岡山大学環境理工学部研究報告
第6巻，第1号，pp.35-39，2001年2月

ランダムビットストリングと抗原多様性 — コンピュータ・シミュレーション

佐々木 徹 *

Random Bit Strings and Antigenic Diversity — Simulations

Toru SASAKI

(Received October 31, 2000)

Transition of random bit strings is simulated by using pseudorandom numbers. Bit strings are considered as RNA of HIV virus here. Transition of random bit strings represents that of antigenic diversity.

Keywords: random bit string, simulation of errors in RNA transcription, antigenic diversity

1 はじめに

HIV(ヒト免疫不全ウイルス) 感染において，約 10 年に渡る無症状期のあと AIDS を発病する仕組みは，まだ知られていない．M. A. Nowak や R. M. May 等は，ウイルスの突然変異により，抗原の多様性が増加していき，その多様性がある閾値を越えたときに発病するという，抗原多様性の閾値による発病モデルを提唱した ([2], [3], [4])．これらの研究では，ウイルスや免疫のダイナミクスを記述する微分方程式系によるモデルにおいて，ウイルス株が突然変異により 1 種ずつ増加するという状況でシミュレーションを行っている．

ウイルスの複製においてエラーが生ずる確率が一定であっても，それによって新しい株のウイルスの生ずる確率は一定ではない．なぜなら，エラーを繰り返すことにより，既に出現している株と同一の株が出来ることもあり得るからである．すなわち，突然変異によって出現するウイルス株が，既に存在するウイルス株と重複する可能性があるのである．すなわち，突然変異を何度も繰り返して行く場合，突然変異によって，抗原の多様性が増加して行く様子も，それほど単純なものではないことが分かる．

本稿では，RNA の塩基の並びをビットストリングで表し，塩基のひとつが複製においてエラーを起こす状況を，ビットの並びのうちのひとつのビットを反転させると解釈する．そして，ランダムに選んだビットを次々と反転させるシミュレーションを行い，その推移を考察する．

ランダムビットストリングのシミュレーションを行うにあたって，岡山大学環境理工学部の洞彰人助教授に助言を頂いたことに感謝します．

2 ランダムビットストリング

ビットが L 個並んだ並び，すなわち 0 または 1 が L 個並んだ並び

$$s = (s_0, s_1, \ldots, s_{L-1}), \quad s_0, s_1, \ldots, s_{L-1} \in \{0, 1\}$$

を考える．考えられるビットの並びの総数は，2^L 個である．例えば，$L = 3$ の場合は，$(0,0,0)$, $(0,0,1)$, $(0,1,0)$, $(1,0,0)$, $(1,1,0)$, $(1,0,1)$, $(0,1,1)$, $(1,1,1)$ の $2^3 = 8$ 個である．

L 個のビットから，無作為にひとつ選び，そのビットを反転させる (すなわち，0 は 1 に，1 は 0 に変える)．このような反転を繰り返していき，この並びがどのように変化していくかを考えよう．例えば $L = 3$ で，最初の並びが $(0,0,0)$ であったとしよう．(左から数えて) 第 2 ビット，第 0 ビット，第 2 ビット，第 1 ビット，第 0 ビットと順に反転させると，この並びは次の様に推移して行く：

$$(0,0,0) \to (0,0,1) \to (1,0,1) \to (1,0,0) \to$$

*Department of Environmental and Mathematical Sciences, Faculty of Environmental Science and Technology, Okayama University, Okayama, 700 Japan.

$\to (1,1,0) \to (0,1,0)$.

本稿では, このように L 個の長さのビットの並びにおいて, L 個のビットのうちひとつを無作為に選んで反転させる. この様な反転を繰り返してゆき, ビットの並びがどのように変化して行くかを調べる.

なお, RNA の塩基は, 4 種あり, ビットの並びのうち 2 ビットが塩基ひとつに対応している. また, 実際には, RNA は 3 個の塩基でひとつのアミノ酸をコードしている. この場合, 6 ビットがひとつのアミノ酸に対応し, $2^6 = 64$ 種類のビットの並びで 20 種類のアミノ酸を表す, という具合に重複があるのだが, ここでは, 簡単のため, それは考えないことにする.

3 シミュレーション

本節では, シミュレーションの方法について述べる.

ビットの長さ L に対して, 0 から $L-1$ までの値を, コンピュータで一様乱数を生成させることにより, ランダムに選ぶ. このようにして選ばれた値 $i \in \{0,1,\ldots,N-1\}$ に応じて, ビットの並び $\boldsymbol{s} = (s_0, s_1, \ldots, s_{L-1})$ のうちの s_i を反転させる.

