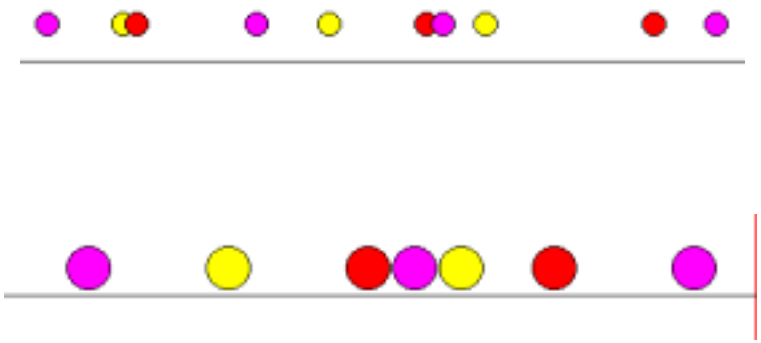
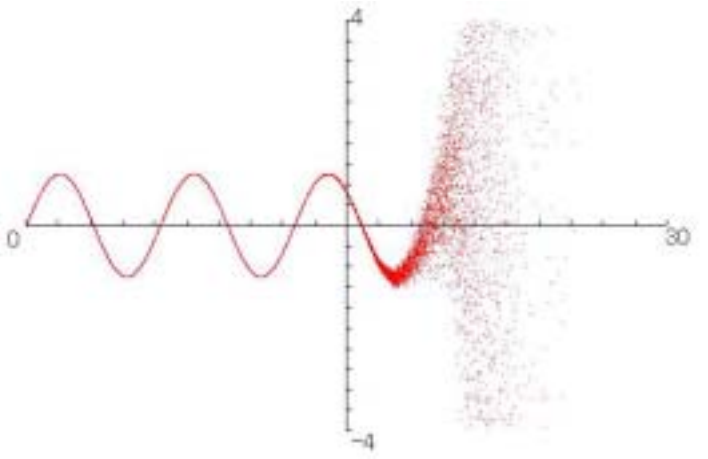
そろそろ まとめの講義をします

～ 無限粒子系，乱数生成とハッシュ関数 ～

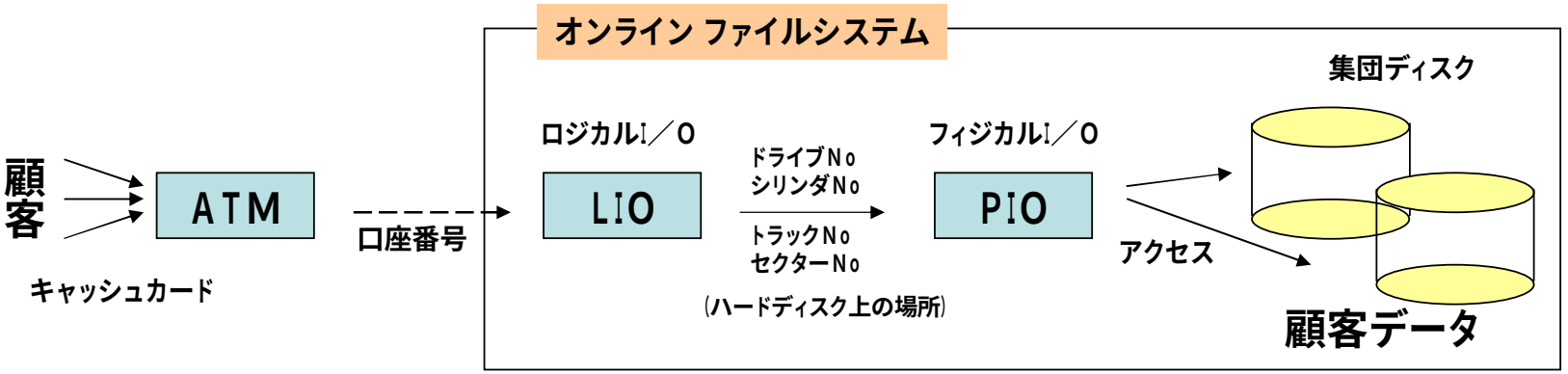
三重大学 教育学部　谷口礼偉





【１．数学との馴れ初め】

東山線が「藤が丘」まで開通して暫くたったころ（４０年以上前），
愛知県のある金融機関のオンラインシステムのファイルシステムを担当。



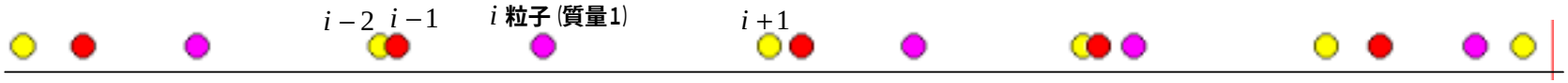
● ディスク内に顧客データを均一に分散させるために，
口座番号を，**乱数** を使って作った。

● このファイルシステムは1秒間に何件のデータを処理できるか？

⇒ハードウェアシミュレーション，モンテカルロシミュレーション，

待ち行列理論　ボレル集合体？　ポアソン入力？
M／M／s？　$\rho/(1-\rho)$ ？

【２．古典力学の運動方程式に従う無限個の粒子】



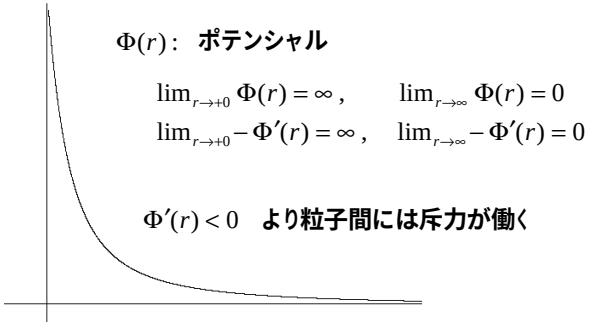
(q_i, v_i) : i 番目の粒子 (質量1) の位置と速度

初期配置 $x = (q_i, v_i)_i, \quad \cdots < q_{-1} < 0 \le q_0 < q_1 < \cdots$

時刻 t での配置 $x(t) = (q_i(t), v_i(t))_i$

ニュートンの運動方程式

$$(3.1)\quad \begin{cases} \dfrac{dq_i(t)}{dt} = v_i(t) \\ \dfrac{dv_i(t)}{dt} = -\Phi'(q_i(t) - q_{i-1}(t)) + \Phi'(q_{i+1}(t) - q_i(t)) \\ (q_i(0), v_i(0))_i = (q_i, v_i)_i \end{cases}$$



？ どのような初期配置 $(q_i(0), v_i(0))_i = (q_i, v_i)_i$ **に対して，(3．1) は永続的な解を持つか？**

[それまでの結果] ハードコア粒子 (一定巾の硬い核を持っている粒子) に対して調べられていた。

(各粒子は一定巾の核を持っているので，有限区間に入る粒子数は制限される)

(続き) 【２．古典力学の運動方程式に従う無限個の粒子】

A

$\Delta_n = [-n, n]$ 区間

区間内のエネルギー

$N(x;\Delta_n) = \#\{i \mid q_i \in \Delta_n\}$ 区間内の粒子数

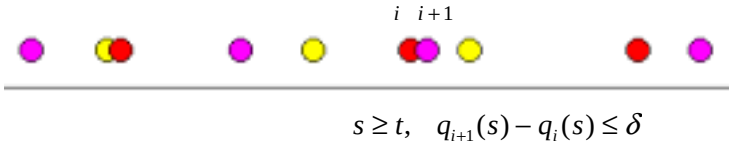
$H(x;\Delta_n) = 2^{-1}\sum_{q_i\in\Delta_n} v_i^2 + \sum_{q_i\in\Delta_n \text{ or } q_{i+1}\in\Delta} \Phi(q_{i+1}-q_i)$

$$X_\gamma = \{\, x \mid \sup_{n=1,2,\cdots}(2n)^{-1}N(x;\Delta_n) < \infty,\ \sup_{n=1,2,\cdots}(2n)^{-1-\gamma}H(x;\Delta_n) < \infty\},\quad 0 \le \gamma < 1$$

