Criminal Cases Forecasting Model using A New Intelligent Hybrid Artificial Neural Network with Cuckoo Search Algorithm

Wullapa Wongsinlatam, and Suntaree Buchitchon

Abstract—Criminal cases are social problems that concern the public order and good morals of the citizens as a whole. Research in criminology has recently focused on finding the root causes of criminal cases in order to identify the factors accelerating crimes and to find preventive methods. Thus, the study on the factors relating to the occurrence of crimes would benefit policy makers in dealing with crimes. This research focuses on a forecasting model of the criminal related factors in the Upper Northeastern Provinces of Thailand by using a New Intelligent Hybrid Artificial Neural Network (NIHANN) with a Cuckoo Search Algorithm (CS). The new CS-NIHANN algorithm, which is a combination of the CS with the NIHANN, is used in this research. In addition, the research selected relevant parameters that related to criminal cases in the Upper Northeastern Provinces of Thailand. The comparison of performances of the forecasting model for historical forecast part was calculated by Mean Absolute Percentage Error (MAPE), Root Mean Squared Error (RMSE) and Correlation Coefficient (R). The result shows that the CS-NIHANN algorithm trains fast, can obtain the global optimal solution, and has a good generalization performance for Multiple Input Single Output (MISO) and Multiple Input Multiple Output (MIMO). Furthermore, the result shows that the CS-NIHANN algorithm has a faster convergence than the CS-ANN algorithm in calculating an optimal solution to the criminal cases.

Index Terms—Forecasting model, Artificial neural network, Metaheuristic optimizations, Applied mathematics, Optimization problem, Criminology.

I. Introduction

CRIMINAL cases are social problems that concern the public order and good morals of the citizens as a whole. Research in criminology has recently focused on the finding of root causes of criminal cases in order to identify the factors accelerating crimes and to find preventive methods. While in the past, government used law sanctions as a mean to prevent crimes in the states. However, it cannot completely demolish criminal cases. Thus, the study of the factors relating to the occurrence of crimes would benefit the policy makers in dealing with crimes.

Time series forecasting models can be used as a great tool for policy planning to deal with criminal cases. There are various forecasting methods that are being used, such as, Exponential smoothing method [1,2], Box-Jenkins method [3,4], Neuro-fuzzy method [5-7] and Artificial neural network method [8-11]. Various studies have shown that, with the ability to learn the pattern of data from the data's past collection, artificial neural network method is more efficient in forecasting results with less errors. Thus, artificial neural network is being used to forecast data in various areas such as in medical matters [12,13], in economics and management [14,15], in agriculture [16,17], and in engineering [18,19].

This research focuses on a forecasting model of the crime related factors in the Upper Northeastern Provinces in Thailand by using artificial neural network (ANN). The purpose of an artificial neural network (ANN) is to create a mathematical model reflecting the system of brain functions that is a collection of neurons with connecting synapses. The collection is organized into the input layer, the hidden layer and the output layer [20,21]. ANN is used in various kinds of problems such as forecasting and classification. In time series applications, ANN has been used in predicting carbon dioxide emissions, power load forecasting, as well as Dow-Jones industrial averages. Still, there are three main limits to ANN: slow training, easy to fall into local minima and very sensitive regarding the choice of the initial weights and bias [22,23]. Recently, back propagation algorithm in neural network (BP) is more understandable and easily programmed. The BP is structured into two main parts: forward pass and backward pass. However, the BP contains big algorithm and requires big data for the training process which takes a long time [24,25]. The main problems of the BP are local minima problems and its slow speed in convergence problem, which lessen the ability to fall into global minimum [26-28]. As a result, there are many works of research focusing on finding a better performed algorithm that has global optimal solution, good generalization performance, as well as better training in order to accelerate training speed. One of the best performed algorithms in improving the BP is Gradient Descent Algorithm (GD) and Levenberg Marquardt Algorithm (LM) [29]. These algorithms used the choice of optimization methods as the activation function to apply to network. Still, it can only solve some part of the problems.

Therefore, a new algorithm that has been proposed to solve optimization problems is metaheuristic optimizations [30,31]. Metaheuristic optimizations are the most popular algorithms that are used to solve local

Manuscript received Dec 10, 2019; revised June 17, 2020. This work was supported in part by National Research Council of Thailand in 2018 (6100118) and the Research Administration Division, Khon Kaen University.

W. Wongsinlatam is with the Faculty of Applied Science and Engineering, Khon Kaen University, Nong Khai Campus, Nong Khai, 43000, THAILAND, e-mail: wullwon@kku.ac.th.

S. Buchitchon is with the Faculty of Integrated Social Sciences, Khon Kaen University, Nong Khai Campus, Nong Khai, 43000, THAILAND, e-mail: suntareebu@kku.ac.th.

minima problems and slow convergence problems for the best performance of network by combining the ANN with metaheuristic optimization [32-38]. Furthermore, a cuckoo search algorithm (CS) was developed in recent years by Yang and Deb as a new metaheuristic optimization [39-44]. In the testing of the CS-ANN algorithm performance in predicting OPEC carbon dioxide emissions from petroleum consumption using neural network and hybrid cuckoo search algorithm, the result shown that the CS-ANN algorithm gave a better result than the PSO-ANN and GA-ANN algorithms [45]. Furthermore, there are several researches that apply neural network with metaheuristic optimizations such as research in radial basis function neural network with particle swarm optimization algorithms for regional logistics demand prediction [46] and research in maximal market potential of feeder bus route design using the particle swarm optimization particle [47]. In addition, a research in the application of GA-BP neural network in the land ecological security evaluation, the genetic algorithm (GA) is introduced to improve the BP neural network, with advantages in solving the problems of slow convergence and getting into local minimum easily when the BP neural network was applied alone in land ecological security evaluation [48].

In this paper, the focus is on developing a new algorithm to address the limits of current models. The research starts with the process of combining the NIHANN algorithm with the CS algorithm to address the NIHANN algorithm limits. The experiment is divided into three parts: first, determining the suitable NIHANN structure to be used in this research, second, CS algorithm is used and developed to obtain the best weight and bias, then, the result is used to develop the criminal cases forecasting model. This research is presented as follows. The part 2 explains preliminary and explains the proposed algorithms which are the NIHANN and CS algorithms. Then the research presents the new algorithm CS-NIHANN and research design in part 3. In part 4, results and discussions on the criminal cases forecasting model of the Upper Northeastern Provinces in Thailand are shown to compare the performances. Finally, the fifth part shows conclusions and presents the points for future work orientation.

