有限体の擬似乱数への応用

松本　眞

平成 22 年 11 月 8 日

目 次

1　擬似乱数

1.1　擬似乱数とは

擬似乱数とは：サイコロを振って得られたかのようなでたらめな数を次々につくる方法、またはそうして生成された数列を指す。[1]

擬似乱数を用いる目的は、大きく分けて二つある。

1. モンテカルロ法

2. 暗号乱数

である。モンテカルロ法とは、確率現象をシミュレート（模倣）するために擬似乱数を用いることである。たとえば、多数の原子核の分裂と連鎖反応、株価の変動のシミュレーションなどがあげられる。また、高次元空間上定義された関数の積分値を近似するのに、その空間内に点を一様ランダムに発生させてそれらの点での関数の値の和をとることも、モンテカルロ法による積分として知られている。この目的で使用される乱数は**モンテカルロ法用乱数**と呼ばれている。

もう一つは、第三者に漏れても内容が解読できない暗号を生成するために乱数を用いることで、**暗号乱数**と呼ばれている。

この両者の要請は微妙に違い、それぞれに適した擬似乱数発生法がある。この書物では、擬似乱数といったらモンテカルロ法用擬似乱数を指す。暗号乱数について記述するときには、その都度暗号乱数と明示する。

[1]「かのような」では数学的定義ではない。現時点では、計算量を用いた暗号論的定義（おおまかに言えば、生成は多項式時間でできるが、数列の一部から他の部分を推測するのは多項式時間ではできないような数列）が妥当な数学的定義と思われる。

1.2 擬似乱数への要請

モンテカルロ法用擬似乱数は、次のような要請を満たす必要がある。

- さいころで言えば１から６までが等確率 (一様) で、数列中の数が互いに独立になっているように見える。（確率・統計的側面）
- 高速性（計算機的側面）

  計算機での確率シミュレーションをするのに使われる際には大量（何百億個）の乱数を使うことも多い。核反応のシミュレーションでは全計算時間の半分近くが乱数発生に消費されている例もある。
- 再現性（計算機的側面）

  もう一度、同じ条件で実験を繰り返したい（例：他のグループによる追試）ことがある。特に、シミュレーションをしつつ何かを最適化したい（たとえば、ある町での車の信号待ち時間の平均値を最小化したい）場合には、同じ乱数列に対してパラメータ（信号の変わる時間など）を増減して影響を見るほうが、毎回違う乱数を用いるよりも効率的なことがある。

これらの要請に対する一つの答として、

有限代数系において漸化式で数列を定義し、生成する

ことが提唱され、今日も広く使われている。通常、擬似乱数といったらこれを指す。

漸化式による擬似乱数の利用は、1943 年ごろフォン・ノイマンが始めたとされる。彼は非線形漸化式を用いた。が、性能は余り良くなかった [4]。大きな成功は、Lehmer による線形合同法であった（60 年代）。

1.3 線形合同法

定義 1.1. (線形合同法) M, a, c を自然数の定数とし、

$$x_{j+1} := ax_j + c \mod M \tag{1}$$

で、x_0 を適当に選んで（初期値と呼ばれる）数列を作る。ここで、 $\mod M$ は M で割った余りを取ることを表す。

このようにして数列を生成し、擬似乱数として使うことを線形合同法 (Linear Congruential Generator, LCG) という。

例 1.2. $M = 7, a = 3, c = 0$ とする。$x_0 = 4$ ではじめると x_j は $4, 5, 1, 3, 2, 6, 4, 5, \ldots$ と周期６で循環する。

注意 1.3. $c = 0$,$x_0 = 0$ だと、x_j はずっと 0。

レポート問題 1. $4/7$ の小数点展開の筆算と、上の計算との間の関係を解明せよ。

一般の場合も周期的になり、周期は M 以下。周期 M を実現できるようなパラメータがある。

例 1.4. $M = 2^n$, $a =$ ４で割って 1 余る数, $c =$ 奇数とすると周期 M を実現する。

レポート問題 2. 上の例 1.4 に述べられた命題を証明せよ。

レポート問題 3. $M = 2^{31} - 1$, $a = 16807$, $c = 0$ とおくと、0 以外の初期値を選べば、周期は $M - 1$ となることを計算機実験で確かめよ。

このような、周期がほとんど M の LCG においては

- 等確率性：よい。（周期 M なら、全ての $0, \ldots, M - 1$ が一周期に 1 回あらわれる。　周期 $M - 1$ でも、一個の例外を除き全部あらわれる。）
- 独立性：ない。次の数は漸化式によってきまるのだから。だが、こうして得られた数列を、6 で割った余りをとって 0,1,2,3,4,5 として、サイコロシミュレーションに使う場合には経験上悪くない（ただし、M と 6 が互いに素でない時には乱数性は著しく損なわれる。）。
- 高速性：そこそこ（足し算、掛け算、割り算 1 回ずつ。だが通常の計算機では、この三つの演算はこの順に遅い。）
- 再現性：よい。M, a, c, x_0 の４つを記録するだけで、M または $M - 1$ の長さの数列が使える。
- 複数の数列の生成：x_0 を変えると違う列が得られる。

注意 1.5. 「一次関数による漸化式より、もうちょっと複雑な漸化式を使えばもっと良い乱数が作れるのではないか？」と思われるのは自然である。実際、フォン・ノイマンの漸化式は１０進展開の一部の桁を取り出して２乗をとるという複雑なものであった。これだと、初期値によっては周期が短くなってしまうなど、乱数性に問題がある。

疑似乱数発生に使う漸化式は、「生成された数列が乱数を模倣した良い性質を持つ」ように選ばれなくてはならないので、なんらかの数学的解析が行える必要性がある。そのため、現時点では代数的な関数（一次式、行列、多項式、分数式など）による漸化式を用いるのが主流である。

1.4　線形合同法の限界

LCG は大きな成功をおさめ、現在も広く使われている。しかし、80 年代ごろから計算機の高速化とシミュレーションの大規模化により、

1. 周期 2^{32} では足りない。
2. もっと高速にしたい。

という要請が強まった。

この二つは、LCG においては両立しにくい。周期は M 以下であるから、周期を長くするには M を大きくする必要がある。M の桁数を 2 倍にすると掛け算、割り算は筆算的アルゴリズムでは 8 倍の時間がかかる。

この事態は、有限代数により解決された。

2 元集合 $\mathbb{F}_2 := \{0, 1\}$ に、通常の積と通常の和を考え、$1 + 1 = 0$ と定義すると、体になる (§4.1 参照)。体なら、線形代数ができる。$\mathbb{F}_2^m$ を m 次元縦ベクトル空間とし、$m \times m$ 行列 A を用いて

$$\mathbf{x}_{j+1} := A\mathbf{x}_j$$

を計算する。A をうまく選ぶと長周期で高速（割り算、掛け算なし）になる。周期、一様分布性、独立性が線形代数（と少しの体論）を用いて解析できる。(これらについては後述???)。

参考：LCG の周期については、次が知られている。

定理 1.6. LCG(定義 1.1) の周期が M となる必要十分条件は、

1. $(c, M) = 1$ (c と M が互いに素であることをこうあらわす)
2. M の全ての素因数が、$a - 1$ を割りきる。
3. M が 4 で割りきれる場合には、$a - 1$ が 4 で割りきれる。

レポート問題 4. 上の定理を証明せよ。(やや難)

2　オートマトンと漸化式

オートマトンとは、「自動機械」の意味である。

2.1 無入力オートマトンとは漸化式である

定義 2.1. 無入力オートマトンとは、集合 S, O、関数 $f : S \to S$、元 $s_0 \in S$、関数 $o : S \to O$ の組である。S が有限集合のとき、有限状態オートマトンという。

現在あるデジタル計算機は、入力がないかぎり、無入力有限状態オートマトンである。メモリが有限なので、S としてメモリの取りうる状態全てを考えればよい。メモリの内容を見て、プロセッサーは次の時刻にメモリをどう変えるか決める。これが関数 f である。

S を状態集合、f を次状態関数、o を出力関数という。

メモリが非常に大きくて無限にあると考えたほうがいい場合もあり、そのような場合にはチューリングマシンと呼ばれるモデルが計算機の良い近似を与える。

使えるメモリが少ない（例えば 100 ビットしかない）ような場合には、有限状態オートマトンと捉えたほうが計算機の良い近似を与えることが多い。

乱数生成ではあまりメモリを使いたくないので、計算機を有限状態オートマトンと捉えることが多い。

以後、無入力有限状態オートマトンのみを扱い、**オートマトンといったら無入力有限状態のものをさすことにする**。

このオートマトンの状態は、**初期状態** s_0 を与えたとき、漸化式

$$s_{i+1} = f(s_i)$$

により $s_0, s_1, s_2, \ldots$ と変化していく。それぞれの状態のときの出力は $o(s_0), o(s_1), o(s_2), \ldots$ である。

従って、**無入力オートマトンとは、漸化式により数列を生成し、それを出力関数で変換して数列を得ること**に他ならない。

例 2.2. $S := \mathbb{Z}/M$, $f(x) := 10x \mod M$, $o(x) := [10x/M]$（切り捨て）とすると、s_0/M の小数展開の小数部を求める。

ここで、

$$\mathbb{Z}/M := \{0, 1, \ldots, M-1\}$$

であり、その元同士の和、差、積はそれぞれ整数と思って演算を行い、結果を M で割った余りをとることでまた $\mathbb{Z}/M$ に落とす、という計算の約束をする。（代数学の言葉で言うと、剰余「環」である。後述。）

2.2 オートマトンの周期

オートマトンに関する最初の定理は次の通りである。

定理 2.3. 有限状態オートマトンの出力列は、準周期的である。その周期は S の元の数 $\#(S)$ を超えない。

定義 2.4. 数列 $x_1, x_2, \ldots$ が (純) 周期的であるとは、ある 1 以上の $p \in \mathbb{N}$ が存在して全ての $n \in \mathbb{N}$ に対して

$$x_{n+p} = x_n \tag{2}$$

が成立すること。性質 (2) を満たす（1 以上の）最小の p を周期 (period) という。

レポート問題 5. 性質 (2) を満たす自然数 p は、周期の倍数となることを示せ。

数列 $x_1, x_2, \ldots$ が準周期的であるとは、ある $n_0 \in \mathbb{N}$ が存在して

$$x_{n_0}, x_{n_0+1}, x_{n_0+2}, \ldots$$

が周期的になること。このような n_0 のうち最小のものをとったとき、$x_1, \ldots, x_{n_0-1}$ を非循環部、$x_{n_0}, x_{n_0+1}, x_{n_0+2}, \ldots$ を循環部という。循環部の周期を、準周期数列の周期と定義する。

証明. (定理 2.3 の証明) まず、状態 $s_0, s_1, \ldots$ が準周期的であることを示す。$N := \#(S)$ とすると、部屋割り論法により $s_1, s_2, \ldots, s_{N+1}$ の中には重複して現れる元がある。すなわち、$1 \leq p < n \leq N+1$ が存在して

$$s_n = s_{n-p}$$

が成立する。両辺の f をとると帰納法により、この式は n が増えても常に成立。よって準周期的。$o(s_n)$ も準周期的となる。ちなみに、$o(s_n)$ の周期は s_n の周期の約数となる（問題 5)。 □

系 2.5. 分数の小数展開は準周期的となる。

系 2.6. $\sqrt{2}, \pi$ の小数展開を計算する有限状態オートマトンは存在しない。
（桁数を増やすにつれて、必要メモリがいくらでも増大していく。）

系 2.7. 周期が $\#(S)$ となるときは、純周期的でかつ s_i は一周期に S の元を丁度一回ずつとる。

レポート問題 6. S は有限集合とする。任意の初期状態に対し列 $\{s_i\}_{i=0,1,2,\ldots}$ が周期的になる必要十分条件を f の言葉であらわせ。

系 2.8. LCG(定義 1.1 参照) の定義する数列は準周期的であり、周期は M 以下。

レポート問題 7. (1) で、M が素数で $a \not\equiv 1 \mod M$ であれば、周期は $M-1$ 以下である。ある a が存在して周期を $M-1$ にできることを示せ。

これは、M が素数であるとき $(\mathbb{Z}/M)^\times$ が位数 $M-1$ の巡回群になることから従う（後述、そうやさしくない)。

3　擬似乱数生成プログラムの実例

本論に入る前に、感じをつかむためにプログラムの実例を眺めておく。以下、この節の中でいくつかのC言語プログラムの実例を挙げている。が、C言語に詳しくない人は無理にプログラムを読む必要はない。

3.1　デジタル計算機

現在、多くの計算機が1ワードを32ビットとしている。これは、計算機のCPUが一度に扱えるデータの単位が

$$\{0,1\}^{32} := \{(x_{31}, x_{30}, \ldots, x_0) | x_i = 0 \text{ または } 1\}$$

だということを意味している。w を1ワードのビット数とし、

$$W := \{0,1\}^w$$

とする。計算機のCPUに組み込まれている演算として、次の二種がある。

算術演算 W の元を2進数だと思っての和、差（非常に高速)。

少し時間がかかるが、積もハードウェアとして組み込まれているCPUも多い。

論理演算 W の元の、ビットごとの演算 (非常に高速)。

簡単のため、$w = 6$ とする。

- AND (C言語では　&) 両方1のときのみ1

$$001101\&010111 = 000101$$

- OR (C言語では　|) どっちか1なら1

$$001101|010111 = 011111$$

- EXOR (C言語では^) 和をとるが、$1 + 1 = 0$ でやる。

$$001101\hat{}010111 = 011010$$

定義 3.1. $\mathbb{F}_2 := \{0,1\}$ とおく。積は普通に、和も $1 + 1 = 0$ と約束するほかは普通にやる。これを2元体と言う。

