純ではない。筆者のチームは多重精度演算プログラムを使って、1byte、2byte の整数空間に演算H(x)とL(x)を定義して、詳細にチェックした。結果を、以下の表２、表３、図４に示す。

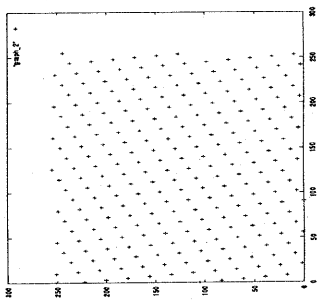

図３：法を 256 とした結晶格子点

表２：8bit 整数空間の解析図

Table2: Analytic arrangement of 8bit-integer space

① 全ての整数を初期値にした結果：

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
00-0f	00	0a	0a	0b	0a	0a	0a	0c	0a	0a	0a	0c	0a	0a	0a	0b
10-1f	0a	0a	0a	0b	0a	0a	0a	0c	0a	0a	0a	0c	0a	0a	0b	0a
20-2f	0a	0a	0b	0a	0a	0a	0b	0a	0a	0c	0a	0a	0a	0b	0a	0a
30-3f	0a	0c	0a	0a	0b	0a	0a	0a	0c	0a	0a	0b	0a	0a	0a	0c
40-4f	0a	0a	0b	0a	0a	0a	0b	0a	0a	0c	0a	0a	0b	0a	0a	0a
50-5f	0b	0a	0a	0c	0a	0a	0b	0a	0a	0c	0a	0a	0a	0b	0a	0a
60-6f	0c	0a	0a	0b	0a	0a	0b	0a	0a	0b	0a	0a	0b	0a	0a	0c
70-7f	0a	0a	0b	0a	0a	0b	0a	0a	0c	0a	0a	0b	0a	0a	0b	0a
80-8f	0b	0a	0a	0b	0a	0a	0b	0a	0a	0c	0a	0b	0a	0a	0b	0a
90-9f	0a	0b	0a	0b	0a	0a	0b	0a	0b	0a	0a	0c	0a	0c	0a	0a
a0-af	0b	0a	0c	0a	0b	0a	0a	0b	0a	0b	0a	0b	0a	0a	0c	0a
b0-bf	0b	0a	0b	0a	0b	0a	0b	0a	0c	0a	0b	0a	0a	0c	0a	0c
c0-cf	0b	0a	0c	0a	0b	0a	0b	0a	0b	0a	0b	0a	0c	0b	0a	0c
d0-df	0a	0b	0b	0a	0b	0a	0b	0b	0a	0b	0b	0a	0c	0b	0a	0c
e0-ef	0b	0b	0a	0b	0b	0b	0c	0a	0b	0b	0b	0c	0b	0b	0c	0b
f0-ff	0c	0b	0c	0b	0b	0b	0b	0b	0b	0c	0b	0b	0c	0c	0b	00

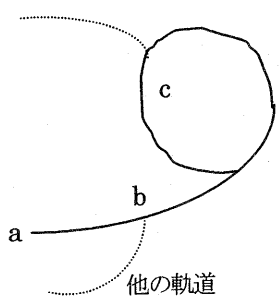

図４：整数が３種類に分類される

Fig4: There three kinds of integer

count_a=142

count_b= 77

count_c= 35

② 全ての整数を初期値にし、その軌道に含まれる周期解の周期-->最大周期
解析図、紙数の都合で省略する。

③ 全ての整数を初期値にし、自分の軌道に辿りつくまでの軌道点の個数—>軌道の平均の長さ
解析図、紙数の都合で省略する。

表３：多重精度演算プログラムによる解析結果

Table2: Analysis based around a calculation program of multi-precision

整数の空間	独立点ａの個数	合流点ｂの個数	周期点ｃの個数	平均の軌道長	最大周期
08bit=254	142	77	35	20.37	23
16bit=65534	36862	28564	108	276.24	104
N bit space	1.125 * 2^(L-1)	0.875 * 2^(L-1)		～2^(L/2) ～	～2^(L/2)/2～

(8bitの場合：0.00, 0.ff, 及び16bitの場合、0.0000, 0.ffffの２点をそれぞれ除く)

