IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 14 May 2023, accepted 28 June 2023, date of publication 5 July 2023, date of current version 3 August 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3292543

RESEARCH ARTICLE

Enhancing Time Series Forecasting With an Optimized Binary Gravitational Search Algorithm for Echo State Networks

ZOHAIB AHMAD[1], TARIQ MAHMOOD[2,3], TEG ALAM[4,5], AMJAD REHMAN[2], (Senior Member, IEEE), AND TANZILA SABA[2], (Senior Member, IEEE)

[1]Department of Computer Science & Information Technology, The Sahara University, Narowal 51600, Pakistan
[2]Artificial Intelligence and Data Analytics (AIDA) Lab, CCIS Prince Sultan University, Riyadh 11586, Saudi Arabia
[3]Faculty of Information Sciences, University of Education, Vehari Campus, Vehari 61100, Pakistan
[4]Department of Industrial Engineering, College of Engineering, Prince Sattam bin Abdulaziz University, Al Kharj-11942, Saudi Arabia
[5]Azad Institute of Engineering and Technology, Azad puram, Chandrawal via Bangla Bazar & Bijnour, Near CRPF Camp, Lucknow-226002, India

Corresponding authors: Teg Alam (t.alam@psau.edu.sa) and Tariq Mahmood (tmsherazi@ue.edu.pk)

This work was supported by Artificial Intelligence and Data Analytics (AIDA) Lab CCIS Prince Sultan University Riyadh Saudi Arabia. The authors would also like to acknowledge the support of Prince Sultan University for paying the Article Processing Charges (APC) of this publication.

ABSTRACT The echo state network (ESN) is a cutting-edge reservoir computing technique designed to handle time-dependent data, making it highly effective for addressing time series prediction tasks. ESN inherits the more precise design of standard neural networks and the relatively simple learning process and has a strong computing capacity for solving nonlinear problems. It can disseminate low-dimensional information cues to high-dimensional areas enabling extracting data. However, this study has proven that not all reservoir output dimensions directly impact model generalization. This study desires to enhance the ESN model's generalization abilities by decreasing the redundant reservoir output feature. A remarkable hybrid model is proposed that optimizes the ESN output association through feature selection. This model is called the binary improved gravitational search algorithm (BIGSA) echo state network (BIGSA-ESN). BIGSA's feature selection approach complements the ESN output connection architecture. In this study, evaluation was performed using root mean square error (RMSE). The experimental findings on the Lorenz and Mackey-Glass benchmark time-series datasets demonstrate that the proposed technique outperforms conventional evolutionary methods. Moreover, empirical findings on predicting a significant water quality parameter from the wastewater treatment process (WWTP) dataset demonstrate that the proposed ensemble of BIGSA models performs very well in real-world scenarios.

INDEX TERMS Network optimization, time series forecasting, binary gravitational search algorithm, root mean square error, echo state network.

I. INTRODUCTION

Time series prediction has attracted numerous scholars in recent decades [1]. This field creates holistic algorithms based on traditional research to predict future values. Time-series estimates exhibit the ability to predict the future by analyzing the history [2]. The economics [3], energy [4], electromechanical systems [5], and engineering [6], [7] are largely depend on predicting. Also, time series estimation is a complex problem since it is a nonlinear dynamic issue. A superior model of functional prediction competencies should be established to boost time series forecasting [8]. Appropriate computer methods were developed to enhance prediction accuracy [9]. However, no universal standards exist for determining the optimal solution to a specific concern [10]. Artificial neural networks (ANNs) have emerged

The associate editor coordinating the review of this manuscript and approving it for publication was Alireza Sadeghian.

as a highly effective and successful methodology in the realm of time series prediction. ANNs' key refinements are data-driven and self-adaptive computational simulation competence. ANNs are intelligent predictors [11] that suggest an ample nonlinear enabling the development of linear series for diverse models and extensively employed forecasting systems [4], [12]. The identical approximation feature that allows ANN to create time-series facts significantly confuses the benchmark formulation. Thus, developing a significant design is vital for neural network application [13], [14]. Recurrent neural networks (RNNs), including Hopfield, Elman, and ESN, propose extensive memory by developing internal network parameters that prompt the system to display intricate temporal patterns. Sophisticated systems relying on complicated rational behavior have proved superior efficacy to standard feed-forward neural networks (FNNs) for modeling nonlinear dynamic systems [15].

Researchers have employed RNNs in a variety of domains, including clustering [16], pattern detection [17], classification [18], and prediction [19]. ESN is among the most prevalent RNNs [20]. The reservoir is ESN's random-initialized hidden layer. The ESN is a recursive design famous for its simple form and accurate predictions. ESN is unique among other recursive networks in two key ways: first, its hidden layer consists of a vast, loosely coupled reservoir, and second, only the weights connecting the reservoir to the system output must be learned [21]. The ESN reservoir plays a crucial role in transforming low-dimensional input into a higher-dimensional space, thereby facilitating the representation of dynamic systems that incorporate a feedback relation within the reservoir and output layer [20]. Despite their extraordinary learning potential, ESN models' complicated and dynamic weight structure remains poorly studied, making reservoir optimization difficult. So, solving this problem is a crucial study. The fruit fly model, particle swarm optimization (PSO), and the hybrid gravitational search approaches have been adopted by researchers to boost crucial ESN parameters and increase overall efficiency as part of the hyper-parameter tuning optimization technique [22], [23], [24].

Network topology is generally intended to improve the competitiveness of ESN further [25]. Identifying the optimal ESN output connection is typically necessary for achieving the best performance. However, it can be challenging as the output layer is fully connected, which may seem incongruous [26]. Investigations have indicated that loosely coupling the ESN output layer increases network efficiency [27], comparable to the sparse links among brain neurons [28]. Yet, optimizing connections is still an issue that must be tackled. Improving the connectivity amongst neurons in the ESN reservoirs and output layer is a feature selection challenge. Frequently, data representation has extra components that may be eradicated [29], [30]. The primary objective of feature selection is to enhance a system's interpretation by downsizing the chosen features' dimensionality.

Recent studies have focused on optimizing the ESN output link. Conventional feature selection techniques, including least angle regression and backward selection, aim to improve network generalization by removing superfluous connections within the ESN reservoir and output layer [27], [31]. Moreover, the greedy feature engineering approach is employed to decrease the high computational cost of the feedback provided [28]. Several feature selection methods are unsuitable for addressing ESN output connections' optimization issues. However, evolutionary computation (EC) algorithms can potentially overwhelm these restrictions. Using a distinct feature selection scheme, Liu et al. [29] presented the binary grey wolf ESN algorithm to enhance the ESN output link. The EC algorithm is inspired by the realistic evaluation process, and a range of global optimization techniques based on the study of biological behavior are provided [32]. A commonly used evolutionary computing method is the genetic algorithm (GA) [33]. It solves discrete issues like feature selection faster than enumeration, heuristic, and other optimization approaches. Genetic algorithm (GA) durability, scalability, programming, parameter tweaking, and deliberate search speed make it challenging to use [34]. Besides, the particle swarm optimization (PSO) approach's quick estimate speed in determining the best configuration may also improve the system's parameter settings [23]. The PSO method functions by iteratively updating the location of each particle using current, global, and specific extremum information. The PSO approach can tackle optimization problems involving continuous functions. PSO can more readily converge in advance, while the potential for local optimization is minimal. Also, it cannot be implemented for arbitrary feature selection challenges. So, a discrete binary PSO method with zero or one trajectory coordinates is created [35]. Gravitational Search Algorithm is among the most recent EC algorithms exploited by Newton's theories of gravity and motion [36]. The search agents in this technique are a swarm of molecular interactions. It has been proven that GSA provides numerous benefits over conventional heuristic optimization techniques. However, the GSA's search feature is very flawed. Due to a quick loss of variation, the first method is susceptible to early convergence [37]. Finding a balance between exploring new options and exploiting known information is challenging. Several measures have been developed to overcome these obstacles. Zandevakili et al. [38] suggested a new gravitational search approach incorporating uniform circular motion and centripetal force to create an attractive-repulsive approach. A chaotic optimization technique is employed, causing the algorithm's parameters to fluctuate in a chaotic fashion [39].

A fused PSO and gravitational search algorithm are used to solve binary optimization concerns [40]. Due to the paucity of memory in the original method, its exploitation step converges slowly. The study presents a novel gravitational search algorithm (GSA) that utilizes global memory to improve exploitation capabilities in the final iterations, achieving

a balance between exploration and exploitation [41], [42]. Also, a binary version of IGSA is developed, referencing the original. The binary GSA [43] is utilized to pick the optimal characteristic subset for grading purposes. BIGSA is proposed in this study to enhance the output connectivity of the ESN. To assess the efficacy of the suggested method, two benchmark time series datasets, namely the Mackey glass and Lorenz time series, were utilized, as well as data from a critical water quality parameter prognostication investigation in the WWTP. The empirical finding reveals that the suggested BIGSA-ESN model exhibits superior performance, reducing the generalization error of the traditional ESN algorithm. These findings suggest the BIGSA algorithm's efficacy in enhancing ESN models' performance. The proposed work's significant intent is as follows:

- This study introduces IGSA, an improved version of GSA that addresses its limitations by balancing exploration and exploitation and using global memory to boost outcome quality.
- To select the optimum feature subset for classification to optimize the ESN output connection and improve its generalization capabilities using novel BIGSA-ESN algorithms
- To evaluate the proposed algorithms on the Lorenz and Mackey-Glass benchmark time series datasets proven the superiority over conventional evolutionary methods
- To forecast the effluent NH4-N concentration in the WWTP, a soft-computing approach based on the BIGSA ESN is developed. Initially, data preprocessing and data preparation methods were used to select auxiliary variables as input variables for the soft computing model. Second, the suggested BIGSA- ESN is used to develop a soft-computing model of effluent NH4-N that can fulfill actual predictive situations.
- Finally, the empirical consequences reveal that the constructed predictive model achieves more promising than other ESN standards regarding prediction accuracy.