W は、$\mathbb{F}_2$ 係数の w 次元横ベクトル空間 $\mathbb{F}_2^w$ とみなすこともできる。その場合、ベクトルとしての和はEXORとなる。

3.2 LCG

定義 1.1 でみた LCG は漸化式

$$x_{j+1} := ax_j + c \mod M$$

で数列を作る。1990 年代まで ANSI-C の標準擬似乱数 `rand` として用いられていたのは LCG で、パラメータは $a = 1103515245$, $c = 12345$, $M = 2^{31}$ で周期は M。

C によるプログラム例は、次のとおり。

```
static unsigned long x=3; /* initial seed */

unsigned long rand(void)
{
    x = x * 1103515245 + 12345;
    x &= 0x7fffffff; /* mod 2^31 */
    return x;
}
```

これでうまく行くのだが、それは C 言語では桁あふれ（足し算・掛け算などの結果が 2^{32} 以上になったときにおこる）が起きたときには警告したり停止したりせずに下位 32 ビットのみを残す仕様になっているからである。数学的に言えば、計算を $\mod 2^{32}$ で行っているということ。

乱数としてはいろいろ問題がある。例えば、最下位 1 ビットは周期 2 で循環する。

レポート問題 8. 下位 m ビットは、周期 2^m で循環することを証明せよ。

3 次元空間での分布を見るため、

1. 3 つ乱数を発生し、それらを xyz 座標とする点を単位立方体にプロット
2. 2^{31} 回繰り返す
3. 単位立方体の原点付近一辺 0.015 の立方体を表示

すると、図 1 のような格子構造を得る。

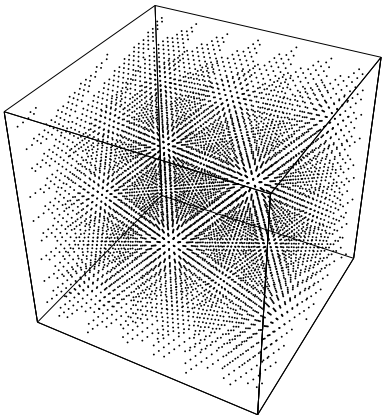


図 1: ANSI 標準 C 言語の擬似乱数 `rand()` により生成された空間内の「ランダム」な点列。ランダムとは言い難い結晶構造が見られる。

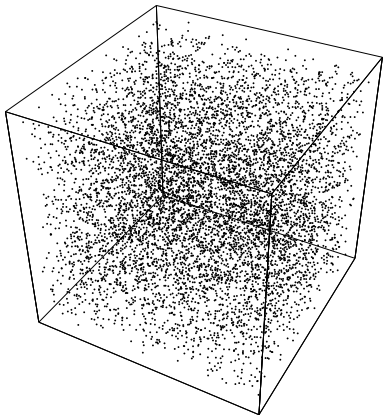


図 2: 松本眞・西村拓士が 97 年に開発したメルセンヌツイスター法により生成されたランダムな点列。

3.3 GFSR

1 ワード=w ビットを $\mathbb{F}_2$ 上の w 次元横ベクトル $\mathbb{F}_2^w$ と同一視し、定数 $n > m > 0$ を選んでワード列を漸化式

$$\mathbf{x}_{j+n} := \mathbf{x}_{j+m} + \mathbf{x}_j \quad (j = 0, 1, \ldots) \tag{3}$$

で生成するのが 3 項 Generalized Feedbacked Shiftregister 法 (GFSR) である。3 項以上の GFSR も使われているが、ここでは 3 項のもののみを扱う。

実際に、

$$\mathbf{x}_{j+3} := \mathbf{x}_{j+1} + \mathbf{x}_j \quad (j = 0, 1, \ldots)$$

を計算してみよう。各ビットごとに計算すればいい。一つのビットだけに着目すると、例えば

$$00101110010111010111\cdots$$

で周期 7 で循環する。一般に、周期 $2^n - 1$ で循環するような (n, m) の組がたくさん見つかっているが、無限にあるかどうかはわかっていない。

(3) のように、n 個前までの元により次の元が決まる漸化式を n 階漸化式という。

定義 3.2. W を集合とし、$g : W^n \to W$ を一つ決めて、$x_0, \ldots, x_{n-1} \in W$ を初期値とし、

$$x_{j+n} = g(x_{j+n-1}, \ldots, x_j)$$

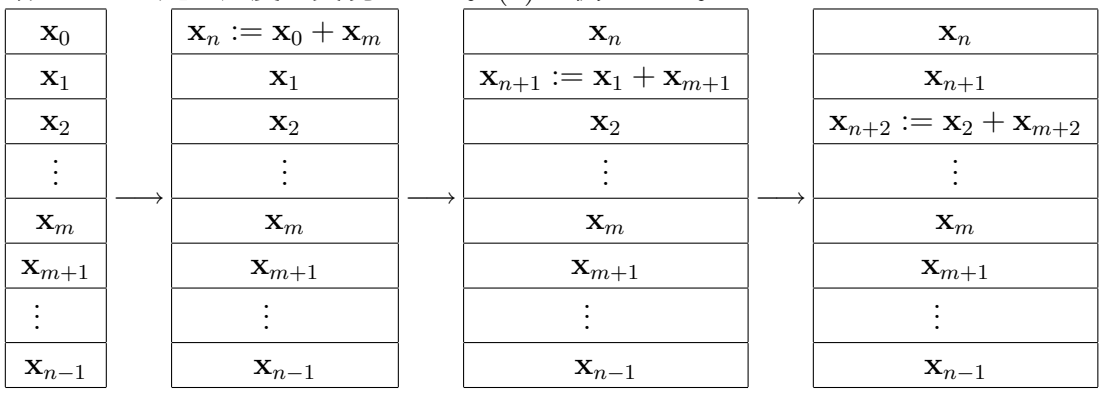
により W の元の列を生成する。このような漸化式を、W 上の n 階の漸化式という。

注意 3.3. n 階漸化式も、一階の漸化式とみなすことができる。$S := W^n$,

$$\begin{aligned} & f(x_{j+n-1}, \ldots, x_j) \\ = \ & (g(x_{j+n-1}, \ldots, x_j), x_{j+n-1}, \ldots, x_{j+1}) \end{aligned}$$

なる $f : S \to S$ により同じ数列が生成される。これを n 階漸化式の**一階化**という。

このような漸化式は、次のような実装（ラウンドロビンという）により、n が増えても一定の速度で実現できる。(3) を例にとる。



コードの例（C プログラム）

以下のプログラムは $n = 1279$, $m = 418$ の例であり、周期は $2^{1279} - 1$ である。関数 `gfsr()` は呼び出される毎に 0 以上 $2^{32} - 1$ 以下の整数を生成する。

初期化関数 `init_gfsr()` は [3, page 31–36] による方法を実装したものであり、$\lfloor 1279/32 \rfloor = 39$ 次元の均等分布性と、位相差 $2^{1279}/32$ までの自己相関を無くしている。

```
#define N 1279
#define M 418
#define W 32 /* W should be power of 2 */

static unsigned long state[N];
static int state_i;

void init_gfsr(unsigned long s)
{
    int i, j, k;
    static unsigned long x[N];

    s &= 0xffffffffUL;

    for (i=0; i<N; i++) {
        x[i] = s>>31;
        s = 1664525UL * s + 1UL;
```

```
        s &= 0xffffffffUL;
    }

    for (k=0,i=0; i<N; i++) {
        state[i] = 0UL;
        for (j=0; j<W; j++) {
            state[i] <<= 1;
            state[i] |= x[k];
            x[k] ^= x[(k+M)%N];
            k++;
            if (k==N) k = 0;
        }
    }

    state_i = 0;
}

unsigned long gfsr(void)
{
    int i;
    unsigned long *p0, *p1;

    if (state_i >= N) {
        state_i = 0;
        p0 = state;
        p1 = state + M;
        for (i=0; i<(N-M); i++)
            *p0++ ^= *p1++;
        p1 = state;
        for (; i<N; i++)
            *p0++ ^= *p1++;
    }

    return state[state_i++];
}
```

3.4 TGFSR

Twisted GFSR 法 [8][9] は、GFSR の改良版である。

$$\mathbf{x}_{j+n} := \mathbf{x}_{j+m} + \mathbf{x}_j A \quad (j = 0, 1, \ldots)$$

ここに A は $\mathbb{F}_2$ を成分とする w 次正則行列で、高速計算できるもの。例えば、A を次のような形の行列（コンパニオン行列という）

$$\begin{pmatrix} & 1 & & & \\ & & 1 & & \\ & & & \ddots & \\ & & & & 1 \\ a_0 & a_1 & \cdots & \cdots & a_{w-1} \end{pmatrix}$$

とする。この行列の最下行のベクトルを $\mathbf{a}$ で表せば

$$\mathbf{x}A = \begin{cases} \text{shiftright}(\mathbf{x}) & (\mathbf{x}\text{ の最下位ビットが }0\text{ の場合}) \\ \text{shiftright}(\mathbf{x}) + \mathbf{a} & (\mathbf{x}\text{ の最下位ビットが }1\text{ の場合}) \end{cases}$$

で右からの積が求まる。ここに、shiftright はワードを右に一ビットずらす（シフトする）ことを表す。（右端のビットは捨てられる。）

周期 $2^{nw}-1$ が実現可能（$n=25, w=32$ のコード TT800 が普及している）であり、初期値への依存性が低い。コードはザルツブルグ大学のホームページにある。

`http://random.mat.sbg.ac.at/ftp/pub/data/tt800.c`

```
/* A C-program for TT800 : July 8th 1996 Version */
/* by M. Matsumoto, email: matumoto@math.keio.ac.jp */
/* genrand() generate one pseudorandom number with double precision */
/* which is uniformly distributed on [0,1]-interval */
/* for each call.  One may choose any initial 25 seeds */
/* except all zeros. */

/* See: ACM Transactions on Modelling and Computer Simulation, */
/* Vol. 4, No. 3, 1994, pages 254-266. */

#include <stdio.h>
#define N 25
#define M 7

double
genrand()
{
    unsigned long y;
    static int k = 0;
    static unsigned long x[N]={ /* initial 25 seeds, change as you wish */
        0x95f24dab, 0x0b685215, 0xe76ccae7, 0xaf3ec239, 0x715fad23,
        0x24a590ad, 0x69e4b5ef, 0xbf456141, 0x96bc1b7b, 0xa7bdf825,
        0xc1de75b7, 0x8858a9c9, 0x2da87693, 0xb657f9dd, 0xffdc8a9f,
        0x8121da71, 0x8b823ecb, 0x885d05f5, 0x4e20cd47, 0x5a9ad5d9,
        0x512c0c03, 0xea857ccd, 0x4cc1d30f, 0x8891a8a1, 0xa6b7aadb
```

```
    };
    static unsigned long mag01[2]={
        0x0, 0x8ebfd028 /* this is magic vector 'a', don't change */
    };
    if (k==N) { /* generate N words at one time */
      int kk;
      for (kk=0;kk<N-M;kk++) {
        x[kk] = x[kk+M] ^ (x[kk] >> 1) ^ mag01[x[kk] % 2];
      }
      for (; kk<N;kk++) {
        x[kk] = x[kk+(M-N)] ^ (x[kk] >> 1) ^ mag01[x[kk] % 2];
      }
      k=0;
    }
    y = x[k];
    y ^= (y << 7) & 0x2b5b2500; /* s and b, magic vectors */
    y ^= (y << 15) & 0xdb8b0000; /* t and c, magic vectors */
    y &= 0xffffffff; /* you may delete this line if word size = 32 */
    y ^= (y >> 16); /* added to the 1994 version */
    k++;
    return( (double) y / (unsigned long) 0xffffffff);
}

/* this main() output first 50 generated numbers */
main()
{ int j;
  for (j=0; j<50; j++) {
    printf("%5f ", genrand());
    if (j%8==7) printf("\n");
  }
  printf("\n");
}
```

3.5 Mersenne Twister

Mersenne Twister は、TGFSR の改良版である [10]。

$$\mathbf{x}_{j+n} := \mathbf{x}_{j+m} + \mathbf{x}_{j+1}B + \mathbf{x}_j C \quad (j = 0, 1, \ldots)$$

によりワード列を生成。

より具体的には、

$$\mathbf{x}_{j+n} = \mathbf{x}_{j+m} + (\mathbf{x}_j \text{の上位 } w-r \text{ ビット}, \mathbf{x}_{j+1} \text{の下位 } r \text{ ビット}) A$$

として、A を w 次コンパニオン行列とする。

- $\mathbf{x}_j$ の下位 r ビットを無視するような（可逆でない）C の採用により、S を $nw - r$ ビットの空間と見なす。

- $2^{nw-r}-1$ が素数になるような r を選ぶことで、
  f の最小多項式の φ_f の原始性判定を容易にした。(§5.3 参照。)
- C プログラム mt19937ar.c が以下よりダウンロード可能
  http://www.math.sci.hiroshima-u.ac.jp/~m-mat/mt.html

4 有限体上の線形漸化式

GFSR, TGFSR, MT のどれもが、$W=\mathbb{F}_2^w$ 上の n 階線形漸化式により擬似乱数を発生させている。

4.1 体と線形代数

(以下で正確な定義を与えるが、荒っぽくいうと) 環とは、2 項演算である和と積が指定されていて、分配法則や単位元の存在などの常識的な性質が満たされているもの。
体とは、0 以外の元が積について可逆なもの。

例 4.1.