平均粒子数　　平均エネルギー

ある γ に対して $x \equiv (q_i, v_i)_i \in X_\gamma$ ならば，(3．1) は 以下を満たす解を持つ。

$$\begin{cases} \text{(i)} & \cdots < q_{i-1}(t) < q_i(t) < q_{i+1}(t) < \cdots, \\ \text{(ii)} & (q_i(t), v_i(t))_i \in X_\gamma, \\ \text{(iii)} & \exists \delta > 0 \ \text{ s.t. } \ \forall i \ \ \forall t \\ & \lim_{k\to\infty} \sigma_{i+k} \circ \cdots \circ \sigma_{i+1} \circ \sigma_i(t) = \infty \qquad \lim_{k\to\infty} \sigma_{i-k} \circ \cdots \circ \sigma_{i-1} \circ \sigma_i(t) = \infty \\ & \text{ただし} \quad \sigma_i(t) = \inf\{s \ge t \mid q_{i+1}(s) - q_i(s) \le \delta\} \end{cases}$$

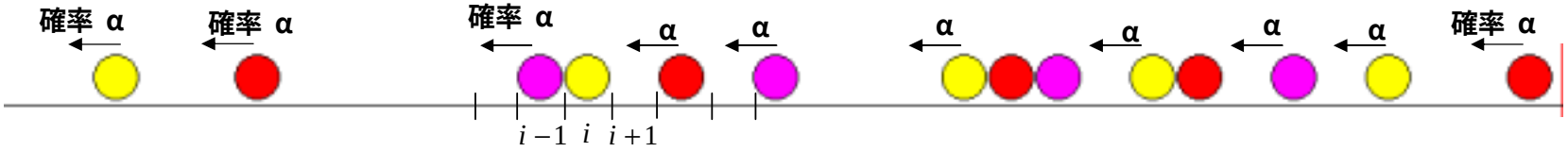


〔証明の考え方〕

有限区間 $\Delta_K = [-K, K]$ の外にある粒子を固定しておいて運動方程式を解く。$(K = 1,2,\cdots)$

それらの解 $\{q_i^K(t)\}_i$ は $t \in [-1,0]$ で，i ごとに一様に有界で，$t = [-1,0]$ に対して同程度連続であるので，Ascoli-Arzela の定理と，対角線論法で，全ての i で収束する部分列を抜き出す。

【３．離散時間 完全非対称 単純 排除過程（交通流の確率モデル）】



無限個の粒子は，時刻が t から t＋1 に変るとき，左側が空いていれば，
それぞれ独立に確率 α (0<α<1) で左に移動する。

Synchronous Totally Asymmetric Simple Exclusion Process, STASEP

$$X=\{0,1\}^{Z} \qquad x\equiv(\cdots x_{-1}x_0x_1\cdots)\in X \quad \begin{cases} x_i=1 & \text{座標 } i \text{ に粒子有り} \\ x_i=0 & \text{座標 } i \text{ に粒子無し} \end{cases}$$

$${}_i[a_i\cdots a_j]_j=\{(\cdots x_{-1}x_0x_1\cdots)\in X \mid x_l=a_l,\ i\le l\le j\}$$

座標 i から j までの配置が $a_i\cdots a_j$ であるような X の要素の全体（筒集合）

$P(x,\ {}_i[a_i\cdots a_j]_j)$ **時刻 t での配置が　　であったとき，**
時刻 t+1 での配置が 集合 ${}_i[a_i\cdots a_j]_j$ に入る確率（推移確率）

X 上の離散時間マルコフ過程 (MP) が定義される

？ 初期配置から出発して，時間が経過したとき，どのような定常状態に落着くか？
（＝ 定常状態に現れる X 上の確率測度 μ を全て決定せよ）

$$\mu(A)=\int_X \mu(dx)P(x,A)$$

［それまでの結果］ない。（ＳＴＡＳＥＰという名称も当時はなかった）

(続き)【 3. 離散時間 完全非対称 単純 排除過程 (交通流の確率モデル) 】

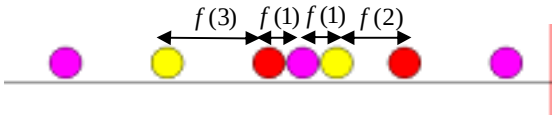
A $\beta,\quad 0<\gamma<\infty$ を, $1-\alpha=(1-\beta)(1-\beta\gamma)$ を満たす数とする。 $(0<\beta<\alpha)$

X 上の確率測度 π_γ を以下で定める:

$$\begin{cases}\pi_\gamma({}_i[a_i\cdots a_{j-1}0]_j)=\left(\dfrac{\gamma}{1+\gamma}\right)(1-\beta)^{\#_{00}}\left(\dfrac{1}{\gamma}\right)^{\#_{10}}\left(\dfrac{1-\beta\gamma}{\gamma}\right)^{\#_{11}}\left(\dfrac{\beta\gamma}{\alpha}\right)^{j-i}\\ \pi_\gamma({}_i[a_i\cdots a_{j-1}1]_j)=\left(\dfrac{1}{1+\gamma}\right)(1-\beta)^{\#_{00}}\left(\dfrac{1}{\gamma}\right)^{\#_{10}}\left(\dfrac{1-\beta\gamma}{\gamma}\right)^{\#_{11}}\left(\dfrac{\beta\gamma}{\alpha}\right)^{j-i}\end{cases}$$

$$\#_{uv}([a_i\cdots a_j]) \;=\; \#\{l\mid a_l a_{l+1}=uv\}$$

$$f(n)=\begin{cases}(1-\beta\gamma)(\beta/\alpha) & n=1\\ \gamma^{-1}(1-\beta)^{n-2}(\beta\gamma/\alpha)^n & n=2,3,\cdots\end{cases}$$



(最終的な状態)

- $\pi_\gamma,\quad 0<\gamma<\infty,$ はマルコフ過程 (MP) の定常確率である。
  - π_γ では, 隣り合う粒子間の距離の分布は $f(n)=\begin{cases}(1-\beta\gamma)(\beta/\alpha) & n=1\\ \gamma^{-1}(1-\beta)^{n-2}(\beta\gamma/\alpha)^n & n=2,3,\cdots\end{cases}$ であり, これらは互いに独立である。(renewal measure)
  - π_γ では, 位置 i に粒子が存在する確率は $\dfrac{1}{1+\gamma}$, 粒子の平均速度は $\beta\gamma$ である。
- $0<\alpha\le 1/2$ の時は以下の定常状態に限られる:

  i) 状態 π_γ ii) 状態 π_0 (粒子が何もない), π_∞ (全ての場所に粒子が詰まっている)

  iii) 状態 Θ_n (座標 n 以下では粒子が全部詰まっている; $n+1$ 以上には粒子が無い)

  iv) 状態 i)〜iii) が混じり合った状態
- $1/2<\alpha<1$ の時は, "に限られる"の部分の証明がまだない。

(a, a, a, a) Synchronous Totally Asymmetric Simple Exclusion Process (STASEP)

This is the discrete-time version of one of the most widely studied interacting particle systems, the Asymmetric Simple Exclusion Process (ASEP). See [Lig] for basic theory of this and related exclusion models in continuous time. Some beautiful recent work on the ASEP appears in [Rez] and [Sep]. The discrete-time STASEP is both a generalization of Rule 184, and a special case of the original Nagel-Schreckenberg model with maximum velocity 1 (see [Nag1], Section IV). The equilibria for the STASEP were first characterized in [Yag]. (See also [SSNI], where these same results were obtained). Under these dynamics there are no persistent jams of the type that interest us. That is, clustering (1.7) cannot arise in the STASEP. Its throughput is given by

$$\theta(\rho) = \frac{1-\sqrt{1-4\alpha\rho(1-\rho)}}{2},$$

which matches (2.1) when $\alpha = 1$.

