

# Parameter Prediction for Lorenz Attractor by using Deep Neural Network

Nurnajmin Qasrina Ann<sup>1</sup>, Dwi Pebrianti<sup>1,4</sup>, Mohammad Fadhil Abas<sup>2</sup>, Luhur Bayuaji<sup>3,4</sup>, Mohammad Syafrullah<sup>4</sup>

<sup>1</sup>College of Engineering, Universiti Malaysia Pahang, Malaysia

<sup>2</sup>Faculty of Electrical and Electronics Engineering Technology, Universiti Malaysia Pahang, Malaysia

<sup>3</sup>Faculty of Computing, Universiti Malaysia Pahang, Malaysia

<sup>4</sup>Magister of Computer Science, Universitas Budi Luhur, Indonesia

---

## Article Info

### Article history:

Received May 1, 2020

Revised Sep 10, 2020

Accepted Sep 23, 2020

---

### Keywords:

Chaos System

Deep Learning

Artificial Neural Network

Lorenz Attractor

Prediction

---

## ABSTRACT

Deep learning models develop from the idea of artificial neural networks. This research presents Deep Neural Network to learn the database, which consists of high precision, a strange Lorenz attractor. Lorenz system is one of the simple chaotic systems, which is a nonlinear and characterized by an unstable dynamic behavior. The research aims to predict the parameter of a strange Lorenz attractor, either yes or no. The method implemented in this paper is the Deep Neural Network by using the Python Keras library. For the neural network, the different number of hidden layers employ to compare the accuracy of the system prediction. As a result, the accuracy of the testing result shows that 100% correct prediction when using the testing dataset. Meanwhile, new random data obtained only a 60% correct prediction.

Copyright © 2020 Institute of Advanced Engineering and Science.  
All rights reserved.

---

### Corresponding Author:

Nurnajmin Qasrina Ann,  
College of Engineering,  
Universiti Malaysia Pahang,  
26300, Gambang, Pahang, Malaysia.  
Email: [qasrinaann@gmail.com](mailto:qasrinaann@gmail.com)

---

## 1. INTRODUCTION

Like most scientific theories, the history of chaos as perceived by a broad audience has its leading contributors: Henry Poincare (1854-1912) and Edward Lorenz (1913-2008) identified in common with such a status. Many scientists contributed to this field like Boris Belousov (1893-1970) and Anatol Zhabotinskii (1938-2008). Nevertheless, Poincare's seminal works consider the foundation of chaos theory [1]. Poincare explored the restricted three-body system. A body is too small compared with the two remaining ones that can omit. He also realized that the solution to that easy problem is so complicated and cannot calculate easily. Shortly afterward, Lorenz discovered the "butterfly effect" when analyzing the forecasting problem. Then, he is known as the father of chaos. However, the official employment of chaos is from Li and Yorke [2] in 1975. Since then, chaos theory started to recognize and accepted by people. It is also dramatically promoted in the research world.

Chaos is characterized by an unstable dynamic behavior which, as an essential part of nonlinear science, exhibits sensitive dependence on initial conditions and includes infinite unstable periodic motions. There are possible applications in chemical reaction areas, power converters, biological systems, information processing, stable communication, and economics. Researchers have gained interest in the management and synchronization of unstable processes over the last two decades [3]–[5]. However, due to the complexities of chaotic processes, current approaches may be hard to decide any parameters in advance. So, the parameter estimation of chaotic systems is essential.

Parameter estimation in atmospheric sciences refers to the determination by data assimilation or other similar techniques of the best values of particular parameters in a numerical model [6]. In controller design and device recognition, the parameter estimation issue of dynamical systems is an important task. There are mathematical and analytical models to explain the behavior of the system within a parameter

estimation problem. The challenges are to estimate the unknown parameters of the models based on measured data from the system [7]. Thus, parameter estimation is a meaningful task in the analysis of chaotic systems.

The cornerstones of the next computer revolution are Artificial Intelligence and machine learning. Such technologies rely on the ability to identify trends and then forecast future results based on data found in the past. A subset of machine learning is deep learning. It is an area focused on learning by analyzing machine algorithms and developing on its own. Deep learning operates with artificial neural networks, which are programmed to mimic how people think and learn, while machine learning uses basic concepts. Until recently, neural networks constrained by computing power and complexity.