This paper's organization is as follows. A concise study of the methods, including the standard ESN, the principle of GSA, the proposed IGSA, and the binary interpretation of IGSA, is provided in Section II. The approach for ESN optimization that makes use of BIGSA is provided in Section III. Section IV explain the experimentation that was carried out using the benchmark datasets. It leads the interpretations of the findings of the investigation, and Section V wraps up the finding of the study.

II. MATERIALS AND METHODOLOGY

A. THE STANDARD ECHO STATE NETWORK

The ESN is a class of recurrent neural networks (RNN) consisting of three layers as input, hidden, and output layer. Contrary to a conventional RNN, the input-to-hidden layer and ESN reservoir link weights are randomly initialized. In an ESN, the links between the hidden and output layers are fixed, and their weights are not taught during training. In general, the training process of ESNs is rapid, particularly in regression tasks in which the network is taught to predict a constant output value. The proposed design of the ESN benchmark is presented in Figure 1. The reservoir in the proposed design is similar to the typical neural network's hidden layer. The connection implications for the reservoir-to-output layer are W^{out}, the weights for the reservoir-to-input layer are W^{in}, and the weights for the reservoir-to-neurons are W. In addition, a link exists from the output to the reservoir layer represented by W^{back}. This connectivity (shown by the dashed arrow in Figure 1 is optional. The ESN can create multi-step projections while W^{back} exists. Otherwise, the prognosis is confined to a single step. Figure 1 depicts the configuration of the ESN without feedback connections, which comprises three components: 1) K input units acquiring external inputs; 2) N core units with associated states; and 3) L result units feeding the results required by the use. ESN training only involves the calculation of the output weights W^{out}, while all other weights are kept fixed.

If $\boldsymbol{f} = [\boldsymbol{f_1}, \boldsymbol{f_2}, \ldots, \boldsymbol{f_N}]^T$ and $\boldsymbol{f^{out}} = [\boldsymbol{f_1^{out}}, \boldsymbol{f_2^{out}}, \ldots, \boldsymbol{f_L^{out}}]^T$ are activation functions of internal and output units, respectively. The internal states $\boldsymbol{x(t)}$ and output $\boldsymbol{y(t)}$ are depicted as:

$$x(t) = dsymbolf(\boldsymbol{W}\boldsymbol{x}(t-1)) + \boldsymbol{W}^{in}\boldsymbol{u}(t) \tag{1}$$

$$\boldsymbol{y}(t) = f^{\text{out}}\left(\boldsymbol{W}^{\text{out}}\boldsymbol{x}(t)\right) \tag{2}$$

In Eqs.1 and 2, $\boldsymbol{W}$ is a weight matrix of ESN's reservoir. $\boldsymbol{S} = [\boldsymbol{x}(1), \boldsymbol{x}(2), \ldots, \boldsymbol{x}(t)]^T$ and $\boldsymbol{D} = [\boldsymbol{d}(1), \boldsymbol{d}(2), \ldots, \boldsymbol{d}(t)]^T$ are inner state vector and selected output, accordingly. The output weights $\boldsymbol{W^{out}}$ can be computed in Eq.2.

$$\boldsymbol{W}^{\text{out}} = \left(\left(\boldsymbol{S}^{\text{T}}\boldsymbol{S}\right)^{-1}\boldsymbol{S}^{\text{T}}\boldsymbol{D}\right)^{\text{T}} \tag{3}$$

In contrast to gradient-based methods, ESN solely uses linear regression to determine its output weights, whereas all other weights stay fixed. Additionally, ESN is advantageous while analyzing one-dimensional time series. To enable the functionality of the ESN, several factors must be considered. For instance, the input from the previous moment reverberates in the ESN reservoir, making it crucial to prevent the reservoir state from exploding. The eigenvalues of the matrix $\boldsymbol{W}$ should ideally be smaller than or equivalent to one to prevent the reservoir from becoming unstable or exploding. $\boldsymbol{W^{out}}$ is the only parameter in the ESN that must be adjusted during learning, and $\boldsymbol{W}$ should be chosen to allow for the incorporation of as multiple relevant data patterns as conceivable. The sparsity of $\boldsymbol{W}$ should be appropriately chosen, usually between 1% and 5%, to ensure optimal performance.

B. THE STANDARD GRAVITATIONAL SEARCH ALGORITHM (GSA)

The GSA is based on optimization strategies boosted by the law of gravity. In this approach, particles are assumed to represent the objects, whereas masses are used to gauge

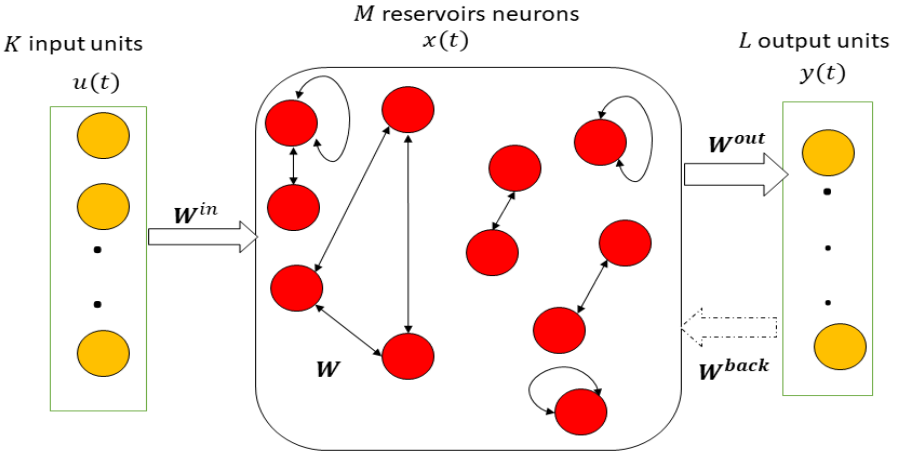


FIGURE 1. The step-wise architecture of Echo State Network comprises varying numbers of input units, reservoirs neurons,s, and output units.

performance. The particles communicate by employing Newton's theory of gravity and the laws of action. Considering a solution contains N particles (masses). Eq.4 calculates the i particle's position.

$$x_i = (x_i^1, \ldots..x_i^d \ldots\ldots x_i^D) \quad for \quad i = 1, 2, 3 \ldots .n \tag{4}$$

Here, x_i^d depicts the location of particle i in dimension d, and the total number of dimensions is denoted by D. Each particle's performance is determined by its mass and assessed by a vigor process. In this approach, the gravity and inertial masses of all particles are equalized and modernized with each iteration by using Eqs.5, 6, and 7

$$M_{ai} = M_{pi} = M_{ii} = M_i \tag{5}$$

$$m_i = \frac{fit_i - worst}{best - worst} \tag{6}$$

$$M_i = \frac{m_i}{\sum_{j=1}^{N} m_j} \tag{7}$$

where, fit_i indicates the particle i fitness value. $best$ and $worst$ represent all particles' highest and lowest fitness scores.

Maximization challenges are characterised as in Eqs.8, and 9

$$best = \max_{j\in\{1,\ldots..N\}} fit_j \tag{8}$$

$$worst = \min_{j\in\{1,\ldots....N\}} fit_j \tag{9}$$

Considering reducing issues, they are completely different and are calculated as in Eqs.10, and 11

$$best = \min_{j\in\{1,\ldots..N\}} fit_j \tag{10}$$

$$worst = \max_{j\in\{1,\ldots....N\}} fit_j \tag{11}$$

The gravity F_{ij}^d exerted on particle i from particle j is calculated as in Eq.12

$$F_{ij}^d = G\frac{M_{pi}\times M_{aj}}{R_{ij}+\varepsilon}\times(x_j^d - x_i^d) \tag{12}$$

where M_{pi} sedentary gravity potential of the particles i. M_{aj} is kinetic gravity energy of particle j,. R_{ij} represents the Euclidean space within two particles, while ε is a teeny invariant. G is designated the gravity acceleration, although it is a sequence of rounds in Eq.13.

$$G = G_0 e^{-\alpha\frac{t}{T}} \tag{13}$$

G_0 and α are adjusted at the beginning and reduced gradually to regulate the search precision, while T denotes the max iterations. The force exerted operating on particle i in size d is an arbitrary weight matrix of other particles' gravitational forces.

$$F_i^d = \sum_{j\in Kbest, j\neq i} rand_j F_{ij}^d \tag{14}$$

In Eq.14, $rand_j$ is a constant arbitrary variable between 0 and 1. Keeping equilibrium is essential to prevent being stuck in local optima and strike a symmetry within exploration and exploitation during the search process. Solely particles $Kbest$ with the most significant fitness weights are used to wield gravitational attraction on other particles. $Kbest$ is initially set to the overall inhabitants and is progressively lowered to 1 with each iteration.

$$Kbest = N \times \frac{per+(1-\frac{t}{T})\times(100-per)}{100} \tag{15}$$

In Eq.15, per indicates particles' proportion that effectively contributes to distinct particles in the final analysis. Using the equation of motion, the rate of particle i in size d at iteration t is determined by the following equation, as shown in Eq.16.

$$a_i^d = F_i^d / M_{ii} \tag{16}$$

Here M_{ii} represents particle's i inertial mass.