Journal of Statistical Physics, Vol. 105, Nos. 3/4, November 2001 (© 2001)

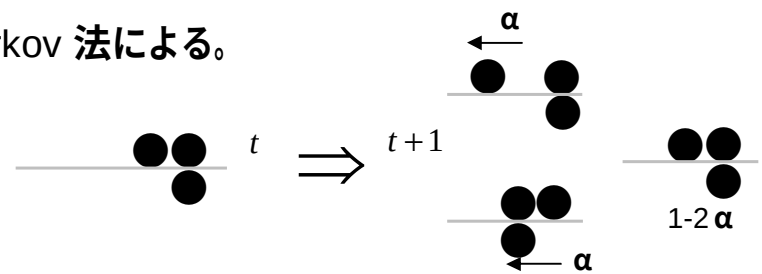
The Ergodic Theory of Traffic Jams

Lawrence Gray[1] and David Griffeath[2]

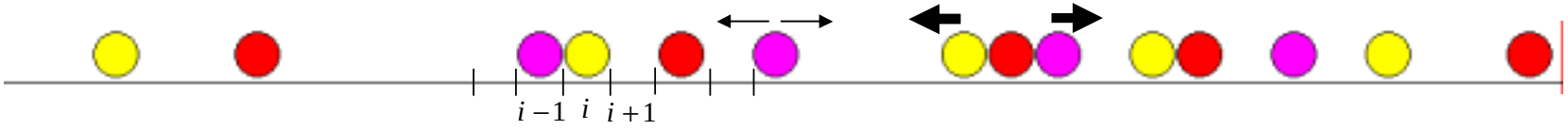
Received November 16, 2000; revised June 26, 2001

〔“**以下の定常状態に限られる**” **の部分の証明の考え方**〕

coupled Markov **法による。**



【 4．1次元格子上の 連続時間 最隣接作用 排除過程 】



$$X = \{0,1\}^Z \qquad x \equiv (\cdots x_{-1} x_0 x_1 \cdots) \in X \qquad \begin{cases} x_i = 1 & \text{座標 } i \text{ に粒子有り} \\ x_i = 0 & \text{座標 } i \text{ に粒子無し} \end{cases}$$

各粒子は，指数分布

$$F(t) = 1 - e^{-\lambda t} \qquad \text{ただし } \lambda = \begin{cases} 2\alpha & \text{(隣り合う粒子数が1の時)} \\ 2\beta & \text{(隣り合う粒子数が0の時)} \end{cases} \qquad \begin{matrix} \alpha > \beta : \text{斥力} \\ \alpha < \beta : \text{引力} \end{matrix}$$

に従う独立なタイマーを持ち，ベルが鳴ったら，コインを振り，ジャンプする方向 (+1, -1) を決める。
そこに粒子がなかったら，実際にジャンプし，タイマーをリセットする。(同時にジャンプする可能性は測度0でない)

X 上の連続時間マルコフ過程$(MP)_c$

$$(\Omega f)(x) = \sum_{i \in Z} \{\alpha \chi_{11}(x_{i-1}x_i) + \beta \chi_{01}(x_{i-1}x_i) + \alpha \chi_{11}(x_{i+1}x_{i+2}) + \beta \chi_{01}(x_{i+1}x_{i+2})\}[f(x^{i,i+1}) - f(x)]$$

$$\chi_{ab}(uv) = \begin{cases} 1 & if \quad uv = ab \\ 0 & if \quad uv \neq ab \end{cases} \qquad x^{i,i+1} \equiv (\cdots x_0 \cdots\cdots x_{i-1} x_{i+1} x_i x_{i+2} \cdots)$$

？ 初期配置から出発して，時間が経過したとき，どのような定常状態 μ に落着くか？

$$\int_X (\Omega f)(x) d\mu(x) = 0$$

[**それまでの結果**] $\alpha = \beta$ (**単純排斥過程**) **の場合が調べられている。**
($\Rightarrow$ exchangeable measure :**ベルヌーイ測度，あるいは，これらが混じり合った状態，になる。**)

問題点： $\alpha \neq \beta$ **の時，** coupled Markov**法が使えない。**

(続き)【 4．1次元格子上の 連続時間 最隣接作用 排除過程 】

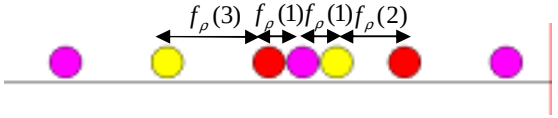
A $0 < \rho < 1$ とし， $q,\ q' \in (0,1)$ を，以下を満たす数とする：

$$qq'/[(1-q)(1-q')] = \beta/\alpha \qquad (1-q')/(1-q') = (1-\rho)/\rho$$

X 上の確率測度 μ_ρ を

$$\begin{cases} \mu_\rho([1]) = \rho \qquad \mu_\rho([0]) = 1-\rho \\ \mu_\rho([a_i \cdots a_j 00]) = q\mu_\rho([a_i \cdots a_j 0]) \\ \mu_\rho([a_i \cdots a_j 11]) = q'\mu_\rho([a_i \cdots a_j 1]) \end{cases}$$

により定める。



● μ_ρ，$0 < \rho < 1$，はマルコフ過程 $(\mathrm{MP})_c$ の定常確率である。

▲ μ_ρ では，隣り合う粒子間の距離の分布は $f_\rho(k) = \begin{cases} q' & k=1 \\ (\alpha/\beta)\, q' q^{k-1} & k = 2,3,\cdots \end{cases}$ であり，これらは互いに独立である。(renewal measure)

▲ μ_ρ では，位置 i に粒子が存在する確率は ρ である。

● $(\mathrm{MP})_c$の定常状態は以下に限られる：

i) 状態 μ_ρ，$0 < \rho < 1$

ii) 状態 μ_0（粒子が何もない），μ_1（全ての場所に粒子が詰まっている）

iii) 状態 i)，ii) が混じり合った状態

〔証明の考え方〕

初期状態 $\nu_0 = \nu$ から出発したとき，$[-n, n]$ 上の相対エントロピー

$$H_{\Delta_n}(\nu_t) = \sum_{\zeta \equiv_{-n}[a_{-n} \cdots a_n]_n} \nu_t(\zeta) \log \nu((\nu_t(\zeta)/\mu_\rho(\zeta))$$

は，境界からの影響を除けば t に関して単調性がある。

【５．エントロピー法の 離散時間 最隣接作用 排除過程への応用】

- 再隣接作用が，左右対称な場合は旨くいった。
- 再隣接作用が，左右非対称な場合は証明の基本になる不等式の証明まで。（見並）

【 6．新しい乱数生成法の起源 】

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \frac{x^9}{9!} - \cdots + (-1)^{n-1}\frac{x^{2n-1}}{(2n-1)!} + \cdots$$

x：ラジアン　（無限に続く）

マクローリン展開

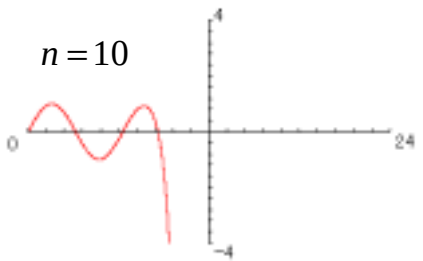
● コンピュータでは無限に続く項を扱えないので，公式を途中で打ち切る：

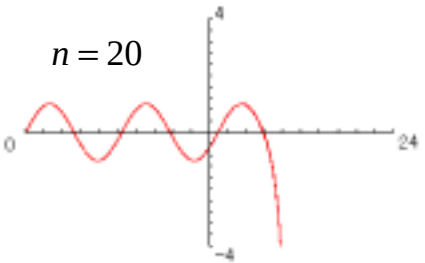
$$\sin x \approx x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \frac{x^9}{9!} - \cdots + (-1)^{n-1}\frac{x^{2n-1}}{(2n-1)!}$$