Due to its complex instability[8], the chaotic system is not easy to estimate. The downside of the conventional strategy is that it easily traps into local minima. A new approach is required to address the problem of being stuck inside the local minima and the complex volatility of chaotic systems. Deep learning can be the answer to the problem. Deep learning methods are representation-learning methods with multiple levels of representation, received by constructing simple. However, the nonlinear modules that each convert representation at one level into a representation at a higher. In high dimensional data, it has turned out to be very good at discovering complex structures. This model's implementation involves research, business, and governance.

Based on the literature, many methods introduce to solve the parameter estimation for chaotic systems, especially for the Lorenz system. From that, a few researchers implement optimization method to solve the parameter estimation such as Particles Swarm Optimization (PSO) [9], [10], Biogeography Based Optimization (BBO) [11], Hybrid quantum-inspired evolutionary algorithm [12], Differential Evolution Algorithm [13], [14], Firefly Algorithm [15], Ant Colony Optimization (ACO) [16] and Cuckoo Search Algorithm (CSA) [8]. Other known approaches, namely the Kalman filter [17] and Synchronization-based approach [18], are also implemented for the parameter estimation.

Over the last ten years, the deployment of optimization algorithm estimates the parameter for chaotic systems. There is a group of researchers from Tsinghua University, China, that implemented many optimizations approaches to solve the problem. According to those journals [11], [14], the researchers claimed the population size impacts the searching efficiency of the solution. The higher the population size assigned, the performances of the algorithm are better, but there is always a threshold to get the optimum result [14]. Wang and Xu introduced a capable hybrid biogeography-based optimization algorithm for parameter estimation of chaotic systems. The method showed to be very useful, efficient, and robust because the optimization algorithm used is undergoing a hybridization process with a differential mutation operator [11]. Besides, Tang and Guan explained parameter estimation for chaotic systems by using the differential evolution approach. The main contribution of this paper is chaotic systems successfully estimated by using the proposed method, and the time-delay of the chaotic system is further estimated [13].

Also, there are related researches about deep learning and chaotic systems, other than the parameter estimation problem. One of the current research is that deep learning can provide a potent tool in data-driven modeling of complex dynamical systems. The technique helped make the right prediction of the nonlinear dynamics by effectively filtering out the noise [19]. Secondly, a group of researcher expects deep learning can work better than machine learning, that may be inherently well suited for tasks which are themselves dynamical in character [20]. Besides, deep hybrid learning, Deep Reinforcement Learning (DRL) successfully controlling the chaotic dynamics governed by well-known nonlinear Kuramoto-Sivashinky equations [21]. Other than that, in a study, authors proposed the standard Deep Neural Networks (DNNs) to classify univariate time series generated by the discrete and continuous dynamical system based on their chaotic and non-chaotic behavior [22]. Also, the Artificial Neural Network (ANN) is useful in memorizing short and long-term temporal information simultaneously [23]. Lastly, Recurrent Neural Network (RNN) is used to handling long and short-term dependencies for simple fluid mechanics problems. Then, the author could further predict a simple flow using another method [24].

In this study, a deep learning model employ to solve parameter estimation for chaotic systems. It is because chaotic systems are an intricate system and complex to solve. The ability of deep learning is no doubt as it can solve high-dimensional data problems such as image and speech recognition. Therefore, this research presents Deep Neural Network to learn the dataset, which consists of high precision, a strange Lorenz attractor.

The rest of this paper is structured as follows: Section 2 explains about Lorenz System and introduces Deep Neural Network (DNN) as the methodology of the research. Then, the experimental setup, dataset, result, and analysis discussed in Section 3. Finally, Section 4 concludes the paper, including future recommendations.

## 2. RESEARCH METHOD

In this section, there are two subtopics to explain the research method to solve the parameter prediction problem for the Lorenz system. Firstly is the explanation of the Lorenz system itself. Then, the second part is about Deep Neural Network by using Keras Phyton Library.

