

リカレンスプロットを用いた擬似的な基底の定義に向けて

城 真範^{1,a)} 大知 正直²

概要: リカレンスプロットは可逆な時系列の表現手法であり、時系列をグラフ表現などの他のデータ構造へ変換することができる。類似の可逆な時系列表現としてフーリエ変換が知られている。フーリエ変換との対比から、リカレンスプロットを利用して非線形時系列でも適切に利用できる擬似的な基底を考える。リカレンスプロットはビット列の集合であるため、各基底もビット列となり、その結合はビット演算となる。本報告では無作為にとった基底の本数と表現力の関係を報告する。

Toward the definition of pseudo basis using a reclamation plot

1. はじめに

データを可逆な変換によって扱いやすくすることは、解析を見通しを良くし、データの深い理解を促してきた。時系列解析においては可逆な変換としてフーリエ変換やウェーブレット変換といった線形変換がよく使われている。それらは時系列を独立な基底と複素数係数に分解することで表現することで、データ圧縮や信号フィルタといった広範な応用を実現してきた。特にフーリエ変換は量子力学の基本的な考え方にも利用されている。しかし線形変換は一般に、基底を波の大きさに応じてとるため、波に含まれる大きな波の成分と小さな波の成分が独立であるという前提がある。もしこれらが独立でない場合、すなわちスケール間の相互作用がある場合、その効果は展開の結果得られる波の位相に反映される。そして相互作用が複雑であれば、位相の変化からそれを復元することは困難である。

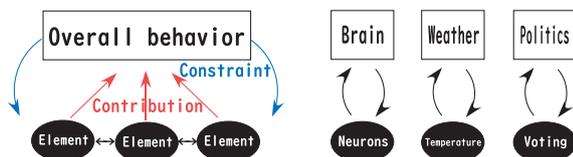


Fig.1: スケール間に相互作用がある例

ところが複雑なシステムでは Fig.1 に示すようなスケール間の相互作用がシステムの重要な特徴になっている場合

がある。例えば脳や経済現象、生体现象などにおいては、個々の構成要素の振る舞いがシステム全体の状態に寄与し、システム全体の状態が個々の要素の振る舞いを規制するといったスケール間の複雑な相互作用が存在する。シンプルな数理モデルであっても、カオス性に関係する立式にはスケール間の相互作用が重要な特徴となっている場合がある。代表的な一例として、次元写像の普遍的定数である第2ファイゲンバウム定数 α は、 $g(\alpha x)/\alpha = g(g(x))$ を満たす偶関数 $g(x)$ の特殊値 ($\alpha = 1/g(1)$) で与えられる [1] が、 α は関数全体の形を決める制約にもなっており、 $g(x)$ を整関数の級数展開と最適化で求めようとしても、せいぜい 1000 桁程度しか求まらない [2]。

こうしたシステムから観測される時系列は非線形時系列として、線形変換の利用が適切ではないものの一つである。もしも非線形性が強い時系列においても利用可能な基底を定義し、それによる可逆な変換を作ることができれば、最近の非線形時系列解析の知見や、うまくいけば線形解析の知見や応用とも対応づけることで、非線形時系列の理解において新しく俯瞰的な視点を提供できるだろう [3]。特に様々な非線形的性質を残したままの情報圧縮は重要な応用の一つである。例えば深層学習で得られたネットワークの状態を適切に圧縮できれば、その本質的な情報を効率的に取り出し、Explainable AI に大きな貢献をしようだろう。

一般的に、非線形時系列の解析においては、まずは遅れ時間座標へのプロットかリカレンスプロット [4] と呼ばれるプロットによってデータの可視化がなされる。本研究では特にリカレンスプロットに着目したい。それは Fig.2

¹ 産総研人間情報インタラクション研究部門
Umezono1-1-1, Tsukuba, Ibaraki 305-8568, Japan

² 東京大学工学系研究科
Graduate School of Engineering, Univ. of Tokyo

a) shiro@aist.go.jp

に示すように、時系列の差の絶対値を2次元上で計算したのち、リカレンスレート ϵ と呼ばれる閾値（この例では $\epsilon = 0.2$ ）で二値化してプロットする（1を黒色の点、0を白色の点に対応させて画像にすることが多い）。