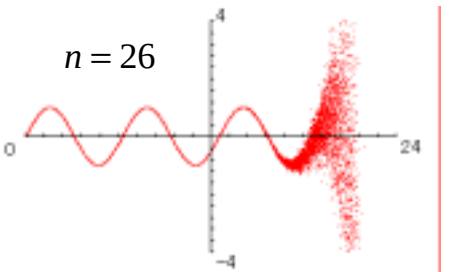
$$= F_1(x) + F_2(x) + F_3(x) + \cdots + F_n(x)$$

単精度で計算

（有効桁数は約６桁）







Aitecsin.exe

(続き)　【 6. 新しい乱数生成法の起源 】

$x = 22$ での $\sin x \approx F_1(x) + F_2(x) + F_3(x) + \cdots + F_{30}(x)$ を見る。

	単精度計算 (有効桁数：約6桁)	倍精度計算 (有効16~17桁)
F1	22.00000000000000	22.00000000000000
F2	-1774.66662597656	-1774.66666666667
F3	42946.93359375000	42946.93333333333
F4	-494912.2812500000	-494912.2793650794
F5	3326910.250000000	3326910.322398589
F6	-14638405.00000000	-14638405.41855379
F7	45416588.00000000	45416591.17038480
F8	-104674424.0000000	-104674429.1736490
F9	186258896.0000000	186258910.7354626
F10	-263594464.0000000	-263594481.8595450
F11	303761248.0000000	303761260.0476660
F12	-290554240.0000000	-290554248.7412460
F13	234380416.0000000	234380427.3179380
F14	-161595616.0000000	-161595622.2533930
F15	96320536.00000000	96320543.31359890
F16	-50128108.00000000	-50128110.71374390
F17	22975382.00000000	22975384.07713260
F18	-9344609.000000000	-9344609.994396810
F19	3395488.500000000	3395488.916882920
F20	-1108918.000000000	-1108918.107807920
F21	327266.0312500000	327266.0757189220
F22	-87705.84375000000	-87705.85860905780
F23	21439.20703125000	21439.20988221410
F24	-4799.526367187500	-4799.527096665880
F25	987.6576538085938	987.6577868989310
F26	-187.4612884521484	-187.4613211212090
F27	32.92136001586914	32.92136408659840
F28	-5.364962577819824	-5.364963036334550
F29	0.8134843111038208	0.8134843701710280
F30	-0.1150573939085007	-0.1150574036127350
計	-16.18064689636230	-0.02237880633948260

有効桁数 ：約6桁

単精度での計算

- 個々の $F_i(x)$ はOK
- $F_1(x) + \cdots + F_{30}(x)$ はダメ

⇓

$F_8, \cdots, F_{14}$ の**有効桁**が，
和をとることにより**消滅**

⇓

桁落ち誤差 (cancellation error) の発生

【7. $\sin x$ をホーナー法で計算する】

桁落ちが起こりにくい
計算法とされている

$$\sin x \approx x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \frac{x^9}{9!} - \cdots + (-1)^{n-1}\frac{x^{2n-1}}{(2n-1)!}$$

$$= x\left(1 - \frac{x^2}{2\cdot 3}\left(\cdots\left(1 - \frac{x^2}{(2n-4)(2n-3)}\left(1 - \frac{x^2}{(2n-2)(2n-1)}\right)\right)\cdots\right)\right) \quad (n=30)$$

G_1

G_2

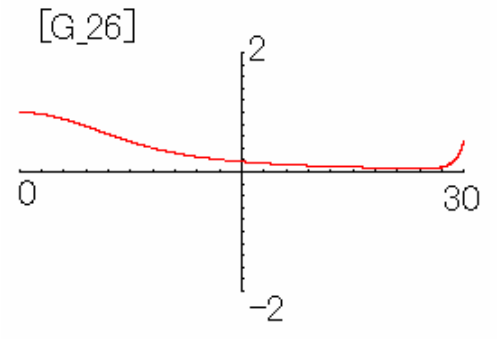
を

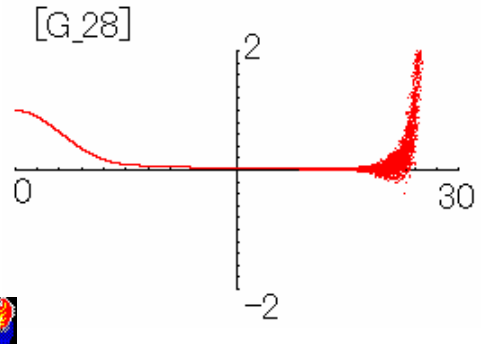
$$G_0 = 1, \qquad G_k = 1 - \frac{x^2}{(2n-2k)(2n-2k+1)} \times G_{k-1}, \qquad G_{30} = x \cdot G_{29}$$

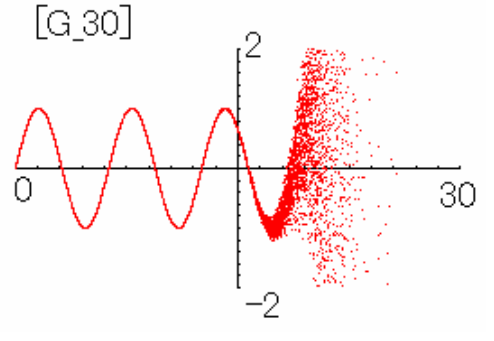
[例]

$$x - \frac{x^3}{3!} + \frac{x^5}{5!} = x\left(1 - \frac{x^2}{2\cdot 3}\left(1 - \frac{x^2}{4\cdot 5}\right)\right)$$

と計算する。 **ホーナー法**







PPTlinkAitec.exe

(続き) 【 7. $\sin x$ をホーナー法で計算する 】

$x = 22, \quad n = 30$

	単精度計算 (有効桁数：約6桁)	倍精度計算 (有効16~17桁)
G0	1.0	1.0
G1	0.858562231063843	0.858562244301578
G2	0.869817018508911	0.869817003057029
G3	0.858252048492432	0.858252043946262
G4	0.849276483058929	0.849276491556607
G5	0.838804006576538	0.838803991406511
G6	0.827388942241669	0.827388974557504
G7	0.814775109291077	0.814775086176766
G8	0.800832748413086	0.800832756712346
G9	0.785380363464355	0.785380368633015
G10	0.768217027187347	0.768217013159525
G11	0.749111294746399	0.749111312841289
G12	0.727800428867340	0.727800393832445
G13	0.703987061977386	0.703987066710165
G14	0.677339255809784	0.677339261091174
G15	0.647492229938507	0.647492255518142
G16	0.614056348800659	0.614056340306921
G17	0.576633512973785	0.576633520358191
G18	0.534848988056183	0.534848960244392
G19	0.488405317068100	0.488405342374929
G20	0.437171012163162	0.437170986406034
G21	0.381313532590866	0.381313574793800
G22	0.321486204862595	0.321486138969856
G23	0.259050846099854	0.259050993993284
G24	0.196278139948845	0.196277685302889
G25	0.136376187205315	0.136378184667289
→ G26	0.0832489654421806	0.0832355364032255
→ G27	0.0406547784805298	0.0408095328771154
→ G28	0.0161543600261211	0.0124093043738067
G29	-0.303118377923965	-0.00101721948707606
G30	-6.66860437393188	-0.0223788287156735

桁落ちが連続して発生

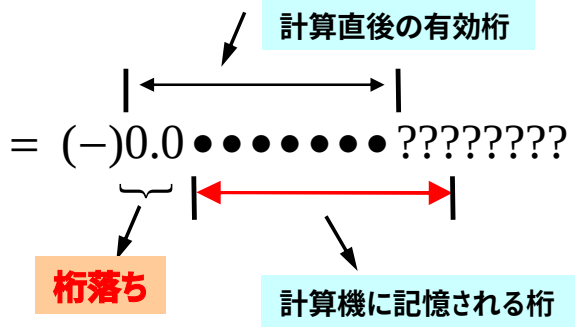
$$x\left(1-\frac{x^2}{2\cdot 3}\left(\cdots\left(1-\frac{x^2}{(2n-4)(2n-3)}\left(1-\frac{x^2}{(2n-2)(2n-1)}\right)\right)\cdots\right)\right)$$

