2023 International Symposium on Nonlinear Theory and Its Applications
NOLTA2023, September 26-29, 2023, Catania and Online

Subseries of Chaotic and Random Neural Networks

Hitoaki YOSHIDA†, Takeshi MURAKAMI† and Satoshi KAWAMURA†

†Iwate University
18-33 Ueda, Morioka, Iwate 020-8550, Japan
Email: hitoaki@iwate-u.ac.jp, mtakeshi@iwate-u.ac.jp, kawamura@iwate-u.ac.jp

Abstract–Chaotic and random neural network (CRNN) generates a time series that consists of various subseries under well-designed conditions. The knowledge of subseries is informative to a fast and secure pseudo-random-number (PRN) generator. Experimental results suggest that the typical time series from CRNN cannot divide into subseries without information on the structure of CRNN. The results are valuable for security applications.

1. Introduction

For the information security of IoT devices and embedded systems, we have studied the computer-generated PRN series in these years [1-3]. PRN series is extracted from the outputs of CRNN with fixed-point arithmetic (Q5.26) and the activation function APLF3 or the function for 10^{13}-bps-order generation speed [4], which is called random identity function (RIF), hereinafter. We have studied the time series generated from CRNN which is composed of 2 cyclic neural networks (NNs) [1-4], and the time series consists of 2 independent subseries. The subseries plays important roles in security applications (*vide infra*). In this paper, we have discussed more general features of subseries from cyclic and multi-cyclic NNs and reported on representative empirical results of new time series composed of 3 subseries.

2. Time Series and Subseries

2.1. Time Series Generated from Linear NN

Let us consider an artificial neural network arranged in a straight line (Figure 1) without a self-regressive output. Artificial neurons are numbered N_0, N_1, …, N_k, …, $\mathrm{N}_{n\text{-}1}$, where n is the number of neurons. The output of N_{k+1} is defined as Equation 1, where f is an activation function, x_k is an output from N_k, w_k is a synaptic weight, I_k is an external input (k = 0, 1, 2, …, n-1). The network generates a discrete time series. If N_0 outputs only at time t = 0 and then following N_k outputs at t = k, it results in a 1-dimensional time series; x_0, x_1, x_2, …, x_k, …, $x_{n\text{-}1}$. A *time series* as a universal set is expressed by U, here. A *subseries* is defined as subsets of U generated from the same unit, and not to mix with each other.

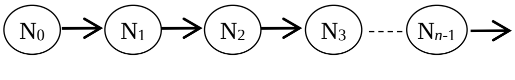


Figure 1: Linear NN Consists of n Neurons (Ln-NN).

$$x_{k+1} = f\,(w_k x_k + I_k) \qquad (1)$$

2.2. Subseries Generated from Linear NN

Let us consider how to generate multiple time series from the Ln-NN besides the time series mentioned in the last section. The time series is called subseries S_0, here.

(*Method-1*)
N_0 outputs at t = 1 also and then following N_k outputs at t = k+1 based on Equation 1, which gives another time series that does not mix with S_0. It is defined as *subseries* that originated from the time difference.

(*Method-2*)
All neurons output at t = 0 and following N_k outputs based on Equation 1 give independent time series, which are also defined as *subseries* that originated from the initial-position difference. In this case, a subseries that starts from N_k at t = 0 is called *Subseries k* (S_k). The maximum number of subseries is n based on the definition.

Both methods are important, but mainly *Method-2* is used in this paper because *Method-1* needs a time limit or appropriate time interval for the finite number of neurons.

The subseries sometimes eventually join the same trajectory. To avoid the confluence some parameters (*e.g.*, the external inputs) should be changed in a unit of time.

2.3. Benefit of Subseries

We have reported valuable applications by using subseries [1-4]. For example, if periods of subseries are designed within a measurable range, the degree of safety can be quantitatively estimated. The higher dimension vector of multiple subseries has implemented the ultra-long period time series, *e.g.*, $10^{100000000}$ with 33554432 subseries.