ここで, 一様乱数列 $\{r_k\}$ $(k = 0,1,\ldots)$ は, 合同式法:

$$r_{k+1} = ar_k + c \mod m \qquad (1)$$

によって生成させる. ただし (1) において, a, c, m は,

$$a = 5^{11}, \quad c = 0, \quad m = 2^{31} \qquad (2)$$

の組合せのものを用いた ([1]).

また, この一様乱数列をもとに, 0 から $L-1$ までの値をとる乱数列を得る際には, もとの乱数の上位ビットを利用した ([5]).

4 初期値からの距離の推移

長さ L のビットストリング全体の集合

$$2^L = \{(s_i)_{i=0}^{L-1};\ s_0 s_1 \ldots s_{L-1} \in \{0,1\}\}$$

に, 次のようにノルムを入れる:

$$|\boldsymbol{s}| = |s_1| + |s_2| + \cdots + |s_{L-1}|.$$

このノルムにより, 2^L の 2 点 $\boldsymbol{s}$, $\boldsymbol{s}'$ の距離 $|\boldsymbol{s} - \boldsymbol{s}'|$ を考える. $\boldsymbol{s}$, $\boldsymbol{s}'$ が, RNA の塩基配列を表すとすると, この距離は, この二つの塩基配列がどの程度異なるかを表している.

さて, $\boldsymbol{s}^{(0)} = (0,0,\ldots,0)$ から, ランダムに選んだビットを次々に反転させていき, 得られたストリング $\boldsymbol{s}^{(i)}$ $(i = 0,1,2,\ldots)$ のノルム $|\boldsymbol{s}^{(i)}|$ がどのように変化していくかを, コンピュータで計算しよう. これは, 突然変異によって, RNA が, 初期の状態からどの程度離れていくかを示すことに相当する.

ここで, まず, $L = 224$ として計算を行う. これは, HIV の env 領域の V3 を想定した値である. V3 は 35 個のアミノ酸からなるので, これに対応する RNA の塩基の数は $35 \times 3 = 105$. よって, この倍の 210 の長さのビットストリングを考えることになる. ここでは, コンピュータプログラミングにおいて, 整数型変数に 32 bit の数を用いるので, この値に近い $L = 224$ を採用した.

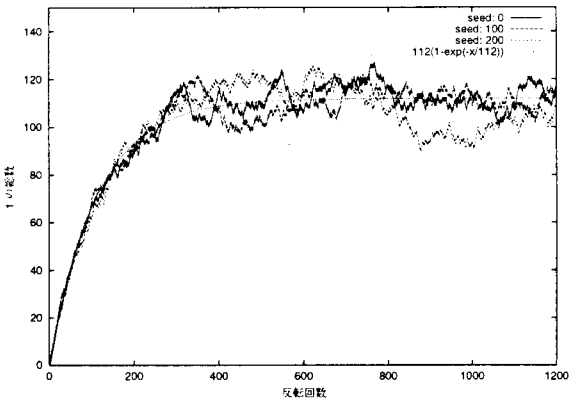

Fig.1 Distance from the original

Fig. 1 は, 乱数のシードを 0, 100, 200 にして反転を繰り返し, ストリング中の 1 の個数の推移をプロットしたものである. 確率的な揺らぎはあるが, ほぼ $L(1-e^{-Lt/2})/2$ (t は反転回数) と一致している.

5 抗原多様性の推移

本節では, ビットをランダムに反転させて得るストリングの列に対して, n 回の反転までに現れるストリングが何種類あるかをシミュレートにより計算する.

すなわち, ランダムな反転によって得られる 2^L の列 $s^{(0)}, s^{(1)}, s^{(2)}, \ldots$ と自然数 k に対して, 集合 $\{s^{(0)}, s^{(1)}, s^{(2)}, \ldots, s^{(k)}\} \subset 2^L$ の元の個数を N_k とおく. k が増加するに従い, N_k がどのように増加するかを, $L = 8$, $L = 16$ $L = 20$ の場合に計算した.

このようにして現れたストリングの多様性を考えることは, RNA の転写エラーを繰り返すことにより, 何種類のポリペプチド (すなわちアミノ酸の配列) ができるかを考えることにほぼ相当し, 抗原多様性の推移のシミュレーションのひとつとなる.