G_1

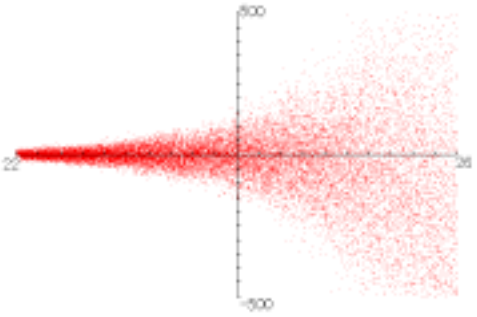
G_2

● $k = 26,27,28$ あたりで，**桁落ちが発生**

$$G_k = 1-\frac{x^2}{(2n-2k)(2n-2k+1)}\times G_{k-1}$$



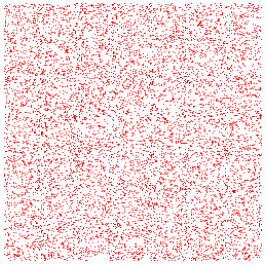
【8．$\sin x$ の Horner 法計算値から乱数を作る】



(1) x=22 から x=26 までを 19999 等分し，
各点で $\sin x$ の 単精度 Horner法 計算をする。

(2) 得られた値の上位3桁の数字を捨て，
残った桁から続けて4桁の数をとる。

```
-6.66860437393188 ⟶ 8604
-7.80057716369629 ⟶ 0577
-7.64557886123657 ⟶ 5578
 4.75384521484375 ⟶ 3845
 .....
```



得られた4桁整数値を，(X座標，Y座標) として点表示

乱数とは？

- **独立性**　出現する数の間には因果関係が無い
- **一様性**　発生可能な数値が同程度に発生される

数学的には，

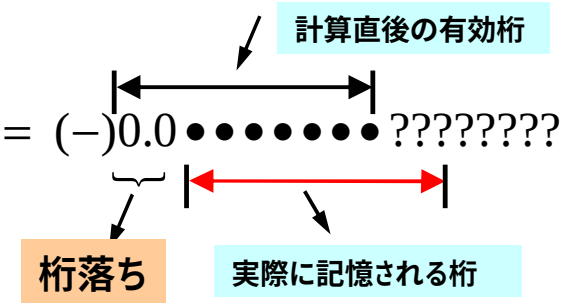
$\Omega = \{0, 1, 2, \cdots, k, \cdots, n-1\}$ (乱数値の空間)　　X_i ，　$i = 0, 1, 2, \cdots$ (i 番目の乱数値を表す確率変数)

としたとき

$$P(X_1 = k_1,\ X_2 = k_2,\ \cdots,\ X_l = k_l) = P(X_1 = k_1)\, P(X_2 = k_2) \cdots P(X_l = k_l) \qquad P(X_i = k) = \frac{1}{n} \quad \forall i,\ \forall k$$

【９．桁落ちを人為的に発生させる】

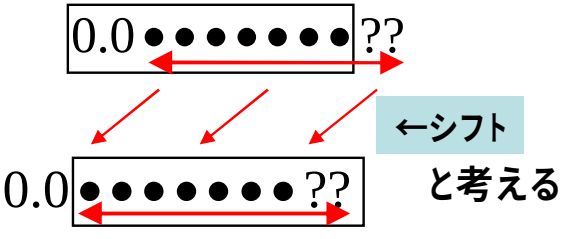
$$G_k = 1 - \frac{x^2}{(2n-2k)(2n-2k+1)} \times G_{k-1}$$



桁落ちのメカニズム

シフト

桁落ち を，



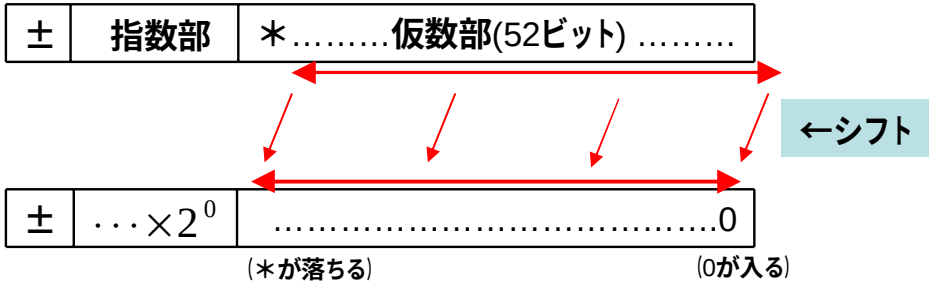
シフトを人為的に発生させる

仮数部　指数部

$0.001234567 = 1.234567 \times 10^{-3}$

（仮数部シフトの例）

$1.4567?? \times 10^{0} \in [1,2)$



【10．人為的な桁落ちを利用した乱数生成法（SSR）】

Simplified Shift-Real random number generator

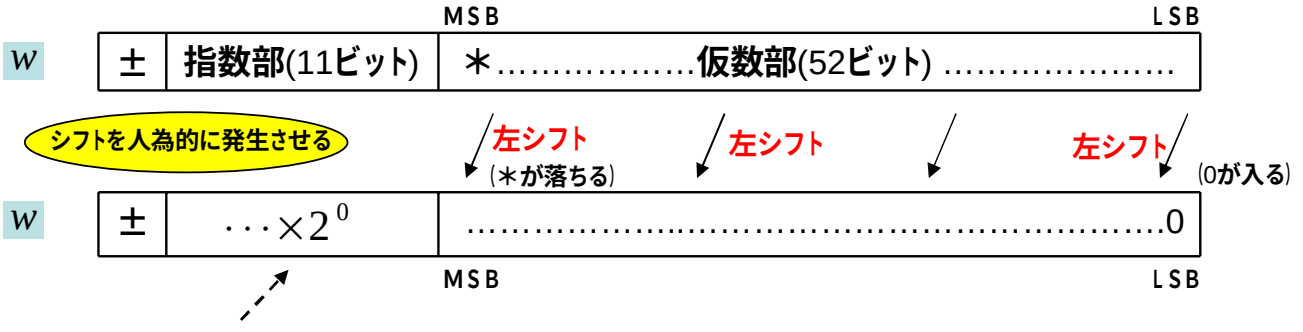
人為的な桁落ちを利用して，新しい乱数生成法を構成する。

$\Phi_x(t)$

SSR計算

i) $t, x \in [1,2)$ に対し $t \times x$ を計算し，（→ 結果を w とおく）

ii) w の仮数部の全ビットを1ビット左にシフト し，

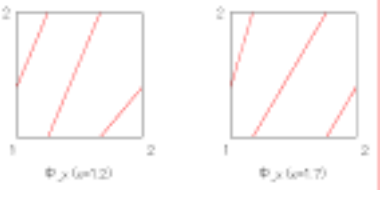


iii) w の指数部を $\times 2^0$ となるように設定する（⇒ $w \in [1,2)$）。

(1) $\Phi_x^{24}(w_0) = \underbrace{\Phi_x(\cdots\Phi_x(\Phi_x(w_0))\cdots)}_{24}$ の上位3桁を棄て，

続く4桁を乱数値とする。 1.77***5845***1 ⟶ ***5845***

$$\Phi_x(t) = \begin{cases} 2xt - \lfloor 2xt \rfloor + 1 & if \quad 1 \le xt < 2 \\ xt - \lfloor xt \rfloor + 1 & if \quad 2 \le xt < 4 \end{cases}$$

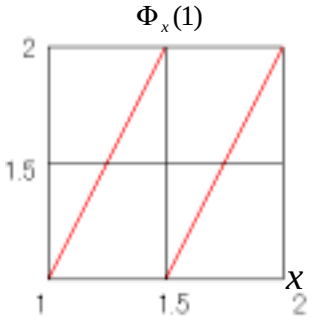


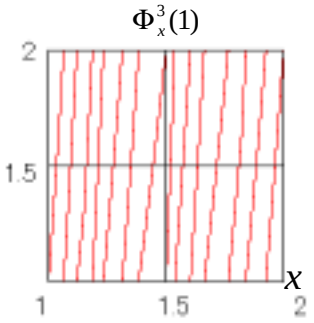
(2) x を $x_1, x_2, x_3, \cdots$ と変化させて，多くの乱数値を得る。