- 整数の集合 $\mathbb{Z}$ は環である。
- M を自然数とするとき、$\mathbb{Z}/M$ は環である (例 2.2)。
- 有理数の集合 $\mathbb{Q}$ は体である。
- $\mathbb{Z}/M$ が体となる必要十分条件は、M が素数であることである。

正確な定義は、次のとおり。

定義 4.2. 集合 S に、二項演算 $+$ が指定されているとする。

G1 $(x+y)+z=x+(y+z)$

が成立するとき $(S,+)$ の組を半群という。
さらに S の元 0 が指定されていて、

G2 $x+0=x, \quad 0+x=x$

を満たすとき $(S,+,0)$ の組をモノイドという。
さらに S の単項演算 $x\mapsto -x$ が指定されて

G3 $x+(-x)=0, \quad (-x)+x=0$

をみたすとき $(S,+,0,-())$ を群という。

さらに

G4 $x+y=y+x$

が満たされるとき、$(S,+,0,-())$ を可換群という。

さらに、S にもう一つの二項演算 $\times$ と S の元 1 が指定され、$(S,\times,1)$ が可換なモノイド（つまり上の G1,G2,G4 が、$+$ を $\times$ にして 0 を 1 にするとなりたつ）であり、$+$ と $\times$ は、分配法則

R1 $a\times(b+c)=a\times b+a\times c$

R2 $(a+b)\times c=a\times c+b\times c$

を満たす（この二つは積の可換性から同値だが）とき、$(S,+,0,-(),\times,1)$ を単位的可換環という。ここでは単に環という。

環 S であって、$0\neq 1$ であり、0 以外の元がどれも積について可逆、すなわち任意の $x\in S$ が $x\neq 0$ ならばある $y\in S$ によって

$$xy=yx=1$$

となるとき、S を体という。

有限集合で体となるものを有限体という。

環 S の、積についての可逆な元の全体を $S^\times$ であらわし、S の乗法群という。

例 4.3. N を自然数とする。

$$\mathbb{Z}/N:=\{0,1,2,\dots,N-1\}$$

は、和・差・積を計算する際に毎回 N で割った余りをとることによって、環となる。これを、N を法とする剰余環という。これが体となる必要十分条件は、N が素数 p であることである。このとき、体であることを強調して

$$\mathbb{F}_p:=\mathbb{Z}/p$$

と記す。この本でもっとも多くあらわれるのは $p=2$ の場合である二元体 $\mathbb{F}_2$ である。

定義 4.4. S を環とする。

V が環 S 上の加群（S 加群ともいう）であるとは、$(V,+,0)$ が可換群であって、スカラー倍と呼ばれる演算

$$S\times V\to V, \quad (a,v)\mapsto a\cdot v$$

が定義されており、任意の $a,b\in S$ と $v_1,v_2\in V$ に対し

M1 $a \cdot (v_1 + v_2) = a \cdot v_1 + a \cdot v_2$

M2 $(a + b) \cdot v = a \cdot v + b \cdot v$

M3 $(ab) \cdot v = a \cdot (b \cdot v)$, $1 \cdot v = v$

を満たすこと。

特に、S が体 K であるとき、K-加群を K 上の線形空間という。

例 4.5. K を体とするとき、n 次元横ベクトルの集合 K^n は成分ごとの和と、一斉に成分を $a \in K$ 倍するという演算によって K 線形空間である。

定義 4.6. K 上の線形空間 V, W を考える。写像 $f : V \to W$ が K 線形写像であるとは、

$$f(v_1 + v_2) = f(v_1) + f(v_2), \quad f(a \cdot v) = a \cdot f(v)$$

が成立すること。

体でさえあれば、線形空間のさまざまな性質、例えば基底の存在や線形写像の行列による表現などは通常の線形代数と同様に定義することができる。有限体でも全く同様に議論することができる。これは、体のもつ著しい特徴である。

この本で扱う限りにおいては、線形空間の例としては $V = K^n$ と、その部分空間 $W \subset V$（すなわち $+$ とスカラー倍に閉じた部分集合）を考えるだけでよい。線形写像とは、行列をかけることだと考えてもほとんど大丈夫である。

4.2 線形漸化式

S が体 K 上の線形空間で、$f : S \to S$ が K 線形写像であるとき、漸化式

$$\mathbf{x}_{j+1} = f(\mathbf{x}_j)$$

を 1 階線形漸化式という。これは S の元の列となる。

S が有限次元のときは、K 上の基底をとることによって

$$S = K^d, f(\mathbf{x}) = F\mathbf{x}$$

であるとしてよい。ここに、F は K 成分の d 次正方行列である。

というか、最初からこの形のもののみを考えていいというわけである。

ところで、この本では、計算機の w ビットのワード（0,1 が計 w 個ならんだもの）を、横ベクトル $\mathbb{F}_2^w$ と同一視する。その都合で、行列をベクトルに掛けるときには

$$\text{行ベクトルに、行列を右から掛ける: } \mathbf{x} \mapsto \mathbf{x}B$$

という記法をしばしばとる。

例：TGFSR

TGFSR では漸化式は n 階の

$$\mathbf{x}_{j+n} := \mathbf{x}_{j+m} + \mathbf{x}_j A \quad (j = 0, 1, \ldots)$$

であった。ここで、各 $\mathbf{x}_j$ は w 次元横ベクトルである。これを 1 階化すると、

$$f : (\mathbf{x}_{n-1}, \mathbf{x}_{n-2}, \ldots, \mathbf{x}_1, \mathbf{x}_0) \mapsto (\mathbf{x}_m + \mathbf{x}_0 A, \mathbf{x}_{n-1}, \ldots, \mathbf{x}_2, \mathbf{x}_1)$$

である。ベクトル列 $\mathbf{x}_j$ の周期は、$f : \mathbb{F}_2^{nw} \to \mathbb{F}_2^{nw}$ が初期値 $(\mathbf{x}_{n-1}, \mathbf{x}_{n-2}, \ldots, \mathbf{x}_1, \mathbf{x}_0)$ に対して生み出すベクトル列の周期と同じである。

f は見るからに線形写像である。具体的には、nw 次元の横ベクトルに

$$B = \begin{pmatrix} & I_w & & & \\ I_w & & I_w & & \\ & & & \ddots & \\ & & & & I_w \\ A & & & & \end{pmatrix}$$

を右からかけている。転置をとれば、nw 次元の縦ベクトルに上の行列の転置をかけていることになる。

レポート問題 9. Mersenne Twister の線形漸化式は

$$\mathbf{x}_{j+n} = \mathbf{x}_{j+m} + (\mathbf{x}_j \text{の上位 } w - r \text{ ビット}, \mathbf{x}_{j+1} \text{の下位 } r \text{ ビット}) A$$

であった。この漸化式を 1 階化するのに、

$$\begin{aligned} f : (\mathbf{x}_{n-1}, \mathbf{x}_{n-2}, \ldots, \mathbf{x}_1, \{\mathbf{x}_0 &\text{の上位 } w - r \text{ ビット }\}) \\ \mapsto (\mathbf{x}_n, \mathbf{x}_{n-2}, \ldots, \mathbf{x}_2, \{\mathbf{x}_1 &\text{の上位 } w - r \text{ ビット }\}) \end{aligned}$$

を考えることで $f : \mathbb{F}_2^{nw-r} \to \mathbb{F}_2^{nw-r}$ なる 1 階漸化式の与える数列と上の数列が本質的に同じであることを示し、具体的にはどのような行列をかけることと対応しているかを求めよ。

4.3 線形漸化式の解数列の周期

先のセクションで、高階であっても線形漸化式は 1 階化され、結局は

$$\mathbf{x}_0 \in K^d, \quad \mathbf{x}_{j+1} = B\mathbf{x}_j \tag{4}$$

の形になることを見た。ここに B は d 次正方行列である。

K が有限体であるとする。B と $\mathbf{x}_0$ が与えられたとき、いつ純周期的になるか、周期を求めるにはどうしたらいいか。

純周期的であったとすると、周期は

$$B^p\mathbf{x}_0 = \mathbf{x}_0$$

となる最小の $p \geq 1$ である。純周期的ではなかったとすると、

$$B^{p+k}\mathbf{x}_0 = B^k\mathbf{x}_0$$

となる最小の $k \geq 0$、$p \geq 1$ が非循環部分の長さと循環部分の周期を与える。

最初に、次の事実に注意しておく。

命題 4.7. K を有限体、B を d 次正方行列とし、d 次元縦ベクトル $\mathbf{x}_0 \in K^d$ を初期値とする漸化式

$$\mathbf{x}_{j+1} = B\mathbf{x}_j$$

を考える。このとき、このベクトル列は準周期的になり、その周期は $\#(K)^d - 1$ を超えない。

もし周期が $\#(K)^d - 1$ になったとしたら、$\mathbf{x}_0 \neq 0$ で、$\mathbf{x}_j$ には 0 以外の全てのベクトルを一周期に一回ずつあらわれる。また、その B に対しては、0 以外のどんな初期ベクトルを選んでも周期は $\#(K)^d - 1$ となる。

証明. 定理 2.3 によれば、有限状態オートマトンの状態遷移は準周期的になり、その周期は状態集合のサイズ以下である。上の場合、状態集合は K^d であり、その元の数は $\#(K)^d$ である。

ここで、零ベクトル $0 \in K^d$ に着目しよう。$B0 = 0$ だから、これは B で不動である。したがって、状態遷移の軌道の長さが $\#(K)^d$ となることはない（そうなったら、K^d の全てのベクトルが $B^j\mathbf{x}_0$ としてあらわれるはずだが、0 があらわれたら以後ずっと 0 であるから、矛盾である）。

よって、軌道の長さは高々 $\#(K)^d - 1$ である。もしこの上限を達成したならば、0 以外の全てのベクトルが一周期にちょうど一回ずつあらわれる。 □

実は、後述するように、周期を $\#(K)^d - 1$ にするような B はたくさん存在する (定理 5.1)。

それを示す前に、$B^p\mathbf{x} = \mathbf{x}$ となる p があるのかないのか判定し、ある場合にはそれを求めるアルゴリズムを考えよう。

単に B を何度も掛けていけば、ある場合には p はもとまるわけである。が、ここで考えている応用では $p = 2^{19937} - 1$ といった大きな数を想定しており、そのような単純な方法では p はもとまらない。

多項式環とそのイデアル論という、代数の基本的な道具がここではうまく使える。

定義 4.8. d 次正方行列 B と、d 次元縦ベクトル $\mathbf{x}$ に対し、

$$\{g(t) \in K[t] | g(B)\mathbf{x} = 0\}$$

とおくと、これは $K[t]$ のイデアルとなる。$K[t]$ は単項イデアル整域（すぐ後で復習する）であるから、これは一個の多項式によって生成される。後で示すように、このイデアルが 0 となることはない。そのため、この多項式はモニック (=最高次の係数が 1 である) 多項式と選ぶことができて、それはただ一通りである。この多項式を、$\mathbf{x}$ の B に関する annihilator 多項式といい、$\varphi_{B,\mathbf{x}}(t)$ で表す。

平たく言えば、

$$g(B)\mathbf{x} = 0 \Leftrightarrow \varphi_{B,\mathbf{x}}(t) | g(t). \tag{5}$$

ここで | は左が右を割り切ることを表す。

ごく簡単に、単項イデアル整域について復習しておく。

定理 4.9. K を体とするとき、K 係数一変数多項式環 $K[t]$ は単項イデアル整域である。

$K[t]$ とは、係数を K に持つ多項式のなす環である。

環 R に対し、その部分集合 $I \subset R$ がイデアルであるとは、加法に閉じていて、かつ R の任意の元をかけても I の外に出ない、すなわち

$$x, y \in I \Rightarrow x + y \in I, \quad x \in I, r \in R \Rightarrow rx \in I$$

が成立するものである。

環 R が整域であるとは、$a, b \in R$ が $ab = 0$ を満たせば a または b が 0 となる、という性質を満たすことである。

環 R が単項イデアル整域であるとは、R が整域であって、全てのイデアル I が単項イデアル、すなわち

$$I = \{ra | r \in R\}$$

の形にかけることである。（このようなイデアルを、R において a が生成する単項イデアルといい、しばしば (a) であらわす。）

レポート問題 10. 定義 4.8 で定義された集合が $K[t]$ のイデアルであることを示せ。$K[t]$ が単項イデアル整域であることを用いて、(5) を示せ。

annihilator 多項式を求めるには、次のアルゴリズムが実用的である。

$$\mathbf{x}, B\mathbf{x}, B^2\mathbf{x}, \ldots$$

を計算していき、これらが一次独立でなくなった瞬間を考えよう。これらはどれも $\mathbb{F}_2^d$ のベクトルであるから、一次独立なベクトルはたかだか d 個しかとれない。ので、一次独立でなくなった際に計算されたベクトルを $B^j\mathbf{x}$ とすると $j \leq d$ である。一次従属になったのだから、ある全てが 0 ではない $a_0, a_1, \ldots, a_j \in K$ が存在して

$$a_0\mathbf{x} + a_1 B\mathbf{x} + a_2 B^2\mathbf{x} + \cdots + a_j B^j\mathbf{x} = 0$$

である。$j-1$ までは一次独立だったのだから、$a_j \neq 0$ である。なので、上の式の両辺を a_j で割って、B^j の係数を 1 としてよい。さて、上の式の係数を用いて

$$\varphi(t) := a_0 + a_1 t + a_2 t^2 + \cdots + t^j$$

とおくと、上の式は $\varphi(B)\mathbf{x} = 0$ を意味する。そして、$\varphi(t)$ は $g(B)\mathbf{x} = 0$ となる $g(t) \neq 0$ のうち、最小の次数のものである。（より次数の低い多項式 $h(t)$ に対して $h(B)\mathbf{x} = 0$ が成立すれば、h の次数個まで $\mathbf{x}, B\mathbf{x}, B^2\mathbf{x}, \ldots,$ を計算すれば一次従属。）

したがって、$\varphi(t)$ は $\{g(t) \in K[t] | g(B)\mathbf{x} = 0\}$ なるイデアルの零以外の元のうちで一番次数の小さいものである。annihilator 多項式の定義から、そのようなモニック多項式は $\varphi_{B,\mathbf{x}}(t)$ でなければならない。よって、$\varphi(t) = \varphi_{B,\mathbf{x}}(t)$ であり、annihilator 多項式が求まった。