### 2.1. Lorenz System

Lorenz system introduces by Edward Lorenz, MIT meteorologist in 1963 [25]. He formulated a highly simplified model of a convection fluid. This straightforward model also shows a broad variant of behavior, and for some parameter values, it is chaotic. The system is given by

$$\begin{aligned}\dot{x} &= \sigma(y - x), \\ \dot{y} &= rx - y - xz, \\ \dot{z} &= xy - bz,\end{aligned}\tag{1}$$

where  $x$  measures the rate of convective overturning,  $y$  measures the horizontal temperature variation,  $z$  measures the vertical temperature variation,  $\sigma$  is the Prandtl number,  $r$  is the Rayleigh number, and  $b$  is a scaling factor. The Prandtl number is related to the fluid viscosity, and the Rayleigh number is related to the temperature difference between the top and bottom of the column.

One of the systems studied by Lorenz is when  $\sigma = 10$ ,  $b = \frac{8}{3}$  and  $r = 28$ . It is known as a strange attractor for the Lorenz system, as shown in Figure 1. The strange attractor has the properties where the trajectory is aperiodic (not periodic), and the trajectory remains on the attractor forever (invariant).

The dataset that had been using for training is from Professor Barrio [26]. This paper is presented a high-precision and validated database of periodic orbits useful to the scientific community. The database consists of hundreds of approximated initial conditions of the periodic orbits of the Lorenz attractor with multiplicities between 2 to 10 for the parameter  $x, y, z$  from Equation (1). This database is a “computational challenge,” and it can use as a benchmark for analyzing state-of-the-art numerical and theoretical approaches in calculation physics and dynamics.

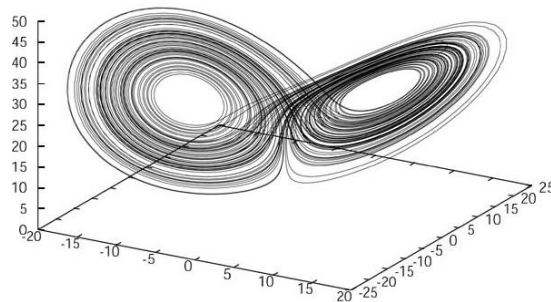


Figure 1. A strange attractor for the Lorenz system

### 2.2 Deep Neural Network

Deep learning represents the subset of Artificial Intelligence (AI). The overview of the process inspired by human brains connected networks of neurons. It consists of three essential layers; input layer, hidden layer, and an output layer, which is the purest form of ANN. When ANN consists of more than three layers, it is called a Deep Neural Network (DNN). The example of DNN, as shown in Figure 2.

In this research, Keras Phyton Libraries employ to build the multi-layer perceptron network models. The Keras Phyton library for deep learning focuses on the creation of models as a sequence of layers.

There are many activation functions that Keras supports, such as *relu*, *softmax*, *rectifier*, *tanh*, and *sigmoid*, that need to be specified for each layer. In this experiment, *relu* and *sigmoid* deploy in the algorithm. The last layer is using *sigmoid*, and another layer is using *relu* (Rectified Linear Unit).

For layer types, there are many core Layer types for the standard neural network. The fully-connected layer had been used in this experiment because ANN needs to be all connected (input, multiples hidden layers, and output layers).

After the model is completely defined, the model needs to compile with three essential attributes. There is a model optimizer, loss function, and metrics. It is the approach used to update weights in the model. In the study, there are some popular gradient descent optimizers, such as Stochastic Gradient Descent (SGD), RMSprop, and Adam. For example, Adam is an adaptive moment estimation that is most popular for adaptive learning rates. The second attribute is model loss functions. The loss function, also called the objective function, calculates the model used by the loss argument. In this research, binary cross-entropy implement, and it is suitable for binary logarithmic loss. Lastly, the accuracy is the model metrics to be measured during the training process.

Then, the model will undergo the training process. For training, the number of epochs and batch size needs to be specified. Epochs are the number of times that the model solves by the training dataset. The batch size is the number of training details shown to the model before a weight update. For this research, both epoch and batch size set to 1000 and 10, respectively.

Lastly, once the model already trains extensively, the model prediction is ready to test. The purpose of the testing dataset is to ensure that the model is right or the need improvisation. The block diagram of network architecture refers to Figure 3. The network starts with the Lorenz dataset as input to the system. Then, the data fed on to the multiple hidden layers and, lastly, expected output. The multiple hidden layers consist of three blocks: the first hidden layer,  $n$  hidden layer, and the last hidden layer.

