e-ISSN: 2687-4539
RESEARCH ARTICLE
Vol.5 / No.2 / 2023 / pp.59-64
https://doi.org/10.51537/chaos.1213070

The Modeling of the Rucklidge Chaotic System with Artificial Neural Networks

Zeynep Keles *,[1], **Guray Sonugur** *,[2] **and Murat Alcın** *,[3]
*Mechatronics Engineering, Faculty of Technology, Afyon Kocatepe University, Afyonkarahisar, Türkiye.

ABSTRACT Chaotic systems are nonlinear systems that show sensitive dependence on initial conditions, and an immeasurably small change in initial value causes an immeasurably large change in the future state of the system. Besides, there is no randomness in chaotic systems and they have an order within themselves. Researchers use chaotic systems in many areas such as mixer systems that can make more homogeneous mixtures, encryption systems that can be used with high security, and Artificial Neural Networks (ANNs) by taking the advantage of the order in this disorder. Differential equations in which chaotic systems are expressed mathematically are solved by numerical solution methods such as Heun, Euler, ODE45, RK4, RK5-Butcher and Dormand-Prince in the literature. In this research, Feed Forward Neural Network (FFNN), Layer Recurrent Neural Network (LRNN) and Cascade Forward Backpropogation Neural Network (CFNN) structures were used to model the Rucklidge chaotic system by making use of the MATLAB R2021A and Neural Network (NN) Toolbox. By comparing the results of different activation functions used in the modeling, the ANN structure that can best model the Rucklidge chaotic system has been determined. The training of the compared ANNs was carried out with the values obtained from the Euler numerical solution method, which can get satisfactory and fast results.

KEYWORDS
Rucklidge chaotic system
Euler algorithm
Artificial neural network

INTRODUCTION

Chaotic systems were discovered in 1960 by a meteorologist named Edward Norton Lorenz based on the meaningful results he obtained when he changed the initial values in the system he used to make weather forecasts by very small proportions. Lorenz's work proves that chaotic systems change unpredictably within certain limits, and one can only know within which probabilities they may act. After Lorenz's studies, many chaotic systems have been presented to the literature and improvements have been made on the systems by working on chaotic systems (Alcin *et al.* 2019; Avaroğlu *et al.* 2015; Liu *et al.* 2020; Prakash *et al.* 2020; Rajagopal *et al.* 2019; Ramakrishnan *et al.* 2022; Vaidyanathan *et al.* 2018).

Chaotic systems have been widely used in the design of chaotic oscillators (Tuna *et al.* 2019a), True Random Number Generators (Koyuncu *et al.* 2020a; Tuna *et al.* 2019b) and Pseudo Random Number Generators (Koyuncu *et al.* 2021; Tuna 2020); the modeling using ANNs (Koyuncu *et al.* 2020b), Image Encryption (Boyraz *et al.* 2022; Kiran *et al.* 2022; Ullah *et al.* 2022), synchronization.

Lee et al. studied the problem of continuous synchronization of a master-slave chaotic system in a sampled data environment by

considering both intermittent coupling and continuous coupling situations. They used the Euler approximation technique to analyze a continuous-time chaotic oscillator containing a nonlinear function. Their experiments with neurons show that using these neurons, ANNs can be implemented rapidly in hardware and the design time can be significantly reduced (Lee *et al.* 2010).

Azzaz et al. have presented to the literature a 3 dimension (3D) chaotic system created with automatically switched numerical resolution of new multiple 3D continuous chaotic systems. The designed chaotic system provides complex chaotic attractors and can automatically change their behavior through a chaotic switching rule. At the same time, some complex dynamic behaviors were investigated and analyzed in the study. The originality of the proposed architecture is that it allows to solve the problem

Manuscript received: 11 December 2022,
Revised: 10 March 2023,
Accepted: 23 March 2023.

[1] zeynepkeles.aku@gmail.com
[2] gsonugur@aku.edu.tr
[3] muratalcin@aku.edu.tr (**Corresponding Author**)

of finite sensitivity due to digital implementation, while providing a good trade-off between high security, performance and hardware resources (low power and cost) (Azzaz *et al.* 2013).

Çavuşoğlu et al. argue that chaotic systems are an alternative to the standard broad spectrum communication systems in the literature, since they can spread the spectrum of information signals to be transmitted over a wide area, simultaneously encode notification signals and perform these operations with simple and inexpensive chaotic circuit mechanisms. They carried out the signal masking application by considering the Lorenz chaotic system (Cavusoglu 2014).