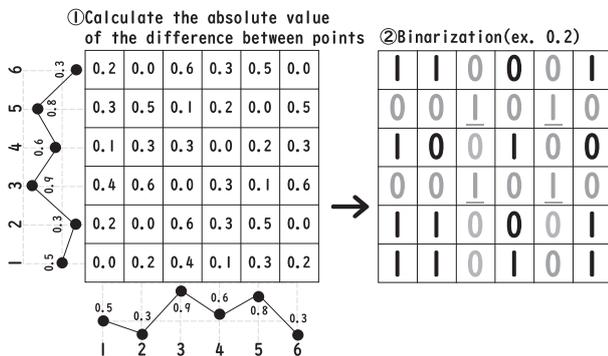


Fig.2: リカレンスプロットの作成方法

リカレンスプロットはもともと決定論的カオス時系列を特徴づけるために導入された情報表現だが、時系列と複雑ネットワークを結びつける方法 [5] が提案されてから情報表現を変換する方法として再認識された [6]。例えばタンパク質構造解析の分野ではアミノ酸の結合を示す Contact map と呼ばれており、脳科学では領野間結合を示す Dynamic functional connectivity として知られている。リカレンスプロットは値を実数ではなく二値表現するため、時系列を言語あるいは、記号力学を結びつけることもできる [7]。興味深いことに、リカレンスプロットはデータを二値に変換してしまっているにも関わらず、リカレンスプロットのみから元の時系列を復元することができる [8]。この方法の妥当性には、時系列が十分に長いという前提のもとで複数の証明も与えられており [9], [10]、つまりリカレンスプロットは数学的に保証された可逆変換といって良い。最近ではこうしたリカレンスプロットの可逆性を積極的に使い、点同士をネットワーク構造に展開し、多次元尺度法 (MDS) を使うことで染色体の3次元構造を復元する [11] といった研究や、ゆっくりとした外力の再構成 [12] といった分野の研究も報告されている。そこで本研究でもリカレンスプロットを使って基底展開を考えてみたい。本研究の目指すところは、Fig.3に示すように、非線形時系列においてフーリエ変換に相当する方法論を確立することである。すなわち、リカレンスプロットを用いてさらにデータ圧縮が可能となる表現方法を構成することである。

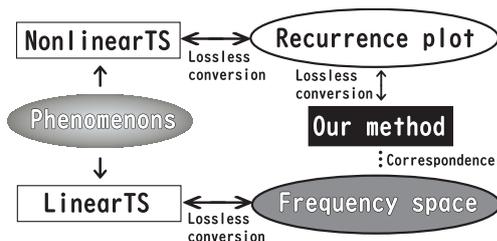


Fig.3: 本研究の位置づけ

なお“基底”は本来線形空間を前提とした概念だが、本研究では基底の概念を拡張し、その要請として独立性と完全性を満たすものとする（これは「擬似的な基底」と呼ぶべきだが以下では単に「基底」と表現する）。まず Fig.2 においてプロット上の左から i 番目の縦の列に着目する。 i 番目の縦の列は時系列の i 番目の点の近傍点集合をビット列（整数）で示したものとなっている。もし時系列のどこかの点とその近傍をまとめて一つの基底として扱い、独立な基底の集合を何かの演算で結合したとき、表現できない点が定数 c 以下である場合、基底のもつべき独立性と完全性を満たしたとみなす。これは Cover と呼ばれる概念である [13]。Cover が成立している場合、基底と演算表現からリカレンスプロットを復元できるため、時系列の可逆な変換を維持できる。こうした基底は時系列の微細な構造と大きな構造を、時系列各点の上で等価に扱うことになり、時系列におけるスケール間の相互作用を自然に維持した変換を実現できる。