The particle's velocity at the next dimension d is the proportion of the present speed and velocity.

$$v_i^d = rand_i \times v_i^d + a_i^d \tag{17}$$

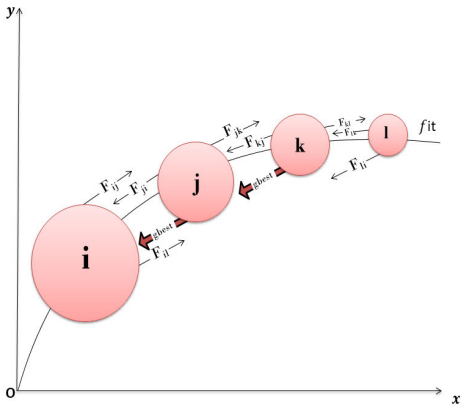


FIGURE 2. Global memory.

In Eq.17, $\boldsymbol{rand_i}$ an invariant arbitrary variable inside an interval [0, 1] is used to give the search an arbitrary characteristic. Additionally, Eq.18 computes the particle's next location in dimension $\boldsymbol{d}$.

$$x_i^d = x_i^d + v_i^d \tag{18}$$

C. IMPROVED GRAVITATIONAL SEARCH ALGORITHM

This section presented the enhanced exploitation capabilities of the updated version of the robust GSA architecture. The primary goal of heuristic algorithms is to locate the approximate optimal solution within a tolerable time frame. A strategy for attaining this aim is to balance exploitation and exploration.

D. GLOBAL MEMORY

The particles explore the search space in the initial iterations in the proposed approach. During subsequent rounds, particles choose the optimal global solution among viable alternatives. Particles get denser and move more slowly as iterations rise as they approach viable solutions. As a result, the algorithm's convergence rate declines during the exploitation phase. Furthermore, the original method lacks the memory to retain the best solution thus far. Consequently, it may be destroyed if the most solemn molecule with the most significant fitness value attracts additional particles. To address these shortcomings, the proposed method includes a global memory ***gbest***, which retains the best solution achieved thus far. All particles are conscious of their location and may move toward it. The introduced technique is illustrated in Figure 2.

It illustrates a straightforward 1-D devaluation issue, where ***fit*** represents a fitness process with a threshold level of 0. According to the Figure 2, particle ***i*** attracts particles j and k. During the same period, these two particles attracted particle ***i***, causing it to diverge slightly from the global optimum. If weighty particles are near the optimal global solution but unable to move toward it, they will drift toward the centroid of surrounding particles. Consequently, the global memory ***gbest*** is implemented to avert particles from becoming immobile under inadequate conditions. This enhances the mobility of heavy particles and the algorithm's capacity to exploit resources. As an instance of intellectual capability, ***gbest*** yields each particle an extra velocity feature that moves it toward the position of the particle with the most mass. Thus, the formula for updating velocity is adjusted in Eq.19

$$.v_i^d = rand_i \times v_i^d + c_1 \times a_i^d + (2 - c_1) \times \left(x_{\text{gbest}}^d - x_i^d\right) \tag{19}$$

where $\boldsymbol{c_1}$ is accelerating factor, and $\boldsymbol{x_{gbest}^d}(t)$ is the location of ***gbest*** in dimension d. The equation's front is comparable to the original method to preserve the program's capacity for exploration. Additionally, the bottom of the equation is introduced to aid particles in pursuing optimal overall development throughout the exploitation phase. Due to the lack of obvious distinction between the exploration and exploitation phases in heuristic approaches, $\boldsymbol{c1}$ is required to achieve a balance between them [24]. It decreases with each repetition, as seen by Eq.20.

$$c_1 = 2 - 2 \times \frac{t^3}{T^3} \tag{20}$$

here $\boldsymbol{t}$ depicts the current iteration, and symbolizes the entire iterations performed. The influence of ***gbest*** on particles is irrespective of their masses and is unrestricted by the rule of sobriety, prohibiting particles from collecting and traveling gradually. This update preserves exploration capabilities in earlier iterations, enriches exploitation capabilities in later iterations, and feeds a seamless transition between the two.

Algorithm 1 Improved GSA

Input : Initialize the algorithm's relevant parameters (N, G_0, α, t_{max})
Output: Optimal solution

Develop the initial population erratically.
while $i = 1$ *to max iterations* **do**
    Assess the fitness weight of all particles applying the fitness function.
    Compute the Worst and Best.
    Compute the masses of all particle using Eq. 7
    Reconfigure the global memory.
    Compute the strength and acceleration of every particle in various proportions as per the specified Eqs. 14 and 16.
    Update velocity V_i^d using Eq. 19
    Update position x_i^d using Eq. 18
end
return S

E. BINARY IGSA

In contrast, this study proposes a binary IGSA related to the original BGSA [43]. The position of each particle represents a subset with potential solutions, as in the PSO method context.

The outcome is that it is a binary vector. Each dimension's outcome is either 1 or 0, indicating the existence or absence of the corresponding characteristic. Each particle's orientation is initialized with a sequence of binary numbers at the start of BIGSA. During a dimension transformation, its value transitions between 1 and 0. In BIGSA, the particle velocity is updated using IGSA, and a probability process is employed to convert the velocity to likelihood is described in Eq.21:

$$S\left(v_i^d\right) = |\tanh(v_i^d)| \tag{21}$$

where $\boldsymbol{S}\left(\boldsymbol{v_i^d}\right)$ is restricted within the interval [0, 1]. As the velocity rises, the probability will increase. Fluctuations in probability determine the trajectory of a particle in BIGSA. To update the location of each particle, the computed likelihood is taken into account, along with a specific rule.

$$\begin{aligned} &if\ rand < S(v_i^d)\ then \\ &x_i^d = complement\ \left(x_i^d\right) \\ &else \\ &x_i^d = x_i^d \end{aligned} \tag{22}$$

In Eq.22 ***rand*** is a uniformly dispersed arbitrary integer between 0 and 1.

III. BIGSA-ESN

This study employs the BIGSA technique to enhance the output connection framework of ESN by eliminating repetitious links within the reservoir and output layer. This is achieved by setting certain network weights in $\boldsymbol{W^{out}}$ to zero, in addition to the standard calculation of connection weights. The output connectivity of ESN can be represented using 1 and 0, where disconnection and connection are indicated by 1 and 0, respectively. By optimizing the parameter, a binary matrix can be constructed that corresponds to the ESN output connection state. This matrix's dimensions typically equal the numeral of neurons in the reservoir, denoted as I. BIGSA particles provide output weights in the $\boldsymbol{W^{out}}$ matrix. RMSE measures each particle's efficacy.

$$RMSE = \frac{1}{N}\sqrt{\sum_{i=1}^{N}(y\,(i) - y_o(i))^2} \tag{23}$$

Eq.23 $\boldsymbol{y\,(i)}$, $\boldsymbol{y_o(i)}$ The predicted and target output are denoted as well as $\boldsymbol{N}$, which represents the size of the training samples. A smaller RMSE corresponds to higher network training or testing accuracy. Since RMSE is an objective function, a smaller RMSE solution offers the optimal solution.

The procedure for designing BIGSA-ESN is outlined as follows:

1) The BIGSA optimization algorithm is initialized using G_0 and α. Here, N denotes the number of exploring particles, and T denotes the highest number of iterations.
2) Initialize the ESN by specifying the reservoir size to $\boldsymbol{M}$ and randomly generating the Input weights $\boldsymbol{W^{in}}$ and internal weights $\boldsymbol{W}$, which will not change throughout further training. The reservoir construction is performed with an appropriately sized architecture, where the ***tanh*** function is utilized as the internal activation process. The sparsity of the reservoir is retained between 1% and 5%. To maintain the ESN model's stability, the weight matrix's spectral radius should be less than or equal to 1.
3) Determine the objective function for every searching particle. This study utilizes the ESN result association to define the particle's position. The RMSE, as shown in Eq.23, is utilized as the objective process for each projected and target value, and the goal of the search is to minimize this value.
4) Calculate the mass of each particle using Equation Eq.7
5) Calculate the gravity coefficient G using equation Eq.13.
6) Determine each particle's force using the equation Eq.14; determine each particle's accelerate rate using the equation Eq.16.
7) Update the global memory
8) Using BINARY IGSA, update the position and velocity according to Eq.22.
9) Perform iterative optimization. **BIGSA** continually updates and modifies the particle's location until achieving the highest number of iterations or the desired level of precision, such as when the optimal connection weight vector has been determined.
10) Once the link weight is optimized, it is adapted to the ESN, and the efficacy of the improved ESN is then assessed.

IV. RESULT ANALYSIS BASED ON BENCHMARK DATABASE

A. ENVIRONMENT CONFIGURATION

The experiments conducted in this study are conducted on a machine integrated with Intel(R) Core(TM)i7-9750H, operating at 2.70 GHz, and a GTX2160Ti GPU. All approaches are developed using Matlab version 2016a and MySQL database devices to ensure standardized training and testing performance.