命題 4.10. B を K 成分 d 次正方行列、$\mathbf{x} \in K^d$ とする。

$$\mathbf{x}, B\mathbf{x}, B^2\mathbf{x}, \ldots$$

を次々に計算し最初に一次従属になったのが

$$\mathbf{x}, B\mathbf{x}, B^2\mathbf{x}, \ldots, B^j\mathbf{x}$$

だったとする。このとき、

$$a_0\mathbf{x} + a_1 B\mathbf{x} + \cdots + a_{j-1} B^{j-1}\mathbf{x} + B^j\mathbf{x} = 0$$

となる $a_i \in K$ が存在する。B の $\mathbf{x}$ に関する annihilator 多項式 $\varphi_{B,x}(t)$ は

$$\varphi_{B,x}(t) = t^j + a_{j-1} t^{j-1} + \cdots + a_1 t + a_0$$

で求まる。特に、その次数は常に d 以下となる。

さて、周期を求める方に戻る。純周期的なら

$$(B^p - I)\mathbf{x} = 0$$

となり、そのような最小の p が周期であるから、それは (5) より

$$\varphi_{B,\mathbf{x}}|t^p - 1$$

なる最小の p といえる。

$$K[t]/\varphi_{B,\mathbf{x}}$$

で、多項式を $\varphi_{B,\mathbf{x}}$ で割った余りの集合に、和差積を入れて環にしたもの（計算後いちいち $\varphi_{B,\mathbf{x}}$ で割って余りをとる）を表す。すると、p はこの環の中での t の乗法的位数に他ならない。

集合としては、この環は $\deg(\varphi_{B,\mathbf{x}})$ 未満の次数を持つ多項式の集合である。加法とスカラー倍に関しては、このような多項式の集合としての加法とスカラー倍に一致する。積をとるときだけは、一旦積の結果の次数が高くなるかも知れないものを、$\varphi_{B,\mathbf{x}}$ で割って余りをとることによりこの集合内で値をとるようにする。K 上の線形空間としての次元は、$\deg(\varphi_{B,\mathbf{x}})$ と一致する。

$(K[t]/\varphi_{B,\mathbf{x}})^\times$ で、この環の積に関する可逆元の集合を表す。これは可換群となり、乗法群とよばれる。$K[t]/\varphi_{B,\mathbf{x}}$ は K 上有限次元の線形空間だから、K が有限体の時は有限集合である。乗法群は有限群となる。

$t \in (K[t]/\varphi_{B,\mathbf{x}})^\times$ となる必要十分条件は、t と $\varphi_{B,\mathbf{x}}$ が互いに素になることである。

定理 4.11. K を有限体とする。$\mathbf{x} \in K^d$ を初期値としたときの $B \in M_d(K)$ による状態変移が純周期的になる必要十分条件は、t と $\varphi_{B,\mathbf{x}}$ が互いに素になることである。このとき、周期は

$$t \in (K[t]/\varphi_{B,\mathbf{x}})^\times$$

の位数となる。

これにより次が言える。

定理 4.12. 上の定理 4.11 において、純周期的であるとき、周期は $\leq \#(K)^d - 1$ である。

等号成立の必要十分条件は $\deg \varphi_{B,\mathbf{x}} = d$ で、$(K[t]/\varphi_{B,\mathbf{x}})^\times$ の位数が $\#(K)^d - 1$ で、かつ t で生成されることである。

証明. $d' := \deg \varphi_{B,\mathbf{x}}$ とおくと、

$$\#((K[t]/\varphi_{B,\mathbf{x}})^\times) \leq \#(K)^{d'} - 1 \leq \#(K)^d - 1$$

である。よって、等号が成立するには上の等号が両方成り立たなければならず、かつ t が左辺の群の生成元でなければならない。 □

さて、一般に K 係数多項式 $\varphi(t)$ に対し、

$$K[t]/\varphi(t)$$

における t の乗法位数はたかだか $\#(K)^{\deg\varphi(t)}-1$ であった。この等号が成立するような多項式を原始多項式という。

定義 4.13. K 係数多項式 $\varphi(t)$ が原始多項式であるとは、

$$K[t]/\varphi(t)$$

における t の乗法位数が $\#(K)^{\deg\varphi(t)}-1$ となること。

系 4.14. $\varphi(t)$ が原始多項式である必要十分条件は、$\varphi(t)$ が既約多項式で、かつ t が乗法群 $(K[t]/\varphi(t))^\times$ を生成すること。

レポート問題 11. 上の系を証明せよ。

この用語を用いれば、定理 4.12 は次のように述べられる。

系 4.15. 定理 4.12 において等号成立の必要十分条件は $\varphi_{B,\mathbf{x}}$ が d 次原始多項式となることである。(d は B のサイズ)

注意 4.16. 「純周期的」という条件は、上の定理 4.12 から省いても実は正しい。純周期的ではないとしよう。t が $K[t]/\varphi_{B,\mathbf{x}}$ の中で何乗しても 1 にならない。有限群においてはその元は何乗かしたら 1 になるのであるから、

$$t\notin(K[t]/\varphi_{B,\mathbf{x}})^\times$$

である。今、$\varphi(t)=t^m\psi(t)$ と、t で割れるだけ割って $\psi(t)$ は t で割り切れないとする。中国式剰余定理により

$$K[t]/\varphi(t)\cong K[t]/t^m\times K[t]/\psi(t)$$

となる。t は $\psi(t)$ と互いに素だから

$$t\in(K[t]/\psi(t))^\times$$

となる。その位数を p とすると、

$$\varphi(t)|t^{m'}(t^{p'}-1)\Leftrightarrow m'\geq m,\quad p|p'$$

となる。よって、非循環部分の長さが m, 周期は p となる。

レポート問題 12. 上の注意の証明をきちんと述べよ。

定義 4.17. K 成分 d 次正方行列 B に対し、その最小多項式 $\varphi_B(t)$ をイデアル

$$\{g(t) \in K[t] | g(B) = 0\}$$

のモニックな単項生成元とする。平たく言えば

$$g(B) = 0 \Leftrightarrow \varphi_B(t) | g(t).$$

定義 4.18. K 成分 d 次正方行列 B に対し、その特性多項式 $\chi_B(t)$ を

$$\chi_B(t) = \det(tI_d - B) \in K[t]$$

で定義する。これは d 次モニック多項式。

定理 4.19.

$$\varphi_{B,\mathbf{x}}(t) | \varphi_B(t) | \chi_B(t).$$

証明. 最初の割り切り関係は、$\varphi_B(B)\mathbf{x} = 0$ を示せばよいが、これはあきらか。次の割り切り関係は、

$$\chi_B(B) = 0$$

を示せばよいが、これは Cayley-Hamilton の定理として知られている。 □

以上をまとめると、次を得る。

定理 4.20. K を有限体とする。B を K 係数 d 次正方行列、$\mathbf{x} \in K^d - \{0\}$ とする。以下は全て同値。

1. $\mathbf{x}$ を初期値とする B による状態遷移の周期が最大値 $\#(K)^d - 1$ を達成する
2. $\varphi_{B,\mathbf{x}}(t)$ が d 次原始多項式
3. $\chi_B(t)$ が原始多項式

証明. 1 と 2 の同値性は系 4.15 そのもの。

2 から 3:定理 4.19 から

$$\varphi_{B,\mathbf{x}} | \chi_B(t)$$

で、右は d 次多項式なので左も d 次ならモニックなので等しい。

3 から 2:同じ割り切り関係で、原始多項式ならば既約 (系 4.14) であることを使えば、$\varphi_{B,\mathbf{x}} = 1$ または $\chi_B(t)$。もし $= 1$ とすると、$I_d\mathbf{x} = 0$ すなわち $\mathbf{x} = 0$ となり仮定に反する。よって $\varphi_{B,\mathbf{x}}(t) = \chi_B(t)$、よって 3 から 2 が言えた。 □

補遺：Cayley-Hamilton の定理

一応 Cayley-Hamilton の定理を復習しておこう。

定理 4.21. (Cayley-Hamilton の定理、一般の K:可換環で成立) A を K 係数 n 次正方行列とすると、

$$\chi_A(A) = 0$$

証明. $tI - A \in M_n(K[t])$ の余因子行列を $Q(t)$ とおき、t について整理して

$$Q(t) = Q_{n-1}t^{n-1} + \cdots + Q_0 \quad (Q_i \in M_n(K))$$

の形に書く。さて、余因子行列と元の行列の積は（順序を逆にしても）、元の行列の行列式×単位行列である（線形代数の標準的教科書参照）。すなわち、

$$(tI - A)Q(t) = Q(t)(tI - A) = \det(tI - A)I = \chi_A(t)I \in M_n(K[t]) \qquad (6)$$

が成立。左端の等号から $AQ(t) = Q(t)A$。各 t^i の係数を比べると、A と Q_i は可換。

乱暴にいうと、Q_i と A が可換であることから、(6) の両辺を展開する際、t に A を代入してから展開しても展開してから代入しても同じであることがわかり、それで $0 = \chi_A(A)$ が言える。

詳しくいうならば、

$$(tI - A)Q(t) = (tQ_{n-1}t^{n-1} + \cdots + tQ_0) - (AQ_{n-1}t^{n-1} + \cdots + AQ_0) = \chi_A(t)I$$

より $Q_{n-1}, Q_{n-2} - AQ_{n-1}, \ldots, Q_0 - AQ_1, -AQ_0$ は全てスカラー行列で、「$\chi_A(t)$ の対応する次数の係数」かける I に一致する。従って、任意の $X \in M_n(K)$ に対して

$$(Q_{n-1}X^n + \cdots + Q_0X) - (AQ_{n-1}X^{n-1} + \cdots + AQ_0) = \chi_A(X)I$$

が成立する。ここで $X = A$ を代入し、A と Q_i の可換性を用いると、前半と後半の各項がそれぞれキャンセルして 0 となる。 □

5 最大周期列の存在、性質、探索

5.1 最大周期列の存在

以上をまとめると、次のようになる。K を有限体とする。

- 状態空間の次元 d を固定したとき、d 次正方行列 B により与えられる線形漸化式の周期は $\leq \#(K)^d - 1$ である。

- 等号が成立する必要十分条件は、初期値が 0 でなく、かつ B の特性多項式 χ_B が原始多項式であることである。

このような B が存在しなければ空論であるのだが、実はたくさん存在する。次の定理と補題による。

定理 5.1. 任意の有限体と任意の自然数 d に対し、d 次の原始多項式が存在する。
実は、$\varphi(\#(K)^d - 1)/d$ 個存在する。(ここに、φ はオイラーの関数。)

補題 5.2. 任意に与えられた d 次多項式に対し、特性多項式がそれになるような d 次行列 B が存在する。

証明. 多項式の係数をもちいて、コンパニオン行列 (§3.4 参照) を使えばよい。 □

レポート問題 13. 上の補題の証明を完成させよ。

定理 5.1 の略証

- 任意の素数ベキ p^m に対し、p^m 元からなる体が同型を除いて唯一つ存在する。これを $\mathbb{F}_{p^m}$ であらわす。

  $$\mathbb{F}_{p^m} \subset \mathbb{F}_{p^n} \Leftrightarrow m|n$$

  が示される。

- このことから、有限体 $K = \mathbb{F}_{p^m}$ と整数 d について、K の d 次拡大体 $L = \mathbb{F}_{p^{md}}$ が存在することがわかる。

- 体の乗法群の有限部分群は巡回群である。したがって、$L^\times$ の生成元 α がある。

- α の最小多項式を $\varphi_\alpha(t)$ とすると、

  $$K[t]/\varphi_\alpha(t) \cong L, \quad t \mapsto \alpha$$

  なる体の同型がある。ここで、右辺の α の位数は $\#(L^\times) = \#(K)^d - 1$ であるから、左辺の t の位数もそうなる。言い換えると、$\varphi_\alpha(t)$ は原始多項式である。

- 最小多項式をとるという写像

  $$L^\times \text{ の生成元} \to d \text{ 次原始多項式}$$

は、$d:1$ の全射である。全射性は d 次拡大体が一つしかないことから従い、$d:1$ であることは L/K が分離拡大であることから従う。

左辺の元の数は $\varphi(\#(K)^d-1)$ だから、これを d で割って右辺の元の数を得る。

レポート問題 14. 定理 5.1 の証明を完成させよ。

5.2 最大周期列の性質

K を有限体とし、状態空間 $S=K^d$, $f:S\to S$ は行列 B を左からかける、というオートマトンを考える。

前節の結果から、このオートマトンの周期は高々$\#(K)^d-1$ であり、それを達成する B は存在することがわかった。

このとき、このオートマトンの軌道は、0 以外の初期値を選べばちょうど

$$K^d-\{0\}$$

を全て一度ずつ通る。

今、線形漸化式が $W=\mathbb{F}_2^w$ 上の n 階線形漸化式

$$\mathbf{x}_{j+n}=g(\mathbf{x}_{j+n-1},\dots,\mathbf{x}_j)$$

であったとする。1 階化して考えれば、これは

$$S=W^n=\mathbb{F}_2^{nw}$$

上の 1 階線形漸化式である。これが最大周期である $2^{nw}-1$ を達成したとする (問題 15 にあるように、そのような g は常に存在する)。

一階化された漸化式においては $(\mathbf{x}_{j+n-1},\dots,\mathbf{x}_j)$ に対して $(\mathbf{x}_{j+n},\dots,\mathbf{x}_{j+1})$ が求められる。最大周期 $2^{nw}-1$ を達成したとする。この n 個の w 次元ベクトルの組が、「全部 0」というパターンを除いて全て一度ずつ一周期にあらわれることを意味している (命題 4.7)。すなわち、連続する n 個の出力の組を一周期にわたり記録したもの (重複度も数える、すなわち multi-set としてみる)

$$\{(\mathbf{x}_{j+n-1},\dots,\mathbf{x}_j)|j=0,1,\dots,2^{nw}-2\}$$

は、2^{nw} 通りある全てのパターンを、全部 0 というパターンをのぞいて全て一回ずつとるということである。

これを、最大周期列の window property と言ったり、n 次元均等分布性と言ったりする。

擬似乱数としてこの数列を用いるときには、この性質がさまざまな量の一周期にわたる分布を求めるのに有用である。例えば、0 と 1 の個数は全体でみてほぼ同じである。また、連続する n 個の出力は、一周期に渡ってみれば独立である。