The activation function RIF (Equation 2) needs PRN (pr) [6]. A possible method is to use the output from NN as pr. Adopting the output from the same subseries as pr, the period of the time series keeps measurable. While, adopting the output from the other subseries as pr, the period is often too long to measure. The knowledge of subseries is informative to implement a faster and securer PRN generator.

$$f\,(x) = x + pr \qquad (2)$$

2.4. Subseries Generated from Cyclic NN

Next, let us consider Cyclic NN which has a closed circulation pathway as shown in Figure 2, where Cn-NN

ORCID iDs First Author: 0000-0002-2255-6937, Second Author: 0000-0002-3189-3265, Third Author: 0000-0001-9436-4371

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International.

generates one-way outputs without mixing among subseries. The number of subseries (σ) is n with *Method-2* without a time limit.

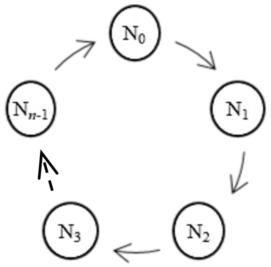


Figure 2: Cyclic NN Consists of n Neurons (Cn-NN).
Cn is named after the largest circulation pathway in a NN.

2.5. Subseries Generated from Multi-Cyclic NNs

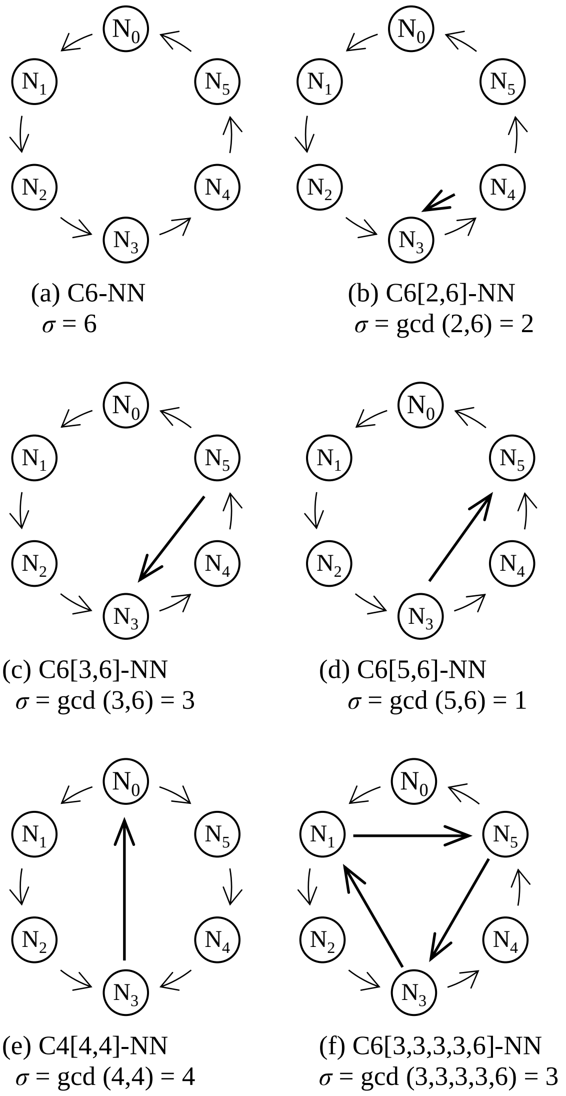
Examples of cyclic and multi-cyclic NNs are shown in Figure3. Figure 3a shows a 6-membered cyclic NN, which generates 6 subseries (S_0-S_5) with *Method-2* without a time limit. C6[i,j]-NN is defined as a 6-membered multi-cyclic NN that is composed of an i-membered cyclic NN and a j-membered cyclic NN. The number of subseries (σ) of C6[i,j]-NN is calculated by a greatest common divisor of i and j because all subseries must repeat periodically through all circulation pathways. The complicated example in Figure 3f is calculated by almost the same way.