II. Preliminary

A. Forecasting model

There are two main types of forecasting methods; qualitative forecasting and quantitative forecasting. Qualitative forecasting is being more subjective than quantitative forecasting, the result is interpreted by researchers based on their experiences. It is normally being used in Subjective Assessment Methods, Salesforce Estimate, Jury of Executive Opinion, Test Market, Exploratory, Scenario Analysis, Delphi, Cross-Impact Analysis, and Analogy. On the other hand, quantitative forecasting technique is being used as future projection by consistently gathering information either daily, weekly, monthly or annually. The examples of quantitative forecasting are Moving Average and Smoothing Technique. In this research, the process of the NIHANN is combined with the CS algorithm to present the new algorithm CS-NIHANN, which is found to conform to the time series of the criminal cases forecasting model for the Upper Northeastern Provinces in Thailand.

B. Artificial Neural Network (ANN)

One of the most widely used forecasting models is an ANN. The ANN algorithm is a common method for training neural network. The topology of the ANN algorithm has input layer, hidden layer and output layer, as shown in Fig. 1. The two parts have the forward and the back transmission and the back transmission of network for the learning and training processes. The purpose of the ANN algorithm is to optimized the weights and the global error function can be defined as:

$$E = [\frac{1}{2}\sum_j (d_j - y_j^P)]^2, \tag{1}$$

where E is global error function, d_j is desired output of neuron j, P is number of layers, and $y_j^P = f(b_j^P)$ is actual output.

The basic equation of the ANN algorithm for the output layer is:

$$\begin{aligned} w_{ji}^P(t+1) &= w_{ji}^P(t) + \alpha \cdot \frac{\partial(f(b_j^P))}{\partial(b_j^P)} \cdot (d_j - y_j^P) \cdot x_i^{(P-1)}, \\ &= w_{ji}^P(t) + \alpha \cdot \delta_j^P \cdot x_i^{(P-1)}, \end{aligned} \tag{2}$$

where δ_j^P is clarified to be

$$\delta_j^P = \frac{\partial(f(b_j^P))}{\partial(b_j^P)}, \tag{3}$$

the weight equation is used to update for hidden layer,

$$w_{ji}^l(t+1) = w_{ji}^l(t) + \alpha \cdot \delta_j^l \cdot x_i^{(l-1)}, \tag{4}$$

where

$$\delta_j^l = \frac{\partial(f(b_j^l))}{\partial(b_j^l)} \cdot \sum_k \delta_k^{l+1} \cdot w_{kj}^{l-1}. \tag{5}$$

Then the weight for the hidden nodes are carried out. The ANN equations provide a way of computing the gradient for the error function. the x_i^l is output of neuron i in the l^{th} layer, y_j^l is actual output of neuron j in the l^{th} layer, w_{ji}^l is weight update value, and α is learning rate.

The concrete steps of the ANN algorithm are the forward part of the input signal of network and the back part of the error signal. The steps start with presenting the inputs at the input layer with a set of the corresponding activation function for the input layer, then feedforward, follows with output error and compute the vector, the error, and output for the gradient of the global error function, respectively [28].

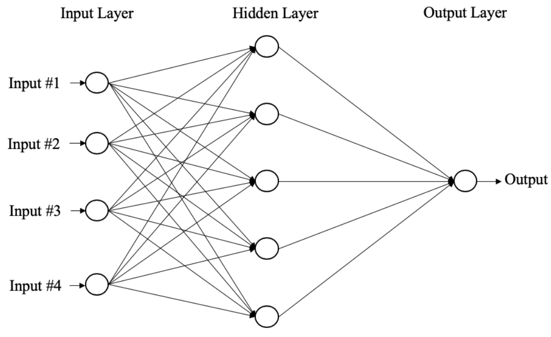

Fig. 1. An Artificial Neural Network Structure

TABLE I
The steps of the CS algorithm

Algorithm of CS
Objective function $f(x)$, $x = (x_1, ..., x_t)^T$;
Initial a population of t host nests x_i, for all $i = (1, 2, ..., t)$;
while (t < MaxGeneration) or (stop criterion);
Get a cuckoo (say i) randomly by Lévy flights;
Evaluate its quality/fitness E_i;
Choose a nest among t (say j) randomly;
if $(E_i > E_j)$
Replace j by the new solution;
end
Abandon a fraction (P_a) of worse nests;
build new ones at new locations via Lévy flights;
Keep the best solutions;
Ranking the solutions and finding the current;
end while

C. Cuckoo search algorithm (CS)

The CS algorithm is proposed by Yang and Deb for the optimization problem [39]. The CS algorithm is a global search algorithm for finding an optimum solution, as shown in Table I. The algorithm of the CS is composed of the following. Firstly, for all of cuckoo in the population, the bird randomly selects a nest to lay the egg where one bird can lay only one egg at a time. In order to make the whole population evolve forward, the birds cannot switch the eggs between the nests. Then, the host bird discovers the cuckoo eggs with probability (P_a) and determine $P_a \in [0, 1]$. In this case, the cuckoo has no other choice and it has to build a fully new nest. In generating the new solution, x_i^{t+1} cuckoo Lévy flight can be defined as:

$$x_i^{t+1} = x_i^t + \alpha_1 \otimes Lévy(\alpha), \tag{6}$$

and

$$Lévy \sim u = t^{-\lambda}; \quad 1 < \lambda \leq 3, \tag{7}$$

where α_1 is the step size and the product $\otimes$ means entry wise multiplications. The CS algorithm initialed the population t for the nest, and randomly select the best nest via Lévy flight. However, the most critical parameters are required to obtain the optimal solution from the CS algorithm are P_a and α_1.

III. Research methodology

A. The New Intelligent Hybrid Artificial Neural Network with Cuckoo Search Algorithm

The new algorithm that is being used in this research is based on CS algorithm. However, this algorithm is developed further to be called New Intelligent Hybrid Artificial Neural Network with Cuckoo Search Algorithm (CS-NIHANN). It is the combination of the algorithm of NIHANN with the CS algorithm for the optimization of weights and bias through the CS algorithm in forecasting problem. The presented steps of the CS-NIHANN algorithm can be explained as the follow:

Step 1: The cuckoo individual is encoded, that is a composed of real number string.

Step 2: The initial weights and bias of the NIHANN algorithm can be determined according to the best individual. After training the NIHANN algorithm, the weights and bias are used to predict the output. The E is the sum of the absolute error as follows:

$$E = c \cdot \sum_{i=1}^{n} |d_i - y_i|, \tag{8}$$

where n is the node number of the output layer in the NIHANN algorithm and c is a coefficient. d_i is the desired output and y_i is the predicted output for the node i in the NIHANN algorithm.