GFSR にはこの性質がなかった（状態空間が nw ビットであるのに、周期は 2^n-1）が、TGFSR はこの性質を持つ。Mersenne Twister も、この性質をもっている。ただし、見る部分を「欠けた窓」

$$\{(\mathbf{x}_{j+n-1},\dots,\mathbf{x}_{j+1},\mathbf{x}_j\text{の上位 }w-r\text{ ビット})\}$$

にしなくてはならない。

レポート問題 15. 上で、任意の n, w に対し

$$\mathbf{x}_{j+n}=g(\mathbf{x}_{j+n-1},\dots,\mathbf{x}_j)$$

が周期 $2^{nw}-1$ を達成するような線形写像 g が存在することを示せ。

レポート問題 16. Mersenne Twister における window property を定式化し、証明せよ。

5.3 多項式の原始性の判定

以上により、$\mathbb{F}_2$ 線形な漸化式により最大周期数列を高速に発生させ（擬似乱数に使い）たい場合、

1. B として、計算機で高速に実現できるものを選ぶ
2. その範疇で、特性多項式が原始的になるものを探索する

ということになる。

多項式が原始的であるかどうかの判定は微妙な問題である。現在のところ、原始多項式になる十分条件も、必要条件もあまり強力なものは知られていない。ので、(1) の範疇内でランダムに B を発生させ、その特性多項式が原始的かどうか判定し、原始的でなければ捨てる、というのが現在普通にとられる戦略である。

原始性をチェックする方法としては、モノイド一般に対して成り立つ、次の低レベルな方法が使われることが多い。

命題 5.3. G を半群とし、$t, a \in S$ をとる。このとき、集合

$$\{n\in\mathbb{N}\cup\{0\}|t^n a=a\}$$

ある $s\in\mathbb{N}\cup\{0\}$ の自然数倍（0 倍も含む）の全体となる。

ここで、$t^0 a := a$ と定義しておく。

証明. この集合が 0 のみからなれば、$s = 0$ で成立。そうでないとすると、この集合には正の整数が含まれる。その最小値を s とすると、この集合の任意の元 n を s で割った余り r に対して

$$a = t^n a = t^r t^{qs} a = t^r a$$

となり、$r < s$ と s の最小性から $r = 0$ となるからである。 □

命題 5.4. G をモノイドとする。e をその単位元とする。$g \in G$ の位数が r である必要十分条件は、

1. $g^r = e$, かつ
2. r の全ての素因子 p について、$g^{r/p} \neq e$。

証明. 上の命題を $t = g$, $a = e$ に対して用いる。$g^n = e$ を満たす最小の正整数 n を s とする。$g^n = e$ ならば $s|n$ であるから、1 から $s|r$ であり、r/s が 1 でなければ、その素因数を一つとって p とすると 2 を満たさなくなる。よって $r = s$。 □

ベキ g^n を計算する際は、n 回掛ける必要はない。n を 2 進展開すると、大体 $1.5 \log_2(n)$ 回の掛け算で g^n の計算ができる。

例えば、n の二進展開が 3 桁で $a_2 a_1 a_0$ であったとしたとき、

1. $x \leftarrow 1$
2. $x \leftarrow g^{a_2} \times x$
3. $x \leftarrow x^2$
4. $x \leftarrow g^{a_1} \times x$
5. $x \leftarrow x^2$
6. $x \leftarrow g^{a_0} \times x$

で x に g^n が求まる。（ここに $\leftarrow$ は左の変数に右の値を代入することをあらわす。）これだと $2 \log_2(n)$ 回必要だが、$a_i = 0$ の時には積をサボってよい。

命題 5.4 を用いて元の位数を確かめる場合、最も難しい計算段階は r の全ての素因数を求めるという部分である。

r が素数であることが分かっている場合には、$g \neq e$, $g^r = e$ を確かめるだけで位数は r となる。

系 5.5. G をモノイドとする。$g \in G$, $g \neq e$, $g^r = e$ で r が素数なら、g の位数は r である。

系 5.6. G を群とし、G の位数が素数 p であるとする。このとき、G の単位元以外の元は全て位数が p であり G を生成する。

系 5.7. K を q 元からなる有限体とする。$q^m - 1$ が素数であるときは、m 次既約多項式はすべて原始多項式である。

証明. $\varphi(t)$ を m 次既約多項式とする。$K[t]/\varphi(t)$ は体であり、$(K[t]/\varphi(t))^\times$ は位数が $q^m - 1$ の群である。ここで、これが素数であると仮定したので 1 以外の元の位数は $q^m - 1$ であり、特に t の位数もそうだから定義 4.13 により $\varphi(t)$ は原始多項式である。 □

もっとも、$q^m - 1 = (q-1)(q^{m-1} + q^{m-2} + \cdots + 1)$ である。それが素数であるには $q = 2$ であるか、または $q \geq 3$ ならば $m = 1$ でなければならない。後者の場合は $\varphi(t)$ は 1 次式であり面白くない。

$2^m - 1$ が素数であるとき、メルセンヌ素数とよび、m をメルセンヌ指数と呼ぶ。このとき m も素数となる。

命題 5.8. m をメルセンヌ指数とし、$\phi(t)$ を $\mathbb{F}_2$ 上の m 次多項式とする。$\phi(t)$ が既約多項式（原始多項式と言ってもこの場合同値）である必要十分条件は、$\mathbb{F}_2[t]/\phi(t)$ において $t^2 \neq t$ かつ $t^{(2^m)} = t$ であること。

証明. t の位数が $2^m - 1$ になればいい。必要性はあきらか。

十分性を示す。命題 5.3 を $a = t$, 素数 $n = 2^m - 1$ について考える。すなわち、この n が命題の集合に含まれると仮定する。このような最小値 s は $n = 2^m - 1$ の約数であるから 1 かそれ自身。1 である可能性は最初に排除したから $2^m - 1$。これは、$t^l t$ が $l = 0, 1, \ldots, 2^m - 2$ まで全部相異なっていることを意味している。これらのうち 0 はないから、$\mathbb{F}_2[t]/\phi(t)$ の元のうち 0 以外を全て t のベキで書いたことになる。よって 1 は t のベキであり、t は可逆で位数は $2^m - 1$ となる。よって $\phi(t)$ は原始多項式。 □

例 5.9. $\phi(t) = t^7 + t + 1$ の原始性を判定する。$2^7 - 1 = 127$ が素数であることは小学校で習う。よって、上の判定法が使える。t^{2^7} を $\mod \phi(t)$ で計算すればよい。$t \to t^2 \to t^4 \to t^8 = t^2 + t \to t^4 + t^2 \to t^8 + t^4 = t^4 + t^2 + t \to t^8 + t^4 + t^2 = t^4 + t \to t^8 + t^2 = t$ となり、確かに自乗を 7 回すると t に戻ったので $\phi(t)$ は既約である。

注意 5.10. $q > 2$ であっても、$(q^m - 1)/(q - 1)$ が素数になることはありうる。この場合にはやはり原始性のチェックは比較的やさしい。

6　擬似乱数のテスト:二項分布からのずれ

擬似乱数をコイン投げのシミュレーションに使うことを考える。例えば、出力の最下位 1 ビットの 0-1 が、コインの表-裏に対応しているものとする。

公平なコインを k 回投げて、うち t 回が表である確率は

$$\binom{k}{t}/2^k$$

である。いまもし擬似乱数の出力列が n 次元均等分布しているとして、$n \geq k$ ならば擬似乱数でシミュレーションしても (経験上も) 余り問題はない。しかし、$k > n$ となったときには問題が出てくることがある。

実際、3 項 GFSR などでは正しい二項分布からかなり偏った分布を得る。この章では、この現象を解析する。

6.1　擬似乱数の確率モデル化

$\{0,1\}$ に値をとる、長さ M の擬似乱数を発生する擬似乱数発生器は初期状態から長さ M の 0-1 列への関数

$$G : S \to \{0,1\}^M$$

とみなせる。(S:状態空間、漸化式で言えば初期値の空間。)

例 ($M = 10$):

$$G(s_0) = (0011001110),$$
$$G(s_1) = (1101100111),$$
$$\vdots$$

確率モデル化するため、次の仮定をおく。

初期値仮定 毎回のシミュレーションごとに、初期状態は S から一様かつ独立にランダムに選ばれるとする。

この仮定により、擬似乱数発生器の出力は $\{0,1\}^M$ に値をとる確率変数となる。[2]

[2]実際のシミュレーションでは、毎回いちいち初期化することは通常しないし、むしろしてはいけない。というのは、多くの擬似乱数で、あるタイプの初期値を選ぶとその後いくつかの出力の分布が偏るからである。これは、大きな状態空間をもつ擬似乱数発生法により顕著に見られ、Mersenne Twister も例外ではない。

従って、[初期値仮定] は少々不自然なモデルである。しかし、多くの擬似乱数発生法で、一連のシミュレーションを終えたあとの状態は極めてランダムに見える。すなわち、状態空間の中で一様かつ独立に選ばれているように見える。ので、無理な仮定ではない。

経験上この仮定をしても実験的統計的検定の結果と良く合致している。

今、m 個の出力のうちの 1 の個数を見て、次の k 個の出力のうちの 1 の個数を当てようとしたとする。

$M := m + k$ とおく。任意の

$$\mathbf{x} \in \{0,1\}^M = \{0,1\}^m \times \{0,1\}^k,$$

に対し、

$$\begin{aligned} \mathrm{wt}_o(\mathbf{x}) &:= \ \mathbf{x} \text{ の左 } m \text{ 成分中の 1 の数} \\ \mathrm{wt}_f(\mathbf{x}) &:= \ \mathbf{x} \text{ の右 } k \text{ 成分中の 1 の数} \end{aligned}$$

とおく (wt は weight、o は observed、f は future)。

$\mathbf{x}$ を擬似乱数発生器が生成する先の確率変数とする。任意の $0 \le s \le m$ と $0 \le t \le k$ に対し、

$$\mathrm{wt}_o(\mathbf{x}) = s \text{ という条件の下で } \mathrm{wt}_f(\mathbf{x}) = t \text{ となる}$$

という条件つき確率を

$$p_{k,m}(t|s) := \mathrm{Prob}(w_f(\mathbf{x}) = t | w_o(\mathbf{x}) = s)$$

であらわす。今から、0,1 をコインの表・裏に対応させて考えることにする。すると、これは、「過去 m 回のうち裏が s 個であった」という条件の下で「次の k 回のうち裏が t 個である」確率である。理想的な乱数では

$$p_{k,m}(t|s) = \binom{k}{t} / 2^k$$

となるべきである。

- $p_{k,m}(t|s)$ を求めるのは一般には困難
- $\mathbb{F}_2$-線形な擬似乱数発生法に対してはある条件のもと、符号理論にあらわれる MacWilliams 恒等式を用いて計算できる (§6.4)。

6.2 確率値の計算結果

`random()` という C 言語の 90 年代に推奨擬似乱数として使われていたものは、次の漸化式により擬似乱数を生成する (ラグつきフィボナッチ法と呼ばれている)。

$$\mathbf{x}_{i+31} = \mathbf{x}_{i+28} + \mathbf{x}_i \bmod 2^{32} \quad (i = 1, 2, \ldots)$$

まぎらわしいが、ここでの + は 2 進整数としての和であり EXOR ではない。しかし、この最下位一ビットは

$$x_{i+31} = x_{i+28} + x_i \bmod 2 \quad (i = 1, 2, \ldots)$$

を満たすので、$\mathbb{F}_2$ 上の線形漸化式とみなせる。

`ran_array()` という擬似乱数は、Knuth が 97 年に [4] で推薦したものである。漸化式

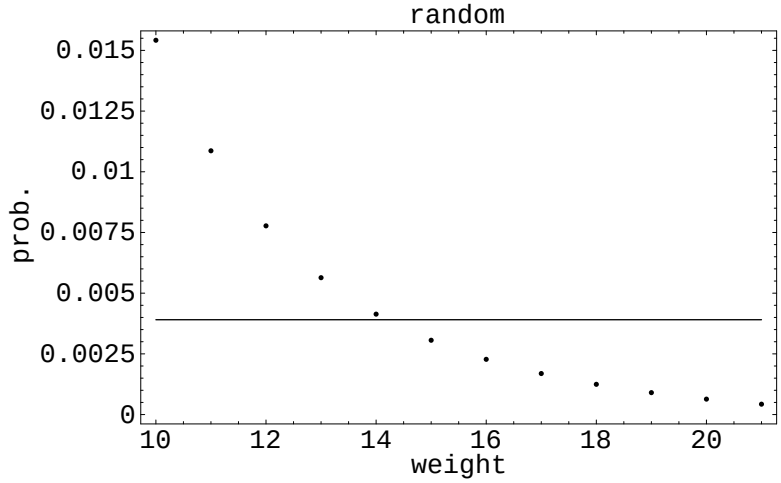
$$\mathbf{x}_{i+100} := -\mathbf{x}_{i+63} + \mathbf{x}_i \bmod 2^{30} \ (i = 1, 2, \ldots)$$