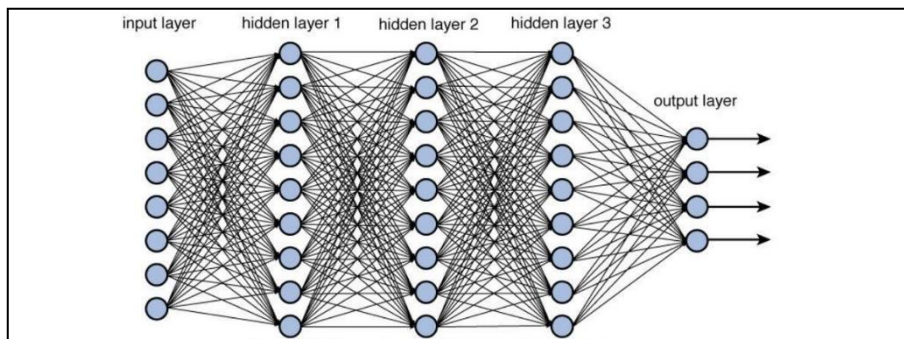


Figure 2. Example of Deep Neural Network

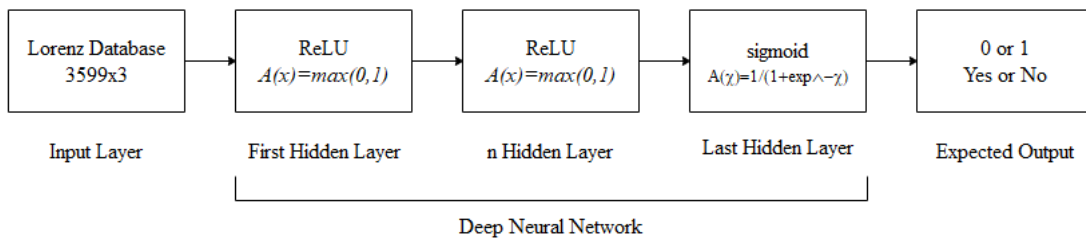


Figure 3. Network Architecture

### 3. RESULTS AND DISCUSSION

In this section, to evaluate the performance of the parameter prediction of Lorenz attractor, the authors need to conduct several experiments. The result is from the developed Deep Neural Network (DNN) architecture with many hidden layers. The result obtained is more likely to explain the parameter prediction for the Lorenz system.

#### 3.1. Dataset and Experimental Setup

The Lorenz system dataset [26] and the setup of the environment describe below: CPU: Intel® Core™ i7-8700 Processor (3.20GHz), RAM: 16GB, GPU: NVIDIA GeForce GTX 1060 3GB. As shown in Table 1, the dataset is 3-dimensional,  $x$ ,  $y$ , and  $z$ , and the last column is the expected output of the system.

Table 1. First 15 Training Dataset

x	y	z	Expected Output
-6.35008	-4.53289	27	1
-6.17844	-4.56347	26.57044	0
-6.0272	-4.61769	26.14751	1
-5.89658	-4.69392	25.73316	0
-5.78666	-4.7907	25.3292	1
-5.69735	-4.90682	24.93724	0
-5.62848	-5.04125	24.55883	1
-5.57981	-5.19312	24.19541	0
-5.55104	-5.36176	23.84841	1
-5.54184	-5.54664	23.51923	1
-5.55189	-5.74734	23.2093	0
-5.58082	-5.96355	22.92006	1
-5.62832	-6.195	22.65304	0
-5.69405	-6.44146	22.40979	1
-5.7777	-6.7027	22.19197	1

Furthermore, the dataset is the set of unstable periodic orbits. This dataset has several advantages. Firstly, it is clear how to use these data as a test of accuracy and try to follow one or several periodic orbits. Next, during the construction of the benchmarks, the authors have reconfirmed some results on the proposed model, the Lorenz system. The dataset divide into two parts, which are 80 percent assign as a training dataset, and the remainder is the testing dataset.