Step 3: A cuckoo is randomly chosen from the cuckoo population and its position is updated according to Eq. (6). The fitness (E_i) of the i^{th} cuckoo at generation t and position x_i^t is evaluated by Eq. (8).

Step 4: Another cuckoo is randomly chosen from the cuckoo population which is $i \neq j$, and its position fitness E_j of the i^{th} cuckoo at generation t and position x_j^t is evaluated by Eq. (8).

Step 5: Replacing operator, where the fitness value of the cuckoo i is bigger than the cuckoo j, that is, $E_i > E_j$, and x_j is replaced by the new solution.

Step 6: When the populations are in the finalized state, ceil $(n * Pa)$, the worst cuckoos are removed in each generation. Then, ceil $(n * Pa)$ cuckoos would randomly regenerated in order to maintain the number of population. Next, the best fitness cuckoos will be passed to the next generation.

The process is repeated to reveal the best weights and bias. In the last part, the NIHANN algorithm with the optimal weights and bias is constructed and is trained to predict the output.

B. Research design

The research design of the NIHANN algorithm is combined with the CS algorithm to address the NIHANN algorithm limit on the criminal cases forecasting model for the Upper Northeastern Provinces in Thailand. There are three main limits to NIHANN, that are, slow training, easy to fall into local minima, and very sensitive for the choice of the initial weights and bias, as shown in Fig. 2.

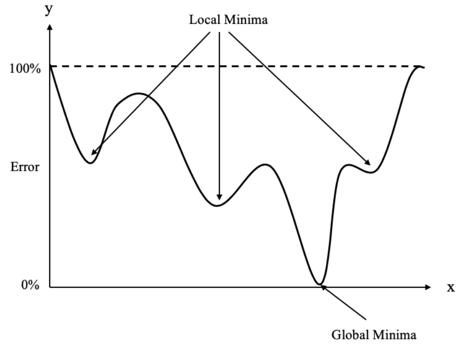

Fig. 2. Local minima problem

TABLE II
The input parameters of criminal cases forecasting model for G1-G3

Input parameters	Time Series		
	Min	Max	Mean
x1	21,775,407	22,015,239	21,914,467
x2	3,795,950	4,034,399	3,914,111
x3	14,690,255	14,743,868	14,721,409
x4	2,785,742	3,250,975	3,010,943
x5	239,530	256,728	247,850
x6	10,856,675	10,939,575	10,910,008
x7	10,918,732	11,075,664	11,004,459
x8	1,930	3,271	2,512
x9	116,878	185,650	148,249

C. Data collection and preparation

There are 3 steps in the process of data collection and preparation of the criminal cases forecasting model for the Upper Northeastern Provinces in Thailand, they are data selection, data cleaning and missing value, and data transformation.

1) Data selection: Knowing the causes of crimes could be the key to the elimination of societal crimes. For this reason, it is appropriate to identify and investigate criminogenic factors of crime occurrence and to forecast crime in the society. It is assumed in this paper that certain social factors, which in themselves, or in a combination of them, can facilitate the emergence or development of crimes. Thus, 9 social factors time series dataset from 2014 to 2019 are used as input parameters for the forecasting model in this research as shown in Table II. The time series dataset in all of the input parameters were collected by the National Statistical Office in Thailand (http://statbbi.nso.go.th).

In this research, the three outputs of the criminal cases forecasting model for the Upper Northeastern Provinces in Thailand are G1: the group of violent crimes, G2: the group of crimes concerning life and bodily harm, and G3: the group of crimes against property. Before data selection in each group, all of the input parameters relating to the crimes are being used, which are; the populations (x1), the childhood ages in the range of 1 - 14 years old (x2), the working ages in the range of 15 - 59 years old (x3), the elderly ages in the range of more than 60 years old (x4), the unidentified age (x5), the number of males (x6), the number of females (x7), the number of poor families (x8) and the number of narcotic cases (x9). The time series dataset in G1 - G3 groups for the input parameters (x1 - x9) are used for the Northeastern Provinces in Thailand.

2) Data cleaning and missing value: It is important to understand the concept of data cleaning and missing value to successfully manage the data. It is commonly found that some part of the data may not be properly reported due to lack of information. Thus, the process of data cleaning and missing value can fill in the gap and complete the data with reliable numbers. The method to fill in the missing value in the research design is to use mean imputation in R programming (freeware). The objective of imputation is to employ known relationships that can be identified in the valid values of the data set to assist in estimating the missing values.

3) Data transformation: A normalization method to transfer the data to fit between $[-1, 1]$ given by equation (9).

$$y_i = 2 \times \frac{[x_i - x_{min}]}{[x_{max} - x_{min}]} - 1, \tag{9}$$

where y_i is a normalization data for the input data to fit between $[-1, 1]$, x_i is the input data, x_{min} is the minimization of data for the input data, and x_{max} is the maximization of data for the input data.

D. Build forecasting model

The time series data for training set (70) and testing set (30) are used 10-fold cross-validation to forecast criminal cases. The activation functions and the parameters are CS-NIHANN and NIHANN to test for the best parameter in forecasting criminal cases. In this research, we present criminal cases forecasting model of the Upper Northeastern Provinces in Thailand. The input parameters data of the Upper Northeastern Provinces in Thailand are collected from 2014 to 2019. In addition, we tested the correlation of the criminal cases in the Upper Northeastern Provinces in Thailand and the forecasting model based on CS-NIHANN to forecast and be compared with CS-ANN. The stopping method of the forecasting model was calculated by error function or objective function and epoch or iterations of the network.

E. Data analysis and statistics

The comparison of the performance of forecasting model was calculated by Mean Squared Error (MSE) for the epoch of the CS-NIHANN and CS-ANN algorithms, Mean Absolute Percentage Error (MAPE), Root Mean Squared Error (RMSE) and Correlation Coefficient (R). All of the errors are used to compare actual values and estimated forecast values.

1) Mean Squared Error (MSE): The equation for MSE was given as follows:

$$MSE = \frac{1}{n} \cdot \sum_{t=1}^{n} (y_t - \widehat{y}_t), \tag{10}$$

where, n is the total number of observations, y_t is actual values , and $\widehat{y}_t$ is the estimated forecast. In all results, the lower values of MSE indicate a better forecasting ability of the algorithm.