線形変換と同様、基底の選択には自由性があり、いくつかの基準が考えられる。演算表現を行わず基底だけですべての点を表現可能にするような基底は従来よりフレームと呼ばれる概念 [14] に相当する。この場合、どのようなアルゴリズムが現実的な時間で適切な基底を選択できるのかについて、検討する必要がある。

演算としてなにを許容するかについても自由性がある。基底がビット列である以上、基底同士をつなぐ演算は論理演算であるべきだが、AND, OR, NOT だけを使うのか、ビットシフトを許容するかどうかにも自由性はある。AND, OR, NOT だけを使う場合、それらは NAND 演算一種類に還元できるため、表現量を定量化しやすい。この場合、NAND を \uparrow で表現すると、

$$\begin{aligned} \text{NOT}(A) & A \uparrow A \\ \text{AND}(A, B) & (A \uparrow B) \uparrow (A \uparrow B) \\ \text{OR}(A, B) & (A \uparrow A) \uparrow (B \uparrow B) \end{aligned}$$

となるので [15]、演算表現の複雑さを定量化するためには、AND と OR について3倍の重みをかけて演算子の数をカウントすれば良い。時系列がカオス的な決定論性をもつ場合、近傍点同士はなかなか離れないため斜めの線が増える傾向にある。そうした場合にはビットシフトを許容することで表現を簡潔にできる可能性がある。ただし本研究では基底選択の方法には踏み込まず、基底を無作為にとって、基底の本数と Error の数（あるいは Cover における c の値）の関係を調べることにする。

2. 方法

2.1 データ作成

リカレンスプロットの作成において、適切なリカレンスレート ϵ のとり方には自由度がある。 ϵ が極端な値（0 や 1）ではリカレンスプロットは真っ黒や真っ白になってしまう

い、時系列の情報は失われる。適切な ε は極端な値であってはならない。

リカレンスプロットが Fig.4 に示すように、時系列を時間方向に点を分類している解釈すれば、リカレンスレートは縦軸の解像度に対応するため小さい方がよい。MDS を用いて時系列を再構成できるためにはリカレンスプロットから得られるネットワーク構造が単連結である必要があるため、これを考慮するなら、 ε は単連結なネットワーク構造を構成する最小の値とすることが適切である。一般に ε は 0.03 などの値がとられ、従ってリカレンスプロットの黒点は白点より少なくなることが多い。

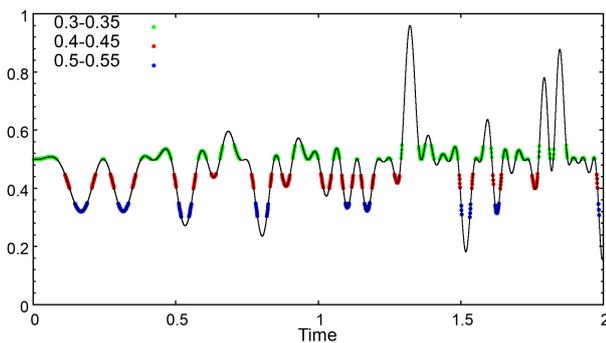


Fig.4: リカレンスレート $\varepsilon = 0.05$ の例

本研究では 0 から 1 の値を取る次の三種類の人工的な時系列を用いてリカレンスプロットを作成した。長さは 100 点から 10000 点までで、 ε は 0.03 とした (Fig.8 と Fig.9 の作成においては、それぞれ 0.05, 0.1 とした)。カッコ内は長さが 500 点、 $\varepsilon = 0.03$ のときのリカレンスプロットの黒点の割合である。

- (1) Logistic 写像 (0.092) : $x_{n+1} = 3.8x_n(1-x_n)$, $x_0 = 0.1$ による作成。
- (2) 乱数列 (0.059) : MT による 0 から 1 までの一様乱数にて作成。
- (3) 正弦波 (0.054) : $x_n = \sin(0.05\pi n)$ による作成。

2.2 演算を構成するアルゴリズム

ε が小さい場合、統計的には黒点が少なくなる。そこである基底に対して NOT 演算を使うことでリカレンスプロットの白黒を反転させ、NOT をとった別の基底との AND を取ることで目的のビット列に向けて貪欲的な方法で演算表現を構成できる。具体的には次のような方法となる。