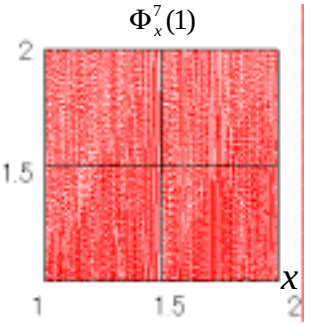
【11. 乱数生成法ＳＳＲの解析】

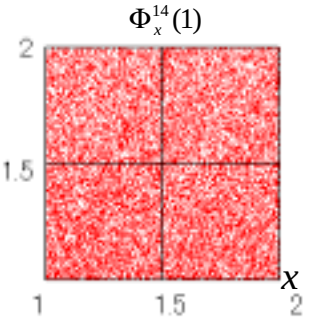
SSR計算

$$\Phi_x(t)=\begin{cases}2xt-\lfloor 2xt\rfloor+1 & if \quad 1\le xt<2\\ xt-\lfloor xt\rfloor+1 & if \quad 2\le xt<4\end{cases}$$









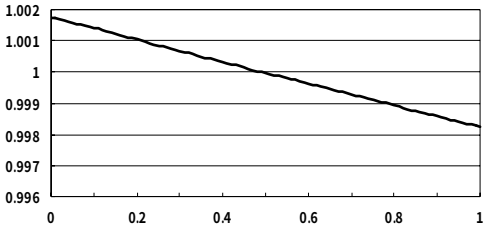
SSR乱数の分布

PPTlinkAitec.exe

$\Phi_x^{24}(1.e)\equiv\Phi_x^{24}(1.271828\cdots)$　**の上位3桁を棄て，続く4桁を乱数値とする。**

● x **を** $x_1, x_2, x_3, \cdots$ **と，区間** [1,2) **内を一様に変化させた時の乱数値の分布**

⇓

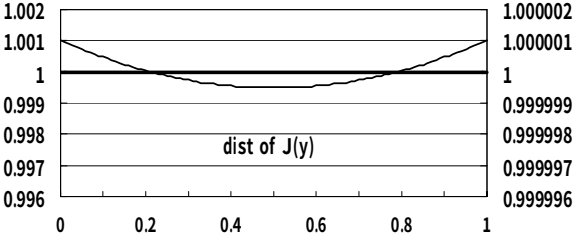


乱数値の分布

K改良

⇒

$\Phi_x^{24}(1.e)-\Phi_x^{24}(1.\pi)$



K改良後(SSRK)の乱数値分布

【12. SSRアルゴリズムの単純化(SSI) と [1,2) 上のβ変換】

(単純化)

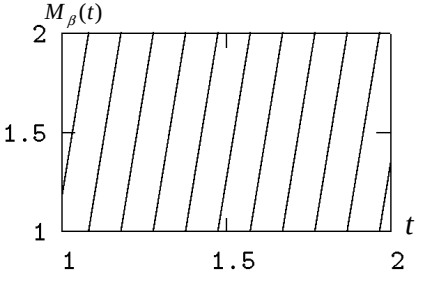
SSI (Simplified Shift-Integer)

$$\Phi_x(t)=\begin{cases}2xt-\lfloor 2xt\rfloor+1 & if \quad 1\le xt<2\\ xt-\lfloor xt\rfloor+1 & if \quad 2\le xt<4\end{cases}$$

の代わりに, $M_\beta:[1,2)\to[1,2)$

$$M_\beta(t)=\beta t-\lfloor \beta t\rfloor+1,\quad \beta>1$$
$$\equiv \beta t \mod [1,2)$$

(**[1,2) 上のβ変換**)



$M_\beta(t)$, $t\in[1,2)$

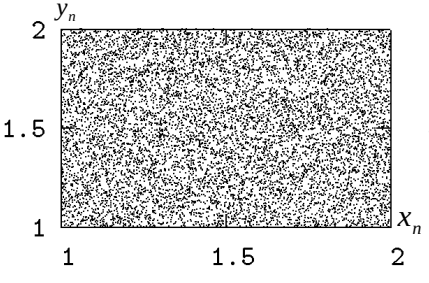
$\beta=2^3\times 1.2781\cdots$

を使う。

$$x_n=1+\frac{n}{20000},\quad n=0,1,\cdots,19999$$

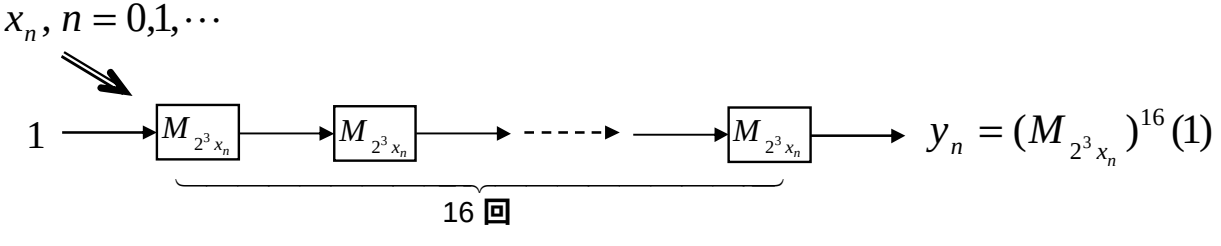
$$\beta=2^3 x_n$$

$$y_n=(M_{2^3x_n})^{16}(1)$$



$y_n=(M_{2^3x_n})^{16}(1)$,

$n=0,1,2,\cdots$



【13．SSI32：[1,2) 上のβ変換を利用した乱数生成】

SSI32K乱数生成法

あらかじめ整数 $x,\ y$ を固定しておく。

$p,\ q,\ r(<p),\ s(<q)$ を素数とし，$k = 0,1,2,3,\cdots$ に対し

$$(r_k, s_k) = (rk \bmod p,\ sk \bmod q)$$

を計算し，

$$x_k = (x \oplus r_k),\quad y_k = (y \oplus s_k)$$

$\oplus$：XOR（排他的論理和）

とおく。

非再帰的乱数生成法
（k番目の乱数を直接生成する）

$M_{2^2 x_k}^{23}(w_0)$ — $M_{2^2 y_k}^{23}(v_0)$ の $b_{17}\cdots b_{48}$ を k 番目の乱数値 として使う。

SSIK乱数 (32 bits)

XXX … XXX	0000000…0000000	XXX … XXX
b_1 … b_{16}	b_{17} … b_{48}	b_{49} … b_{64}

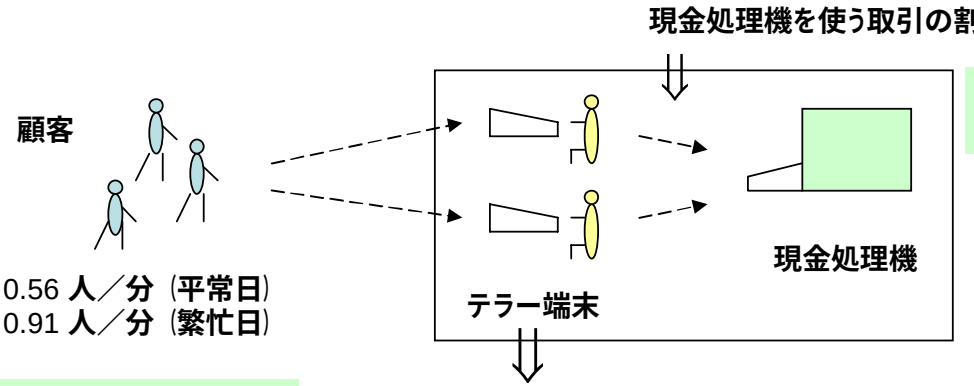
SSIK乱数の周期

$$= \text{period of } \{x_k\}\times\{y_k\}$$
$$= pq$$
$$\approx 1.18\times 10^{21}$$