According to the work of Koyuncu et al. in 2017, the digital implementation of the hardware and the experimental results of the Field Programmable Gate Array (FPGA) circuit show that a promising technique can be applied in efficient embedded cryptographic communication systems. ANN-based Rössler system was created to demonstrate the effectiveness of using neurons in fast realization of ANNs in embedded systems (Koyuncu *et al.* 2017).

In 2018, Koyuncu and colleagues, who added the numerical modeling of the new 3-D Jerk chaotic system with the fifth-order Runge-Kutta-Butcher algorithm to their work on Matlab, trained a multi-layered feed-forward ANN with the data set obtained from the modeling and analyzed the results obtained from this network. The weights and bias values taken as reference from the numerical solution are used for the design and implementation of the ANN-based 3-D New Jerk Chaotic oscillator on FPGA (Koyuncu *et al.* 2020b).

In the continuation of 2018, Koyuncu et al. argued that the most basic structure used in chaos-based applications such as cryptology, secure communication, industrial control, ANN, Random Number Generators (RNGs) and image processing is a chaotic oscillator structure that generates the chaotic signal, and they performed an oscillator design that has not been presented in the literature before on FPGA in 32-bit IEEE-754-1985 floating point number standard (Koyuncu *et al.* 2020b).

In this study, different ANN structures of the Rucklidge Chaotic System and different activation functions in these structures were trained by using them. Success results of each training were compared and ANN structures that could best model the system were presented.

In Section 2, information is given about the Euler method used in this study, and the time series and phase portraits obtained from the solution of the Rucklidge Chaotic System with the Euler method are presented. In Section 3, general information about ANN structures and activation functions is given. In Section 4, the results of different ANN structures and the trainings performed using different activation functions in these structures are presented. In the conclusion part, the success results of the trainings are compared and discussed.

MATERIAL AND METHODS

Rucklidge Chaotic System

The Rucklidge system is a model of a double convection process in which motion is limited to long thin coils that models convection in an applied vertical magnetic field and a smoothly rotating fluid layer (Dong *et al.* 2021). The second-order nonlinear Rucklidge chaotic system is defined by the following equations.

$$dx/dt = -ax + by - yz \tag{1}$$

$$dy/dt = x \tag{2}$$

$$dz/dt = y^2 - z \tag{3}$$

a and b, which are in the differential equation sets of the Rucklidge chaotic system, are the system parameters. Besides, x, y and z represent the dynamic variables of the system. The system parameters based on this study are $a = 2$ and $b = 6.7$ and the initial conditions are $x_0 = 2, y_0 = 2, z_0 = 2$ and the system shows a chaotic behavior. In addition, since this system is an Ordinary Differential Equation (ODE), this equation can also be solved using MATLAB function libraries.

Euler Method

Euler's method is one of the methods used in the numerical solution of differential equations. Given first-order ODE as follows:$y' = f(x, y)$ and initial values $y(x_0) = y_0$ and trying to solve this equation in the range of x-values $[x_0, x_n]$, our goal is to get a $P = [x_0, x_1, x_2, ..., x_n]$ is to approximate the value of the y(x) solution at each of the x values. Given $y(x)$, the first value we have to guess is $y(x_1)$. The symbol y' represents the derivative of the function f, where x is the independent variable and y the dependent variable. When expressed by Taylor's theorem:

$$y(x_1) = y(x_0) + y'(x_0)(x_1 - x_0) + \frac{y'(c)}{2}(x_1 - x_0)^2 \tag{4}$$

Since $c \in (x_0, x_1), y'(x_0) = f(x_0, y(x_0))$

$$y(x_1) = y(x_0) + f(x_0, y(x_0))(x_1 - x_0)\frac{y'(c)}{2}(x_1 - x_0)^2 \tag{5}$$

Here, $\frac{y'(c)}{2}(x_1 - x_0)^2$ is a small error value and may not be taken into account. Then, the Equation 6 is obtained.

$$y(x_1) \approx y(x_0) + f(x_0, y(x_0))(x_1 - x_0) \tag{6}$$

Similarly, for $k = 1, 2, \ldots, n-1$, $y(xk+1)$ can be calculated approximately.

