IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 6, 2020, accepted June 21, 2020, date of publication July 1, 2020, date of current version July 20, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3005247

A Hybrid Improved MVO and FNN for Identifying Collected Data Failure in Cluster Heads in WSN

THI-KIEN DAO[1], JIE YU[2], TRONG-THE NGUYEN[1,3], AND TRUONG-GIANG NGO[3]
[1]Fujian Provincial Key Laboratory of Big Data Mining and Applications, Fujian University of Technology, Fuzhou 350118, China
[2]College of Mechanical and Automotive Engineering, Fujian University of Technology, Fuzhou 350118, China
[3]Faculty of Computer Science and Engineering, Thuyloi University, Hanoi 100000, Vietnam

Corresponding authors: Trong-The Nguyen (vnthe@hpu.edu.vn) and Truong-Giang Ngo (giangnt@tlu.edu.vn)

This work was supported in part by the Fujian Provincial Buses and Special Vehicles Research and Development Collaborative Innovation Center Project under Grant 2016BJC012.

ABSTRACT This study proposes a new classifying approach for identifying collected failure data of cluster head (CH) in wireless sensor networks (WSN) based on hybridizing improved multi-verse optimizer (MVO) and feedforward neural network (FNN). An improvement of the MVO is proposed based on enhancing diversity agents for avoiding it's disadvanced of the local optimal. The data failure is detected for aggregating data in CH to forward to the base station (BS) based on classification by applying hybrid improved MVO and FNN. Twelve selected benchmark functions and the problem of identifying failure data in WSN are used in conducting comprehensive experiments to evaluate the performance of the proposed method. The experimental results are investigated and compared with the other approaches in the literature. The compared result exhibits the proposed technique that provides the alternative tool with the anticipation of influence on data sets and an effective way of forwarding the correct data from CH to BS in WSN applications.

INDEX TERMS Improved multi-verse optimizer, feedforward neural network, wireless sensor networks.

I. INTRODUCTION

The meta-heuristic algorithms have been one of the most considered active areas of research in the field of artificial intelligence (AI) for recent decades [1], [2]. The algorithms can be suitable for the sophisticated structural design of optimization problems due to their non-linear solution behavior [3], [4]. The object of algorithms is to find out the best-obtained values or/and suitable ways for solving complex challenges by evaluating the fitness functions under constraints [5], [6]. Almost meta-heuristic algorithms often developed based on inspiration from natural phenomena, e.g., animal behavior, social events, and physical facts [7]. It means the genetic processes of the algorithms govern the rules that are inspired by the natural expression, e.g., swarms, insects, or other facts to deal with global optimization problems [8]. For examples: Genetic algorithm (GA) [9] derived from Darwinian evolution, Particle swarm optimizer (PSO) [10], [11] mimicked the foraging behavior of flocks of birds, Ant colony optimization (ACO) [12] mimicked ants' behavior in searching and foraging for food in the nature that they leave pheromones as smelled chemicals on the path going, Grey wolves optimizer (GWO) [13] mimicked the hunting principles of gray wolves in nature and Multi-verse optimizer (MVO) [14] inspired by the multi-verse theory in physics. The primary reason for implementing evolutionary techniques on the vast amount of data to extract knowledge is the robustness and adaptive search-ability of global search methods [15]. In population-based as swarm intelligence, the stochastic manner multiplicity, messiness, and randomness often used to attribute to a kind of interpretive capability that appears in the communication of processing units. It can be described ability intelligence that can prove the solution successful in some ways [16]. The individuals or agencies in the meta-heuristic algorithms can be characterized as non-sophisticated solutions to have the ability to deal with complex problems [17].

The feed-forward neural network (FNN) [18] is the most common neural network that has been widely used for classification and approximations in practical applications [19]. The learning phase is an essential part of a neural network that can approximate its continuous and discontinuous functions. Most FNN applications use Back-propagation (BP) as training methods [18], [19]. BP is popularly applied in the neural networks that algorithm was developed based on a slope

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

scheme (gradient algorithm); however, it existed with several disadvantages, e.g., the tendency to fall into a local minimum and slow convergence [20]. In FNN, the goal during learning is to find the best combination of connection weights and deviations to achieve the smallest error.

However, most of the time, FNN converges to the point where the best solution is local rather than global optimal [21]. The factors such as the weights, biases (known as deviations), and parameter values are the most influential to the convergence of the BP algorithm [22]. It means that BP convergence depends on the optimization of the initial costs of parameters of weights and deviations. In the literature, the meta-heuristic algorithms are promising ways for a universal solution to deal with the problem of learning algorithms upon BP.