### 3.2. Result and Discussion

In this section, the experimental results will explain and discuss. All the experiment runs with a different number of hidden layers. Those experiments aim to find whether the number of the hidden layer will affect the accuracy of DNN. In total, five different sets of the experiment run dependently, as shown in Table 2. The very first column is the list of the experiment, which is five in total. Next, the second column shows the multiple hidden layers in every experiment. The third column presents the number of neurons for each layer and, lastly, the number of parameters used for every neuron.

Table 2. Multiple Hidden Layers for Five Different Experiments

No. of Experiment	Multiple Hidden Layers	No. of neurons	Parameters
#1	Layer 1	16	64
	Layer 2	2	34
#2	Layer 1	16	64
	Layer 2	8	136
	Layer 3	2	18
#3	Layer 1	16	64
	Layer 2	16	272
	Layer 3	8	136
	Layer 4	2	18
#4	Layer 1	16	64
	Layer 2	16	272
	Layer 3	16	272
	Layer 4	8	136
	Layer 5	2	18
#5	Layer 1	16	64
	Layer 2	16	272
	Layer 3	16	272
	Layer 4	16	272
	Layer 5	8	136
	Layer 6	2	18

The first analysis is about the value of accuracy and loss for every epoch record during the training process. Figure 4 shows the loss and accuracy during training for every experiment. All the graphs represent the performance index for the different multiple hidden layers.

In particular, from experiment 2 and experiment 4, which are 3 and 5 hidden layers—the performance index graphs are illustrated in Figure 4(b) and Figure 4(d). From the graphs, there is an accuracy

and loss curve that represents the performance of the system. Hence, the lower the loss value and the higher the accuracy value, the system is in an excellent performance. To sum up, from Figure 4(b) and Figure 4(d), an increase in the hidden layer, the decrease in the loss value, and the higher the accuracy value.