目的となるビット列を O とし、基底の候補を B_i ($2 \leq i \leq n$) とする。 $\bar{B}_i \cap B_j$ と $B_i \cap B_j$ を計算し O と一致するビットが最も多い i と j を選択する。さらにその結果 $\bar{B}_i \cap B_j$ (あるいは $B_i \cap B_j$) に対して、別の基底候補 B_k を選び、同様に $(\bar{B}_i \cap B_j) \cup B_k$ と $(B_i \cap B_j) \cup B_k$ を計算し、 O と一致するビットが最も多い k を選択する。これを繰り返し、選択可能なすべての基底を考慮したら、 O の中で既に表現できている部分をマスクし、表現できていない部分につい

て同様の AND 表現を探索し、最後に AND 表現同士を OR で結ぶことで基底表現を完成させる。それでも合致しないビットの個数を Error の数として終了する。実装は C++ に行った。図示のため libpng++[16] を利用している。

3. 結果

3.1 リカレンスプロット

まず、500 点の時系列から作成したリカレンスプロットそのものを Fig.5 に示す。どの図にも見られる斜めの黒い線はリカレンスプロットの特徴であり、同じ点同士の差が常に 0 であるためである。Logistic 写像においては短い斜めの線が多数見られるが、これは系の決定論性を示している。この短い斜めの線は乱数列では見られない。正弦波は周期的な波でありリカレンスプロットにおいてもその周期性を見ることができる。

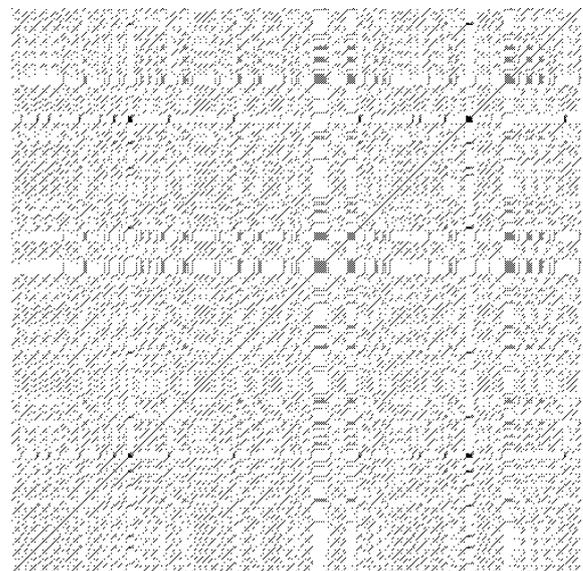


Fig.5-1: リカレンスプロット (Logistic 写像)

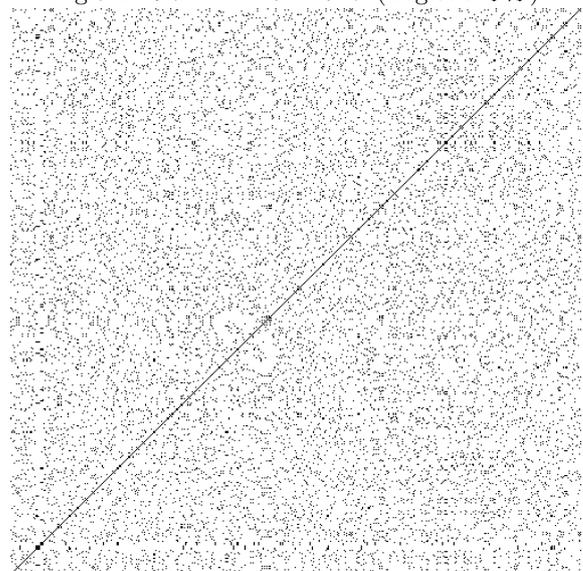


Fig.5-2: リカレンスプロット (乱数列)