B. PREPARATION AND PREPROCESSING OF THE DATA

The experimental data for this study were acquired from a short-scale wastewater treatment plant located in Beijing, China. The data set consists of 1,000 input-output samples extracted from the treatment plant's operating report spanning from June 2016 to July 2022. The first 900 instances are utilized for training, while the remaining 100 are for validation. The model uses water temperature (T), dissolved oxygen (DO) concentration, effluent pH, effluent oxidation-reduction potential (ORP), and effluent nitrate nitrogen (NO3-N) concentration as auxiliary variables of the **BIGSA-ESN** to signify the WWTP effluent NH4-N. Since the order of magnitude of wastewater data varies significantly, the ESN model is susceptible to data scaling. To address potential issues such

Algorithm 2 BIGSA-ESN Algorithm

Input: M, The number of reservoir nodes; T: Maximum number of iterations; N: Population size; any other relevant parameters;
Output: Trained BIGSA-ESN;

```
Randomly initialize T and the population of N search particle's positions from the interval [0, 1];
Randomly initialize an ESN, including setting the value of M and generating the input weight matrix W^in, the reservoir weight matrix W and the feedback weight matrix W^back;
while the stopping criterion is not met do
    for each search particle i in population do
        Set the output connection of the ESN to the position of search particle i;
        Compute the output weight matrix, W^out;
        Estimate the objective function for search particle i
        Predict the output of ESN for the given input;
        Compute the root mean square error and the mean absolute error between the predicted and actual values;
        Use the minimum value of the objective function as the search target;
    end
    Calculate the mass of each search particle M_i using Eq.7;
    Compute the gravitational force F_i^d and acceleration a_i^d of every particle in various proportions as per the specified Eq.14; and Eq.16;
    Update global memory;
    Update the velocity V_i^d and position x_i^d of each search particle using Eq.19; and Eq.18;
    Using BINARY IGSA, update the position and velocity according to Eq.22;
end
Obtain the optimal W^out weight matrix based on the best search particle position found during the search;
```

as slower convergence of the model caused by variations in the magnitude of the data, the wastewater data must be normalized using a suitable scaling method, as in Eq.24.

$$\acute{x_i} = (x_i - x_{i,\ \min})/(x_{i,\ \max} - x_{i,\ \min}) \tag{24}$$

where $\acute{x_i}$, $x_{i,min}, x_{i,max}$ are the normalized wastewater data value, value, minimum value, and maximum value of each dimensional data x_i. Each data sample is placed into the range [0, 1].

C. SETTING THE MODEL PARAMETER

The appropriate configuration of the reservoir size (N) and connectivity rate is crucial in the network architecture of BIGSA-ESN during the training method. It depends on the specific tasks being performed. The remaining parameters, including reservoir sparseness (SP) and spectral radius (SR) of the inner link matrix, are set according to the methodology proposed by [20]. They are explicitly specified as sparseness. They are explicitly specified as follows: the SP of W is selected as 5%, and the SR is set to 0.8. The weight matrices W and W^{in} are not modified and initialized with random values before training. The weights of all elements in W and W^{in} are assigned randomly within the range of -1 to 1.

The efficacy of the offered strategy is highly dependent on the parameters used in the BIGSA method. These parameters are determined through a sequence of investigations or following current literature. The inhabitant's dimension is adjusted to 20, the maximum iterations are set to 200, G_0 is fixed to 100, and α is fixed to 20. Each study is carried out ten times to reduce the influence of arbitrary initialization of certain system parameters, and the overall average is recorded.

D. THE CUTTING-EDGE APPROACHES

This section provides an overview of the relevant methodologies employed in this study. It encompasses the ARIMA model, the extreme learning machine (ELM), the Classical Radial Basis Function (RBF) model, and the Simple Recurrent Deep Neural Networks (SRDNNs) model. Additionally, it delves into a comprehensive explanation of the proposed ESN optimized by an improved BGSA algorithm.

1) ARIMA MODEL

The ARIMA method is a widely-adopted time series forecasting method that integrates three key elements: autoregressive, integrated, and moving average. The autoregressive component estimates current values based on a linear combination of previous values. The integrated part makes the time series stationary by differencing data points, which is essential for dealing with trends or seasonality. The moving average aspect utilizes a weighted average of past errors to enhance accuracy. In water waste forecasting, ARIMA is instrumental in making predictions by using historical data, identifying trends, and accounting for errors. This enables effective planning and policy-making for water management by anticipating seasonal changes or the impact of events on water waste.

2) ELM MODEL

The ELM algorithm is a variant of single-layer feed-forward neural networks known for its unique and efficient training process. It randomly assigns weights to the input layer, and in the second phase, employs the Moore-Penrose inversion technique to calculate weights for the output layer using the hidden layer's output matrix. This two-stage process enables ELM to efficiently handle non-linear problems and produce accurate models. In water waste forecasting, ELM's speed and accuracy make it invaluable for modeling intricate relationships between historical data, weather conditions, and infrastructure, thereby facilitating more informed water management strategies.

3) CLASSICAL RBF MODEL

The conventional RBF model represents a neural network that employs radial basis functions as activation functions Radial Basis Function Neural Networks (RBFNNs) employ radial basis functions, usually Gaussian, as activation functions and are particularly effective in learning and approximating non-linear relationships. With a three-layer structure (input, hidden, and output), RBFNNs are similar to other neural networks in architecture. What distinguishes RBFNNs is the way the hidden layer processes input data by transforming it into a higher-dimensional space, facilitating linear separability, which is advantageous for classification and regression tasks. In the hidden layer, radial basis functions calculate the "distance" between input data and certain reference points, generally cluster centers. The network's final output is generated by weighting and summing the outputs from these functions. For water waste forecasting, which involves complex, non-linear relationships, RBFNNs are valuable due to their ability to capture intricate patterns and associations in data, such as historical water consumption, weather conditions, and infrastructure status.

4) SIMPLE RECURRENT DEEP NEURAL NETWORKS (SRDNNs) MODEL

Short-term Recurrent Dynamic Neural Networks (SRDNNs) are particularly effective for predicting water waste due to their dynamic architecture and ability to recognize time-based patterns. SRDNNs consist of three layers: input, hidden, and output. The network's neurons are interconnected within and across layers, imparting short-term memory to the network by retaining past information. The input layer collects data such as historical water usage, sensor readings, and weather conditions. The hidden layer is central to the network's ability to identify temporal patterns, which is vital for water waste prediction. The output layer then converts this information into predictions, such as the estimated amount of water waste over time. Utilizing SRDNNs enables utility companies and authorities to effectively manage water resources through informed planning and monitoring, contributing to more sustainable practices.

E. PERFORMANCE ESTIMATION

This research used RMSE and the Mean Absolute Percentage Error (MAPE) among the anticipated and actual weights to evaluate the efficacy of the time series prediction approach. The formula for RMSE has been discussed in Eq.23 and the MAPE computational formula is depicted in Eq.25.

$$MAPE = \frac{1}{N}\sqrt{\sum_{i=1}^{N}\frac{|y(i) - y_o(i))|}{|y(i)}} \times 100\% \tag{25}$$

where N refers to the size of the training dataset. The efficacy of the suggested **BIGSA-ESN** approach will be assessed through the prognosis of two-time series problems: the forecast of the Lorenz system and the Mackey-Glass time series, as well as the forecast of the effluent NH4-N concentration in a WWTP. Also, several other algorithms are compared with our proposed model.

F. LORENZ TIME SERIES PREDICTION

The Lorenz system, which is an extensively employed benchmark task in time series forecasting, is described by a set of equations known as the Lorenz equations, as shown in Eq.26.

$$\begin{cases} \frac{d_x}{d_i} = a(-x + y) \\ \frac{d_y}{d_i} = bx - xz - y \\ \frac{d_z}{d_i} = xy - c \end{cases} \tag{26}$$

The model parameters for the Lorenz system, a classic benchmark task for time series prediction, are fixed to $a = 10$, $b = 28$, and $c = 8/3$. The Kutta-Runge strategy with a degree of 0.01 yields the Lorenz time series to calculate values. The model is trained using $y(k-3)$, $y(k-2)$, and $y(k-1)$ to forecast $y(k)$ for all learning parameters. The initial 2000 values are excluded to eliminate the starting point's impact. The samples of y-dimension with $i \in [2001, 4000]$ are elected as the learning dataset. The subsequent 1000 instances, with $i = 4001$ *to* 5000, are classified as the testing instances. The reservoir size (N) used in the Lorenz system forecast was set to 300. The rate of connection was set at 5%. To demonstrate the significance of **BIGSA-ESN**, the benchmark's output and the corresponding targets are plotted on the exact figure, showcasing the degree of overlap over them for a sequential structure from the Lorenz standard. Figure 3a–b illustrates these two signals for testing data samples. The RMSE evolution was tracked from the first pattern to the last one to assess the significance of the presented **BIGSA-ESN** approach. The resulting predicted RMSE signal for the testing samples was plotted in Figure 3a–b.

It is clearly illustrated that the margin of most testing errors for **BIGSA-ESN** is limited into the range [-0.02, 0.06], which implies the proposed method has better testing performance. Comparison based on RMSE with other methods including GA-ESN [33], BPSO-ESN [35], LAR-ESN [44], Extreme Learning Machine (ELM) [45], simple recurrent deep artificial neural network based PSO algorithm (SRDNN-PSO) [46], Classical RBF [47], ARIMA model [48], and the standard ESN applied for the Lorenz attractor. Based on the consequences presented in Table 1, it can be figured that the suggested **BIGSA-ESN** algorithm is highly competitive, achieving accurate predictions compared to other existing models. Although BPSO-ESN and GA-ESN outperformed ESN in the tests, they had a slower average runtime compared to BIGSA-ESN.