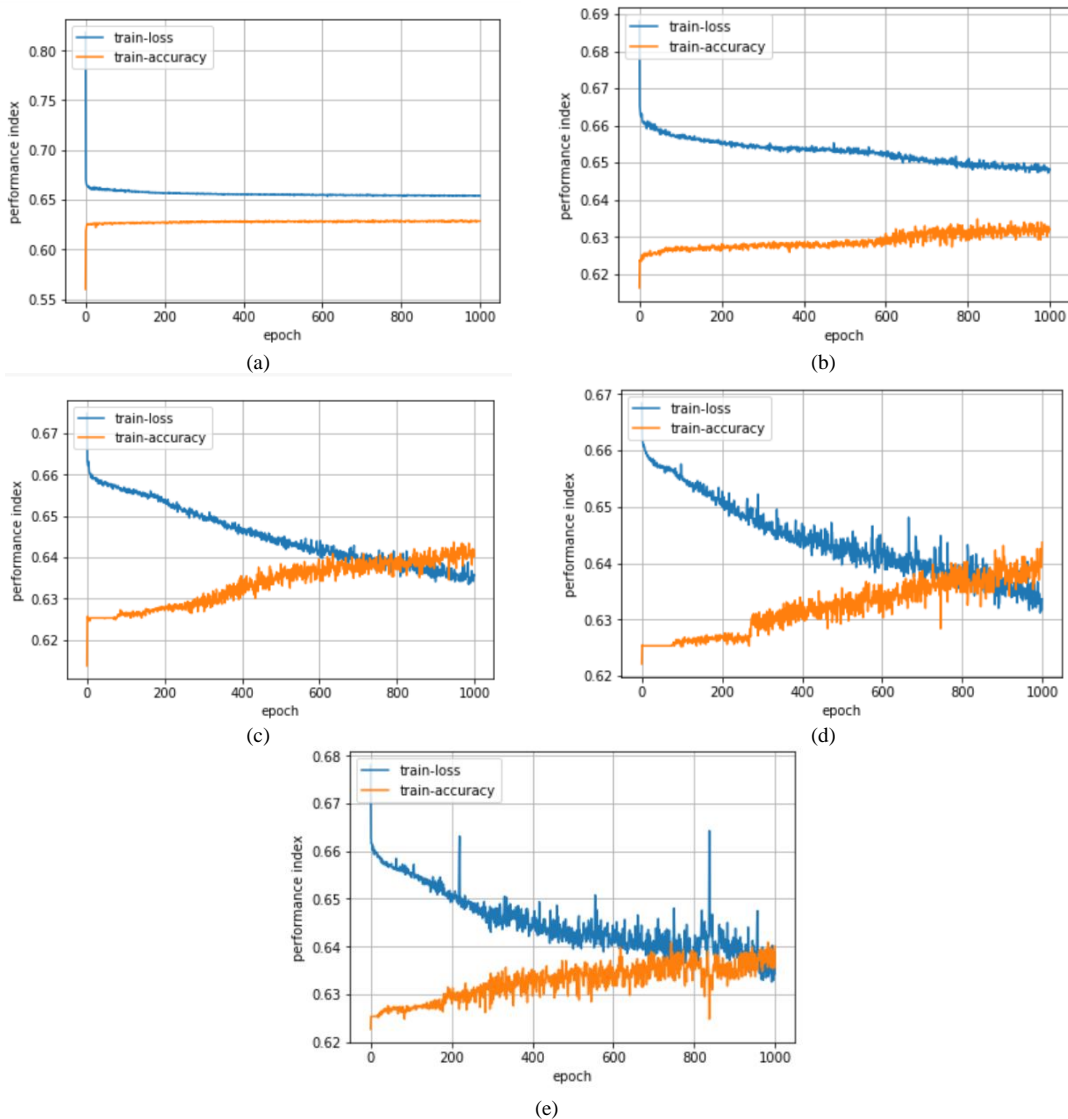


Figure 4. Performance index for (a) Experiment #1 (b) Experiment #2 (c) Experiment #3 (d) Experiment #4 (e) Experiment #5

Moreover,

Table 3 is the summary of the accuracy and loss value for each experiment. The highest accuracy is 64.3% from experiment number 4 compared with others. The minimum percentage of training loss is 63.25% in experiment number 3 and 4. Therefore, experiment 4 is the optimum model made up of five hidden layers of DNN, as explained earlier. For experiment 5, the percentage of accuracy decrease, and the percentage of loss increase. The earlier hypotheses cannot be applied for experiment 5 because six hidden layers are too much for this system. Thus, the system may result in overfitting.

Table 3. Summary of the value of accuracy and loss of each experiment

No. of Experiment	Percentage of Accuracy (Maximum)	Percentage of Loss (Minimum)
#1	64.00%	65.30%
#2	63.25%	64.80%
#3	64.25%	63.25%
#4	64.30%	63.25%
#5	64.00%	63.50%

After the model undergoes training, the testing process performs. The random data and the testing dataset deploy to measure the robustness of the model. All the results for accuracy of parameter prediction summarize in Table 4. In particular, the testing dataset shows very high accuracy. This result is because all the data has a similar pattern and trend with the training dataset. For the new data, random numbers use. It shows that the accuracy is lower, which is 60% for experiments 2, 3, and 4 compared with the prediction accuracy by using a testing dataset, experiment 4 shows 100% accuracy. The random data is an outlier of the data from the training process and exceeds the training in searching space.

Table 4. Testing Result for All Experiments

No. of Experiment	Prediction Accuracy (%)	
	Train Data	New Data
#1	80	50
#2	80	60
#3	90	60
#4	100	60
#5	90	50

Base on the performance result, it can be concluded that the developed DNN architecture is not yet suitable for estimation purposes. For future works, hybrid Deep Learning architecture will develop to solve the estimation problem by increasing the accuracy and efficiency of the result. Also, the stability of the chaotic system will consider in the next study. The findings of the study could be the fundamental of this research field.

#### 4. CONCLUSION

In conclusion, this paper presents the deep learning in Artificial Neural Networks called Deep Neural Network to train an enormous database consists of high-precision Lorenz system parameters. The developed model compares with the varied number of hidden layers and their index performance analyses. The accuracy of the testing result shows that 100% correct prediction obtain when using the testing dataset while only 60% correct prediction for the ten new random data.

For recommendations, the extensive work will be on training the dataset by semi-supervised learning instead of only supervised deep learning to increase the ability of the model to estimate the given parameters of a system [27]. Another work will be implementing the optimization algorithm to optimize the hyperparameters involved in deep learning [28].