2) Mean Absolute Percentage Error (MAPE): The equation of MAPE was given as follows:

$$MAPE = \frac{1}{n} \cdot \sum_{t=1}^{n} |\frac{(y_t - \widehat{y}_t)}{y_t}|, \tag{11}$$

where, n is the total number of observations, y_t is actual values , and $\widehat{y}_t$ is the estimated forecast.The average of the absolute percentage errors and the values of MAPE indicates the better forecasting.

3) Root Mean Squared Error (RMSE): The equation of RMSE was given as follows:

$$RMSE = \sqrt{\frac{1}{n} \cdot \sum_{t=1}^{n} (y_t - \widehat{y}_t)^2}, \tag{12}$$

where, n is the total number of observations, y_t is actual values , and $\widehat{y}_t$ is the estimated forecast. The residual of error was calculated to compare between the actual values and the estimated forecast values. A best model is the model with the values of RMSE equal to zero which indicates a perfect forecasting.

4) Correlation Coefficient (R): The equation for R was given as follows:

$$R = \frac{\sum_{t=1}^{n} (y_t - \overline{y}) \cdot (\widehat{y}_t - \widehat{y})}{\sqrt{\sum_{t=1}^{n} (y_t - \overline{y})^2 \cdot \sum_{t=1}^{n} (\widehat{y}_t - \widehat{y})}}, \tag{13}$$

where, n is the total number of observations, $\overline{y} = \frac{1}{n} \cdot \sum_{t=1}^{n} y_t$ and $\widehat{y} = \frac{1}{n} \cdot \sum_{t=1}^{n} \widehat{y}_t$ are the average values of y_t and $\widehat{y}_t$, respectively. The strength of relationships was calculated to compare between actual values and the estimated forecast values.

In this paper, the time series dataset for input parameters of criminal cases forecasting are tested by Pearson Correlation Coefficient with R programming (freeware). The comparison performance of forecasting model was calculated by the equations error in data analysis and statistics. The equations error are shown in equations (10) - (13). All of the errors are used to compare the actual values and the estimated forecast values.

IV. Results and Discussions

The workstation uses to carry out the result equipped with SCILAB. The SCILAB new version 6.0.2 is a free software that is used to carry out simulation of the algorithm. In part of data collection and preparation, the volume of criminal cases of the Upper Northeastern Provinces in Thailand is devided into three output parameters of the neural network which are the volumes of G1: the group of violent crimes, G2: the group of crimes concerning life and bodily harm, and G3: the group of crimes against property. Then they are tested with ADF where the numerical set to value of 0.05 level of significant in order to check for the non-stationary time series. The normalization is then transferred to the non-stationary time series data of the volume of criminal cases to be stationary time series to fit within the limits of the transfer function. The new algorithm called CS-NIHANN, that is the combination of the NIHANN and the CS algorithm, is used for the optimization of weights and bias through the CS algorithm in the forecasting problem.

TABLE III
The input parameters for criminal cases forecasting model for G1-G3

Groups	Input parameters [0.47,0.72]	Input parameters [-0.93,-0.87]
G1:	x4	x1
	x5	x2
	x8	x3
	x9	x6
		x7
Groups	Input parameters [0.38,0.85]	Input parameters [-0.78,-0.65]
G2:	x3	x1
	x5	x2
	x6	x4
	x8	x7
	x9	
Groups	Input parameters [0.45,0.82]	Input parameters [-0.74,-0.53]
G3:	x3	x1
	x5	x2
	x8	x4
	x9	x6
		x7

A. Multiple Input Single Output (MISO) for G1-G3

1) The results of historical forecast for G1-G3: The input parameters of criminal cases forecasting in G1-G3 for the Upper Northeastern Provinces in Thailand from 2014 to 2019 are tested by Pearson Correlation Coefficient with R programming is shown in Table III. First, it is found that there is correlation between the input parameters of x4, x5, x8, and x9 at the number between 0.47 and 0.72. The numerical value of correlation coefficient is between -0.93 and -0.87 and is the opposite direction of the number of the input parameters (x1, x2, x3, x6, and x7) in G1. Second, it is found that there is correlation between the input parameters of x3, x5, x6, x8, and x9 at the number between 0.38 and 0.85. The numerical value of correlation coefficient is between -0.78 and -0.65 and is the opposite direction of the number of the input parameters (x1, x2, x4, and x7) in G2. Third, it is found that there is correlation between the input parameters of x3, x5, x8, and x9 at the number between 0.45 and 0.82. The numerical value of correlation coefficient is between -0.74 and -0.53 and is in the opposite direction of the number of the input parameters (x1, x2, x4, x6, and x7) in G3. The time series data for training set (70) and testing set (30) are used 10-fold cross-validation to forecast criminal cases in G1-G3. We use 10-fold cross-validation to reduce the overfitting problem in the network.

The activation functions and the parameters are used in the CS-NIHANN and CS-ANN algorithms to define the best forecasting results for criminal cases in G1-G3. The accuracy of the CS-ANN and CS-NIHANN

TABLE IV
The accuracy of the CS-ANN and CS-NIHANN algorithms in G1-G3

Algorithms	G1: Train set			
	Mean	Best	Worst	Std.
CS-ANN	87	88	53	2.63
CS-NIHANN	88	89	87	0.45
	G1: Test set			
	Mean	Best	Worst	Std.
CS-ANN	85	88	49	3.58
CS-NIHANN	87	89	85	0.95
	G2: Train set			
	Mean	Best	Worst	Std.
CS-ANN	86	87	59	2.42
CS-NIHANN	88	89	86	0.58
	G2: Test set			
	Mean	Best	Worst	Std.
CS-ANN	86	87	52	3.63
CS-NIHANN	87	89	86	0.94
	G3: Train set			
	Mean	Best	Worst	Std.
CS-ANN	87	88	60	2.75
CS-NIHANN	88	89	87	0.57
	G3: Test set			
	Mean	Best	Worst	Std.
CS-ANN	86	88	52	3.58
CS-NIHANN	87	89	85	0.91

algorithms in G1-G3 is shown in Table IV. First, the results of the CS-NIHANN algorithm in G1 show that the train and test set (train value, test value) are (88,87) mean, (89,89) best, (87,85) worst, and (0.45,0.95) Std., respectively. Second, the results of the CS-NIHANN algorithm in G2 show that the train and test set (train value, test value) are (88,87) mean, (89,89) best, (86,86) worst, and (0.58,0.94) Std., respectively. Third, the results of the CS-NIHANN algorithm in G3 shows that the train and test set (train value, test value) are (88,87) mean, (89,89) best, (87,85) worst, and (0.57,0.91) Std., respectively. Therefore, the accuracy of the CS-NIHANN algorithm in G1-G3 is more effective in finding an optimal solution in relation to criminal cases of the Upper Northeastern Provinces in Thailand than of the CS-ANN algorithm.