G. MACKAY GLASS (MG) TIME-SERIES PROGNOSIS

The benchmark for this time series, as described by Yang et al. [49], employs a time-delay differential structure

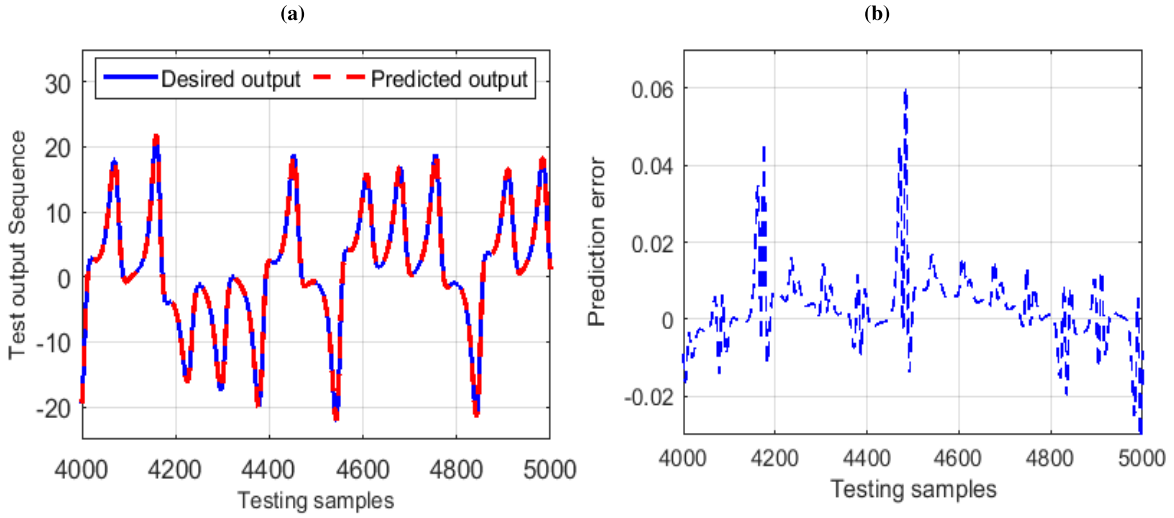


FIGURE 3. a-b). Prediction outcomes during the testing process.

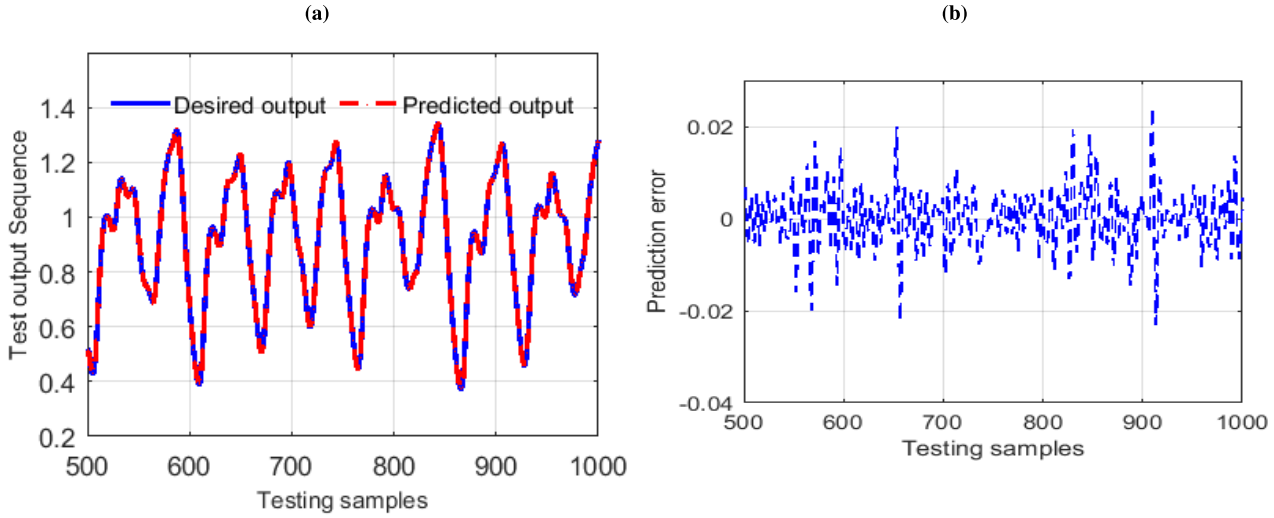


FIGURE 4. a) Projection outcomes during the testing method, b). Prognosis results during the testing method.

TABLE 1. Comparison of numerous approaches of 50 trails for (Lorenz time series).

Methods	Reservoir size (M)	Testing RMSE	Testing MAPE	CPU time(s)
BIGSA-ESN	300	0.0086	0.0059	450
BPSO-ESN	300	0.0182	0.0071	469
GA-ESN	300	0.0254	0.0120	1937
LAR-ESN	300	0.0287	0.0132	-
ESN	300	0.0298	0.0140	-
SRDNN-PSO	300	0.0302	0.0149	-
ELM	300	0.0376	0.0152	-
Classical RBF	-	0.0462	-	-
ARIMA model	-	0.0805	0.0193	-

similar to the one represented in Eq.27. This refers to a specific time series analysis approach incorporating time lags and differential equations to model complex dynamic systems. The reference to Eq.27 suggests that the article includes a specific equation or formula related to this approach.

$$\frac{dx(t)}{dt} = \frac{\propto . \; x\,(t-\tau)}{1+x^{c}\,(t-\tau)} - \beta . \, x(t) \tag{27}$$

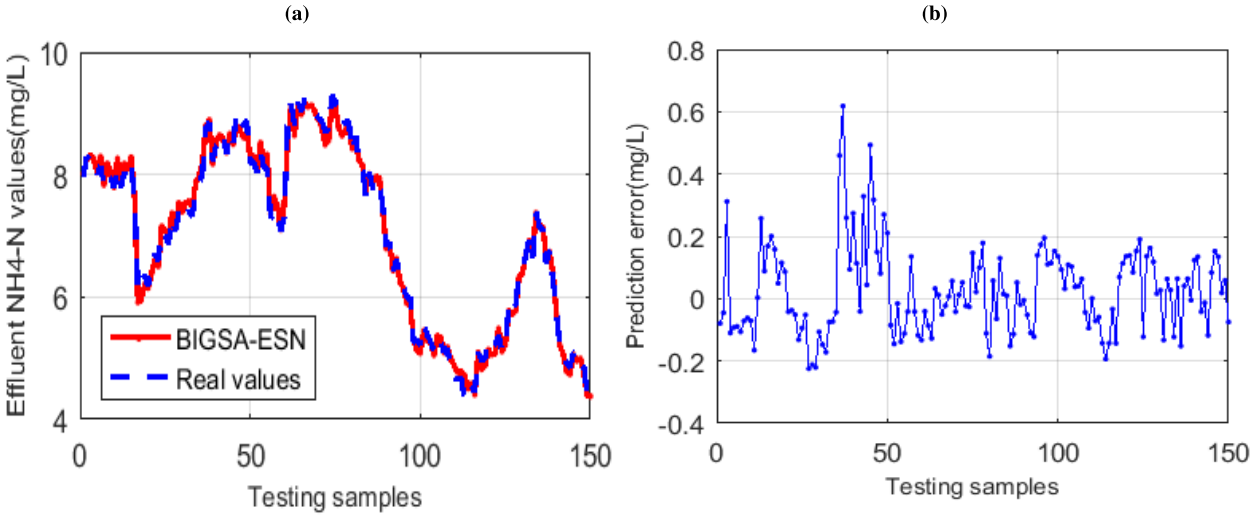


FIGURE 5. a). The prediction results of BIGSA-ESN for the testing samples are shown. b) The prediction errors of BIGSA-ESN for the testing samples are also presented.

The weights of the parameters $\alpha = 0.2$, $\beta = 0.1$, and $c = 10$ have been chosen for this system. Additionally, the parameter τ, a crucial factor in the MG system, has been extensively examined in prior work by Nezamabadi et al. [35]. This describes the specific parameter values used for the system in question and references a prior work that delves into a related parameter. The parameter τ, which controls the time delay in the system, can be adjusted and is typically set to values such as 17 or 30, as discussed in an analysis by Cherif et al. [16]. For the current study, a value of 17 was chosen to achieve optimal system performance. The data sets were generated using the Runge-second order Kutta method with a step size of 0.1. This analysis used one thousand instances, with 500 instances reserved for model training and 500 for testing. The reservoir size was adjusted to 200. We aim to demonstrate the significance of **BIGSA-ESN** by visualizing the relationship between the network results and their affiliated targets on a single graph. This graph will enable us to measure the similarity between predicted and actual values for a particular sequence structure from the MG time series standard.