Subseries generated from C6[3,6]-NN (Figure 3c) are shown in Table 1 as the typical example. The initial value of N_k is defined as $x_k(0)$ and the subseries originating from $x_k(0)$ is expressed as S_k (k=0,1,…,5). Some subseries mix each other as time advances so that the unified subseries which originated from $x_k(0)$ and $x_l(0)$ is expressed as $S_{k,l}$. According to Table 1 the subseries from C6[3,6]-NN consists of three 2-dimensional subseries, $S_{0,3}$, $S_{1,4}$, and $S_{2,5}$. If $t = 3k$ (k = 0,1,2,…), three subseries are expressed as 2-D vector, $S_{0,3} = (x_0, x_3)$, $S_{1,4} = (x_1, x_4)$, $S_{2,5} = (x_2, x_5)$, similarly if $t = 3k+1$, as $S_{0,3} = (x_1, x_4)$, $S_{1,4} = (x_2, x_5)$, $S_{2,5} = (x_0, x_3)$, and if $t = 3k+2$, as $S_{0,3} = (x_2, x_5)$, $S_{1,4} = (x_0, x_3)$, $S_{2,5} = (x_1, x_4)$. Examples of various multi-cyclic NNs are shown in Figure 3, which generate many types of subseries.

Table 1: Subseries Generated from C6[3,6]-NN.

neuron	Time / t							
	0	1	2	3	4	5	6	7
N₀	S_0	S_5	S_4	S_3	$S_{2,5}$	$S_{1,4}$	$S_{0,3}$	$S_{2,5}$
N₁	S_1	S_0	S_5	S_4	S_3	$S_{2,5}$	$S_{1,4}$	$S_{0,3}$
N₂	S_2	S_1	S_0	S_5	S_4	S_3	$S_{2,5}$	$S_{1,4}$
N₃	S_3	$S_{2,5}$	$S_{1,4}$	$S_{0,3}$	$S_{2,5}$	$S_{1,4}$	$S_{0,3}$	$S_{2,5}$
N₄	S_4	S_3	$S_{2,5}$	$S_{1,4}$	$S_{0,3}$	$S_{2,5}$	$S_{1,4}$	$S_{0,3}$
N₅	S_5	S_4	S_3	$S_{2,5}$	$S_{1,4}$	$S_{0,3}$	$S_{2,5}$	$S_{1,4}$

If we monitor outputs from N_k without any information, can we find the structure of NN or divide the time series into subseries? If it is possible, it may be a threat to security applications. The dimension of the time series is informative because it becomes the hints of the subseries.



(a) C6-NN
σ = 6

(b) C6[2,6]-NN
σ = gcd (2,6) = 2

(c) C6[3,6]-NN
σ = gcd (3,6) = 3

(d) C6[5,6]-NN
σ = gcd (5,6) = 1

(e) C4[4,4]-NN
σ = gcd (4,4) = 4

(f) C6[3,3,3,3,6]-NN
σ = gcd (3,3,3,3,6) = 3

Figure 3: Examples of Cyclic and Multi-Cyclic NNs.

2.6. Dimension of Subseries

The dimension of the subseries is not always constant, *e.g.*, Figure 3e is a 4-membered cyclic NN which is composed of two circulation pathways. The subseries from the NN are shown in Table 2. The dimension of the subseries is 2 on N₁ and N₅ (or N₂ and N₄), yet the dimension of the subseries is 1 on N₀ (or N₃). In other words, the time series consists of 4 subseries, $S_{0,}$ S_3, $S_{1,5}$, and $S_{2,4}$.