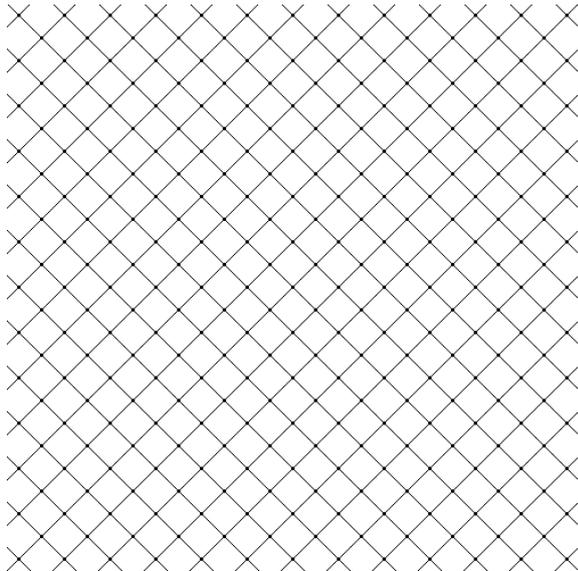


Fig.5-3: リカレンスプロット (正弦波)

次に, Logistic 写像にて演算表現の例を示す. 以下は 160 点の時系列から作ったリカレンスプロットにおいて, その 9 点めのビット列を, 無作為に選定した 20 本の基底の論理演算で表現した結果である. ビット列は 16 進数にて表示し, チルダが NOT を, & が AND を示している.

ターゲット:

0002000104000010004010080205000000080020

無作為に選んだ 20 本の基底:

- A:020000000080000002000000000080420000001
- B:0001400001000008002008024180000002020010
- C:0084000208001000400020000400000000100240
- D:0041400001040808202000024000000000020000
- E:2000082080000001000400000000804000002000
- F:040801041000210084000000000100800200400
- G:000480020A08002000800014000A000000140000
- H:9020249040428404101204812040402001811008
- I:4010104000210202000802400020008100000000
- J:80000000040000401104480284004025100808A
- K:0400010410002000840040000800100840000482
- L:0002000104000010004010088205000000080020
- M:0080000008001080400020000400020108004240
- N:000480020A08002000800014000A000000140000
- O:0400010000002000850040000800140A50008482
- P:0002000104000010004010088205000000080020
- Q:4000000000200080000822400420028108004000
- R:4000000000200082000822400420028108004000
- S:4000104000210082000802400020028108004000
- T:0400010000002000850040000800140A50008482

貪欲的な方法によって構成された演算表現:

~A&~B&~C&~D&~E&~F&~G&~H&~I&~J&~K& L&~M&~N&
 ~O& P&~Q&~R&~S&~T

= 0002000104000010004010088205000000080020

表現できなかったビット数:1

この演算表現ではたまたま OR を取ることなく, 無作為にとった 20 個の基底で 160 点をほとんど Error なしに表現できた.

Fig.6 は 500 点の時系列に対して, 無作為にとった基底の本数と Error の数の関係を調べたものである. 各点 (Error の数) は安定化のため各時系列 400 回無作為に基底を取り直した場合の平均である. 指数関数でよく近似できることが分かった.

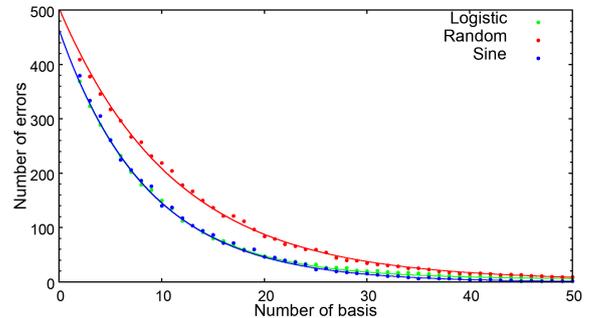


Fig.6: 無作為にとった基底の本数と Error の数の関係

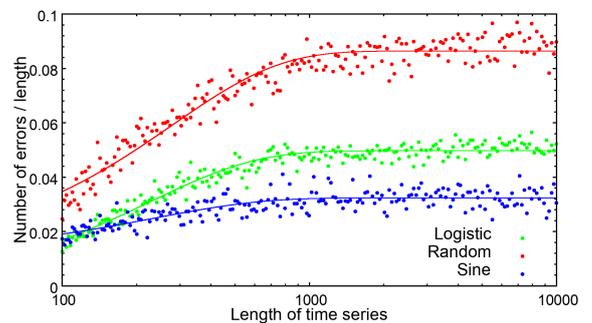


Fig.7: 時系列の長さ と Error 数の関係 ($\epsilon = 0.03$)