Figure 5a displays a graphical representation of the two signals, namely the network results and the affiliated targets. In addition, we analyzed the evolution of the RMSE across all patterns in the test set. This analysis is shown in Figure 5b. The results demonstrate that most of the testing errors for **BIGSA-ESN** fall within the range of [-0.02, 0.02], indicating the superior testing execution of the suggested strategy. The empirical determination is presented in Table 2, which illustrates the effect of the training process on the errors within the network and target output. The Table 2 demonstrates that the optimization of weights considerably impacts test errors. Additionally, we compared the interpretation of the presented strategy with other existing methods to assess its efficacy. These methods have used different techniques, such as GA-ESN [33], BPSO-ESN [35], LAR-ESN [44], ELM [45], SRDNN-PSO [46], Classical RBF [47], ARIMA model [48], and the standard ESN. The outcomes presented in Table 2 exhibit that the presented method surpasses the cutting-edge methods in terms of testing RMSE and testing MAPE. Moreover, GA-ESN has a longer runtime compared to BIGSA-ESN and BPSO-ESN due to the GA algorithm's requirement for a larger population and a greater number of iterations.

H. TESTED USING REAL-WORLD EFFLUENT $NH_4 - N$ PREDICTION IN WWTP DATASET

Effluent $NH_4 - N$, the most critical water differentia parameters in the WWTP, is usually used to evaluate treatment efficiency. Excessive effluent $NH_4 - N$ will cause deterioration of the water environment, resulting in the eutrophication of water bodies. Therefore, seeing flowing $NH_4 - N$ in the WWTP helps take necessary measures in time to ensure treatment efficiency. Measuring the effluent $NH_4 - N$ can be challenging because of the biological properties of the activated sludge reaction approach. The existing instrumentation is expensive, problematic to maintain, and tedious to operate. It is challenging to meet the demand for online detection of effluent $NH_4 - N$ in the WWTP.

This study used the presented **BIGSA-ESN** to signify the effluent $NH_4 - N$ in WWTPs. The soft-sensing system designed for this purpose is illustrated in Figure 6, where the infusion variables are easily measurable, while the outcome variable is the effluent $NH_4 - N$. In this experiment, the input variables for predicting the effluent $NH_4 - N$ in WWTPs are the water temperature (T), dissolved oxygen (DO) concentration, effluent pH, effluent oxidation-reduction potential (ORP), and effluent nitrate nitrogen ($NO_3 - N$) concentration. Out of the 750 input-output data samples available, 600 instances are utilized for training, and 150 instances are

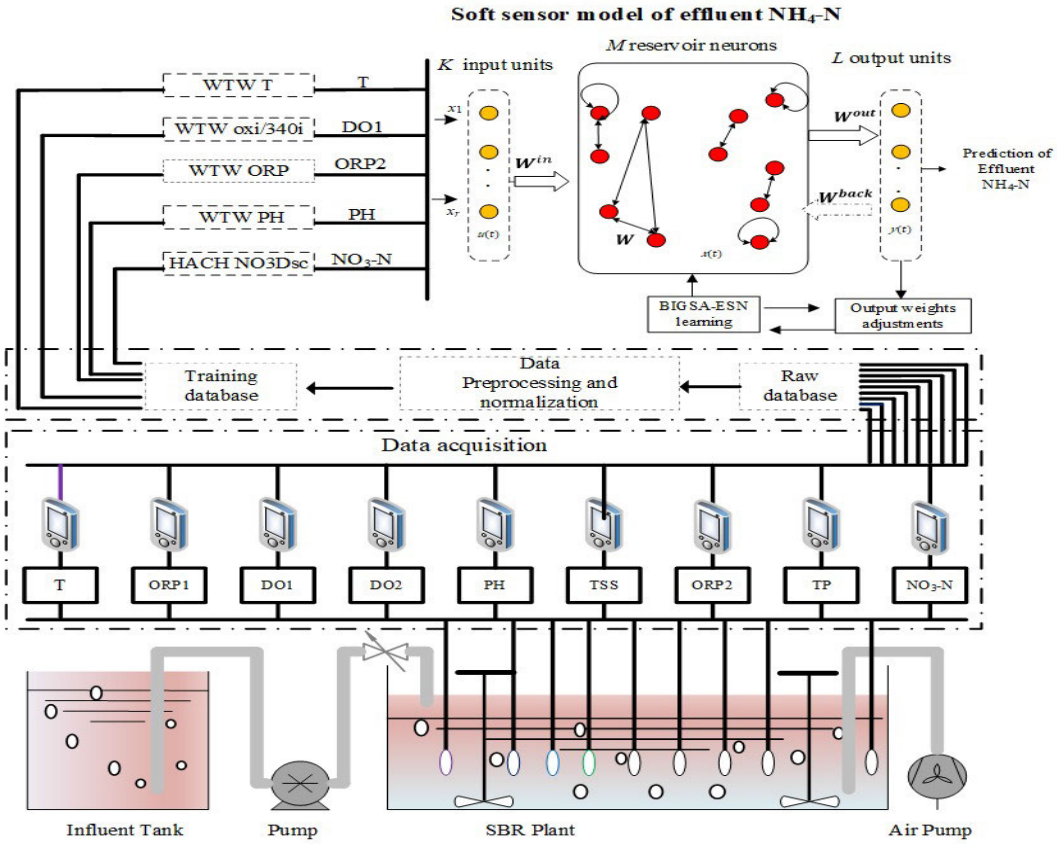


FIGURE 6. Representation of the BIGSA-ESN based prediction model of effluent $NH_4 - N$.

TABLE 2. Comparison of numerous approaches of 50 trails for (Mackey-Glass time-series).

Methods	Reservior Size (M)	Testing RMSE	Testing MAPE	CPU Times (s)
BIGSA-ESN	200	0.0064	0.0056	372
BPSO-ESN	200	0.0107	0.0076	388
GA-ESN	200	0.0127	0.0070	1705
LAR-ESN	200	0.0131	0.0090	-
ESN	200	0.0165	0.0142	-
SRDNN-PSO	200	0.0179	0.0173	-
ELM	200	0.0214	0.0184	-
Classical RBF	-	0.0142	-	-
ARIMA model	-	0.0277	0.0234	-

kept aside for testing. The reservoir size has been adjusted to 200, and the connection rate is specified at 5%. The testing RMSE and testing accuracy are used to measure performance. The mathematical formula for testing accuracy is described as in Eq. 28.

$$Accuracy = \frac{1}{N}\sqrt{\sum_{i=1}^{N} 1 - \frac{|y(i) - y_o(i))|}{|y(i)} \times 100\% \quad (28)$$

The modeling curves for testing data samples 0-150 of BIGSA-ESN, BPSO-ESN, and GA-ESN are compared in Figure 7a–b, while the corresponding modeling error for testing data samples is compared in Figure 7c–d. It can be found that all methods can fit the output well, but BIGSA-ESN has a better fitting effect at some infection points. As a remark, it is clearly illustrated that the margin of most testing errors for BIGSA-ESN is limited to the range [-0.2, 0.6] mg/L, the margin of most testing errors for BPSO-ESN is limited to the range [-0.1] mg/L, and the margin of most testing error for GA-ESN is limited to the range [-1, 1.5] mg/L, which implies the proposed method BIGSA-ESN has better prediction effect than other evolutionary strategies. The relation between **BIGSA-ESN** with other existing methods ESN along with the testing and training RMSE values is shown in Table 3. All **BIGSA-ESN** training and testing RMSE weights are lower than GA-ESN [33], BPSO-ESN [35], HPSO-ESN [50], Mathematical Method [51] and the standard ESN, suggesting that the

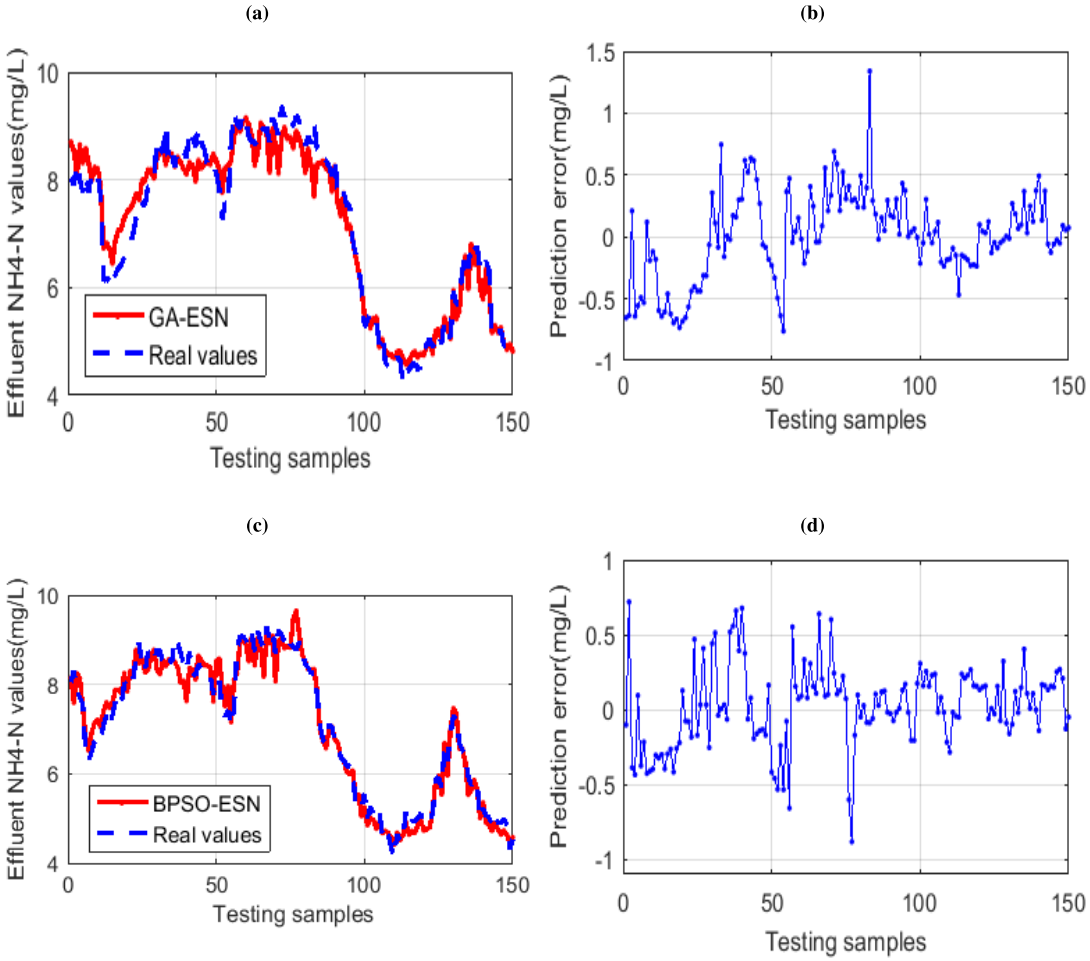


FIGURE 7. a-b). Prediction results and of GA-ESN for the testing samples, c-d) Prediction results and errors of BPSO-ESN for the testing samples.