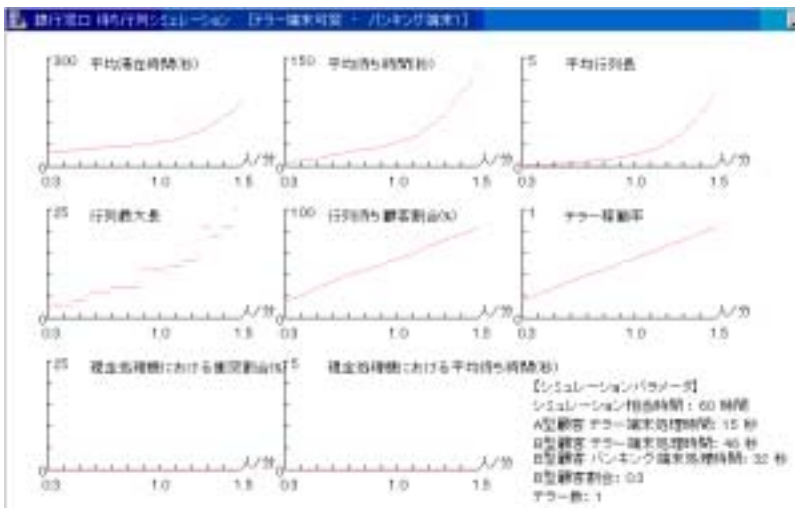
$$\begin{pmatrix} p = 34359738337 \\ q = 34359738319 \end{pmatrix}$$

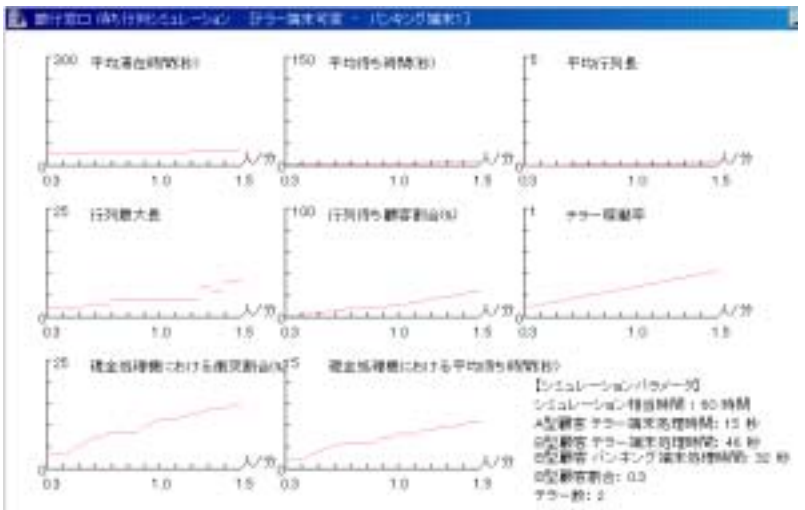
【14．乱数の応用：銀行の窓口端末（モンテカルロシミュレーション）】

ある銀行の窓口では，2人のテラーがそれぞれの端末を通して，1台の現金処理機を共有している。

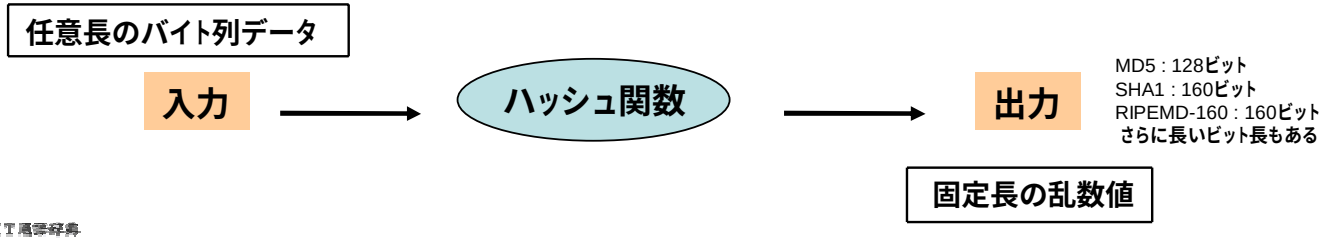


YOSI2008_4.exe





【15. ハッシュ関数への応用】



IT用語辞典
e-Words

ハッシュかんすうハッシュ関数【hash function】
メッセージダイジェスト関数 / message digest function
ハッシュアルゴリズム / hash algorithm

与えられた原文から固定長の疑似乱数を生成する演算手法。

生成した値は「ハッシュ値」と呼ばれる。「要約関数」「メッセージダイジェスト」とも呼ばれる。

通信回線を通じてデータを送受信する際に，経路の両端でデータのハッシュ値を求めて両者を比較すれば，データが通信途中で改ざんされていないか調べることができる。

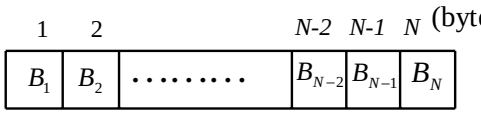
不可逆な一方向関数を含むため，ハッシュ値から原文を再現することはできず，また同じハッシュ値を持つ異なるデータを作成することは極めて困難である。

通信の暗号化の補助や，ユーザ認証やデジタル署名などに応用されている。

【16. 小型ハッシュ関数 MB32hash】

MB*n*hash, *n*=160, ..., 4096 に拡張可能

入力バイト列 $B =$



out put → MB32hash → ζ：32ビット擬似乱数

$$1.\ e \equiv 1+\frac{1}{10}\left(1+\frac{1}{1!}+\frac{1}{2!}+\frac{1}{3!}+\cdots\right)=1.27181\cdots$$

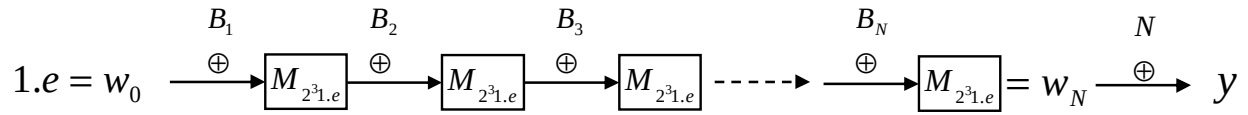
$$e = 2.7181\cdots$$

有理係数の代数方程式の解とならないような数（超越数）

MB32hash

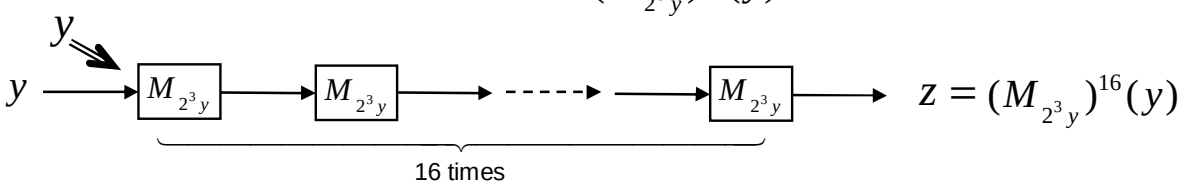
Stage 1. *B* **の圧縮**　　$\oplus$：XOR（**排他的論理和**）

$$w_0 = 1.e\,,\quad w_k = M_{2^3 1.e}(w_{k-1}\oplus B_k)\,,\quad y = w_N \oplus N$$