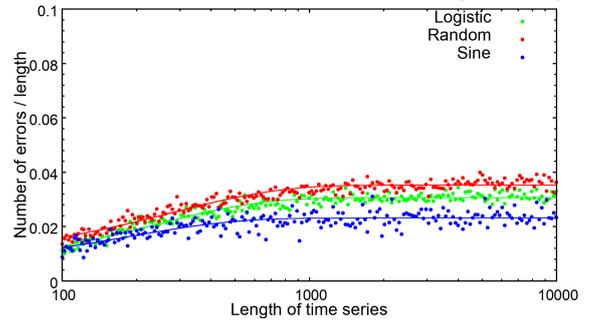


Fig.8: 時系列の長さ と Error 数の関係 ($\epsilon = 0.05$)

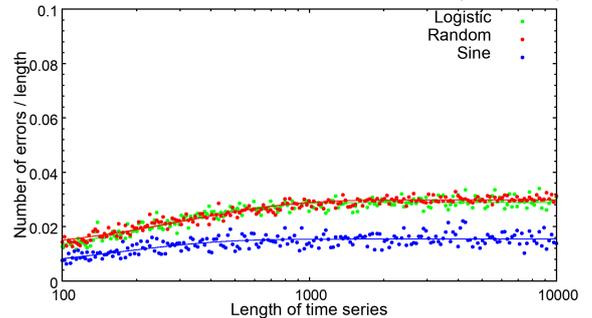


Fig.9: 時系列の長さ と Error 数の関係 ($\epsilon = 0.1$)

Fig.7 は無作為に選択する基底を 30 本に固定し, 時系列の長さを変えて Error の数を時系列の長さで規格化した値を調べたものである. リカレンスレート ϵ は 0.03 である. Error の数は各時系列 400 回の試行の平均である. これも指数関数でよく近似できた. Fig.8 と Fig.9 はリカレンス

レート ε を 0.05 と 0.1 にとったものである。近似は同様に指数関数である。 ε が大きくなるほど Error 数は全体的に減っていることが分かる。

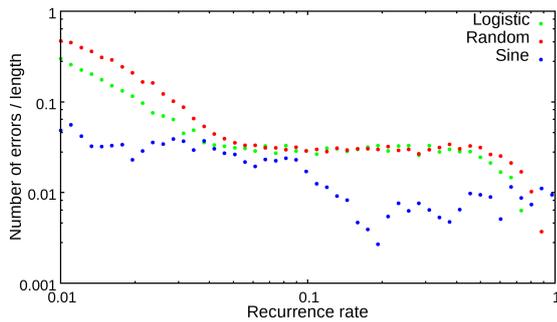


Fig.10: ε と Error 数の関係の関係 (時系列は 10000 点)

Fig.10 はこの減り方について調べたものである。時系列の長さを 10000 点に固定し、基底の数が 30 本のとき、横軸をリカレンスレート ε にとって Error の数を両対数表記した。関数による簡単な近似はできない結果だが、決定論的な時系列 (Logistic 写像, 正弦波) では Error の数が乱数列を超えることはほとんどなく、 ε に対して減少傾向にあることが分かる。また乱数列については、 $0.05 < \varepsilon < 0.2$ でほとんど Error の数に変化がないことも分かる。