$$y(x_{k+1}) \approx y(x_k) + f(x_k, y(x_k))(x_{k+1} - x_k) \tag{7}$$

Here $y(x_k)$ will be known from previous calculations. As with numerical integration methods, it is practical to take the division to consist of sub-intervals of equal width. If we express it in this way, our equation will be as it is expressed in Equation 8.

$$(x_{k+1} - x_k) = \Delta x = \frac{(x_n - x_0)}{n} \tag{8}$$

In the study of numerical methods for differential equations, this quantity is usually denoted by h. Here is our general relationship

$$y(x_{k+1}) \approx y(x_k) + f(x_k, y(x_k))\Delta x \tag{9}$$

If we show our approximations for $y, y_0, y_1, \ldots, y_n$ values, $x_0, x_1, \ldots, x_n, (y_0 = y(x_0), y_1 \approx y(x_1), etc.),$ then approximately $y(x)$ can be calculated iteratively in the P part.

$$y_{k+1} = y_k + f(x_k, y_k)\Delta x \tag{10}$$

The reason why the Euler method is preferred in this study is to obtain good results in a short time by reducing the processing load in modeled ANNs. The time series and the phase portraits obtained from the solution of the Rucklidge chaotic system by Euler's method are shown in Fig.1 and Fig. 2, respectively.

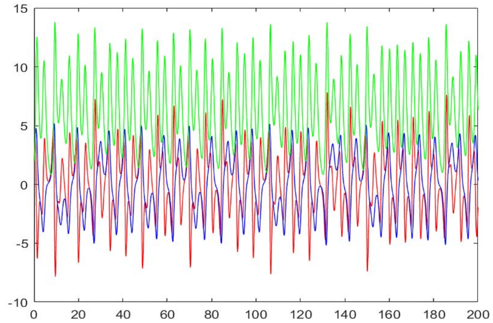

Figure 1 The time series of Rucklidge chaotic system using Euler's numerical solution.

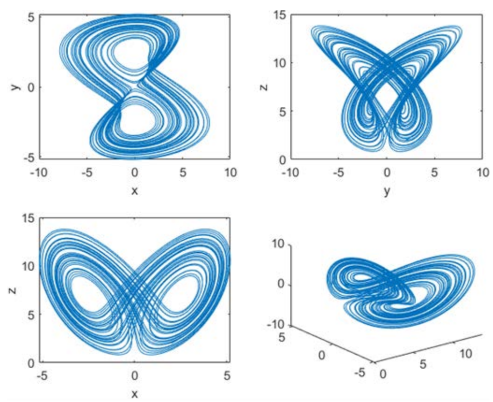

Figure 2 The phase portraits of Rucklidge chaotic system using Euler's numerical solution.

ARTIFICIAL NEURAL NETWORKS (ANNS)

Artificial Neural Network (ANN), is a computer algorithm inspired by the neuron system in order to imitate the process of producing new information with learning in the human brain. It has been developed so that machines can recognize the desired pattern in complex data and generally performs better than other algorithms when recognizing audio, image or video segments. An ANN consists of inputs (X), weights (W), addition function, activation function (Tansig, Purelin, Satlins etc.) and outputs. In ANN modeling, the relationship between Inputs (X) and Outputs (Y) is $Y = f(X) + b$. Here Weight (W) information is used to reduce the error (b).

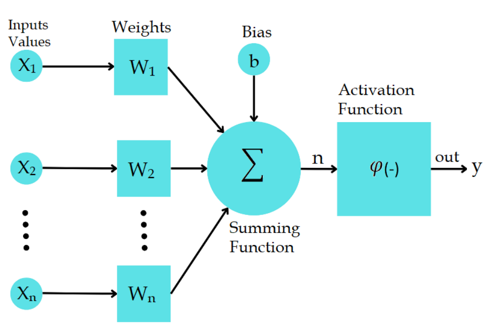

Figure 3 Artificial Neuron Structure

ANNs can also consist of one hidden layer or more than one hidden layer, and different activation functions can be used in these hidden layers. In this project, multi-layer ANNs were studied and single-layer ANNs were excluded. In ANNs, there is no certain rule such as how many hidden layers will be found or how many neurons will be used, zthey are usually created according to the needs of the problem and the best model is tried to be reached by using trial and error method.

In this network structure, the information received from the inputs is transmitted to the hidden layer and has a one-way working principle. The output value is determined by processing the information in the hidden layers and the output layer.ANNs can be created in various structures such as FFNN, LRN, and CFBN. Since MATLAB R2021A program has functions that allow the modeling of the above-mentioned ANNs, the MATLAB R2021A program was used in this study and various variations of the specified ANNs were created and the results were compared.