TABLE 3. Comparison of numerous approaches of 50 trials for (effluent NH4-N concentration prediction in WWTP.

Methods	Reservoir size (M)	Testing RMSE	Testing MAPE	Accuracy (%)
BIGSA-ESN	200	0.104	0.117	98.07
BPSO-ESN	200	0.170	0.244	94.74
GA-ESN	200	0.210	0.246	95.56
HPSO-ESN	200	0.421	0.471	93.15
Mathematical model	-	-	0.635	90.52
ESN	200	0.605	0.644	90.87

proposed system has improved predictive efficiency. The results obtained from the experiments confirm that **BIGSA-ESN** can efficiently forecast the effluent NH_4-N in WWTPs, providing a precise, trustworthy, and online soft computing approach to detect critical variables. The efficacy of the presented approach outperformed other current models, demonstrating its capability of online prediction in real-time applications.

V. CONCLUSION

The proposed strategy seeks to solve the problem of duplicate connectivity between the reservoir and output layer in ESNs. To do so, a combination of ESN and BIGSA is used, where the data is first used to train the ESN reservoir, and then BIGSA optimizes the output connection. The presented scheme predicts two-time series models and a key parameter in the wastewater treatment strategy dataset. The interpretation of BIGSA-ESN has corresponded to GA and BPSO, and the results indicate that it reduces generalization error compared to conventional ESNs. Future work could explore testing the proposed method on other benchmarks to test its robustness and apply it to Deep ESN, a network incorporating multiple smaller ESNs.

CONFLICT OF INTEREST

The authors declare that they have no Conflict of interest.

ACKNOWLEDGMENT

This work was supported by Artificial Intelligence and Data Analytics (AIDA) Lab, CCIS Prince Sultan University, Riyadh, Saudi Arabia. The authors would also like to acknowledge the support of Prince Sultan University for paying the Article Processing Charges (APC) of this publication.

REFERENCES

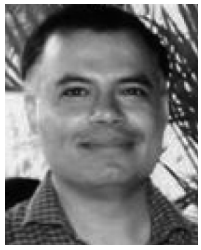
[1] L. Wang, H. Hu, R. Liu, and X. Zhou, "An improved differential harmony search algorithm for function optimization problems," *Soft Comput.*, vol. 23, no. 13, pp. 4827–4852, Jul. 2019.
[2] H.-W. Peng, S.-F. Wu, C.-C. Wei, and S.-J. Lee, "Time series forecasting with a neuro-fuzzy modeling scheme," *Appl. Soft Comput.*, vol. 32, pp. 481–493, Jul. 2015.
[3] P. Esling and C. Agon, "Time-series data mining," *ACM Comput. Surv.*, vol. 45, no. 1, pp. 1–34, 2012.
[4] Y.-R. Zeng, Y. Zeng, B. Choi, and L. Wang, "Multifactor-influenced energy consumption forecasting using enhanced back-propagation neural network," *Energy*, vol. 127, pp. 381–396, May 2017.
[5] R. Liu, Y.-R. Zeng, H. Qu, and L. Wang, "Optimizing the new coordinated replenishment and delivery model considering quantity discount and resource constraints," *Comput. Ind. Eng.*, vol. 116, pp. 82–96, Feb. 2018.
[6] T.-C. Fu, "A review on time series data mining," *Eng. Appl. Artif. Intell.*, vol. 24, no. 1, pp. 164–181, Feb. 2011.
[7] S. Akhtar, M. Ramzan, S. Shah, I. Ahmad, M. I. Khan, S. Ahmad, M. A. El-Affendi, and H. Qureshi, "Forecasting exchange rate of Pakistan using time series analysis," *Math. Problems Eng.*, vol. 2022, pp. 1–11, Aug. 2022.
[8] S.-X. Lv, L. Peng, and L. Wang, "Stacked autoencoder with echo-state regression for tourism demand forecasting using search query data," *Appl. Soft Comput.*, vol. 73, pp. 119–133, Dec. 2018.
[9] Z. Ahmad, J. Li, and T. Mahmood, "Adaptive hyperparameter fine-tuning for boosting the robustness and quality of the particle swarm optimization algorithm for non-linear RBF neural network modelling and its applications," *Mathematics*, vol. 11, no. 1, p. 242, Jan. 2023.
[10] L. Wang, S.-X. Lv, and Y.-R. Zeng, "Effective sparse AdaBoost method with ESN and FOA for industrial electricity consumption forecasting in China," *Energy*, vol. 155, pp. 1013–1031, Jul. 2018.
[11] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Netw.*, vol. 2, no. 5, pp. 359–366, Jan. 1989.
[12] S. F. Crone, M. Hibon, and K. Nikolopoulos, "Advances in forecasting with neural networks? Empirical evidence from the NN3 competition on time series prediction," *Int. J. Forecasting*, vol. 27, no. 3, pp. 635–660, Jul. 2011.
[13] L. Wang, Z. Wang, H. Qu, and S. Liu, "Optimal forecast combination based on neural networks for time series forecasting," *Appl. Soft Comput.*, vol. 66, pp. 1–17, May 2018.
[14] T. Mahmood, J. Li, Y. Pei, F. Akhtar, S. A. Butt, A. Ditta, and S. Qureshi, "An intelligent fault detection approach based on reinforcement learning system in wireless sensor network," *J. Supercomput.*, vol. 78, no. 3, pp. 3646–3675, Feb. 2022.
[15] T. Saba, A. Rehman, K. Haseeb, T. Alam, and G. Jeon, "Cloud-edge load balancing distributed protocol for IoE services using swarm intelligence," *Cluster Comput.*, vol. 2023, pp. 1–11, Jan. 2023.
[16] A. Cherif, H. Cardot, and R. Boné, "SOM time series clustering and prediction with recurrent neural networks," *Neurocomputing*, vol. 74, no. 11, pp. 1936–1944, May 2011.
[17] S. P. Chatzis and Y. Demiris, "The copula echo state network," *Pattern Recognit.*, vol. 45, no. 1, pp. 570–577, Jan. 2012.
[18] S. E. Lacy, S. L. Smith, and M. A. Lones, "Using echo state networks for classification: A case study in Parkinson's disease diagnosis," *Artif. Intell. Med.*, vol. 86, pp. 53–59, Mar. 2018.
[19] Q. Cao, B. T. Ewing, and M. A. Thompson, "Forecasting wind speed with recurrent neural networks," *Eur. J. Oper. Res.*, vol. 221, no. 1, pp. 148–154, Aug. 2012.
[20] H. Jaeger, "The 'echo state' approach to analysing and training recurrent neural networks-with an erratum note," German Nat. Res. Center for Inf. Technol., Bonn, Germany, GMD Tech. Rep. 34, 2001, vol. 148, p. 13.
[21] T. Saba, A. Rehman, K. Haseeb, S. A. Bahaj, and R. Damaševičius, "Sustainable data-driven secured optimization using dynamic programming for green Internet of Things," *Sensors*, vol. 22, no. 20, p. 7876, Oct. 2022.
[22] J. Liu, T. Sun, Y. Luo, Q. Fu, Y. Cao, J. Zhai, and X. Ding, "Financial data forecasting using optimized echo state network," in *Proc. Int. Conf. Neural Inf. Process.* Cham, Switzerland: Springer, 2018, pp. 138–149.
[23] N. Chouikhi, B. Ammar, N. Rokbani, and A. M. Alimi, "PSO-based analysis of echo state network parameters for time series forecasting," *Appl. Soft Comput.*, vol. 55, pp. 211–225, Jun. 2017.
[24] Z. Ahmed, M. Memon, A. Memon, P. Munshi, and M. Memon, "Echo state network optimization using hybrid-structure based gravitational search algorithm with square quadratic programming for time series prediction," *Int. Arab J. Inf. Technol.*, vol. 19, no. 3A, pp. 530–535, 2022.
[25] A. Rodan and P. Tino, "Minimum complexity echo state network," *IEEE Trans. Neural Netw.*, vol. 22, no. 1, pp. 131–144, Jan. 2011.
[26] T. Saba, A. R. Khan, T. Sadad, and S.-P. Hong, "Securing the IoT system of smart city against cyber threats using deep learning," *Discrete Dyn. Nature Soc.*, vol. 2022, pp. 1–9, Jun. 2022.
[27] X. Dutoit, B. Schrauwen, J. Van Campenhout, D. Stroobandt, H. Van Brussel, and M. Nuttin, "Pruning and regularization in reservoir computing," *Neurocomputing*, vol. 72, nos. 7–9, pp. 1534–1546, Mar. 2009.
[28] H.-U. Kobialka and U. Kayani, "Echo state networks with sparse output connections," in *Proc. Int. Conf. Artif. Neural Netw.* Cham, Switzerland: Springer, 2010, pp. 356–361.
[29] J. Liu, T. Sun, Y. Luo, S. Yang, Y. Cao, and J. Zhai, "Echo state network optimization using binary grey wolf algorithm," *Neurocomputing*, vol. 385, pp. 310–318, Apr. 2020.
[30] A. ElShafee, W. El-Shafai, A. D. Algarni, N. F. Soliman, and M. H. Aly, "Statistical time series forecasting models for pandemic prediction," *Comput. Syst. Sci. Eng.*, vol. 47, no. 1, pp. 349–374, 2023.
[31] T. Saba, A. Rehman, K. Haseeb, S. A. Bahaj, and G. Jeon, "Energy-efficient edge optimization embedded system using graph theory with 2-tiered security," *Electronics*, vol. 11, no. 18, p. 2942, Sep. 2022.
[32] J. Liu, X. Huang, Y. Huang, Y. Luo, and S. Yang, "Multi-objective spiking neural network hardware mapping based on immune genetic algorithm," in *Proc. Int. Conf. Artif. Neural Netw.* Cham, Switzerland: Springer, 2019, pp. 745–757.
[33] W. Guo, M. Jiang, X. Li, and B. Ren, "Using a genetic algorithm to improve oil spill prediction," *Mar. Pollut. Bull.*, vol. 135, pp. 386–396, Oct. 2018.
[34] G. Shao, Y. Shangguan, J. Tao, J. Zheng, T. Liu, and Y. Wen, "An improved genetic algorithm for structural optimization of Au–Ag bimetallic nanoparticles," *Appl. Soft Comput.*, vol. 73, pp. 39–49, Dec. 2018.
[35] H. Nezamabadi-Pour, M. Rostami-Shahrbabaki, and M. Maghfoori-Farsangi, "Binary particle swarm optimization: Challenges and new solutions," *CSI J. Comput. Sci. Eng.*, vol. 6, no. 1, pp. 21–32, 2008.
[36] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "GSA: A gravitational search algorithm," *Inf. Sci.*, vol. 179, no. 13, pp. 2232–2248, Jun. 2009.
[37] K. M. Fouad, M. M. Ismail, A. T. Azar, and M. M. Arafa, "Advanced methods for missing values imputation based on similarity learning," *PeerJ Comput. Sci.*, vol. 7, p. e619, Jul. 2021.
[38] H. Zandevakili, E. Rashedi, and A. Mahani, "Gravitational search algorithm with both attractive and repulsive forces," *Soft Comput.*, vol. 23, no. 3, pp. 783–825, Feb. 2019.
[39] H. Mittal, R. Pal, A. Kulhari, and M. Saraswat, "Chaotic Kbest gravitational search algorithm (CKGSA)," in *Proc. 9th Int. Conf. Contemp. Comput. (IC3)*, Aug. 2016, pp. 1–6.
[40] S. Mirjalili, G.-G. Wang, and L. D. S. Coelho, "Binary optimization using hybrid particle swarm optimization and gravitational search algorithm," *Neural Comput. Appl.*, vol. 25, no. 6, pp. 1423–1435, Nov. 2014.
[41] L. Zhu, S. He, L. Wang, W. Zeng, and J. Yang, "Feature selection using an improved gravitational search algorithm," *IEEE Access*, vol. 7, pp. 114440–114448, 2019.
[42] H. Inbarani, A. T. Azar, K. M. Fouad, and S. F. Sabbeh, "Modified dominance-based soft set approach for feature selection," *Int. J. Sociotechnol. Knowl. Develop.*, vol. 14, no. 1, pp. 1–20, Jan. 2022.
[43] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "BGSA: Binary gravitational search algorithm," *Natural Comput.*, vol. 9, no. 3, pp. 727–745, Sep. 2010.
[44] H. Wang and X. Yan, "Optimizing the echo state network with a binary particle swarm optimization algorithm," *Knowl.-Based Syst.*, vol. 86, pp. 182–193, Sep. 2015.