4. 議論

4.1 黒点の割合

乱数列と正弦波について黒点の割合を解析的に示す。まず乱数列については 0 から 1 の値を取る二つの一様乱数の差 z の分布が、

$$\begin{cases} z + 1 & (-1 \leq z < 0) \\ 1 - z & (0 \leq z < 1) \end{cases}$$

で与えられるので、 $\varepsilon = 0.03$ のとき、黒点の割合は $|z| < 0.03$ におけるこの分布の面積に等しいため、 $(1 + (1 - 0.03)) \cdot 0.03 = 0.0591$ となる。これはリカレンスプロットから得た値に近い。

正弦波の場合、 $|\sin(0.05\pi n) - \sin(0.05\pi(n+1))| < 0.03$ となる n は $-11.7246 < n < -9.27536$ または $8.27536 < n < 10.7246$ に入る必要があるので $n = -11, -10, 9, 10$ の 4 点である。今回のサンプルは 40 点 ($= 2/0.05$) で 1 周期となるが、この中の 36 点は近傍点が 2 個で、4 点は 4 個である。ゆえに近傍点数の期待値は $(36 \cdot 2 + 4 \cdot 4) / 40 = 2.2$ となり、黒点の割合はこれを 40 で割って 0.055 となる。これもリカレンスプロットから得た値に近い。

Logistic 写像の場合、こうした簡便な考察は難しい。リカレンスプロットが示すようにパラメトリックな分布で示される一様構造も、点を列挙可能な周期構造も存在しないためである。

4.2 時系列の長さ と Error 数, リカレンスレートの関係

正弦波や Logistic 写像など単純な生成アルゴリズムで

作られた時系列は、乱数列よりも少ない数の基底で表現できることが確認できた。これは合理的な結果と考えられる。現実的な観測時系列は、ある程度の規則性にノイズが入ったものが想定されるので、各条件下において Fig.6 から Fig.10 までの赤色の線より下側の挙動を示すことが予想される。

Fig.6 により、500 点の時系列では、たとえ乱数列であっても 30 本程度の無作為な基底で 90% 以上のビット列を表現可能なが示された。Fig.7 からは、30 本程度の無作為な基底があれば、時系列が 10000 点程度であっても 90% 以上のビット列を表現可能であることが示された。グラフの形状からは、リカレンスプロットが扱える限り、もっと長い時系列でも同様であると考えられる。

逆に、たとえ正弦波のような単純な時系列であっても、基底を無作為にとっている限り 2% 程度は再現できないビットが出現することが分かる。今回利用した正弦波は $x_n = \sin(0.05\pi n)$ によって作成されているため、連続した 40 本の基底を取れば時系列のサイズに関係なく演算なしにすべてのビット列を完全に表現可能である。

Logistic 写像の挙動は興味深い。時系列が短くリカレンスレート ε も小さな場合 (Fig.6), Error の数は正弦波のそれに近い。しかし、時系列が十分に長くなると正弦波とは異なる Error の数を示すようになる。より多くの時系列で系統的にこの値を調べることで、時系列の背後にある決定論性に関して何らかの指標を開発することができる可能性がある。また ε が大きくなれば Error の数は乱数列のそれに近くなる。これは ε が大きいと Fig.4 における縦の解像度が悪くなり、時系列の微細な構造が取りにくくなるためだと考えられる。

Fig.8, Fig.9 からはリカレンスレート ε が大きくなるにつれて Error の数が減っていくことが分かる。これは $\varepsilon = 0.5$ で最小になるわけではなく、Fig.10 によれば、一方的に減少するのみである。この理由についての考察は今後の課題である。

5. 結論

フーリエ変換との対比から、リカレンスプロットを利用して非線形時系列でも適切に利用できる擬似的な基底を考えた。リカレンスプロットはビット列の集合であるため、各基底もビット列となり、その結合として AND, OR, NOT からなるビット演算を考えた。無作為にとった基底の本数と表現力の関係を調べた。基底の本数に応じて指数関数的に Error の数は減り、時系列の長さに対しても規格化した Error の数は指数関数で近似できることを確かめた。今後は基底を無作為ではなく、効率的に選択する方法を考え、また時系列の種類と基底の関係を横断的に調べてみたい。

ところで、ある列挙可能な自然数の集合には、対応する整

数係数のディオファントス方程式が存在することが保証される [17]. 時系列をリカレンスプロットでビット表現 (自然数列表現) し, 各点を基底と式で表現できれば, それは列挙可能な自然数の集合となるため, 何らかのディオファントス方程式へと対応づけられる. これは時系列やネットワーク, 言語などを数論と結びつける端緒になるだろう.