#### ACKNOWLEDGMENTS

The work presented in the paper has been supported by Universiti Malaysia Pahang Research Grant RDU 170378 and RDU 1703142.

#### REFERENCES

- [1] C. Letellier and V. Messaeger, "Influences on Otto E. Rössler'S Earliest Paper on Chaos," *Int. J. Bifurc. Chaos*, vol. 20, no. 11, pp. 3585–3616, 2010.
- [2] T.-Y. Li and J. A. Yorke, "Period Three Implies Chaos," *Am. Math. Mon.*, vol. 82, no. 10, p. 985, Dec. 1975.
- [3] Y. Tang, M. Cui, L. Li, H. Peng, and X. Guan, "Parameter identification of time-delay chaotic system using chaotic ant swarm," *Chaos, Solitons and Fractals*, vol. 41, no. 4, pp. 2097–2102, 2009.
- [4] F. Sun, H. Peng, Q. Luo, L. Li, and Y. Yang, "Parameter identification and projective synchronization between different chaotic systems," *Chaos*, vol. 19, no. 2, 2009.
- [5] B. Rakshit, A. R. Chowdhury, and P. Saha, "Parameter estimation of a delay dynamical system using synchronization in presence of noise," *Chaos, Solitons and Fractals*, vol. 32, no. 4, pp. 1278–1284, 2007.
- [6] A. Aksoy, "NUMERICAL MODELS | Parameter Estimation," *Encycl. Atmos. Sci.*, pp. 181–186, Jan. 2015.
- [7] H. Xu, F. Ding, and J. Sheng, "On some parameter estimation algorithms for the nonlinear exponential autoregressive model," *Int. J. Adapt. Control Signal Process.*, vol. 33, no. 6, pp. 999–1015, Jun. 2019.
- [8] X. T. Li and M. H. Yin, "Parameter estimation for chaotic systems using the cuckoo search algorithm with an orthogonal learning method," *Chinese Phys. B*, vol. 21, no. 5, 2012.
- [9] J. Sun, J. Zhao, X. Wu, W. Fang, Y. Cai, and W. Xu, "Parameter estimation for chaotic systems with a Drift Particle Swarm Optimization method," *Phys. Lett. Sect. A Gen. At. Solid State Phys.*, vol. 374, no. 28, pp. 2816–2822, 2010.
- [10] Q. He, L. Wang, and B. Liu, "Parameter estimation for chaotic systems by particle swarm optimization," *Chaos, Solitons and Fractals*, vol. 34, no. 2, pp. 654–661, Oct. 2007.
- [11] L. Wang and Y. Xu, "An effective hybrid biogeography-based optimization algorithm for parameter estimation