（１）３種類に分けられる整数

全ての整数を初期値にして演算させた結果が表２である。これによると、整数値は、独立点ａ、合流点ｂ、周期点ｃという３種類の整数に分類される。独立点ａという初期値には、他の初期値に始まる軌道が合流しない。一方、合流点ｂ、周期点ｃには、それを初期値にしても、そこへ他の軌道が合流する。この様子を図４に示した。表３は、表２_①_②_③のデータの集約結果である。これを詳細に見てゆく。

（２）最大周期解、および平均の軌道長

まず、我々は最大周期解に関心がある。これに関する研究を探した結果、唯一、見つかった報告[4]は、計算機固有の４捨五入演算などを利用した解析である。この報告に拠れば、Logistics Loopに含まれる軌道値の個数は、10^(L/2)に比例するという：Lは10進桁数である。２進の場合も同様に、周期解は2^(L/2)に比例するという仮説が成り立つ；Lは２進桁数。我々の解析によれば、報告[4]の指摘する最大周期解は、表１における軌道の平均の長さ（一つの軌道に含まれる整数の個数の平均）に相当する。実際の最大周期解は、それより若干小さく、～2^(L/2-1)個に比例する。

（３）周期点ｃの個数

極めて短い周期解、１，４，５周期が存在し、8bit空間では、相対的に大きい比重を占める：周期点の個数にも大きく効いている。いずれにしろ、周期点ｃの個数は、最大周期解（～2^(L/2)/2～）のオーダと予想する。

（４）独立点ａの個数、及び軌道点ｂとｃの合計値

独立点ａの個数を表３から評価した：独立点ａの個数 = 1.125 * 2^(L-1) である。
試しに、8bitでは、126*1.125 = 141.75となり、16bitでは、32768 * 1.125=36864、となり、それぞれ実測値の142, 36862と比較して、極めて正確である。これは全整数空間の半分強に相当する。ここから軌道点ｂとｃの合計が与えられる = 2^(L) - 1.125 * 2^(L-1)=0.875 * 2^(L-1)。実測値を入れると、= 0.875 * 32768 = 28672 = 28564 + 108、となり、これも、驚くほど正確である。この周期点108個に比べ、合流点ｂの個数28564が圧倒的に大きいことに注目している。これは、周期点から枝分かれ分岐する「発散の姿」を予想させるからである。

４．２．情報エントロピーの場、発散の場

全ての整数値がL(x)演算によって３種類に分類されることを示した。また、周期点に比べ、合流点ｂの個数が圧倒的に大きい：16bitで、既に、108対28564である。周期点から逆演算すると軌道が枝分かれ分岐してゆく「発散解の姿」を裏付ける。言い換えると、全ての軌道は周期解に落ちて行く。他に例が見当たらない現象だ。

この発散の整数場を実測値で確認してみた。アルゴリズムL(x)で定義された整数空間では、任意の整数値ｔを指定すると $(2^{2n}-t)$ の整数値が必ず見つかる。ゆえに、ｂ点の値 = t(2nビット)とすると、$(2^{2n}-t)$ の値を持つ別の整数値が存在し、二つの軌道値が $4t(2^{2n}-t) = Y$ の形で4nビットに伸張される。この4nビットにL(x)演算が作用すると、2nビットのロジステイック演算値になるし、同じく、H(x)演算を行なうと、ハーフスライド演算

値になる。同じことが、周期点ｃにも言える。

ここから、1byte整数空間の中の軌道値eb（16進）の逆演算を考える。直前の軌道値ｔ＝a2と5dを逆演算から割り出したとする。4nビットに伸張されたYの値＝$4t(2^{2n}-t)=4(a2)(2^{2n}-a2)=4(a2)(5d)$＝eb68に戻ることができる。点ebは、写像L(x)により68が切り捨てられた結果に他ならない。H(x)演算では中ほどの整数値b6を取り出す。結局、異なる二つのH(x)軌道もここで一致する。次に、軌道値ｔ＝a2と5dから同じように逆演算してやれば、それぞれ4nビットに拡張されたY(a2)、Y(5d)の値に戻る。ここから先、計算機の演算対象が2^2個に増える。計算機はどこで停止するか？ ハーフスライドH(s)演算の種ｓに到達した所で停止するか？