The parameters of the CS-NIHANN algorithm in G1-G3 are shown in Table V. In G1, the learning algorithm is CS-NIHANN, the activation function is hyperbolic tangent sigmoid in hidden layer of the CS-NIHANN algorithm for solving the forecasting problem, and linear in the output layer. In the process of training function of the CS algorithm in G1, the learning rate is [0,1], the maximum of epoch is 2,000 iterations, the number of hidden neurons is between 1 and 10, where 5 is the best number of hidden neurons.

In G1, the CS-NIHANN and CS-ANN algorithms in G1 define the optimal neurons in input layer, hidden layer, output layer as 4, 5 and 1, respectively. The length of encoded string number for each cuckoo individual is 31 for the computation equation. The CS algorithm found the minimum of a 31 dimension function. The parameters of the CS-NIHANN algorithm in CS-NIHANN network used epoch or iterations = 2,000, learning rate = 0.001, and objective = 0.0004. The learning algorithm is CS-ANN, the activation function is a hyperbolic tangent sigmoid in [-1,1] for hidden layer, and linear in the output layer in G1. The process of training function is the CS part.

The parameters of the the CS-NIHANN algorithm in CS network are; the best discovery rate = 0.1 [0:0.1:1.0], the best of population size = 50 [10:10:100)], and maximum generations = 50 [10:10:100]. The number of hidden neurons is between 1 and 10, where 5 is the best number of hidden neurons in G1.

In G2, the learning algorithm is CS-NIHANN, the activation function is hyperbolic tangent sigmoid in hidden layer of the CS-NIHANN algorithm for solving the forecasting problem, and linear in the output layer. In the process of training function of the CS algorithm, the learning rate is [0,1], the maximum of epoch is 1,000 iterations, the number of hidden neurons is between 1 and 10, where 7 is the best number of hidden neurons.

In G2, the CS-NIHANN and CS-ANN algorithms define the optimal neurons in input layer, hidden layer, output layer as 5, 7, and 1, respectively. The length of encoded string number for each cuckoo individual is 50 for the computation equation. The CS algorithm found the minimum of a 50 dimension function. The parameters of the CS-NIHANN algorithm in CS-NIHANN network used epoch or iterations = 1,000, learning rate = 0.0001, and objective = 0.0003. The learning algorithm is CS-ANN, the activation function is hyperbolic tangent sigmoid in [-1,1] for hidden layer, and linear in the output layer. The process of training function is the CS part.

The parameters of the the CS-NIHANN algorithm in CS network are; the best discovery rate = 0.1 [0:0.1:1.0], the best of population size = 30 [10:10:100)], and maximum generations = 30 [10:10:100]. The number of hidden neurons is between 1 and 10, where 7 is the best number of hidden neurons in G2.

In G3, the learning algorithm is CS-NIHANN, the activation function is hyperbolic tangent sigmoid in hidden layer of the CS-NIHANN algorithm for solving forecasting problem, and linear in the output layer. In the process of training function of the CS algorithm, the learning rate is in [0,1], the maximum of epoch is 2,000 iterations, the number of hidden neurons is between 1 and 10, where 6 is the best number of hidden neurons.

In G3, the CS-NIHANN and CS-ANN algorithms define the optimal neurons in input layer, hidden layer, output layer as 4, 6 and 1, respectively. The length of encoded string number for each cuckoo individual is 37 for the computation equation. The CS algorithm found the minimum of a 37 dimension function. The parameters of the CS-NIHANN algorithm in CS-ANN network used epoch or iterations = 2,000, learning rate = 0.001, and objective = 0.0003. The learning algorithm is CS-ANN, the activation function is hyperbolic tangent sigmoid in [-1,1] for hidden layer, and linear in the output layer. The process of training function is the CS part.

The parameters of the the CS-NIHANN algorithm in CS network are; the best discovery rate = 0.1 [0:0.1:1.0],

TABLE V
The parameters of the CS-NIHANN algorithm for criminal cases forecasting model in G1-G3

Parameters	Value
G1:	
Activation Function	Hyperbolic Tangent Sigmoid
Learning Algorithm	NIHANN
Training Function	CS
Leaning Rate	0.001
Epoch	2,000
Number of Hidden Neurons	5
G2:	
Activation Function	Hyperbolic Tangent Sigmoid
Learning Algorithm	NIHANN
Training Function	CS
Leaning Rate	0.0001
Epoch	1,000
Number of Hidden Neurons	7
G3:	
Activation Function	Hyperbolic Tangent Sigmoid
Learning Algorithm	NIHANN
Training Function	CS
Leaning Rate	0.001
Epoch	2,000
Number of Hidden Neurons	6

the best of population size = 50 [10:10:100)], and maximum generations = 50 [10:10:100]. The number of hidden neurons is between 1 and 10, where 6 is the best number of hidden neurons in G3.

The MSE in equation (10) is used to select the optimization of the neurons in hidden layer for the CS-NIHANN network.

B. Multiple Input Multiple Output (MIMO) for G1-G3

All input parameters of criminal cases forecasting for MIMO are x1, x2, x3, x4, x5, x6, x7, x8 and x9 for the Upper Northeastern Provinces in Thailand from 2014 to 2019. The input parameters for criminal cases forecasting model is shown in Table II. In MIMO, there are 9 inputs in the input layers (x1-x9) and 3 outputs for output layer (G1-G3). The time series data for training set (70) and testing set (30) are used 10-fold cross-validation to forecast criminal cases in G1-G3. We use 10-fold cross-validation to reduce the overfitting problem in the network. The activation functions and the parameters are used in the CS-NIHANN and CS-ANN algorithms for the best forecasting result in G1-G3. The accuracy of the CS-ANN and CS-NIHANN algorithms is shown in Table VI.

It can be seen that the results of the CS-NIHANN algorithm show that the train and test set (train value, test value) are (86,86) mean, (87,88) best, (86,86) worst and (2.43,1.97) Std., respectively. Therefore, the accuracy of the CS-NIHANN algorithm is more effective in finding an optimal solution in relation to criminal cases forecast than of the CS-ANN algorithm.