Table 2: Subseries Generated from C4[4,4]-NN.

neuron	Time / t							
	0	1	2	3	4	5	6	7
N₀	S_0	S_3	$S_{2,4}$	$S_{1,5}$	S_0	S_3	$S_{2,4}$	$S_{1,5}$
N₁	S_1	S_0	S_3	$S_{2,4}$	$S_{1,5}$	S_0	S_3	$S_{2,4}$
N₂	S_2	S_1	S_0	S_3	$S_{2,4}$	$S_{1,5}$	S_0	S_3
N₃	S_3	$S_{2,4}$	$S_{1,5}$	S_0	S_3	$S_{2,4}$	$S_{1,5}$	S_0
N₄	S_4	S_5	S_0	S_3	$S_{2,4}$	$S_{1,5}$	S_0	S_3
N₅	S_5	S_0	S_3	$S_{2,4}$	$S_{1,5}$	S_0	S_3	$S_{2,4}$

However, the dimensions of the subseries from C6[2,6]-NN, C6[3,6]-NN, C6[5,6]-NN, C6[3,3,3,3,6]-NN are constant every time. The time series U from C6[2,6]-NN consists of two 3D subseries, and that from C6[3,6]-NN and C6[3,3,3,3,6]-NN consist of three 2D subseries. The time series U from C6[5,6]-NN consists of only one 5D time series, *i.e.*, which cannot divide into subseries.

3. Time Series Analysis

This section aims to confirm whether the chaos time series analysis can give any information on the subseries of CRNN. The structure of CRNN is common to the NN, but the activation function and the calculation method are unique as follows: the output from CRNN is calculated with fix-point arithmetic Q5.16, which allows overflow and underflow, and the activation function is APLF or RIF.

In this work, the time series from C6[3,6]-NN (Figure 3c) is analyzed as a typical example. The lower 30 b of the same subseries is utilized for the pseudo-random number pr of RIF. External inputs (I_k) are constant values, but a small perturbation (I_D) is added to I_k (k = 3, 4, 5) when t = 3q (q = 0, 1, 2, …) due to generating subseries that have different trajectories. 7-bit-rotate-left instruction is additionally executed before the iteration only for x_1, x_2, and x_3. The following Figures show the results on the time series from N_5 with an embedded dimension, m = 2, and with a lag time τ = 3 as theoretical values for $S_{2,5}$.

3.1. Attractor of the Time Series

The attractor of the time series that is embedded in time-delay coordinates with m = 2 and τ = 3 is shown in Figure 4. The number of points in the target time series is 5000. The outputs are randomly distributed in the whole domain of Q5.16. The results of τ = 1 and τ = 2 give similar attractors, which are nearly indistinguishable.

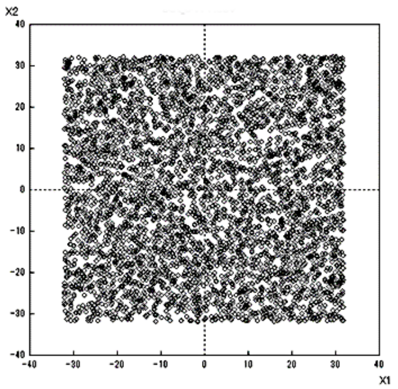


Figure 4: Attractor of Time series Generated from N5.

3.2. Local vs. Global Plot of Time Series

The basic approaches of chaos time series analysis are to estimate the fractal dimension or the Lyapunov exponent of the target time series. Owing to confirmation of the difficulty, the result of the Local vs. Global Plot [5] of the time series is shown in Figure 5, which shows no flat region. The result suggests that high randomness of the time series interrupts to obtain the accurate Lyapunov exponent. Results with different m and τ shows also similar tendencies.

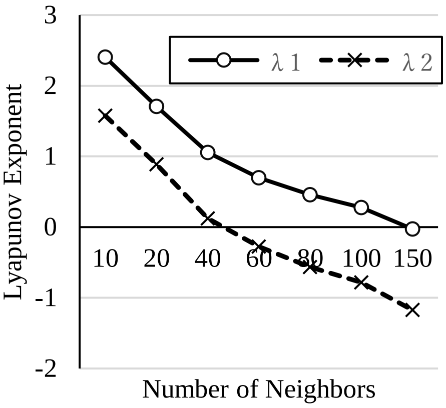


Figure 5: Local vs. Global Plot of the Time Series.