MVO is a new meta-heuristic optimizer inspired by the multi-verse theory to find the best planet in the universe as solutions based on phenomena physics of white holes, black holes, and wormholes for global optimization [14]. The MVO demonstrated to be a competitive optimizer in the literature [22]. It has been used widely in dealing with the optimization problem in the field of engineering and finance because of its advantages, e.g., simplicity with fewer control parameters, robustness, and outcome performance [23]. However, MVO still also has some drawbacks when dealing with complicated situations as the problem in identification collecting data failure in WSN.

The disadvantage of the MVO facing is such as premature convergence, search stagnation, or falling into the optimal local trap [3], [24]. The enhanced diversity planet of the universe is one of the best-suggested ways to overcome the mentioned deficiencies of the MVO. In this paper, we develop an improved new version of the MVO (namely IMVO) based on the enhanced diversity technique of the universe to overcome the mentioned cons of the MVO. The proposed IMVO would be validated a very competitive algorithm through some experimental results.

Moreover, applications of the wireless sensor networks (WSN) [25] have become widely popular spectrums of life, such as in smart homes, monitoring, surveillance, etc [26], [27]. WSNs consist of hundreds of deployed sensor nodes that require different metrics to maintain correctness transferring data in the system [28], [29]. WSN application of the smart home is a building integrating intelligent systems that control existing features with the adaptability to meet progression in terms of energy, efficiency, longevity, comfort, and satisfaction [30]. These terms of metrics are defined as the quality of service (QoS) of a WSN application that is the most used solution in apps dedicated to the buildings. The applications would have become more successful once they had been optimized in the deployment phase [31]. The optimization application is often implemented for the complicated problem by applying the meta-heuristic algorithm. Many WSN applications have also been implemented optimization successfully by using the meta-heuristic algorithms and machine learning, e.g., A combining machine learning and GA for WSN topology control [32], hybrid PSO and ACO for NN [33], hybrid PSO and BA for sensor net topology [34], FPA for optimizing node layouts [16], BA for unequal clustering [5], Naive Bayes classification for data correcting [35]. However, the studies of aggregating correctness data for Cluster heads (CH) in WSN seems the humbles.

Collecting data failure detection and classification for aggregating data inspection is one of the complicated issues that deployed in the CHs then transfer data to the base station (BS) of WSN. The importance of various parameters in the detector of a neural network is figured out according to application nature that is strongly affected by the optimization method [36].

In this paper, a new method of detecting collected failure data in cluster heads (CHs) in Wireless sensor networks (WSN) is proposed based on a hybrid Improved MVO and Feedforward neural network (FNN). The improved MVO (IMVO) is combined with FNN for increasing further detecting accuracy. Due to the improvement in IMVO performance makes a significant impact when solving the real-life classification problem or in training classifiers. The experimental results are compared with the other approaches to test the performance of the proposed method.

The contributions behind the proposed method are listed as follows:

- Improving the MVO algorithm (namely IMVO) based on enhancing its diversity planets population to balance the exploiting and exploring search space.
- Hybridizing the IMVO and FNN (namely NN-IMVO) for increasing further detecting accuracy and processing speed.
- Applying the NN-IMVO for identifying failure data in cluster heads (CHs) in WSN applications.

The rest of the structured paper organizes as follows. Section 2 reviews the MVO algorithm. The proposed improvement of MVO and the experimental results of the IMVO for the selected test functions are presented in Section 3. A hybridizing IMVO and FNN for identifying data failure in CHs in WSN are presented in Section 4. The final is a conclusion presented in Section 5.

II. CANONICAL MULTI-VERSE OPTIMIZER (MVO)

The MVO is a recent meta-heuristics algorithm that has been considered a promising optimization algorithm because of its successful implementation with fewer parameters [14]. The multi-verse theory with three main concepts of white holes, black holes, and wormholes were taken inspiration for MVO success. The black holes and white holes are considered to search for exploration strategy, while wormholes are used in search space for exploitation action.

There are some following rules in the optimization process of the MVO algorithm that needs to be obedient.

1. The higher the expansion rate, the higher the probability of white holes, and reversed, the lower the likelihood of black holes.

2. Universes with high expansion rates tend to send objects through holes, while worlds with flat expansion rates tend to receive objects through black holes.

3. No matter what of the effect of the expansion rate, all objects in the universe are moved randomly towards the best universe through the wormhole.