Feed-Forward Back-propagation Neural Network (FFNN)

In this network structure, the information received from the inputs is transmitted to the hidden layer and has a one-way working principle. The output value is determined by processing the information in the hidden layers and the output layer.

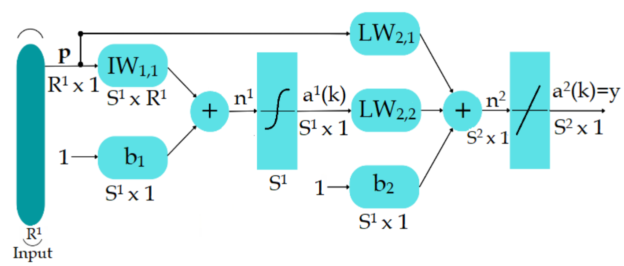

Figure 4 The structure of Feed-Forward Backpropagation Neural Network

Layered-Recurrent Neural Network (LRN)

In Layered-Recurrent Neural Networks, the outputs in the hidden layers and the output layers are also fed back as inputs. Thus, it has a bidirectional working principle. Since there is feedback, this type of ANNs have memory. It is shown in Fig 5

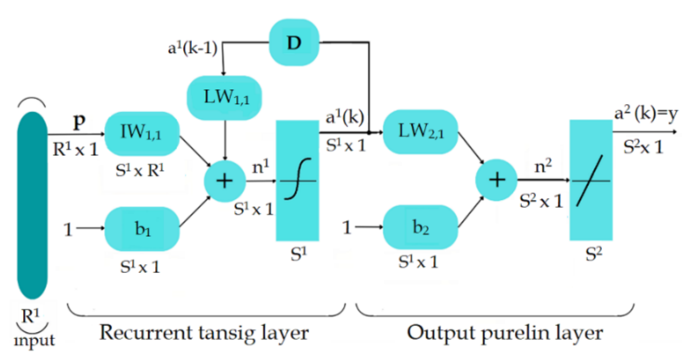

Figure 5 The structure of Layered-Recurrent Neural Network

Cascade Forward Back-propagation Neural Network (CFNN)

Cascade Forward Back-propagation Neural Networks are similar to feed-forward networks. The difference is that the data from the input contains a link to each hidden layer.

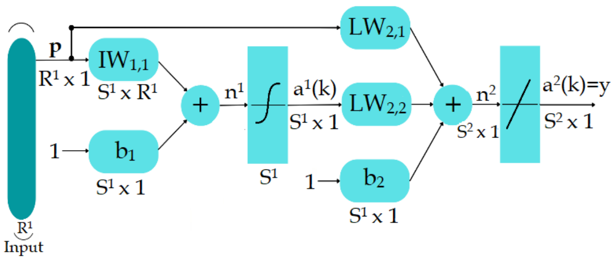

Figure 6 The structure of Cascade Forward Back-propagation Neural Network

In this research, the solution values produced by applying the Euler numerical solution algorithm were given as input to the modeled ANNs, and the ability to model the Rucklidge chaotic system of FFNN, LRN, CFBN were analyzed and the results obtained are presented.

Activation Functions

Activation Functions are used to decide whether neurons will be active or not by processing the information from the summing function. For this reason, it is important to choose an appropriate activation function for the solution of the problem. In this study, the most suitable activation function options for the problem were determined by using different activation functions.

Hyperbolic Tangent Sigmoid Activation Function (tansig)

Hyperbolic tangent sigmoid activation function is an S-shaped activation function that compresses the input values in the infinite space range to the range of -1 and 1 and is expressed mathematically as follows:

$$f(n) = \frac{2}{1 + e^{-2\pi}} - 1 \tag{11}$$

The input-output relationship of the hyperbolic tangent sigmoid transfer function is demonstrated in Fig. 7, where n is the input value and a is the output value for this activation function.