- of chaotic systems,” *Expert Syst. Appl.*, vol. 38, no. 12, pp. 15103–15109, 2011.
- [12] L. Wang and L. po Li, “An effective hybrid quantum-inspired evolutionary algorithm for parameter estimation of chaotic systems,” *Expert Syst. Appl.*, vol. 37, no. 2, pp. 1279–1285, 2010.
- [13] L. W. Bo Peng, Bo Liu, Fu-Yi Zhang, “Differential evolution algorithm-based parameter estimation for chaotic systems,” *Chaos, Solitons and Fractals*, vol. 39, pp. 2110–2118, 2009.
- [14] Y. Tang and X. Guan, “Parameter estimation of chaotic system with time-delay: A differential evolution approach,” *Chaos, Solitons and Fractals*, vol. 42, no. 5, pp. 3132–3139, 2009.
- [15] Y. Mousavi and A. Alfi, “Fractional calculus-based firefly algorithm applied to parameter estimation of chaotic systems,” *Chaos, Solitons and Fractals*, vol. 114, pp. 202–215, 2018.
- [16] J. A. Lazzús, M. Rivera, and C. H. López-Caraballo, “Parameter estimation of Lorenz chaotic system using a hybrid swarm intelligence algorithm,” *Phys. Lett. Sect. A Gen. At. Solid State Phys.*, vol. 380, no. 11–12, pp. 1164–1171, 2016.
- [17] J. D. Annan and J. C. Hargreaves, “Efficient parameter estimation for a highly chaotic system,” *Tellus, Ser. A Dyn. Meteorol. Oceanogr.*, vol. 56, no. 5, pp. 520–526, 2004.
- [18] R. Konnur, “Synchronization-based approach for estimating all model parameters of chaotic systems,” *Phys. Rev. E - Stat. Physics, Plasmas, Fluids, Relat. Interdiscip. Top.*, vol. 67, no. 2, p. 4, 2003.
- [19] K. Yeo, “Model-free prediction of noisy chaotic time series by deep learning,” no. 10, pp. 1–5, 2017.
- [20] J. Pathak, B. Hunt, M. Girvan, Z. Lu, and E. Ott, “Model-Free Prediction of Large Spatiotemporally Chaotic Systems from Data: A Reservoir Computing Approach,” *Phys. Rev. Lett.*, vol. 120, no. 2, p. 24102, 2018.
- [21] M. A. Bucci, O. Semeraro, A. Allauzen, G. Wisniewski, L. Cordier, and L. Mathelin, “Control of chaotic systems by deep reinforcement learning,” *Proc. R. Soc. A Math. Phys. Eng. Sci.*, vol. 475, no. 2231, p. 20190351, 2019.
- [22] N. Bouille, V. Dallas, Y. Nakatsukasa, and D. Samaddar, “Classification of chaotic time series with deep learning,” *Phys. D Nonlinear Phenom.*, vol. Volume 403, no. 132261, 2020.
- [23] S. Lahmiri and S. Bekiros, “Cryptocurrency forecasting with deep learning chaotic neural networks,” *Chaos, Solitons and Fractals*, vol. 118, pp. 35–40, 2019.
- [24] P. Dubois, T. Gomez, L. Planckaert, and L. Perret, “Data-driven predictions of the Lorenz system,” *Phys. D Nonlinear Phenom.*, vol. 408, p. 132495, 2020.
- [25] S. Lynch, “Three-Dimensional Autonomous Systems and Chaos,” in *Dynamical Systems with Applications using MATLAB®*, Cham: Springer International Publishing, 2014, pp. 283–309.
- [26] R. Barrio, A. Dena, and W. Tucker, “A database of rigorous and high-precision periodic orbits of the Lorenz model,” *Comput. Phys. Commun.*, vol. 194, pp. 76–83, 2015.
- [27] Gao Huang, Shiji Song, J. N. D. Gupta, and Cheng Wu, “Semi-Supervised and Unsupervised Extreme Learning Machines,” *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2405–2417, Dec. 2014.
- [28] I. Loshchilov and F. Hutter, “CMA-ES for Hyperparameter Optimization of Deep Neural Networks,” Apr. 2016.

## BIOGRAPHY OF AUTHORS



Nurnajmin Qasrina Ann was born in Kuantan, Pahang, Malaysia. Currently, she is studying in Ph.D. in the Department of Electrical and Electronics Engineering, College of Engineering, Universiti Malaysia Pahang (UMP). She is interested in many research fields, such as image processing, optimization, deep learning, and chaos theory.



Dwi Pebrianti received her Ph.D. (Engineering) in Artificial System Science from Chiba University, Japan. Currently, she is a Senior Lecturer in the College of Engineering, Universiti Malaysia Pahang (UMP), major in Robotic and Unmanned System.





Mohammad Fadhil bin Abas received his Ph.D. from Chiba University, Japan. Currently, he is a Senior Lecturer in the Faculty of Electrical and Electronics Engineering, Universiti Malaysia Pahang (UMP).



Luhur Bayuaji had received his Doctor of Philosophy in Remote Sensing and Earth Observation from Chiba University, Japan. His field of interest is including computer engineering, computer network, satellite data image processing, remote sensing for environmental monitoring using combined optical and synthetic aperture radar satellite data, Geographic Information System (GIS), Image Processing, Neural Network, and Vision. Currently, he is a Senior Lecturer in the Faculty of Computing, Universiti Malaysia Pahang (UMP).



Mohammad Syafrullah had received his Ph.D. from Universiti Teknologi Malaysia, Malaysia. Currently, he is a Lecturer in the Magister of Computer Science, Universitas Budi Luhur, Indonesia.