により数列を生成し、一部分を捨て去る（Lüscher による改良）ことで改良を得るのだが、ここでは捨て去りについては全く扱わない。これも整数としての和（というより差）をとっているが、その下一桁は

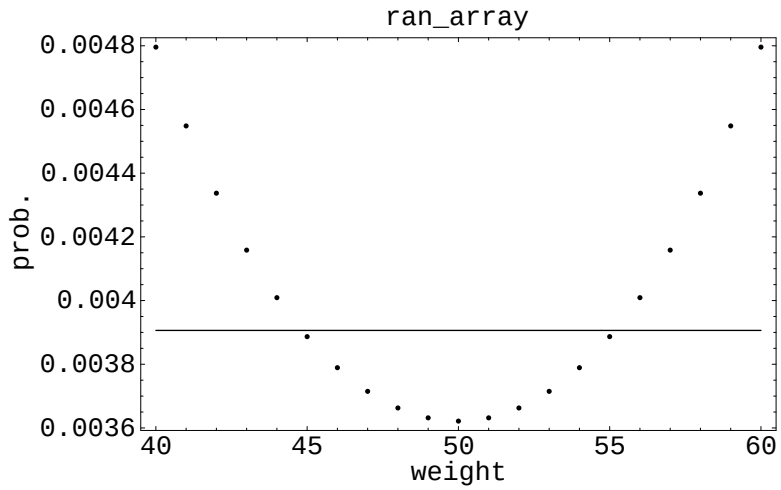
$$x_{i+100} := x_{i+63} + x_i \bmod 2 \ (i = 1, 2, \ldots)$$

なる、$\mathbb{F}_2$ 上の線形漸化式で定義された数列となる。

これらに対し、後述の §6.4 の方法により $p_{k,m}(0|t)$ を計算してみたのが以下の図である。



`random()` 最下位 1 ビットに関する $p_{8,31}(0|s)$ $(10 \leq s \leq 22)$ のグラフ。すなわち、直前の 31 回のコイン投げのうち丁度 s 回が裏だったという条件の下で、次の 8 回が全て表となる確率。横線は真乱数を用いたときの確率 1/256=0.00390625



ran_array() 最下位1ビットに関する $p_{8,100}(0|s)$ $(40 \leq s \leq 60)$ のグラフ。すなわち、直前の100回のコイン投げのうち丁度 s 回が裏だったという条件の下で、次の8回が全て表となる確率。横線は真乱数を用いたときの確率 $1/256$=0.00390625

このように、これらの生成法には大きな偏りがある。では、どうやってこれらの確率を求めたのかを次に述べる。

6.3 分離重み数え上げ

前の節のとおり、

$$\{0,1\}^M = \{0,1\}^m \times \{0,1\}^k$$

で、擬似乱数の可能な出力は

$$G(S) \subset \{0,1\}^M$$

であるとする。

今、疑似乱数出力 $\mathbf{x} \in G(S)$ は $G(S)$ 上一様ランダムに分布していると仮定する。すなわち、$G(S)$ のどの元も等確率で出現するとする。これは、G が線形であれば [初期値仮定] より従う。

レポート問題 17. なぜ従うか、証明せよ。

すると、

$$p_{k,m}(t|s) := \mathrm{Prob}(w_f(\mathbf{x}) = t | w_o(\mathbf{x}) = s)$$

は

$$A_{ij} := \#\{\mathbf{x} \in G(S) | \mathrm{wt}_o(\mathbf{x}) = i, \mathrm{wt}_f(\mathbf{x}) = j\} \quad (0 \leq i \leq m,\ 0 \leq j \leq k)$$

から、次式で求まる。

$$p_{k,m}(t|s) = A_{st}/(A_{s0} + A_{s1} + \cdots + A_{sk}).$$

A_{ij} を $G(S)$ の分離重み数え上げという。

さて、以下の 2 条件が満たされる場合には A_{ij} が計算できる。

1. 擬似乱数の出力集合
$$G(S) \subset \mathbb{F}_2^M$$
が $\mathbb{F}_2$-線形部分空間である。(例えば、漸化式が $\mathbb{F}_2$ 線形ならよい。)

2. $M = k + m$ が $G(S) \subset \mathbb{F}_2^M$ の次元に比べて大きすぎない。

6.4 MacWilliams 恒等式

一般の部分線形空間 $C \subset \mathbb{F}_2^{m+k}$ ($G(S)$ を念頭においている) に対し、その分離重み数え上げとは上でも定義した二次元の数表

$$A_{ij} := \#\{\mathbf{x} \in C | \mathrm{wt}_o(\mathbf{x}) = i, \mathrm{wt}_f(\mathbf{x}) = j\} (0 \le i \le m, 0 \le j \le k)$$

のこと。

A_{ij} を求めるのは、一般には $\dim C$ に関して NP-完全問題である (最小重みを求める、すなわち $A_{ij} > 0$ となる最小の $i + j$ を求めることさえ NP-完全である。A, Vardy 1997 reference???)。しかし、$M - \dim C$ が大きすぎなければ、分離 MacWilliams 恒等式という反転公式により求めることができる。

C の直交補空間 $C^\perp \subset \mathbb{F}_2^M$ を次で定義する。

$$C^\perp := \{\mathbf{y} \in \mathbb{F}_2^M | < \mathbf{x}, \mathbf{y} >= 0 \text{ for all } \mathbf{x} \in C\}.$$

ここで、内積は次のように定義されている。

$$< (x_1, \ldots, x_M), (y_1, \ldots, y_M) >:= \sum_{i=1}^{M} x_i y_i.$$

C の分離重み数え上げ多項式を

$$W_C(x, y, X, Y) := \textstyle\sum_{0 \le i \le m, 0 \le j \le k} A_{ij} x^{m-i} y^i X^{k-j} Y^j,$$

で定義する。

定理 6.1. (分離 MacWilliams 恒等式)

$$W_C(x, y, X, Y) = \tfrac{1}{\#(C^\perp)} W_{C^\perp}(x + y, x - y, X + Y, X - Y).$$

もし $\dim C^{\perp}(= M - \dim C)$ が小さければ、右辺は総当り法で計算できる。よって左辺の $W_C(x, y, X, Y)$ がもとまり、A_{ij}、$p_{k,m}(t|s)$ がもとまる。

いままで、そしてこれから扱う例はすべてこのタイプであり、

$$\dim C^{\perp} \leq 8$$

であるが $\dim C$ は 521 など大きな値をとる。（全数チェックは不可能）

- 実験的検定では捕らえにくい、生成法の微妙な優劣をつけることができる
- $G(S)^{\perp}$ に入る、重みの小さなベクトルが分布を損ねる主要項

  $\Rightarrow$ 3 項、5 項式の分布が悪い

  悪いことは古くから知られていたが、

  - 悪さの度合い
  - 項数や次数を増やしたときの改良度

  を定量的に測る方法はなかった

これらは [5] にて発表されている。MacWilliams 恒等式を使って、条件付きではない 1 の個数の二項分布からのずれをテストする話は [11] に出ている。

6.5 MacWilliams 恒等式の証明

6.5.1 離散フーリエ変換

$V := \mathbb{F}_2^M, W := \mathbb{F}_2^M$ とおく。そして

$$e : V \times W \to \{\pm 1\}, \quad (v, w) \mapsto e(v|w) := (-1)^{<v,w>}$$

と定義する。ここに、$< v, w >$ は上で定義した内積である。

R を任意の環とする。任意の写像

$$f : V \to R$$

に対し、その離散フーリエ変換 $\widehat{f}$ を

$$\widehat{f} : W \to R, \quad \widehat{f}(w) := \sum_{v \in V} f(v) e(v|w)$$

によって定義する。($\pm 1 \in R$ により、$e(v|w) \in R$ とみなせることに注意。)

例 6.2. $V = W = \mathbb{F}_2$ とし、$R = \mathbb{Q}[x, y]$, $f : V \to R$ を

$$f(0) = x, \quad f(1) = y$$

で定義する。すると、

$$\widehat{f}(0) = \sum_{v=0,1} f(v)e(v|0) = f(0) + f(1) = x + y,$$
$$\widehat{f}(1) = \sum_{v=0,1} f(v)e(v|1) = f(0) - f(1) = x - y$$

となる。

$C \subset V$ を部分線形空間とし、$C^\perp \subset W$ を先と同じように

$$C^\perp := \{w \in W | < v, w >= 0 \text{ for all } v \in C\}$$

で定義する。

定理 6.3. (Poisson の公式)

$$\sum_{v \in C} f(v) = \frac{1}{\#C^\perp} \sum_{w \in C^\perp} \widehat{f}(w)$$

証明.

$$\begin{aligned} \sum_{w \in C^\perp} \widehat{f}(w) &= \sum_{w \in C^\perp} \sum_{v \in V} f(v)e(v|w) \\ &= \sum_{v \in V} f(v)(\sum_{w \in C^\perp} e(v|w)) \\ &= \sum_{v \in C} f(v)\#(C^\perp) \end{aligned}$$

で、両辺を $\#(C^\perp)$ で割ればよい。

最後の等式には、次の補題を用いた。 □

補題 6.4.

$$\sum_{w \in C^\perp} e(v|w) = \begin{cases} 0 & \text{もし } v \notin C \\ \#(C^\perp) & \text{もし } v \in C \end{cases}$$

証明. 下半分は、$e(v|w) = 1$ となることから従う。上半分を示す。もし $v \notin C$ ならば、線形写像

$$C^\perp \to \mathbb{F}_2, \quad w \mapsto < v, w >$$

は 0 写像でない。よって像は $\mathbb{F}_2$ 全体となり、0 の逆像と 1 の逆像のサイズは同じとなる。すなわち $e(v|w)$ の和は 0。よって上半分が言える。 □

レポート問題 18. 上の証明を完成させよ。ヒント：群準同型 $f : G_1 \to G_2$ があるとき、$x \in G_1$ に対して $f(x)$ の f による逆像は $x + \mathrm{K}erf$ である。

さて、環 R として

$$\mathbb{Q}[x_1, x_2, \ldots, x_M, y_1, y_2, \ldots, y_M]$$

をとり、関数 $f : V \to R$ として

$$f(v_1, \ldots, v_M) := f_1(v_1)f_2(v_2)\cdots f_M(v_M)$$

とおく。（各 v_i は $\mathbb{F}_2$ の元。）ここに、$f_i : \mathbb{F}_2 \to R$ は

$$f_i(0) = x_i, f_i(1) = y_i$$

で定義される (例 6.2 参照)。すると

$$\tilde{W}_C(x_1, y_1, \ldots, x_M, y_M) := \sum_{v \in C} f(v)$$

は、なんというか、C の各元について対応する単項の M 次式を作って足したものである。

命題 6.5. $\tilde{W}$ に次のような代入

$$\begin{aligned} &x_1 = x_2 = \cdots = x_m := x, \quad y_1 = y_2 = \cdots = y_m := y, \\ &x_{m+1} = x_{m+2} = \cdots = x_{m+k} = X, y_{m+1} = y_{m+2} = \cdots = y_{m+k} = Y \end{aligned} \tag{7}$$

を行うと

$$W_C(x, y, X, Y)$$

が得られる。

証明. $\tilde{W}$ の項

$$f_1(v_1)f_2(v_2)\cdots f_M(v_M)$$

のうちで、上の代入を行って

$$x^{m-i}y^iX^{k-j}Y^j$$

という形になるものとは、

「$v_1, \ldots, v_m$ の中にちょうど i 個の 1 を持ち、$v_{m+1}, \ldots, v_{m+k}$ の中にちょうど j 個の 1 を持つもの」

すなわち $\mathrm{wt}_o(v) = i, \mathrm{wt}_f(v) = j$ となるような v である。$\tilde{W}$ は $v \in C$ に渡るこれらの項の和であるから、このような項の数は A_{ij} に一致する。従って、代入後は

$$\sum A_{ij}x^{m-i}y^iX^{k-j}Y^j = W_C(x, y, X, Y)$$

に一致する。 □

Poisson の公式 6.3 によれば、

$$\tilde{W} := \sum_{v \in C} f(v) = \frac{1}{\#C^{\perp}} \sum_{w \in C^{\perp}} \widehat{f}(w)$$

である。よって、右辺の $\sum_{w \in C^{\perp}} \widehat{f}(w)$ に同様の代入をしたものが

$$W_{C^{\perp}}(x+y, x-y, X+Y, X-Y)$$

に一致することを示せば、分離 MacWilliams 恒等式が導き出される。

今から右辺を計算する。

ここで再び一般の $f : V \to R$ を考え、$V = V_1 \oplus V_2$ と直和に分解されているとする。f もこれにそって分解される、すなわちある $f_1 : V_1 \to R$ と $f_2 : V_2 \to R$ が存在して

$$f(v_1 \oplus v_2) = f_1(v_1) f_2(v_2)$$

となっているとする。

補題 6.6.

$$\widehat{f}(w_1 \oplus w_2) = \widehat{f_1}(w_1) \widehat{f_2}(w_2)$$

ここに、

$$W = W_1 \oplus W_2 = V_2^{\perp} \oplus V_1^{\perp}$$

であり、$\widehat{f_i} : W_i \to R$ である。

レポート問題 19. 上の事実を証明せよ。

さっきまで考えていた f に戻る。分解

$$f(v_1, \ldots, v_M) = f_1(v_1) f_2(v_2) \cdots f(v_M)$$

$$V = \mathbb{F}_2 \oplus \mathbb{F}_2 \oplus \cdots \oplus \mathbb{F}_2 = V_1 \oplus \cdots \oplus V_M$$