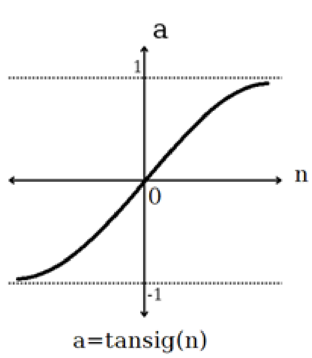

Figure 7 Hyperbolic Tangent Sigmoid Activation Function

Linear Activation Function (Purelin)

Purelin is a linear transfer function used by neural networks and it is mathematically expressed as follows:

$$f(n) = n \tag{12}$$

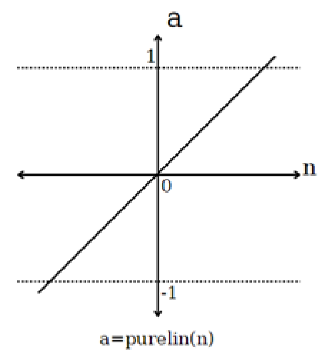

Figure 8 Purelin Activation Function.

Symmetric Saturating Linear Activation Function (Satlins)

The Satlins function is an inverse Z-shaped activation function that transmits to the output in the space interval [-1 1], gives an output of -1 for values between -1 and infinity, and gives an output of 1 for values between 1 and infinity. This function is mathematically expressed as follows:

$$f(n) = \begin{cases} -1, & n \leq -1 \\ n, & -1 < n < 1 \\ 1, & n \geq 1 \end{cases} \tag{13}$$

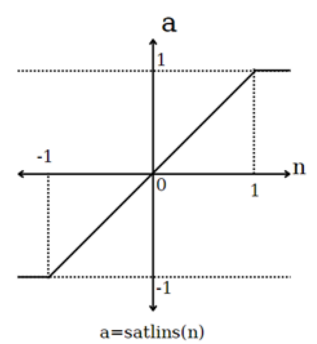

Figure 9 Satlins Activation Function.

In this study, 70% of the $3x10.000$ data obtained by the Euler method of the Rucklidge chaotic system was reserved for training, 15% for validation and 15% for testing, and was used in network training of 14 different ANNs. These ANN structures, Trainlm, Trainbr, Trainscg training functions; Various hidden layer numbers and sequences of Tansig, Purelin, Satlins activation functions have been created on different ANN types such as FFNN, LRN, and CFBN.

Table 1 Training results on modeling the Rucklidge chaotic system with different ANN structures

No	Network Dimension	Model	Training Function	1^{st} Activation Function	2^{nd} Activation Function	3^{rd} Activation Function	Best Performance
1	8x8x3	CFNN	Trainlm	Tansig	Purelin	Satlins	15.2867
2	8x5x3	CFNN	Trainbr	Tansig	Purelin	Purelin	$1.57x10^{-11}$
3	8x5x3	CFNN	Trainlm	Tansig	Purelin	Purelin	$4.4519x10^{-15}$
4	8x3	CFNN	Trainlm	Tansig	Purelin	-	$5.13x10^{-12}$
5	8x5x3	FFNN	Trainbr	Tansig	Purelin	Satlins	0.0012001
6	5x3	FFNN	Trainlm	Tansig	Purelin	-	$6.6562x10^{-9}$
7	5x5x3	FFNN	Trainlm	Tansig	Satlins	Purelin	$4.3206x10^{-9}$
8	5x5x3	FFNN	Trainlm	Tansig	Purelin	Satlins	16.3083
9	8x3	FFNN	Trainlm	Tansig	Purelin	-	$2.297x10^{-11}$
10	8x8x3	FFNN	Trainbr	Tansig	Purelin	Satlins	15.2869
11	8x8x3	FFNN	Trainlm	Tansig	Satlins	Purelin	$5.4601x10^{-7}$
12	8x8x3	FFNN	Trainscg	Tansig	Purelin	Satlins	15.287
13	8x5x3	LRN	Trainlm	Tansig	Purelin	Purelin	$3.7906x10^{-12}$
14	8x3	LRN	Trainlm	Tansig	Purelin	-	$2.94x10^{-10}$

FINDINGS AND DISCUSSION

In this study, 14 different ANN structures have been trained with respect to different Network Dimension, Training Function, Activation Function. In these ANN structures, there are 2 hidden layers in 8x3 and 5x3, structures. Here, the first and second numbers express the number of neurons in the first and the second hidden layer, respectively. Also, the third refers to the number of neurons in the output layer. Apart from these, there are 1 hidden layers in 8x3 and 5x3 structures. Here, the first and second numbers denote the number of neurons in the hidden layer and output layer, respectively.