3.3. Estimation of Embedding Dimension

The result of the GP algorithm [6] with various embedding dimensions (2 ≤ m ≤ 9) is shown in Figure 6. The ordinate is the correlation dimension of the embedded attractor, r is a scale factor, and the lag time τ is 3. The correlation dimension increases with embedded dimension (m = 2, 3, …, 9), which is the distinctive feature of random time series. It is also true to analyze by the box-counting dimension or with other lag times; τ = 1, 2, 3, 4, 5, 6, which is related to the interval time of the subseries. It suggests that the method is hard to determine the fractal dimension and the lag time for the attractor. Therefore, it also suggests that to obtain the number of time series is difficult without additional information.

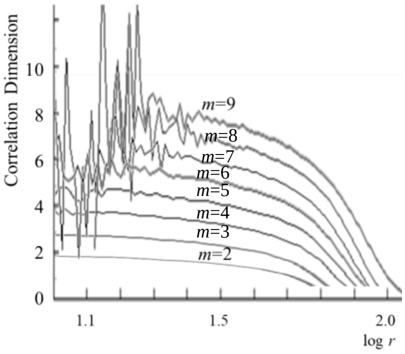
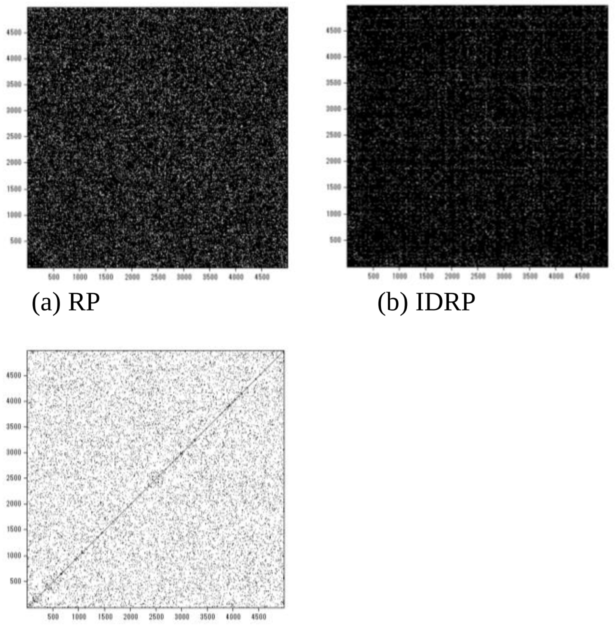


Figure 6: Correlation Dimension by GP algorithm.

3.4. Determinism Analysis of the Time Series

The above results suggest that the basic approach is hard to afford any information on subseries, but many efficient methods are known on deterministic nonlinear prediction. Next, owing to the denial of predictability with the prediction method, the result of the determinism analysis of the time series is shown in Figure 7. Figure 7a shows the recurrence plot (RP) of the time series and Figure 7b shows the iso-directional recurrence plots (IDRP), and Figure 7c shows the Iso-directional Neighbors Plots

(IDNP). IDNP is the intersection of RP and IDRP, the cardinality of which is known as an index of determinism of the time series [7]. The low cardinality of the point set in Figure 7c suggests that the deterministic character of the time series is low and therefore it is not predictable.



(a) RP

(b) IDRP

(c) IDNP (RP∩IDRP)

Figure 7: (a) the Recurrence Plots (RP), (b) the Iso-directional Recurrence Plots (IDRP) of the time series from N_5, and the Iso-directional Neighbors Plots (IDNP).