The learning algorithm is CS-NIHANN, the activation function is hyperbolic tangent sigmoid in hidden layer of the CS-NIHANN algorithm for solving forecasting problem and the hyperbolic tangent sigmoid in the output layer. The process of training function is the CS algorithm, the learning rate is [0,1], the maximum of epoch is 2,000 iterations, the number of hidden neurons is between 1 and 10, where 5 is the best number of hidden neurons.

TABLE VI
The accuracy of the CS-ANN and CS-NIHANN algorithms in G1-G3 in MIMO

Algorithms	G1-G3: Train set			
	Mean	Best	Worst	Std.
CS-ANN	85	86	53	3.73
CS-NIHANN	86	87	86	2.43
	G1-G3: Test set			
	Mean	Best	Worst	Std.
CS-ANN	86	87	45	4.68
CS-NIHANN	86	88	86	1.97

TABLE VII
The parameters of the CS-NIHANN algorithm in MIMO for criminal cases forecasting model in G1-G3

Parameters	Value
Activation Function	Hyperbolic Tangent Sigmoid
Learning Algorithm	NIHANN
Training Function	CS
Leaning Rate	0.001
Epoch	2,000
Number of Hidden Neurons	10

In this research, the CS-NIHANN and CS-ANN algorithms show the optimal neurons in input layer, hidden layer, output layer at 9, 10, and 3, respectively. The length of encoded string number for each cuckoo individual is 133 for the computation equation. The CS algorithm found the minimum of a 133 dimension function. The parameters of the CS-NIHANN algorithm in CS-ANN network use epoch or iterations = 2,000, learning rate = 0.001, and objective = 0.0001. The learning algorithm is CS-NIHANN, the activation function is hyperbolic tangent sigmoid in [-1,1] for hidden layer, and hyperbolic tangent sigmoid in the output layer. The process of training function is the CS part. The parameters of the the CS-NIHANN algorithm in CS network are; the best discovery rate = 0.1 [0:0.1:1.0], the best population size = 50 [10:10:100)], and maximum generations = 50 [10:10:100]. The number of hidden neurons is between 1 and 10, where 5 is the best number of hidden neurons. The MSE in equation (10) is used to select the optimization of the neurons in hidden layer for the CS-NIHANN network. The parameters of the CS-NIHANN algorithm are shown in Table VII.

C. The results of ex-ante forecast

In MISO part, the forecasting models were also implemented in SCILAB version 6.0.2. All of the errors are used to compare actual values and estimated forecast values in the CS-ANN and CS-NIHANN algorithms in all three groups (G1 - G3). The comparison performance of forecasting model for historical forecast part was calculated by Mean Absolute Percentage Error (MAPE),

TABLE VIII
Performance of forecasting model in G1-G3 for MISO

Iterations	Algorithms	G1: Performance		
		MAPE	RMSE	R
500	CS-ANN	0.1432	135,287	0.7198
	CS-NIHANN	0.0859	85,120	0.8717
1000	CS-ANN	0.0827	78,598	0.7953
	CS-NIHANN	0.0574	59,172	0.9281
1500	CS-ANN	0.0805	75,513	0.7986
	CS-NIHANN	0.0567	57,724	0.9385
2000	CS-ANN	0.0795	73,395	0.8817
	CS-NIHANN	0.0475	53,317	0.9391
		G2: Performance		
		MAPE	RMSE	R
500	CS-ANN	0.2732	147,635	0.7531
	CS-NIHANN	0.0863	87,532	0.7953
1000	CS-ANN	0.0815	73,368	0.8873
	CS-NIHANN	0.0486	53,324	0.9492
1500	CS-ANN	0.0827	75,242	0.7973
	CS-NIHANN	0.0573	57,763	0.9385
2000	CS-ANN	0.0893	75,463	0.7978
	CS-NIHANN	0.0527	57,727	0.9365
		G3: Performance		
		MAPE	RMSE	R
500	CS-ANN	0.1501	136,019	0.7214
	CS-NIHANN	0.0865	85,352	0.8675
1000	CS-ANN	0.0856	78,612	0.7825
	CS-NIHANN	0.0569	59,256	0.9291
1500	CS-ANN	0.0897	75,679	0.7974
	CS-NIHANN	0.0559	57,745	0.9215
2000	CS-ANN	0.0763	73,438	0.8898
	CS-NIHANN	0.0467	53,375	0.9497

Root Mean Squared Error (RMSE) and Correlation Coefficient (R).

The CS-NIHANN algorithm in G1-G3 has smaller MAPE and RMSE values as well as a bigger R value than the CS-ANN algorithm for the 500 - 2,000 iterations. In G1 part, at the 2,000 iteration, the MAPE value is 0.0475, the RMSE value is 53,317, and the R value is 0.9391. In G2 part, at the 1,000 iteration, the MAPE value is 0.0486, the RMSE value is 53,324, and the R value is 0.9492. In G3 part, at the 2,000 iteration, the MAPE value is 0.0467, the RMSE value is 53,375, and the R value is 0.9497, respectively. All of the errors are used to compare the actual values with the estimated forecast values in the CS-ANN and CS-NIHANN algorithms. Therefore, it can be concluded that the CS-NIHANN algorithm trains faster, can obtain the global optimal solution, and has a good generalization performance as shown in Table VIII.

In MIMO part, from Table IX, the CS-NIHANN algorithm has smaller MAPE and RMSE values as well as a bigger R value than the CS-ANN algorithm for the 500 - 2,000 iterations. At the 2,000 iteration, the MAPE value is 0.1425, the RMSE value is 57,259, and the R value is 0.8959, respectively. Therefore, it can be concluded that the CS-NIHANN algorithm trains faster, can obtain the global optimal solution, and has a good generalization performance in MIMO part as well.

TABLE IX
Performance of forecasting model in G1-G3 for MIMO

Iterations	Algorithms	Performance		
		MAPE	RMSE	R
500	CS-ANN	0.1592	155,186	0.8619
	CS-NIHANN	0.1499	86,207	0.8917
1000	CS-ANN	0.1857	88,694	0.6951
	CS-NIHANN	0.1574	79,172	0.7285
1500	CS-ANN	0.1827	75,513	0.7986
	CS-NIHANN	0.1567	73,128	0.8192
2000	CS-ANN	0.1793	82,497	0.8953
	CS-NIHANN	0.1425	57,259	0.8959

The result shows that the CS-NIHANN algorithm has a faster convergence than the CS-ANN algorithm in calculating an optimal solution in the criminal cases forecasting model of the Upper Northeastern Provinces in Thailand G1-G3. The results obtained by these algorithms were used to forecast the data and are shown in Table VIII-IX. The CS-NIHANN algorithm has a smaller MAPE and RMSE values as well as a bigger R value than the CS-ANN algorithm. Therefore, it can be concluded that the CS-NIHANN algorithm trains faster, can obtain the global optimal solution, and has a good generalization performance in all three groups (G1 - G3) in both MISO and MIMO.