ここで, L を高々 20 としたのは, コンピュータの能力の関係である. 長さ L のストリングの組合

せは，全部で 2^L 種類ある．この全ての組合せに対して，既に出現したか否かを記録するためには，記憶容量が 2^L bit 必要である．第 4 節で採用した $L = 224$ の場合，これはおよそ 2.7×10^{67} bit，すなわち 3.4×10^{57} Gbyte の容量が必要になる．これは，直接計算するのは不可能な値である．

また，後述するよう結果からわかるように，ほぼ 5×2^L 回の反転をする必要があり，演算回数も莫大なものになる．今回利用した DEC の Alpha Station (CPU クロック 500MHz) における，$L = 20$ の場合のシミュレーションでは 5242880 回反転を繰り返したが，この計算にはおよそ 11 時間要した．L をわずかに増やすだけでも，計算時間は膨大に増加する．

そこで，今回は $L = 8, 16, 20$ の三つの場合について計算し，比較を行う．

これらを計算した結果のグラフが Fig. 2 から Fig. 4 である．$L = 16, 20$ の場合は，シードを変えてもグラフに差異が見られなかったので，シード 0 の場合のみをグラフにした．

Fig. 2 から Fig. 4 を比べると，これらの曲線が，ほぼ同じ形状をしていることに気がつく．そこで，$L = 8, 16, 20$ に対して，横軸 (反転回数)，縦軸 (現れた種類の総数) の両方を 2^L で割ったもののグラフを描いた (Fig. 5)．

Fig. 5 から，$L = 8$ の場合にはランダム性によるゆらぎが見られるが，ほぼ一致しており，特に $L = 16$ の場合と $L = 20$ の場合には一致の度合が高いことがわかる．