(すなわち V_i は i 番目の成分と対応) に上の補題を用いると、内積の定義から

$$W = W_1 \oplus W_2 \oplus \cdots \oplus W_M$$

も成分の分解に一致し、例 6.2 でみたように

$$\widehat{f_i}(0) = f_i(0)e(0|0) + f_i(1)e(1|0) = x_i + y_i$$
$$\widehat{f_i}(1) = f_i(0)e(0|1) + f_i(1)e(1|1) = x_i - y_i$$

となる。いいかえると、$\widehat{f_i}(w_i)$ は、$f_i(w_i)$ に代入：$x_i \leftarrow x_i + y_i$, $y_i \leftarrow x_i - y_i$ を行ったものである。

上の補題と合わせて

$\widehat{f}(w_1,\ldots,w_M) = \widehat{f_1}(w_1)\cdots\widehat{f_M}(w_M)$
$= f_1(w_1)\cdots f_M(w_M)$ の x_i に x_i+y_i を、y_i に x_i-y_i を代入したもの

である。従って、

$\sum_{w\in C^\perp}\widehat{f}(w) = \sum_{w\in C^\perp}(f(w)$ に上の代入をしたもの$)$
$= \tilde{W}_{C^\perp}(x_1+y_1, x_1-y_1,\ldots,x_M+y_M,x_M-y_M)$

であるから次がいえた。

定理 6.7. (一般化 MacWilliams 恒等式)

$$\tilde{W}_C(x_1,y_1,\ldots,x_M,y_M) = \frac{1}{\#(C^\perp)}\tilde{W}_{C^\perp}(x_1+y_1,x_1-y_1,\ldots,x_M+y_M,x_M-y_M).$$

(7) のような代入をすると、分離 MacWilliams 恒等式が得られる。

レポート問題 20. 上の証明を完成させよ。

注意 6.8. $x_1=\cdots=x_M=x, y_1=\cdots=y_M=y$ とおいたものがもともとの MacWilliams 恒等式であるが、どのバージョンも MacWilliams 恒等式と呼ぶのが普通である。一般化 MacWilliams 恒等式については [7, P.147, Theorem 14] 分離 MacWilliams 恒等式については [7, P.158, Eq.(52)] を参照。

6.5.2 どこがフーリエ変換やねん

f を実数上定義された周期 1 の可積分複素数値関数とする。言い換えると、

$$f:\mathbb{R}/\mathbb{Z}\to\mathbb{C}$$

である。このとき、f のフーリエ展開は

$$f(x)=\sum_{n\in\mathbb{Z}}a_n\exp(2\pi inx)$$

であった。

注意 6.9. f が実数値関数である場合には $f(x)=\overline{f(x)}$ より $a_n=\overline{a_{-n}}$ であり、

$$f(x)=C+\sum_{n\in\mathbb{N}}(s_n\cos nx+t_n\sin nx)$$

という通常のフーリエ展開が得られる。

さて、a_n は

$$a_n = \int_{\mathbb{R}/\mathbb{Z}} f(x)e^{-2\pi inx}dx$$

で求まるのであった。これは、

$$\int_{\mathbb{R}/\mathbb{Z}} e^{2\pi imx}e^{-2\pi inx}dx = \begin{cases} 0 & (m \neq n) \\ 1 & (m = n) \end{cases}$$

に起因している。a_n を

$$a : \mathbb{Z} \to \mathbb{C}, \quad a(n) := a_n$$

なる関数だと思えば、

$$f : \mathbb{R}/\mathbb{Z} \to \mathbb{C}$$

の情報と

$$a : \mathbb{Z} \to \mathbb{C}$$

の情報とが変換公式

$$a(n) = \int_{\mathbb{R}/\mathbb{Z}} f(x)e^{-2\pi inx}dx$$

$$f(x) = \sum_{n\in\mathbb{Z}} a(n)e^{2\pi inx}$$

で移りあうのである。

さて、$\mathbb{C}_1$ で絶対値 1 の複素数が積についてつくる群を表し、

$$e : \mathbb{R}/\mathbb{Z} \times \mathbb{Z} \to \mathbb{C}_1, \quad (x, n) \mapsto e(x|n) := \exp(2\pi inx)$$

とおくとこれは well-defined である。そして、次の性質を持つ。

1. e は連続双準同型である、すなわち任意に $x_0 \in \mathbb{R}/\mathbb{Z}$ を固定したとき

   $$e(x_0|-) : \mathbb{Z} \to \mathbb{C}_1, \quad n \mapsto e(x_0|n)$$

   は連続群準同型であり、また n_0 を固定したとき

   $$e(-|n_0) : \mathbb{R}/\mathbb{Z} \to \mathbb{C}_1, \quad x \mapsto e(x|n_0)$$

   は連続群準同型である

2. $$\mathbb{R}/\mathbb{Z} \to \mathrm{H}om(\mathbb{Z}, \mathbb{C}_1), \quad x_0 \mapsto e(x_0|-)$$

   は一対一写像である。（上から、群同型である）

3.

$$\mathbb{Z} \to \mathrm{H}om(\mathbb{R}/\mathbb{Z}, \mathbb{C}_1), \quad n_0 \mapsto e(-|n_0)$$

は一対一写像である。（上から、群同型である）

このような性質をもつ位相可換群 V, W と e を、直交群対という。(上で、$V = \mathbb{R}/\mathbb{Z}$, $W = \mathbb{Z}$ として抽象化したもの。)

ある位相可換群 G がもし直交群対になるとしたら、お相手は連続群準同型全体のなす群

$$\widehat{G} := \mathrm{H}om(G, \mathbb{C}_1)$$

にならざるを得ない（いわゆる連続指標群)。

定理 6.10. (Pontryagin の双対定理) G が局所コンパクトアーベル群ならば G と $\widehat{G}$ は直交群対になる。

例としては先の $\mathbb{R}/\mathbb{Z}, \mathbb{Z}$ のほかに、

$$\mathbb{Z}/M, \mathbb{Z}/M, e(n|m) := \exp(2\pi i nm/M)$$

があげられる。また、

$$\mathbb{F}_2^M, \mathbb{F}_2^M, e(v|w) := \exp(2\pi i < v, w > /2) = (-1)^{<v,w>}$$

があげられる (§6.5.1 離散フーリエ変換を参照)。

注意 6.11. 一般に、有限アーベル群の指標群はそれ自身と同型である。が、これらの間に標準的な同型はない。

位相群といったら、ハウスドルフは仮定する。

さて、局所コンパクトアーベル群には Haar 測度と呼ばれる群作用不変な測度が定数倍を除いて唯一つさだまる。大体、「コンパクトな部分集合に対して体積を定める方法」があると思えばよい。$\mathbb{R}/\mathbb{Z}$ では通常の長さが体積であり、$\mathbb{Z}$ では有限集合の元の個数が体積である。

直交群対 V, W, e を考える。$f : V \to \mathbb{C}$ に対して $\widehat{f} : W \to \mathbb{C}$,

$$\widehat{f}(w) := \int_V f(v)e(v|w)dv$$

を f のフーリエ変換という。ここで dv が V の Haar 測度である。

$V = \mathbb{F}_2^M$ のとき、この式は §6.5.1 で定義された $\widehat{f}$ に一致している。（離散集合上の積分は、単なる和であるから。）

$V = \mathbb{R}/\mathbb{Z}$ のとき $W = \mathbb{Z}$ であり、$f : V \to \mathbb{C}$ に対し

$$\widehat{f}(n) = \int_{\mathbb{R}/\mathbb{Z}} f(x)e(x|n)dx \quad = a(-n) \text{ を求める式}$$

である。

$V = \mathbb{Z}$ のとき $W = \mathbb{R}/\mathbb{Z}$ であり、$a : V \to \mathbb{C}$ に対し

$$\widehat{a}(x) = \sum_{n \in \mathbb{Z}} a(n) e(x|n)$$

である。

直交群対の枠組みで、Poisson の公式は次のように述べられる。$C \subset V$ を閉部分群とすると、あるゆるい条件の下で

定理 6.12. (Poisson の公式)

$$\int_C f(v) dv|_C = \int_{C^\perp} \widehat{f}(w) dw|_{C^\perp}.$$

この式の正確な定式化はここではしない (測度の定数倍の自由度と、正確な条件が面倒、[14, Theorem 5.5.2] とその前後を参照) が、これの有限群の場合が定理 6.3 である。

以上、フーリエ変換は局所コンパクトアーベル群一般にあらわれる理論であり、離散フーリエ変換とは有限アーベル群に対してそれを用いたものなのである。

なお、この $\mathbb{R}/\mathbb{Z}$ と $\mathbb{Z}$ の双対性に関する Poisson の公式は、`random()` や `ran_array()` の出力を実数だと思って M 個ずつ足したものの分布を計算するのに用いられている [12]。

ついでに、フーリエの反転公式は

$$\widehat{\widehat{f}}(x) = f(-x)$$

と述べられる。これが、関数のフーリエ展開を与える式である。

7 乱数のテスト：高次元均等分布性と数の幾何

7.1 v ビット精度での k 次元均等分布性

定義 7.1. ある擬似乱数発生器が v ビット精度で k 次元均等分布しているとは、擬似乱数発生器を一周期走らせたとき、出力の連続する k ワードの上位 v ビットが、可能な 2^{kv} 種類全部を同じ回数ずつ実現すること。

線形漸化式を用いる都合により、「全部 0」というパターンは他より一回少なくてもいいとする。

全ビット見れば、すなわち $v = w$ ならば単なる k 次元均等分布に一致する (§5.2 参照)。

もし出力を w ビット整数だと思い、2^w で割って $[0, 1)$ 区間に入れるとすると、「k 次元単位立方体に一周期に渡って点を打ったとき、各辺を 2^v 等分して得られる小立方体に打たれる点の数がどの小立方体でも同じ」ということ。

定理 7.2. 擬似乱数の出力のうち、上記の kv ビットだけを見る。これは、状態の集合から kv ビットへの写像

$$G : S \to \mathbb{F}_2^{kv}$$

である。もし S が $\mathbb{F}_2$ 線形空間で G が線形写像であり、状態遷移が最大周期 (0 を除く全てを通る) ならば、v ビット精度 k 次元均等分布性は G の全射性と同値である。

レポート問題 21. 上の定理を証明せよ。

v ビット精度での均等分布の次元とは、上が全射となる最大の k をいい $k(v)$ であらわす。

自明な上限として、

$$k(v)v \le \dim S$$

を得る。各 $v = 1, 2, \ldots, w$ について

$$k(v) = \lfloor \dim S/v \rfloor$$

のとき、各ビットでの高次元均等分布性が最良であるという。

TGFSR, MT はそのままでは上の上限から程遠い。それぞれ $\dim S = nw, nw - r$ であるが、$v = 1$ では $k(1) = \dim S$ であるが、$v = 2$ では $k(2) = n < nw/2$ である。

また、ワード中のどの連続する 3 ビットも n 次元しか均等分布していない。

そのため、TGFSR, MT では調律 (tempering) と呼ぶ変換をおこなう。これは、ある w 次正方行列 T を用意し、生成されたワード $\mathbf{x}$ に対し $\mathbf{x}T$ を出力するというものである。実際には、

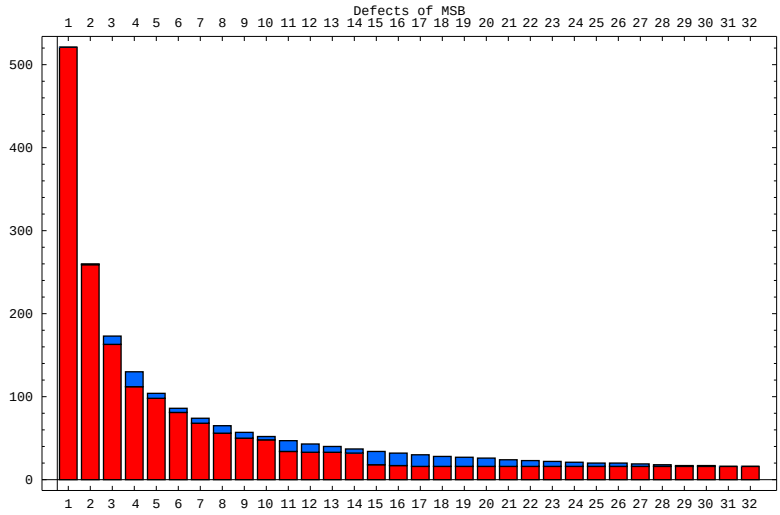
$$\mathbf{y} \leftarrow \mathbf{x} + (\mathbf{x} \text{ の数ビット左シフト}) \ \& \ \text{あるワード}$$

という形の変換を二度施し、最後に

$$\mathbf{y} \leftarrow \mathbf{x} + (\mathbf{x} \text{ の数ビット右シフト})$$

を施す。

これをしても最良にはならないが、かなり最良に近づいている。



2^{16} 個の小 MT(MT521) について、tempering を行った。各 $v = 1, \ldots, 32$ につき、2^{16} 個中最悪の $k(v)$ をグラフにした。