V. Conclusions

In this paper, the new algorithm, called a New Intelligent Hybrid Artificial Algorithm Neural Network with Cuckoo Search Algorithm (CS-NIHANN) is presented as a criminal forecasting model. The research uses the CS algorithm to optimize weights and bias in the basic of the NIHANN algorithm to solve the forecasting problem. The time series dataset for input parameters of criminal cases forecasting are tested by Pearson Correlation Coefficient with R programming. The comparison performance of forecasting model was calculated by the equations error in data analysis and statistics. The results of the CS-NIHANN algorithm shows that the accuracy of the train and test set of the CS-NIHANN algorithm is more effective in finding an optimal solution in relation to criminal cases of the Upper Northeastern Provinces in Thailand than of the CS-ANN algorithm. As an illustration, the comparison performance of forecasting model for historical forecasting part was calculated by Mean Absolute Percentage Error (MAPE), Root Mean Squared Error (RMSE) and Correlation Coefficient (R). All of the errors are used to compare the actual values with the estimated forecast values in the CS-ANN and CS-NIHANN algorithms. The CS-NIHANN algorithm has a smaller MAPE and RMSE values as well as a bigger R value than the CS-ANN algorithm. The result confirms that the CS-ANN algorithm trains faster, can obtain the global optimal solution, and has a good generalization performance. In MISO and MIMO parts, the results show that the CS-NIHANN algorithm has a faster convergence than the CS-ANN algorithm in calculating an optimal solution to the criminal cases (G1: the group of violent crimes, G2: the group of crimes concerning life and bodily harm, and G3: the group of crimes against property) forecasting model of the Upper Northeastern Provinces in Thailand. This study suggested social structure factors could have an impact on crime rates in G1: the group of violent crimes, G2: the group of crimes concerning life and bodily harm, and G3: the group of crimes against property. Interestingly, the number of poor families (x8) and the number of narcotic cases (x9) are associated with the numbers of all three groups of crime types. This illustration would benefit in raising awareness to the policy makers to be prepared for the future possibilities. Still, this new algorithm can also be applied to other forecasting scenario as well. In the future, we would like to focus our research in three prospects which are; to improve other algorithms of neural networks, to modify the groups of metaheuristic optimizations, and to combine those results to solve other forecasting models and pattern recognition.

Appendix A

ANN:	Artificial Neural Network
CS:	Cuckoo Search Algorithm
CS-ANN:	Artificial Neural Network with Cuckoo Search Algorithm
CS-NIHANN:	New Intelligent Hybrid Artificial Neural Network with Cuckoo Search Algorithm
MISO:	Multiple Input Single Output
MIMO:	Multiple Input Multiple Output
MSE:	Mean Squared Errors
MAPE:	Mean Absolute Percentage Errors
RMSE:	Root Mean Squared Error

References

[1] N. B. Adrian, ”Data transforms with exponential smoothing methods of forecasting,” International Journal of Forecasting, vol. 30, pp. 918-927, 2014.

[2] B. Billah and et al., ”Exponential smoothing model selection for forecasting,” International Journal of Forecasting, vol. 22, pp. 239–247, 2006.

[3] Y. Lu and S. M. AbouRizk, ”Automated Box–Jenkins forecasting modelling,” Automation in Construction, vol. 18, pp. 547–558, 2009.

[4] S. Buhamra, N. Smaoui, and M. Gabr, ”Applied Mathematical Modelling (Netherlands),” Amsterdam: Netherlands/American Elsevier BV, vol. 27, pp. 805–815, 2003.

[5] M. Kolcak and A. Keles, ”The adaptive neuro-fuzzy model for forecasting the domestic debt,” Knowledge-Based Systems, vol. 21, pp. 951–957, 2008.

[6] S. Karimi and et al., ”Neuro-fuzzy and neural network techniques for forecasting sea level in Darwin Harbor, Australia,” Computers and Geosciences, vol. 52, pp. 50–59, 2013.

[7] H. Peng and et al., ”Time series forecasting with a neuro-fuzzy modeling scheme,” Applied Soft Computing, vol. 32, pp. 481–493, 2015.

[8] M. Khashei and M. Bijari, ”An artificial neural network (p, d, q) model for time series forecasting,” Expert Systems with Applications, vol. 37, pp. 479–489, 2010.

[9] S. H. Elsafi, ”Artificial Neural Networks (ANNs) for flood forecasting at Dongola Station in the River Nile Sudan,” Alexandria Engineering Journal, vol. 53, pp. 655–662, 2014.

[10] G. S. Thakur, M. R. Bhattacharyya and S. S. Mondal, ”Artificial neural network Based Model for forecasting of Information in India” Fuzzy Information and Engineering, vol. 8, pp. 87-100, 2016.

[11] S. Taghi, A. Niaki and S. Hoseinzade, ”Forecasting S and P 500 index using artificial neural networks and design of experiments,” Journal of Industrial Engineering International, vol. 9, no. 1, 2013.

[12] S. Stauba and et al., ”Artificial Neural Network and Agility,” Procedia - Social and Behavioral Sciences, vol. 195, pp. 1477–1485, 2015.

[13] K. Kourou K and et al., ”Machine learning applications in cancer prognosis and prediction,” Computational and Structural Biotechnology Journal, vol. 13, pp. 8–17, 2015.

[14] J. P. Teixeira and P. O. Fernandes, ”Tourism time series forecast with artificial neural networks,” Journal of Review of Applied Management Studies, vol. 12, pp. 26–36, 2014.

[15] L. Falat and et al., ”Application of Neural Network Models in Modelling Economic Time Series with Non-constant Volatility,” Journal of Procedia Economics and Finance, vol. 34, pp. 600–607, 2015.

[16] N. Tehlah, P. Kaewpradit and I. M. Mujtaba, ”Artificial Neural Network based Modelling and Optimization of Refined Palm Oil Process,” Journal of Neurocomputing In Press, vol. 216, pp. 489-501, 2016.

[17] H. C. Co and R. Boosarawongse, ”Forecasting Thailand’s rice export: Statistical techniques vs. artificial neural networks,” Journal of Computers and Industrial Engineering, vol. 53, pp. 610–627, 2007.