[45] N. Garg, K. Soni, T. K. Saxena, and S. Maji, "Applications of AutoRegressive integrated moving average (ARIMA) approach in time-series prediction of traffic noise pollution," *Noise Control Eng. J.*, vol. 63, no. 2, pp. 182–194, Mar. 2015.
[46] E. Bas, E. Egrioglu, and E. Kolemen, "Training simple recurrent deep artificial neural network for forecasting using particle swarm optimization," *Granular Comput.*, vol. 7, no. 2, pp. 411–420, Apr. 2022.
[47] W. Guo, T. Xu, and Z. Lu, "An integrated chaotic time series prediction model based on efficient extreme learning machine and differential evolution," *Neural Comput. Appl.*, vol. 27, no. 4, pp. 883–898, May 2016.
[48] K. B. Cho and B. H. Wang, "Radial basis function based adaptive fuzzy systems and their applications to system identification and prediction," *Fuzzy Sets Syst.*, vol. 83, no. 3, pp. 325–339, Nov. 1996.
[49] C. Yang, J. Qiao, L. Wang, and X. Zhu, "Dynamical regularized echo state network for time series prediction," *Neural Comput. Appl.*, vol. 31, no. 10, pp. 6781–6794, Oct. 2019.
[50] Z. Ahmad, K. Nie, J. Qiao, and C. Yang, "Hybrid structure based PSO for ESN optimization," in *Proc. IEEE Symp. Product Compliance Eng.-Asia (ISPCE-CN)*, Oct. 2019, pp. 1–5.
[51] Z. Wang, J. Chu, Y. Song, Y. Cui, H. Zhang, X. Zhao, Z. Li, and J. Yao, "Influence of operating conditions on the efficiency of domestic wastewater treatment in membrane bioreactors," *Desalination*, vol. 245, nos. 1–3, pp. 73–81, Sep. 2009.

ZOHAIB AHMAD received the Ph.D. degree from the School of Electronics and Information Engineering, Beijing University of Technology, Beijing. He is currently with the Department of Computer Science and Information Technology, Sahara University, Narowal. His research interests include artificial intelligence, pattern recognition, machine learning, algorithms, intelligent computation, and intelligent optimization.

TARIQ MAHMOOD received the Ph.D. degree in software engineering from the Beijing University of Technology, China, and the master's degree in computer science from the University of Lahore, Pakistan. He is currently an Assistant Professor/the HOD of the Faculty of Information Sciences, University of Education, Vehari Campus, Vehari, Pakistan. He is also a renowned expert in image processing, healthcare informatics and social media analysis, ad-hoc networks, and WSN. He is contributed with more than 35 research articles in well-reputed international journals and conferences. He is an Editorial Member and a Reviewer of various journals, including *Plos One*, *Journal of Supercomputer*, *Journal of Digital Imaging*, *International Journal of Sensors*, and *Wireless Communications and Control*. His research interests include image processing, social media analysis, medical image diagnosis, machine learning, and data mining. He aims to contribute to interdisciplinary research of computer science and human-related disciplines.

TEG ALAM is presently working as a faculty member at the Industrial Engineering Department, College of Engineering, Prince Sattam Bin Abdulaziz University, Al Kharj, Kingdom of Saudi Arabia. His areas of expertise include Operations Research, Predictive Analysis, Multi-Objective Decision Making Approach, Data Science, and Statistical Analysis. In addition to publishing numerous types of research in reputed peer-reviewed journals, he has attended several prestigious national and international conferences. Additionally, he actively participates in community work.



AMJAD REHMAN (Senior Member, IEEE) received the Ph.D. and Postdoctoral degrees (Hons.) from the Faculty of Computing, Universiti Teknologi Malaysia, with a specialization in forensic documents analysis and security, in 2010 and 2011, respectively. He is currently a Senior Researcher with the Artificial Intelligence and Data Analytics Laboratory, College of Computer and Information Sciences (CCIS), Prince Sultan University, Riyadh, Saudi Arabia. He is the author of more than 200 ISI journal articles and conferences. Currently, he is a PI in several funded projects and also completed projects funded from MOHE, Malaysia, and Saudi Arabia. His research interests include data mining, health informatics, and pattern recognition. He received the Rector Award for the 2010 Best Student from Universiti Teknologi Malaysia.

TANZILA SABA (Senior Member, IEEE) received the Ph.D. degree in document information security and management from the Faculty of Computing, Universiti Teknologi Malaysia (UTM), Malaysia, in 2012. She is currently the Associate Chair of the Information Systems Department, College of Computer and Information Sciences, Prince Sultan University, Riyadh, Saudi Arabia, where she is also the Leader of the Artificial Intelligence and Data Analytics Research Laboratory. Her primary research interests include medical imaging, pattern recognition, data mining, MRI analysis, and soft-computing. She is an Active Professional Member of ACM, AIS, and IAENG organizations. She is also the PSU Women in Data Science (WiDS) Ambassador with Stanford University and the Global Women Tech Conference. She was a recipient of the Best Student Award from the Faculty of Computing, UTM, in 2012.