発散の整数場に対する計算機の停止プロセスを考える：① $4t(2^{2n}-t)$の形に書ける整数解ｔが見つからないとき停止する。これは独立点ｔ値と独立点(1-t)値に到達した時である。ほぼ1/2の確率でこの停止イベントになる。独立点には、他の軌道が合流して来ないから、ここにはY(4nビット)が存在しない。他に1/2確率停止イベントが有るはずだ：② 出発点と同じ整数値に合流したとき停止する。③ 初期値を総当りにしてランウドの種を確定したとき、NEXTへ進む。例えば、H(s)＝b6の場合、Y＝eb68を見つけ出し、このランウドの種ｓ＝a2と$(2^{2n}-t)=5d$を確定し、H(s)＝b6になるかどうか、チェックする。否なら、演算対象をt=a2と$(2^{2n}-t)$へ広げNEXTへ進む。この場合、総当り数は8ビットである。整数空間がｎ倍になると計算量は指数関数的に増大する。他に、計算機の停止プログラムを書けるだろうか？

４．３．種ｓの確率論的な推測に関する

計算機の1/2確率停止イベント③を考える。合流点ｂと周期点ｃには1bitの情報エントロピーの場が存在する、合流点ｂは全て周期解に落ちて行く。また合流点ｂからの逆演算は比較的独立点に到達し易い。そこで周期解に始まる発散の場を考える：周期解の点ｃの総数は概算～2^(L/2)個である。そこに合流する軌道点は、4nビットのYを構成する点ｂとｃである。周期解の一点から枝分かれ分岐してゆく「発散解の整数」の個数は幾らか？

周期解の個数で合流点ｂと周期点ｃの総数を割り算すると、0.875＊2^(L-1)／2^(L/2)＝0.875＊2^(L-1)／2^(L/2)＝0.875＊2^(L/2-1)個である：L＝256bitにすると、ほぼ127bitとなる。

言い換えると、任意の一点から、ハーフスライドH(s)演算の種ｓを探す過程をクログラムすると、その候補者が2^127bit個以内に絞り込めるということ。すなわち、総当り数と比較して計算量を2^129回も減らせる。

どのような確率で種ｓの推測が可能になるか？ 初期値を一様に選んだ時、それが独立点ａか、合流点ｂと周期点ｃなのか、その軌道解析は32bit整数を超えた辺りで困難になる。初期値は、ほぼ1/2の確率で独立点ａ、1/2の確率で軌道点ｂと点ｃである、と言うのみである。これを、種ｓを探す攻撃者から見たとき、H(s)の種は、128bit[=127+1]の情報エントロピーの中に存在すると見える：256bit整数空間の半分に相当する。

４．４．ロジステイック演算[UC]の確率論的な推測

ハーフスライド演算[SC]_2の種は、ロジステイック写像[RC]であり、さらに[RC]演算の種は[UC]演算のL(s)である。前の2.2節に述べたようにL(s)⟶H(L(s))の方向の計算は大変効率良く、逆にH(L(s))から種L(s)を求める計算には、その関数と解法が発見されていない。計算機は、二度、この計算困難なパスに出会う：L(s)⟶One-Way⟶H(L(s))。しかし、[UC]演算を推測する手は有る。上記考察から、H(s)の種はH(s)の値から見て128bit[=127+1]の情報エントロピーの中に存在すると見えるから、計算機がH(s)の値からスタートして、コイン投げを128回行なえば、1/2^128の確率で正解を得る。

ある種の関数は実質1/2より高い確率で推測するのが困難であるとする理論[1][2]と比べて、どちらが厳しい基準か、この議論が妥当か？ 周期解に比べ「発散の場」が桁違いに大きいことから、このような議論に至った。

線形合同法では、独立点、合流点、周期点などの区別が無い：任意の一点における情報エントロピー＝zero場であるから、確率１の推測ができる。また、同等の周期解が二つ存在し、互いに独立である整数の場が有ったとしたら、情報エントロピー＝１ビット場である。この場合なら、1/2の高い確率で対象の推測が可能になる。このように「情報エントロピーの場」の存在が計算困難性の原因に間違いないが、なお定量的な解析が望まれる。

以上、初等数学の範囲で有限精度の軌道解析を行い、計算量的な予測困難性の議論を報告した。

5．公開可能な擬似乱数、公開パッド

我々の関心事は擬似ランダム性に関するが、それは統計的検定にパスすることや大きな線形複雑度を持つことを意味するのではなく、自然乱数の性質と計算量的に識別困難な性質に関するものである。予測困難と識別困難とはイコールであるとする報告も在るが[1]、そこまで議論を尽くせなかった。識者のご教示を期待している。