Table II. Parameters and k-distribution of Mersenne Twisters

Generator		The order of equidistribution					
ID	Parameters	$k(1)$	$k(2)$	$k(3)$	$k(4)$	$k(5)$	$k(6)$
		$k(7)$	$k(8)$	$k(9)$	$k(10)$	$k(11)$	$k(12)$
(the number of		$k(13)$	$k(14)$	$k(15)$	$k(16)$	$k(17)$	$k(18)$
terms in the		$k(19)$	$k(20)$	$k(21)$	$k(22)$	$k(23)$	$k(24)$
characteristic		$k(25)$	$k(26)$	$k(27)$	$k(28)$	$k(29)$	$k(30)$
polynomial)		$k(31)$	$k(32)$				
	Upper bounds	11213	5606	3737	2803	2242	1868
	$\lfloor \frac{nw-r}{v} \rfloor$ for	1601	1401	1245	1121	1019	934
	$(w, n, r) = (32, 351, 19)$	862	800	747	700	659	622
	$1 \le v \le 32$	590	560	533	509	487	467
		448	431	415	400	386	373
		361	350				
MT11213A	$(w, n, m, r) = (32, 351, 175, 19)$	11213	5606	3560	2803	2111	1756
	a = E4BD75F5	1405	1401	1055	1053	709	704
	u = 11	703	702	701	700	356	352
	s = 7,b = 655E5280	351	351	351	350	350	350
(177)	t = 15,c = FFD58000	350	350	350	350	350	350
	l = 17	350	350				
MT11213B	$(w, n, m, r) = (32, 351, 175, 19)$	11213	5606	3565	2803	2113	1759
	a = CCAB8EE7	1408	1401	1056	1053	715	704
	u = 11	702	702	701	700	355	352
	s = 7,b = 31B6AB00	351	351	351	351	350	350
(151)	t = 15,c = FFE50000	350	350	350	350	350	350
	l = 17	350	350				
	Upper bounds	19937	9968	6645	4984	3987	3322
	$\lfloor \frac{nw-r}{v} \rfloor$ for	2848	2492	2215	1993	1812	1661
	$(w, n, r) = (32, 624, 31)$	1533	1424	1329	1246	1172	1107
	$1 \le v \le 32$	1049	996	949	906	866	830
		797	766	738	712	687	664
		643	623				
MT19937	$(w, n, m, r) = (32, 624, 397, 31)$	19937	9968	6240	4984	3738	3115
	a = 9908B0DF	2493	2492	1869	1869	1248	1246
	u = 11	1246	1246	1246	1246	623	623
	s = 7, b = 9D2C5680	623	623	623	623	623	623
(135)	t = 15, c = EFC60000	623	623	623	623	623	623
	l = 18	623	623				
TT800	$(w, n, m, r) = (32, 25, 7, 0)$	800	400	250	200	150	125
	a = 8EBFD028	100	100	75	75	50	50
	u : not exist	50	50	50	50	25	25
	s = 7, b = 2B5B2500	25	25	25	25	25	25
(93)	t = 15, c = DB8B0000	25	25	25	25	25	25
	l =16	25	25				
ran_array	Knuth's new recommendation.	129	64	43	32	25	21
		18	16	14	12	11	10
	Here we list the trivial	9	9	8	8	7	7
	upper bounds.	6	6	6	5	5	5
		5	4	4	4	4	4

tempering のパラメータは、$k(v)$ を計算しながら上位ビットより順次 try-and-error で決めていく。

全射性は、行列の掃き出しを行えばよい。が、19937 次正方行列の掃き出しは意外に計算機でも大変である。

形式冪級数体の格子の幾何を用いると効率よく計算できる ([1])。

7.2 形式冪級数体の利用

$\mathbb{F}_2$ 線形な擬似乱数を考える。初期状態が $s \in S$ であったとしよう。その出力のとある v ビットに着目し、$k(v)$ を求めたい。

出力が

$$(x_{10}, x_{20}, \ldots, x_{v0}), (x_{11}, x_{21}, \ldots, x_{v1}), \ldots$$

であるとき、形式冪級数環 $A := \mathbb{F}_2[[t]]$ 係数のベクトル

$$w(s) := (\sum_{i=0}^{\infty} x_{1i}t^i, \sum_{i=0}^{\infty} x_{2i}t^i, \ldots, \sum_{i=0}^{\infty} x_{vi}t^i)$$

を考える。状態集合を S とすれば、

$$w : S \to A^v$$

である。

$F := \mathbb{F}_2((t))$ とおく。F は A の商体であり、

$$|\sum_{i=-m}^{\infty} a_i t^i| := 2^m \quad (a_{-m} \neq 0)$$

と定義するとノルムとなる。F^v には sup ノルム

$$||(x_1, \ldots, x_v)|| := \max_{i=1,2,\ldots,v} \{|x_i|\}$$

を導入する。単位球は立方体で A^v となる。三角不等式より強い、次の不等式（ウルトラノルムという）をみたす。

$$||\mathbf{x} + \mathbf{y}|| \leq \max\{||\mathbf{x}||, ||\mathbf{y}||\}$$

v 次元ベクトル $e_i := (0, \ldots, 0, 1/t, 0, \ldots, 0)$ (i 番目が $1/t$) を考えると、F^v は $\mathbb{F}_2$ 線形空間としての直和

$$F^v = A^v + \mathbb{F}_2[t^{-1}] < e_1, e_2, \ldots, e_v >$$

となる。ここに右辺の後半は、e_i たちの生成する $\mathbb{F}_2[t^{-1}]$-自由加群。

定理 7.3. 擬似乱数が最大周期性を持つとする。0 でない $s_0 \in S$ を一つ取り、$\{e_1, e_2, \ldots, e_v, w(s_0)\}$ が生成する $\mathbb{F}_2[t^{-1}]$ 加群を $L \subset F^v$ とする。すると

$$w(S) = L \cap A^v$$

が成立する。

証明. s_0 から m 回動いた状態を s_m とすると

$$w(s_m) = t^{-m} w(s_0) \text{ の負冪の項を除いたもの}$$

である。よって

$$w(s_m) \subset L.$$

$w(s_m) \in A^v$ は自明。最大周期の仮定より m を動かすと $w(s_m)$ は $w(S) - \{0\}$ の元を全てつくす。よって $\subset$ が言えた。

逆に、$\mathbb{F}_2$ 線形空間としての直和

$$L' := w(S) + \mathbb{F}_2[t^{-1}] < e_1, e_2, \ldots, e_v >$$

は t^{-1} 倍について閉じている。(右の項は $<>$ の中身を基底とする $\mathbb{F}_2[t^{-1}]$ 加群。) よって $\mathbb{F}_2[t^{-1}]$ 加群。$w(s_0) \in w(S)$ より $L' \supset L$。両辺と A^v の共通部分をとると $\supset$ が言える。 □

F^v の元の有限集合にたいし、そのノルムをその中の元のノルムの最大値と定義する。

L の $\mathbb{F}_2[t^{-1}]$ 基底 B のなかで、ノルムが最小のものを最短基底とよぶ。

定理 7.4. L の最短基底のノルムは $2^{-k(v)+1}$ である。

証明のためいくつか準備をする。

定義 7.5. $\mathbb{F}_2[t^{-1}]$ 格子 $L \subset F^v$ に対し、各 L の元をそれを中心とする半径 r の閉球に置き換えると全空間 F^v を覆うとき、L の被覆半径は r 以下であるという。(そのような r の最小値を L の被覆半径という。)

補題 7.6. 最大周期の仮定のもとで、出力が少なくとも k 次元均等分布するということと、L の被覆半径が 2^{-k} 以下であることとは同値である。

証明. 被覆半径が 2^{-k} 以下だとしよう。いま、任意の kv ビットパターンを考える。これを冪級数化して (k 項め$=t^{k-1}$ の項で打ち止めになっている) A^v の元 x をつくる。L の元 x' で誤差 2^{-k} で x が近似できる。すると自動的に $x' \in A^v$、すなわち

$$x' \in L \cap A^v, |x - x'| \leq 2^{-k}.$$

このとき x' の k 項目までは x に一致している。先の定理により $x' \in w(S)$。すなわち、いつかは与えられた kv ビットパターンが現れる。よって k 次元均等分布している。

逆に、任意の kv ビットパターンを実現できるということは、$L \cap A^v$ の各元を半径 2^{-k} の球で膨らませると A^v を覆うということに他ならない。$\mathbb{F}_2$ 線形空間としての直和

$$F^v = A^v + \mathbb{F}_2[t^{-1}] < e_1, e_2, \ldots, e_v >$$

と $L \supset \mathbb{F}_2[t^{-1}] < e_1, e_2, \ldots, e_v >$ を用いれば、L の被覆半径は 2^{-k} 以下とわかる。 □

証明. (定理 7.4 の証明)

上の補題により、

$$\text{最短基底のノルムが } 2^{-k+1} \text{ 以下} \Leftrightarrow L \text{ の被覆半径が } 2^{-k} \text{ 以下}$$

を言えばよい。よって、

$$\text{ノルムが } 2^{-k+1} \text{ 以下の基底 } B \text{ が存在} \Leftrightarrow L \text{ の被覆半径が } 2^{-k} \text{ 以下}$$

を言えばよい。左から右をしめす。B は基底であり、L は最大次元格子であるから F 上 B は F^v を生成する。よって任意のベクトル $x \in F^v$ は

$$x = \sum_i a_i b_i, \quad a_i \in F,\ B = \{b_1, \ldots, b_v\}$$

と書ける。a_i に最も近い $\mathbb{F}_2[t^{-1}]$ の元を α_i とすれば、$|a_i - \alpha_i| \leq 1/2$ で

$$||x - \sum_i \alpha_i b_i|| = ||\sum_i (a_i - \alpha_i) b_i|| \leq 1/2||B|| \leq 2^{-k}$$

であり、右が言えた。

右から左を言う。原点中心で半径 2^{-k+1} の球を考える。まず、この球に入る L の点であって F^v の基底になるものがあることを示す。$e_i = (0, \ldots, 0, 1/t, 0, \ldots, 0)$ とおく。被覆半径の定義から、

$$||t^k e_i - \ell_i|| \leq 2^{-k} \tag{8}$$

なる $\ell_i \in L$ がある。$||t^k e_i|| = 2^{-k+1}$ だから超ノルム性より

$$||\ell_i|| = 2^{-k+1}$$

が成立する。$\ell_1, \ldots, \ell_v$ が一次独立であることを示せばよい。(8) より

$$t^k e_i \equiv \ell_i \mod t^k,$$

よって　mod t^k で見て ℓ_i は $(0,0,\ldots,0,t^{k-1},0,\ldots,0)$ に一致する。

今 ℓ_i たちが一次従属であったとして、非自明な一次結合を

$$\sum_i a_i \ell_i = 0, \quad a_i \in F$$

とおく。t 冪と定数を適当に賭けることで $a_i \in A$ で、どれか一つは定数項が 1 としてよい。a_1 がそれであるとしても一般性を失わない。すると、上の等式を mod t^k でみると、左辺は $(t^{k-1},*,\ldots,*)$ となり、矛盾である。

こうして、中心 0 半径 2^{-k+1} の球の中の L の点で、一次独立なものが v 本とれることがわかった。そういったものの中で、ノルムの和が最小のもの B をとる (なぜ存在するか？$w(S)$ の元は、0 が続くとしても特性多項式の次数は超えないからである。別の言い方をすると、$w(S) \in \varphi(t)^{-1}A^v$ より従う。ここに、$\varphi(t)$ は漸化式の特性多項式である。よって、0 でないベクトルのノルムは $2^{-\deg(\varphi(t))}$ 以上になる)。

これが L の基底となることを示そう。L の任意の元 x をとる。これが B の元の $\mathbb{F}_2[t^{-1}]$ 係数一次結合でかければよい。B の元の F 係数一次結合で x を書く。B のはる $\mathbb{F}_2[t^{-1}]$ 格子の元で x を平行移動してもよい。よって、平行移動した結果 x は B の元の tA 係数一次結合であるとしてよい。$x = 0$ ならばそれでよい。そうでないとする。係数が 0 でない B の元のうち、ノルムの最大なものを考える。係数が tA に入るのだから、それは x よりノルムが大きい。B からそれを抜いて x を加えても、一次独立なままでノルムは短くなっている。よって最小性に反する。 □

格子の生成元が与えられたとき、その最小基底は Lenstra[6] のアルゴリズムによって求まる。このときかかる計算時間は行列の掃き出しよりもずっと短い。

参考文献

[1] R. Couture, P. L'Ecuyer, and S. Tezuka, On the distribution of k-dimensional vectors for simple and combined Tausworthe sequences, Math. Comp. 60 (1993), 749–761.

[2] Luc Deveroye, Nonuniform random variate generation. Springer-Verlag, 1986.

[3] 伏見正則, 乱数, 東京大学出版会, 1989.

[4] Knuth, D. E. The Art of Computer Programming. Vol. 2. Seminumerical Algorithms 3rd Ed. Addison-Wesley, 1997.

[5] Haramoto, H., Matsumoto, M., Nishimura, T. "Computing conditional probabilities for $\mathbb{F}_2$-linear pseudorandom bit generator by splitting MacWilliams identity", International Journal of Pure and Applied Mathematics, Vol.38 No.1, 2007.

[6] Lenstra, A. K. Factoring multivariate polynomials over finite fields. J. Comput. System Sci. 30, 235-248.

[7] F.J. MacWilliams and N.J.A. Sloane, The theory of error correcting code. North-Holland, 1977.

[8] Matsumoto, M. and Kurita, Y. "Twisted GFSR Generators," ACM Transactions on Modeling and Computer Simulation **2** (1992), 179–194.

[9] Matsumoto, M. and Kurita, Y. "Twisted GFSR Generators II," ACM Transactions on Modeling and Computer Simulation **4** (1994), 254–266.

[10] Matsumoto, M. and Nishimura, T. "Mersenne Twister: a 623-dimensionally equidistributed uniform pseudo-random number generator" ACM Trans. on Modeling and Computer Simulation **8** (1998), 3–30.

[11] M. Matsumoto and T. Nishimura "A Nonempirical Test on the Weight of Pseudorandom Number Generators" 381–395 in: Monte Carlo and Quasi-Monte Carlo methods 2000, Springer-Verlag 2002.

[12] M. Matsumoto and T. Nishimura "Sum-discrepancy test on pseudorandom number generators" Mathematics and Computers in Simulation, Vol. 62 (2003), pp 431-442.

[13] H. Niederreiter, Random Number Generation and Quasi-Monte Carlo Methods. SIAM, 1992.

[14] Reiter, S. and Stegeman, J.D.: Classical harmonic analysis and locally compact groups. Oxford Science Publications, Oxford, 2000.