Levenberg-Marquardt backpropagation (Trainlm), Bayesian regularization backpropagation (Trainbr) and Scaled conjugate gradient backpropagation (Trainscg) functions have been used as Training Function in these structures. Hyperbolic tangent sigmoid transfer function (Tansig), Linear transfer function (Purelin) and Symmetric saturating linear transfer function (Satlins) functions have been used as Activation Function in these structures.

The results obtained by changing the training and activation functions in different ANN structures of the Rucklidge chaotic system are given in Table 1.

According to the data obtained from Table 1, the ANN structures that can best model the Rucklidge chaotic system are marked with red in the table, and the best result is the CFBN 3rd architecture, which is the 8x5x3 hidden layer Trainlm training function and has the tansig-purelin-purelin activation function order and the best test performance is 4.4519x10-15. The worst test performance is the 8th architecture, an FFNN with 5x5x3 hidden layer Trainlm training function and Tansig- Purelin- Satlins activation function sequence. Based on this comparison, it can be concluded that FFNN gives better results for the Rucklidge chaotic system among network structures under the same conditions. For this reason, the network structure no. 9, which has fewer neurons and gives satisfactory accuracy values, was preferred in the error analysis.

For 100 iterative values produced by the Rucklidge Chaotic System with the Euler algorithm and 100 iterative values produced by the selected reference number 9 FFNN, Mean Squared Error (MSE), Root Mean Squared Error (RMSE) and Normalized Mean Squared Error (NMSE) values are obtained for 3 outputs, namely X, Y and Z. Here, X, Y and Z represent the produced outputs of the 9th FFNN structure of Rucklidge Chaotic System. The comparison of their outputs for 100 input values is presented in Table 2.

Table 2 MSE, RMSE and NMSE values for 100 produced outputs between Rucklidge Chaotic System with the Euler algorithm and the FFNN network structure no. 9.

	MSE	RMSE	NMSE
X	$3.8965E-04$	$1.9740E-02$	$1.2065E-03$
Y	$2.4895E-04$	$1.5778E-02$	$1.1937E-03$
Z	$6.0393E-04$	$2.4575E-02$	$1.3986E-03$

CONCLUSION

In this study, the Rucklidge chaotic system, which has not been modeled using ANN before in the literature, has been solved by Euler numerical algorithm and network trainings with different architectures have been carried out with these solution values, making use of the proof of the rapid applicability of ANNs to the hardware implementation in the literature. In this context, network training results were compared by using different ANN structures and different activation functions in these structures, and ANN structures that could best model the Rucklidge Chaotic System were specified. The FFNN with 8x3, which get more satisfactory results in terms of MSE as $2.297x10^{-11}$ in Table 1 in a shorter time than the others, was preferred for error analysis. In error analysis, 100 output values generated by Rucklidge Chaotic System with the Euler algorithm and 9th FFNN structure have been used and MSE, RMSE and NMSE values have been obtained. MSE, RMSE and NMSE values for X are $3.8965E-04$, $1.9740E-02$ and $1.2065E-03$, respectively. MSE, RMSE and NMSE values for Y are 2.4895E-04, 1.5778E-02 and $1.1937E-03$, respectively. MSE, RMSE and NMSE values for Z are $6.0393E-04$, $2.4575E-02$ and $1.3986E-03$, respectively. In future studies, an application can be made about the hardware implementation of the ANN-based Rucklidge chaotic system.

Conflicts of interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Availability of data and material

Not applicable.

LITERATURE CITED

Alcin, M., İ. Koyuncu, M. Tuna, M. Varan, and İ. Pehlivan, 2019 A novel high speed artificial neural network–based chaotic true random number generator on field programmable gate array. International Journal of Circuit Theory and Applications **47**: 365–378.

Avaroğlu, E., T. Tuncer, A. B. Özer, B. Ergen, and M. Türk, 2015 A novel chaos-based post-processing for trng. Nonlinear Dynamics **81**: 189–199.

Azzaz, M. S., C. Tanougast, S. Sadoudi, R. Fellah, and A. Dandache, 2013 A new auto-switched chaotic system and its FPGA implementation **18**: 1792–1804.

Boyraz, O. F., E. Guleryuz, A. Akgul, M. Z. Yildiz, H. E. Kiran, *et al.*, 2022 A novel security and authentication method for infrared medical image with discrete time chaotic systems **267**: 169717.