3.5. Results of NIST SP800-22 Tests

For a security application, the upper-2-bit has been removed from the time series according to the previous study [1-4]. Owing to the evaluation of the randomness the results of the NIST SP800-22 tests are shown in Table 3 and 4, which show only the representative result due to space constraints. The fail rates (%) of repeating 100 tests are listed in the Tables [8]. The result is acceptable for PRN generators for cryptographic applications.

Table 3: Result of "Proportion of Sequences Passing a Test" on the NIST SP800-22 Test Suite. [a)]

FR	FB	CS	RU	LR	RK	OT	AE	RE	RV
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.38	0.17

Table 4: Result of "Uniform Distribution of P-values Test" on NIST SP800-22 Test Suite. [a)]

FR	FB	CS	RU	LR	RK	OT	AE	RE	RV
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

a) Abbreviations of test names: FR; Frequency Test, FB; Frequency Test within a Block, CS; Cumulative Sums Test, RU; Runs Test, LR; Test for the Longest Run of Ones in a Block, RK; Binary Matrix Rank Test, OT; Overlapping Template Matching Test, AE; Approximate Entropy Test, RE; Random Excursions Test, RV; Random Excursions Variant Test.

4. Conclusion

CRNN generates a time series that consists of various subseries under well-designed conditions. The knowledge of subseries is informative to a fast and secure PRN generator. Experimental results suggest that the typical time series from CRNN cannot divide into subseries without detailed information on the structure of CRNN. The experimental results are valuable for security applications.

5. Future Work

The features of the subseries will be studied in detail due to the implementation of the far-longer period of the time series and to the further acceleration of PRN generation under various conditions on GPUs.

Acknowledgments

Part of the experimental results in this research was obtained using supercomputing resources at Cyberscience Center, Tohoku University. Special thanks to the staff members of Iwate University Super-Computing and Information Sciences Center.

References

[1] H. Yoshida, Y. Akatsuka and T. Murakami, "Implementation of High-Performance Pseudo-Random Number Generator by Chaos Neural Networks using Fix-Point Arithmetic with Perturbation," *Proceedings of Papers, NOLTA 2018*, pp.46-49, 2018.

[2] H. Yoshida, H. Fukuchi and T. Murakami, "Implementation of High-Speed Pseudo-Random-Number Generator with Chaotic and Random Neural Networks," *Proceedings of Papers, HICSS53 2020*, pp.6418-6425, 2020.

[3] Z. Liu, T. Murakami, S. Kawamura, and H. Yoshida, "Fast Stream Cipher based Chaos Neural Network for Data Security in CAN Bus," *Advances in Science, Technology and Engineering Systems Journal*, 5:5, pp.63-68, 2020.

[4] H. Yoshida and T. Murakami, "Activation Functions for Chaotic and Random Neural Networks," *Proceeding of Papers, JSST2022*, pp.351-354, 2022. *Idem*, "Implementation of Secure and Fast Pseudo-Random-Number Generator on GPU," *Proceedings of Papers, NOLTA 2022*, pp.5-8, 2022.

[5] T. Ikeguchi, K. Aihara, "Lyapunov spectral analysis on random data," *Int. J. Bifurc. Chaos Appl. Sci. Eng.*, vol.7, pp.1267-1280, 1997.

[6] M. Ding, C. Grebogi, E. Ott, T. Sauer and J. A. Yorke, "Estimating correlation dimension from a chaotic time series: when does plateau onset occur?", *Physica D* 69, pp.404-424, 1993.

[7] S. Horai, T. Yamada and K. Aihara, "Determinism Analysis with Iso-Directional Recurrence Plots," *IEEJ Trans. Electron. Inf. Syst.*, vol.122, pp.141-147, 2002.

[8] H. Yoshida, T. Murakami, and S. Kawamura, "Study on Testing for Randomness of Pseudo-Random Number Sequence with NIST SP800-22 rev. la," *Technical Reports of IEICE*, vol.110, pp.13-18, 2012. And references cited therein.