[18] A. Marvuglia and A. Messineo, ”Using Recurrent Artificial Neural Networks to Forecast Household Electricity Consumption,” Journal of Energy Procedia, vol. 14, pp. 45-55, 2012.

[19] A. Sarkar and P. Pandey, ”River Water Quality Modelling using Artificial Neural Network Technique,” Journal of Aquatic Procedia, vol. 4, pp. 1070–1077, 2015.

[20] W. S. Mcculloch and W. Pitts, ”A logical calculus of the ideas immanent in Nervous activity,” Journal of Bulletin of Mothemnticnl Biology, vol. 52, pp. 99-115, 1990.

[21] R. Law, ”Back-propagation learning in improving the accuracy of neural network-based tourism demand forecasting,” Journal of Tourism Management, vol. 21, pp. 331–340, 2000.

[22] S. Kulkarni and I. Haidar, ”Forecasting model for crude oil price using artificial neural networks and commodity futures prices,” International Journal of Computer Science and Information Security, vol. 2, pp. 1–8, 2009.

[23] H. Yonaba, F. Anctil and V. Fortin, ”Comparing sigmoid transfer functions for neural network multistep ahead streamflow forecasting,” Journal of Hydrologic Engineering, vol. 15, pp. 275–283, 2010.

[24] D. H. Wolpert, ”The lack of a priori distinctions between learning algorithms,” Neural Computation, vol. 8, pp. 1341–1390, 1996.

[25] D. H. Wolpert, ”The existence of a priori distinctions between learning algorithms,” Journal of Neural Computation, vol. 8, pp. 1391–1420, 1996.

[26] S. E. Fahlman and C. Lebiere, ”The cascade-correlation learning architecture,” Advances in Neural Information Processing, vol. 2, pp. 524–532, 1990.

[27] A. Wam, S. Esm and A. Esa, ”Modified Back Propagation Algorithm for Learning Artificial Neural Networks,” Eighteenth National Radio Science Conference (NRSC), pp. 345-352, 2001.

[28] W. Leigh, R. Hightower and N. Modena, ”Forecasting the New York Stock exchange composite index with past price and invest rate on condition of volume spike,” Journal Expert System with Applications, vol. 28, pp. 1-8, 2005.

[29] M. N. Nazri, K. Abdullah and M. Z. Rehman, ”A New Levenberg Marquardt based Back Propagation Algorithm Trained with Cuckoo Search,” Journal of Procedia Technology, vol. 11, pp. 18-23, 2013.

[30] R. Lewis, ”A survey of metaheuristic-based techniques for University Timetabling problems,” Or Spectrum, vol. 30, pp. 167-190, 2008.

[31] E. G. Talbi, ”Metaheuristics: From Design to Implementation,” John Wiley and Sons, vol. 1, pp. 1-593, 2009.

[32] J. Zhang, T. Lok and M. Lyu, ”A hybrid particle swarm optimization back propagation algorithm for feed forward neural network training,” Journal Applied Mathematics and Computation, Elsevier, vol. 185, pp. 1026-1037, 2007.

[33] R. Mendes and et al., ”Particle swarm for feed forward neural network training,” Proceedings of the International Joint Conference on Neural Networks, vol. 2, pp. 1895-1899, 2002.

[34] H. Shah and et al., ”Global hybrid ant bee colony algorithm for training artificial neural networks,” Computational Science and Its Applications, vol. 7333, pp. 87-100, 2012.

[35] H. Shah, R. Ghazali, and N. M. Nawi, ”Hybrid ant bee colony algorithm for volcano temperature prediction,” Communications in Computer and Information Science, vol. 281, pp. 453-465, 2012.

[36] D. Karaboga, B. Basturk and C. Ozturk, ”Artificial bee colony (ABC) optimization algorithm for training feed-forward neural networks,” Journal Modeling Decisions for Artificial Intelligence, vol. 4617, pp. 318-319, 2007.

[37] X-S. Yang, ”A New Metaheuristic Bat-Inspired Algorithm,” Nature Inspired Cooperative Strategies for Optimization, Springer Berlin Heidelberg, vol. 284, pp. 65-74, 2010.

[38] X-S. Yang, ”Firefly algorithm, stochastic test functions and design optimization,” International Journal of Bio-Inspired Computation, vol. 2, pp. 78-84, 2010.

[39] X-S. Yang and S. Deb, ”Cuckoo Search via Lévy flights,” Proceedings of World Congress on Nature and Biologically Inspired Computing, pp. 210-214, 2009.

[40] X-S. Yang and S. Deb, ”Engineering optimisation by cuckoo search,” International Journal of Mathematical Modelling and Numerical Optimisation, vol. 1, pp. 330-343, 2010.

[41] T. T. Nguyen, D. N. Vo and B. H. Dinh, ”Cuckoo search algorithm for combined heat and power economic dispatch,” Journal of Electrical Power and Energy Systems, vol. 81, pp. 204–214, 2016.

[42] M. K. Marichelvama, T. Prabaharanb and X-S. Yang, ”Improved cuckoo search algorithm for hybrid flow shop scheduling problems to minimize makespan,” Journal of Applied Soft Computing, vol. 19, pp. 93–101, 2014.

[43] M. N. Nazri, K. Abdullah and M. Z. Rehman, ”A New Levenberg Marquardt based Back Propagation Algorithm Trained with Cuckoo Search,” Journal of Procedia Technology, vol. 11, pp. 18-23, 2013.

[44] J. Wang and et al., ”Forecasting solar radiation using an optimized hybrid model by Cuckoo Search algorithm,” Journal of Energy, vol. 81, pp. 627-644, 2015.

[45] H. Chiroma and et al., ”Global Warming: Predicting OPEC Carbon Dioxide Emissions from Petroleum Consumption Using Neural Network and Hybrid Cuckoo Search Algorithm,” PLOS ONE, vol. 10, pp. 1-21, 2015.

[46] Z. Hu, Y. Zhang and L. Yao, ”Radial Basis Function Neural Network with Particle Swarm Optimization Algorithms for Regional Logistics Demand Prediction,” Hindawi Publishing Corporation, vol. 2014, no. 414058, 2014.

[47] W-C. Ho, J-M. Su, and C-Y. Chang, ”Maximal Market Potential of Feeder Bus Route Design Using Particle Swarm Optimization,” IAENG International Journal of Applied Mathematics, vol. 49:4, pp. 470-477, 2019.

[48] L. Wu, J. Zhou, and Z. Li, ”Applying of GA-BP Neural Network in the Land Ecological Security Evaluation,” IAENG International Journal of Computer Science, vol. 47:1, pp. 11-18, 2020.