この擬似乱数生成法はアルゴリズム公開の暗号系である。その出力n個は、初期値依存の、公開可能なデータとして扱える。そういう適用も視野に入れて、実用的で、攻撃に強い、パッド算出ソフトウエアとしての提供を考えている（当ソフトウエアの試用可能）：

秘密情報：初期値s、

UC(J)，計算ステップJ = parameter 1,2,3,,,,,n

パッド(Pad) = SC_2(J) 、

パッド算出ソフトをどのように運用するかは上位システムが決める。もし、①One-Time Pad暗号として運用する場合、パッドを公開しないで、パラメーター(J)を公開し、これをパッドの配送に利用する。ある種の通信方式もこのカテゴリーに入る。②パッドを初期値依存の、公開データとして扱うことが可能となった。公開データ部をパラメーター(J)とパッドSC(J)_2にする一方、初期値sとUC(J)は秘密の扱いになる。この公開パッドは、管理すべき秘密情報を激減させたいとするセキュリテイ問題とか、ゼロ知識対話証明プロトコル[5]などに役立つ。

6．謝辞

筆者の研究グループとは、それぞれ別世界の住人である。ご芳名を載せる許可を下さった方々の役割を述べ、感謝の意を表明させて頂く。大阪大学・藤原融教授に助言を頂いている。技術士・武弘司氏は多重精度演算プログラムと統計検定を受け持っている。今回、計算量の問題に絞り統計検定を割愛した。パッド算出プログラムのバグつぶしに、武氏とは別に当社スタッフ関口浩が開発に関わり、照合確認をした。IN4S社山田社長・戸嶋両氏のご教示、他、参考書の著者、ご芳名を伏せた方々、多くの方々に感謝します。

2003年7月17日

参考文献：

[1] Oded Goldreich: “Modern Cryptography, Probabilistic, Proofs and Pseudo-randomness” 「現代暗号・確率的証明・擬似乱数」岡本龍明他訳、シュプリンガー・フエアラーク東京株式会社、東京、2001

[2] J Hastad, R. Impagliazzo, L. A. . Levin and M. Luby: ″Construction of Pseudorandom Generator from any One-Way Function″ In the 21st ACM Symposium on the theory of Computing, pages 12-24, 1989 and pages 396-404, 1990.

[3] Patent Corporation Treaty /JP99/00476: 1999.02.04
“IP KEY MANAGEMENT MECHANISM WITH DIVERGENCE BARRIER INCREASING ENTROPY AGAINST COMPUTATIONAL CRYPTO-ANALYSES”

[4] Anastasios A. Tsonis, Department of Geosciences University of Wisconsin-Milwaukee, Milwaukee, WI 53201: “The Effect of Truncation and Round-off on Computer Simulated Chaos Trajectories”, Computers Math. Applic. Vol.21 No 8 pp93-94 1991

[5] T. Wu. The Secure Remote Password Authentication and key exchange protocol. Stanford University 2000. RFC-2945

付録：①計算量的に予測困難なサンプルの定義：一様に選ばれた種と伸張関数から生成されたビット列を複数部分に区分けした場合、その一つのビット列から他のビット列を計算量で予測することが困難、及びその初期値や種を計算量で推定することも困難。この際、伸張関数と構成法は公開される。②情報エントロピーの場：整数空間において計算機が演算対象を選択する情報に不足する。

4．2．節の8bit実測値

j	4X(1-X)	L(x)	H(x)	4X(1-X)	L(x)	H(x)
0		02		0500	05	
1	07e8	07	7e	1388:	13	38
2	1b20	1b	b2	4610	46	61
3	6030	60	03	ca58	ca	a5
4	ee80	ee	e8	a748	a7	74
5	3f38	3f	f3	e5a0	e5	5a
6	bd00	bd	d0	5d08:1-5d=a2	5d	d0
7	c2e8	c2	2e	**eb68**=4(5d)(a2)	**eb**	**b6**
8	b8e8	b8	8e			
9	cc20	cc	c2			
10	a290:1-a2=5d	a2	29			
11	**eb68**=4(a2)(5d)	**eb**	**b6**			
12	4970	49	97			
13	cf98	cf	f9			
14	9b40	9b	b4			
15	f230	f2	23			
16	3128	31	12			
17	9db8	9d	db			
18	f068	f0	06			
19	3840	38	84			
20	ae20	ae	e2			
21	dc38	dc	c3			
22	7850	78	85			
23	fd20	fd	d2			
24	07e8	07	7e			