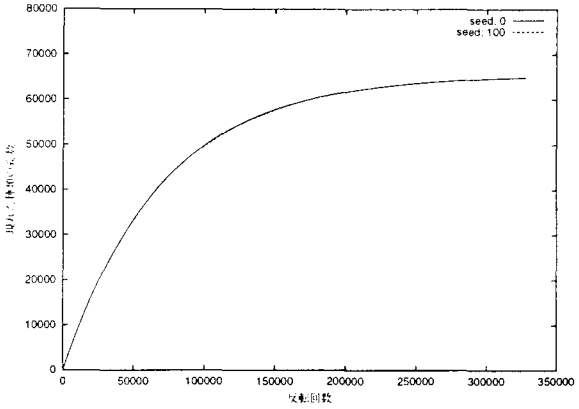

Fig.3 Diversity of strings. Bit length: 16

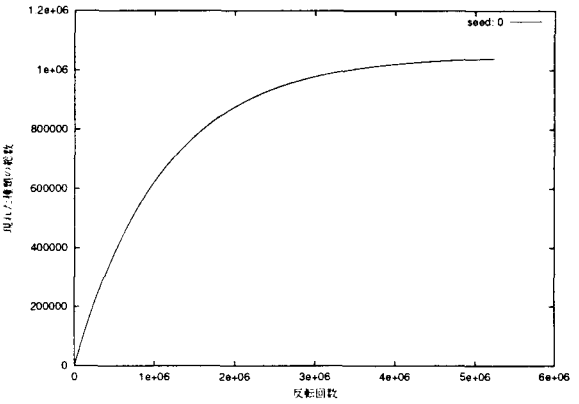

Fig.4 Diversity of strings. Bit length: 20

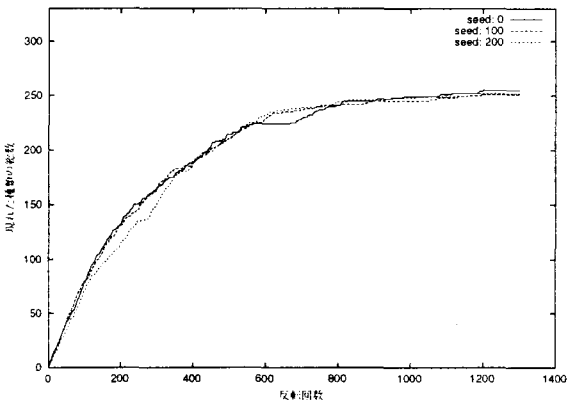

Fig.2 Diversity of strings. Bit length: 8

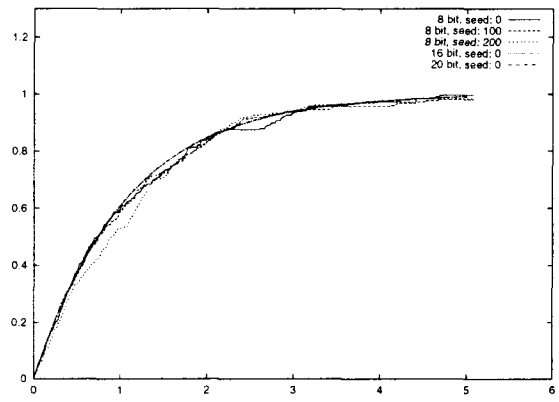

Fig.5 Diversity of strings: Scaled results

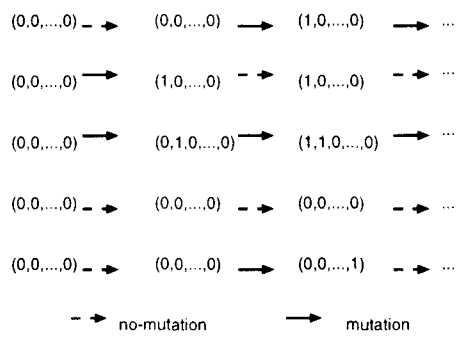

Fig.6 An example of change of five strings

6 L 個のストリングの反転

第 5 節では, ひとつのストリングが反転によってどのように推移するかを調べた. しかし, 体内にはウイルスは無数に存在し, 変異が起こった場合, 少なくともその直後には, 変異前の株と変異後の株の両者が存在する. ウイルス株間の競争による淘汰で絶滅する株がないとすると, 変異は, 既に出現した株全てに対して起こり得ることになる (**Fig.6**).

しかし, このように現れた全ての種類にたいして, 変異を行うシミュレーションは, 非常に大規模になるので, ここでは L 個のストリングを反転させる過程を考える.

最初のストリングが $(0,0,\ldots,0)$ であった場合, 最初の反転で生じ得るストリングは, $(1,0,\ldots,0)$ から $(0,0,\ldots,1)$ の L 種類ある. この L 種のストリングを出発点にして, L 個のストリングのビットを次々に反転させ, 現れるストリングの種類の総数の推移を計算する.

L 個のストリングのそれぞれに対してランダムに反転する操作を 1 回ずつ行う操作を, このストリングのセットに対する反転 **1** 回と数えることにする. ストリングのセットを k 回反転するまでに現れたストリングの種類の総数を M_k とする. すなわち, L 個のストリング $s_0^{(0)}, s_1^{(0)}, \ldots, s_{L-1}^{(0)}$ を

$$\begin{aligned} s_0^{(0)} &= (s_{0,i}^{(0)})_{i=0}^{L-1} = (1,0,\ldots,0) \\ s_1^{(0)} &= (s_{1,i}^{(0)})_{i=0}^{L-1} = (0,1,0,\ldots,0) \\ &\vdots \\ s_{L-1}^{(0)} &= (s_{L-1,i}^{(0)})_{i=0}^{L-1} = (0,\ldots,0,1) \end{aligned}$$

とし, 自然数 l に対して, $s_0^{(l-1)}, s_1^{(l-1)}, \ldots, s_{L-1}^{(l-1)}$ のそれぞれのストリングをランダムに反転させたものを $s_0^{(l)}, s_1^{(l)}, \ldots, s_{L-1}^{(l)}$ とする. 自然数 k に対して, 集合

$$\bigcup_{l=0}^{l=k} \{s_0^{(l)}, s_1^{(l)}, \ldots, s_{L-1}^{(l)}\}$$

の元の個数を M_k とおく.

$L = 8.16.20$ のそれぞれの場合について, コンピュータで M_k を計算したものが, **Fig. 7** から, **Fig. 9** である. 横軸は, ストリングのセットの反転回数 k, 縦軸は現れたストリングの種類の総数 M_k である.

$L = 8$ の場合は, ランダム性によるゆらぎがグラフに現れるているが, $L = 16, 20$ の場合には乱数のシードを変えてもグラフに変化が現れないので, これらの場合はシードが 0 のグラフのみプロットしている.

これらの結果を 第 5 節と同様に, スケールを取り直してプロットしたのが **Fig. 10** である. ここで, 横軸は $kL/2^L$, 縦軸は $M_k/2^L$ である.

このときも, $L = 8, 16, 20$ の場合のグラフがほぼ一致していることがわかる.

7 考察

初期値からの距離は, $L = 224$ の場合には, ランダム性によるゆらぎがみられたが, ほぼ $L(1-e^{-Lt/2})/2$ (t は反転回数) と一致している. ここには計算結果を書いていないが, L を大きくすると, このゆらぎは漸近値 $L/2$ と比較すると相対的に小さくなってくる.

抗原多様性は, 計算機の能力の関係で, $L = 8, 16, 20$ という小さな値の L に関してのみしかシミュレーションを行うことができなかった. このような小さな値の L に対しては, スケールを取り直すことによって, L の値が異なっていても, 抗原多様性の推移のグラフはほぼ一致した.