Stage 2. y **の撹乱**（**擬似乱数の作成**）

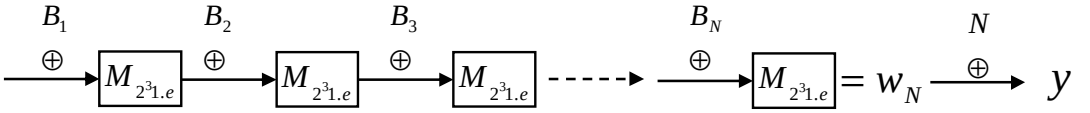
$$z = (M_{2^3 y})^{16}(y)$$



→ **32ービット ハッシュ値の取り出し**　$\zeta = \left\lfloor 2^{32}\left(2^{11}z - \left\lfloor 2^{11}z \right\rfloor\right)\right\rfloor$

【17．アルゴリズムの観点から見たMB32hashの安全性】

圧縮過程



2つの 入力 B , $B\hat{}$ に対して何時 $w_N = \hat{w}_{\hat{N}}$ となるか？

$w_N = \hat{w}_{\hat{N}}$ **となるためには，$\gamma \equiv 2^3 1.e$ は，変数 $\tilde{\gamma}$ に関する以下の代数方程式群の解の中に含まれている必要がある**：

$$\tilde{\gamma}^N(1.e+t_1)+\tilde{\gamma}^{N-1}(\tilde{p}_1+t_2)+\cdots+\tilde{\gamma}(\tilde{p}_{N-1}+t_N)+\tilde{p}_N$$
$$=\tilde{\gamma}^{\hat{N}}(1.e+\hat{t}_1)+\tilde{\gamma}^{\hat{N}-1}(\tilde{q}_1+\hat{t}_2)+\cdots+\tilde{\gamma}(\tilde{q}_{\hat{N}-1}+\hat{t}_{\hat{N}})+\tilde{q}_{\hat{N}}$$

t_i and $\hat{t}_j$ are of the form $\pm 0.0000000 b_8 b_9 \cdots b_{15} 000\cdots$,
$\tilde{p}_i$ and $\tilde{q}_j$ are in $\{-9,-10,\cdots,-19\}$.
($\rightarrow$ totally, $(2^8\times 2)^{N+\hat{N}} \times 11^{N+\hat{N}}$ equations)

しかしながら， $\gamma = 2^3\left\{1+\frac{1}{10}\left(1+\frac{1}{1!}+\frac{1}{2!}+\frac{1}{3!}+\cdots\right)\right\}$ **は超越数であるので，**

$B=B\hat{}$ 以外で $w_N = \hat{w}_{\hat{N}}$ となることはない。(⇒ 圧縮過程は安全である)

撹乱過程 $z = (M_{2^3 y})^{16}(y)$ **ハッシュ値の取り出し** $\zeta = \left\lfloor 2^{32}\left(2^{11}z - \left\lfloor 2^{11}z \right\rfloor\right)\right\rfloor$

2つの $y, \hat{y}$ に対して何時 $\zeta = \hat{\zeta}$ となるか？

$|y-\hat{y}| > 2^{-91}$ **を満たす $y, \hat{y}$ を見つけようとすると $y, \hat{y}$ に関する5次以上の代数不等式を解かなければならない。**

⇒ 5次以上の代数方程式を解くことは一般に困難 (⇒ 撹乱過程は安全)

- **仮に $y, \hat{y}$ を見つけても， $y, \hat{y}$ を生じる B , $B\hat{}$ を見つけることは大変難しい。**

【18．アルゴリズムの実装という観点から見たMB32hashの安全性】

● 圧縮過程での

$w_{k-1} \oplus B_k$ の取り方 ⇒

$1.b_1 \cdots b_7 (b_8 \oplus c_1) \cdots (b_{15} \oplus c_8) b_{16} b_{17} \cdots b_{31}$ (MBnhash)

⊕で触られない部分を確保する

$1.b_1 \cdots b_{23} (b_{24} \oplus c_1) \cdots (b_{31} \oplus c_8)$ の時，$\zeta = \hat{\zeta}$ とできる

B=0x0000

$w_0 = a2cb4411 \xrightarrow{\oplus B_1} a2cb44(1 \oplus 0)(1 \oplus 0) = a2cb4411 \xrightarrow{mul.\ x} 6785e38a890f0921$

$\xrightarrow{shift\ 4} 785e38a890f0921 \xrightarrow{OR} f85e38a890f0921 \xrightarrow{cut} f85e38a8 = w_1$

$\xrightarrow{\oplus B_2} f85e38(a \oplus 0)(8 \oplus 0) = f85e38a8 \xrightarrow{mul.\ x} \cdots\cdots \xrightarrow{cut} df0d49ac = w_2$

B^=0x0136

$w_0 = a2cb4411 \xrightarrow{\oplus \hat{B}_1} a2cb44(1 \oplus 0)(1 \oplus 1) = a2cb4410 \xrightarrow{mul.\ x} 6785e389e643c510$

$\xrightarrow{shift\ 4} 785e389e643c510 \xrightarrow{OR} f85e389e643c510 \xrightarrow{cut} f85e389e = \hat{w}_1$

$\xrightarrow{\oplus \hat{B}_2} f85e38(9 \oplus 3)(e \oplus 6) = f85e38a8$（$B$ の時と同じ）$\xrightarrow{mul.\ x} \cdots \rightarrow \hat{w}_2\ (= w_2)$

● 撹乱過程で y が $y + \eta$ に変化するとき，

$\eta > 2^{-91}$ ならばハッシュ値 ζ に影響がでる

⟹ 数値計算で $\zeta = \hat{\zeta}$ となる $y, \hat{y}$ を精密に求めても，

32ビットに丸める時点で $\zeta = \hat{\zeta}$ が狂ってしまう

⟹ 数値計算は役に立たない

【19. [1,2) 上のβ変換の性質】

元々のβ変換 $\beta > 1$

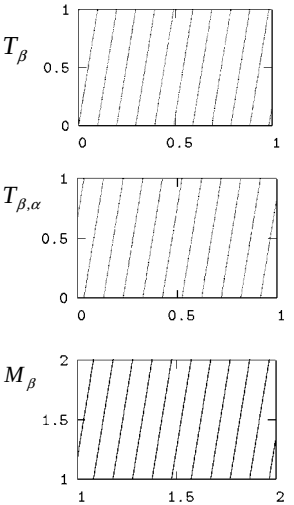
$$T_\beta : [0,1) \to [0,1), \qquad T_\beta(s) = (\beta s \mod 1) = \beta s - \lfloor \beta s \rfloor$$

Linear mod 1 変換 $\beta > 1, \quad 0 \le \alpha < 1$

$$T_{\beta,\alpha} : [0,1) \to [0,1), \qquad T_{\beta,\alpha}(x) = (\beta x + \alpha \mod 1) = \beta x + \alpha - \lfloor \beta x + \alpha \rfloor$$

[1,2) 上のβ変換 $\beta > 1$

$$M_\beta : [1,2) \to [1,2), \qquad M_\beta(t) = (\beta t \mod [1,2)) = \beta t - \lfloor \beta t \rfloor + 1$$



$\beta = \lfloor \beta \rfloor + \hat{\beta}$ **とすると** $M_\beta(t) = T_{\beta,\hat{\beta}}(t-1) + 1, \quad t \in [1,2)$.

$T_{\beta,\alpha}$ の性質 $X = [0,1)$, (X, B, λ): **確率空間**

(Parry) $h_{\beta,\alpha}(x) = \sum_{x < T^n_{\beta,\alpha}(1),\, n \ge 0} \frac{1}{\beta^n} - \sum_{x < T^n_{\beta,\alpha}(0),\, n \ge 1} \frac{1}{\beta^n}$ **とすると,**

$T^n_{\beta,\alpha}(t),\ n = 1,2,3,\cdots,$ **の分布を記述する**

$$\nu_{\beta,\alpha}(E) = \int_E h_{\beta,\alpha}(x)\, d\lambda(x)$$ **は,** $T_{\beta,\alpha}$ **の不変測度である。**

- $h_{\beta,\alpha}(x)$ **を利用してSSI乱数の分布の関数表現を調べる (西村)**
- **SSR写像の不変測度について** $h_{\beta,\alpha}(x)$ **のような表現を調べ, 乱数の分布を調べる (山口)**

まとめの講義を終わります