謝辞 本研究は平田祥人准教授 (筑波大学) の多大な助言を得て進めた. 本研究は JSPS 科研費 21K12068 の助成を受けた.

参考文献

- [1] Strogatz, S. H.: *Nonlinear dynamics and chaos with student solutions manual: With applications to physics, biology, chemistry, and engineering*, CRC press (2018).
- [2] Briggs, K. M.: Feigenbaum Scaling in Discrete Dynamical Systems, PhD Thesis, University of Melbourne, Melbourne (1997).
- [3] 城真範, 平田祥人, 合原一幸: リカレンスプロットとフーリエ変換の類似性と相違性, 生産研究, Vol. 72, No. 2, pp. 137–138 (2020).
- [4] Eckmann, J.-P., Kamphorst, S. O., Ruelle, D. et al.: Recurrence plots of dynamical systems, *World Scientific Series on Nonlinear Science Series A*, Vol. 16, pp. 441–446 (1995).
- [5] Zhang, J. and Small, M.: Complex network from pseudoperiodic time series: Topology versus dynamics, *Physical review letters*, Vol. 96, No. 23, p. 238701 (2006).
- [6] Marwan, N., Romano, M. C., Thiel, M. and Kurths, J.: Recurrence plots for the analysis of complex systems, *Physics reports*, Vol. 438, No. 5-6, pp. 237–329 (2007).
- [7] beim Graben, P. and Hutt, A.: Detecting recurrence domains of dynamical systems by symbolic dynamics, *Physical Review Letters*, Vol. 110, No. 15, p. 154101 (2013).
- [8] Hirata, Y., Horai, S. and Aihara, K.: Reproduction of distance matrices and original time series from recurrence plots and their applications, *The European Physical Journal Special Topics*, Vol. 164, No. 1, pp. 13–22 (2008).
- [9] Hirata, Y., Komuro, M., Horai, S. and Aihara, K.: Faithfulness of recurrence plots: a mathematical proof, *International Journal of Bifurcation and Chaos*, Vol. 25, No. 12, p. 1550168 (2015).
- [10] Khor, A. and Small, M.: Examining k-nearest neighbour networks: Superfamily phenomena and inversion, *Chaos: An Interdisciplinary Journal of Nonlinear Science*, Vol. 26, No. 4, p. 043101 (2016).
- [11] Hirata, Y., Oda, A., Ohta, K. and Aihara, K.: Three-dimensional reconstruction of single-cell chromosome structure using recurrence plots, *Scientific reports*, Vol. 6, No. 1, pp. 1–9 (2016).
- [12] Hirata, Y. and Aihara, K.: Dimensionless embedding for nonlinear time series analysis, *Physical Review E*, Vol. 96, No. 3, p. 032219 (2017).
- [13] Vershynin, R.: *High-dimensional probability: An introduction with applications in data science*, Vol. 47, Cambridge university press (2018).
- [14] Giuliani, A. and Manetti, C.: Hidden peculiarities in the potential energy time series of a tripeptide highlighted by a recurrence plot analysis: a molecular dynamics simulation, *Physical Review E*, Vol. 53, No. 6, p. 6336 (1996).
- [15] Wikipedia: Functional completeness, Wikipedia (online), available from (https://en.wikipedia.org/wiki/Functional_completeness) (accessed 2021-06-02).
- [16] Shulgin, A.: libpng++ Documentation, (online), available from (<http://charette.no-ip.com:81/programming/doxygen/libpng++/>) (accessed 2021-06-02).
- [17] Matiyasevich, Y. V., Garey, M. R. and Meyer, A.: *Hilbert's tenth problem*, MIT press (1993).