しかし, さらに大きな値の L について, このような性質が成り立つか否かは, わからない. 非常に大きな L に対しては, 多様性が飽和状態に達するまでの間は増加率がほぼ一定になる可能性もある.

L が大きい場合の問題の解決には, より多くの工夫が必要である. また, 異なるタイプのランダムウオークに対して, 同様の考察を行い, その性質を調べるなど, 課題も多い.

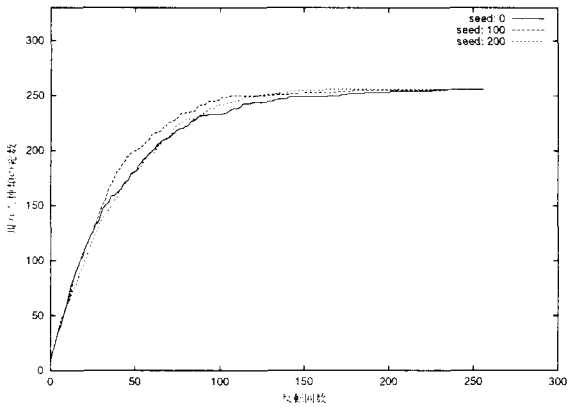

Fig.7 Diversity of strings. Bit length: 8

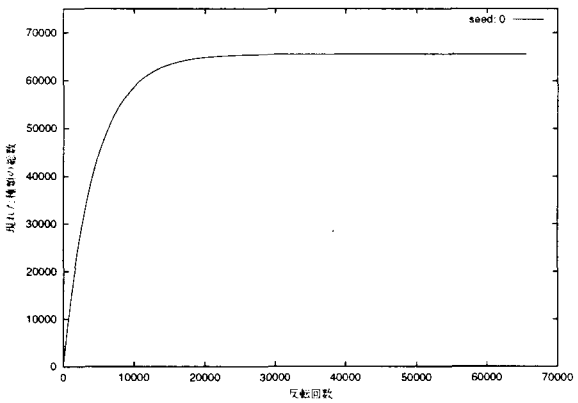

Fig.8 Diversity of strings. Bit length: 16

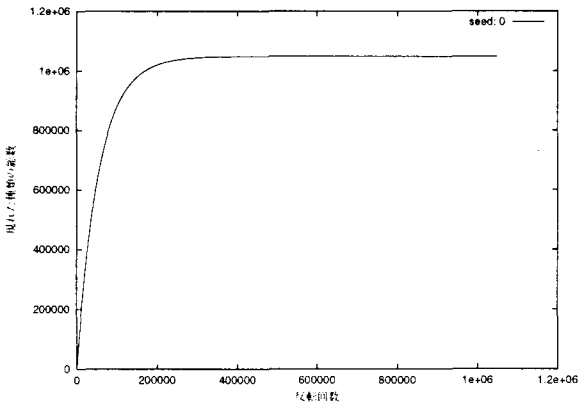

Fig.9 Diversity of strings. Bit length: 20

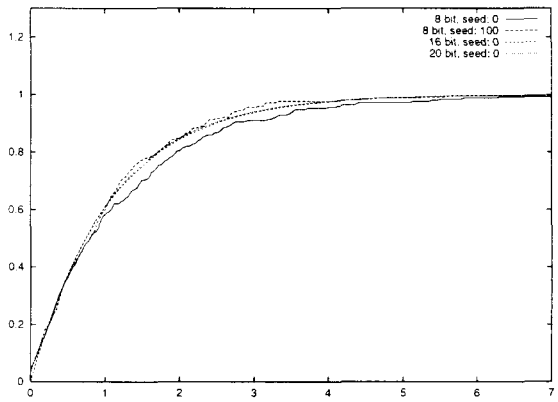

Fig.10 Diversity of strings: Scaled results

参考文献

[1] 森正武 (1986): **FORTRAN 77 数値解析プログラミング, 岩波書店**

[2] **M. A. Nowak(1992):** *Variability of HIV infection*, **J. Theoret. Biol. 155, 1–20**

[3] **M. A. Nowak et al.(1991):** *Antigenic diversity threshold and the develpment of AIDS*, **Science 254, 963–969**

[4] **M. A. Nowak and R. M. May(1991):** *Mathematical biology of HIV infections: Antigenic variation and diversity threshold*, **Math. Biosci. 106, 1–21**

[5] **W. H. Press et al.(1988): Numerical recipes in C , Cambridge University Press**