Cavusoglu, U., 2014 Surekli zamanli otonom kaotik devre tasarimi ve sinyal gizleme uygulamasi. Gazi Universitesi Muhendislik Mimarlik Fakültesi Dergisi **29**.

Dong, C., L. Jia, Q. Jie, and H. Li, 2021 Symbolic encoding of periodic orbits and chaos in the rucklidge system **2021**: e4465151, Publisher: Hindawi.

Kiran, H. E., A. Akgul, and O. Yildiz, 2022 A new chaos-based lightweight encryption mechanism for microcomputers. In *2022 10th International Symposium on Digital Forensics and Security (ISDFS)*, pp. 1–5.

Koyuncu, I., M. Alcin, P. Erdogmus, and M. Tuna, 2020a Artificial neural network-based 4-d hyper-chaotic system on field programmable gate array **8**: 102–108, Number: 2.

Koyuncu, I., K. Rajagopal, M. Alcin, A. Karthikeyan, M. Tuna, *et al.*, 2021 Control, synchronization with linear quadratic regulator method and FFANN-based PRNG application on FPGA of a novel chaotic system **230**: 1915–1931.

Koyuncu, s., b. Şahin, C. Gloster, and N. K. Sarıtekin, 2017 A neuron library for rapid realization of artificial neural networks on FPGA: A case study of rössler chaotic system **26**: 1750015, Publisher: World Scientific Publishing Co.

Koyuncu, s., M. Tuna, h. Pehlivan, C. B. Fidan, and M. Alçın, 2020b Design, FPGA implementation and statistical analysis of chaos-ring based dual entropy core true random number generator **102**: 445–456.

Lee, S.-H., V. Kapila, M. Porfiri, and A. Panda, 2010 Master–slave synchronization of continuously and intermittently coupled sampled-data chaotic oscillators **15**: 4100–4113.

Liu, J., K. Rajagopal, T. Lei, S. Kaçar, B. Arıcıoğlu, *et al.*, 2020 A novel hypogenetic chaotic jerk system: Modeling, circuit implementation, and its application. Mathematical Problems in Engineering **2020**.

Prakash, P., K. Rajagopal, I. Koyuncu, J. P. Singh, M. Alcin, *et al.*, 2020 A novel simple 4-d hyperchaotic system with a saddle-point index-2 equilibrium point and multistability: design and fpga-based applications. Circuits, Systems, and Signal Processing **39**: 4259–4280.

Rajagopal, K., M. Tuna, A. Karthikeyan, İ. Koyuncu, P. Duraisamy, *et al.*, 2019 Dynamical analysis, sliding mode synchronization of a fractional-order memristor hopfield neural network with parameter uncertainties and its non-fractional-order fpga implementation. The European Physical Journal Special Topics **228**: 2065–2080.

Ramakrishnan, B., M. E. Cimen, A. Akgul, C. Li, K. Rajagopal, *et al.*, 2022 Chaotic oscillations in a fractional-order circuit with a josephson junction resonator and its synchronization using fuzzy sliding mode control. Mathematical Problems in Engineering **2022**.

Tuna, M., 2020 A novel secure chaos-based pseudo random number generator based on ANN-based chaotic and ring oscillator: design and its FPGA implementation **105**: 167–181.

Tuna, M., M. Alçın, İ. Koyuncu, C. B. Fidan, and İ. Pehlivan, 2019a High speed fpga-based chaotic oscillator design. Microprocessors and Microsystems **66**: 72–80.

Tuna, M., A. Karthikeyan, K. Rajagopal, M. Alcin, and s. Koyuncu, 2019b Hyperjerk multiscroll oscillators with megastability: Analysis, FPGA implementation and a novel ANN-ring-based true random number generator **112**: 152941.

Ullah, A., A. A. Shah, J. S. Khan, M. Sajjad, W. Boulila, *et al.*, 2022 An efficient lightweight image encryption scheme using multichaos **2022**: e5680357, Publisher: Hindawi.

Vaidyanathan, S., A. Akgul, S. Kaçar, and U. Çavuşoğlu, 2018 A new 4-d chaotic hyperjerk system, its synchronization, circuit design and applications in rng, image encryption and chaos-based steganography. The European Physical Journal Plus **133**: 1–18.

How to cite this article: Keles, Z., Sonugur, G., and Alcin, M. A The Modeling of the Rucklidge Chaotic System with Artificial Neural Networks. *Chaos Theory and Applications*, 5(2), 59-64, 2023.