The MVO uses a roulette wheel technique to select a new one universe over a generation among worlds according to the expansion rates as a white hole. The exchange of objects between the white/black holes tunnels in the cosmos is used to mathematically model for process optimizing. The MVO assumes that the solution space as the universe as follows.

$$U = \begin{bmatrix} u_{11} & u_{12} & \dots & u_{1p} \\ u_{21} & u_{22} & \dots & u_{2p} \\ \vdots & \vdots & \vdots & \vdots \\ u_{n1} & u_{n2} & \dots & u_{np} \end{bmatrix}, \tag{1}$$

where p is several variables as a dimension of space search, and n is a number of universes as candidate solutions.

$$u_{ij} = \begin{cases} u_{kj} & r_1 < NI\,(U_i) \\ u_{ij} & r_1 \geq NI\,(U_i) \end{cases} \tag{2}$$

where U_i is the i^{th} universe, $NI\,(U_i)$ is a rate of normalized expansion of the i^{th} universe, r_1 is a random number $\in [0, 1]$, u_{ij} is the universe as of the ij^{th} solution, u_{kj} indicates the kj^{th} solution of the universe selected by the roulette wheel scheme.

A wormhole tunnel is established among universes due to local search changes in each universe and increasable expansion rates probability. Then, the best universe would be obtained so far in processing optimization. The wormholes usually are forwarded randomly to the universe without the effect of expansion rates. The detailed mechanism is modeled as follows.

$$u_{ij} = \begin{cases} \begin{cases} u_j + TDR \times \left(\left(ub_j - lb_j\right) \times r_4 + lb_j\right) & r_3 < 0.5 \\ u_j - TDR \times \left(\left(ub_j - lb_j\right) \times r_4 + lb_j\right) & r_3 \geq 0.5 \end{cases} \\ \quad if\,(r_2 < WEP) \\ u_{ij} \\ \quad otherwise\,(r_2 \geq WEP) \end{cases} \tag{3}$$

where TDR and WEP are coefficients which indicate the acronyms for traveling distance rate and wormhole existence probability, respectively, u_j is the j^{th} variables of the best universe obtained so far, ub_j represents the upper bound, and lb_j indicates the lower bound of j^{th} parameter, r_2, r_3, and r_4 are the random number in [0, 1].

The TDR is an essential factor in defining the distance rate for an object, which is helpful to teleport the object through a wormhole around the best universe that occurred so far. TDR is increased over the optimization process to achieve more explicit exploitation around the best planet.

$$TDR = 1 - \frac{T^{\frac{1}{w}}}{F^{\frac{1}{w}}} \tag{4}$$

where, T is the current iteration, and F denotes the maximum repetition, w describes the local search accuracy over the iteration, a high w means the sooner and more accurate local/exploitation search, and w is set to 6 in this paper.

The WEP represents the existence probability of wormhole and is defined to increase linearly during the optimization process. Therefore, the MVO algorithm emphasizes exploitation over the iterations.

$$WEP = W_{min} + T \times \left(\frac{W_{max} - W_{min}}{F}\right) \tag{5}$$

where W_{min} denotes the minimum and W_{max} denotes the maximum of the WEP. In this paper, W_{min} is set to 0.2, W_{max} is set to 1.

The necessary steps of the MVO algorithm are summarized as follows. First, the initialization population size is generated randomly for a set of universes as a solution. Then, in each iteration, obtained the objective function values a planet with high expansion rates by sorting from low expansion planets through white holes / black holes. Next, updating the universes that have the best chance of transmitting their random objects to the universe through wormholes over time regardless of the expansion rate. Finally, the optimization process is terminated until the last criteria, e.g., maximum iteration is met, and the best universe and its expansion speed are obtained as global results.

III. IMPROVING MULTI-VERSE OPTIMIZER

A. IMPLEMENTING IMOV

The MVO simulates the trend forwarding to the best planet in the universe of the multi-verse theory that generates implemented solutions based on phenomena physics of white holes, black holes, and wormholes for global optimization [14]. This algorithm can effectively solve many optimization problems in real life [37]. However, when solving complex optimization problems, the MVO algorithm shows some inherent deficiencies. For example, it loses population diversity early in the search process, is easy to fall into the local optimal solution space, and converges prematurely. In order to reduce the above shortcomings, a new version of its improved optimization algorithm is proposed called IMVO (improved multi-verse optimizer). The parallel processing plays an essential role in practical optimization computations [17]. In order to enhance diversity swarms for the MVO algorithm, the parallel can be created by dividing the population into subgroups that are evaluated with the original algorithm. A multi-group communication mechanism is designed to enhance the universe's diversity. In the multi-group strategy, the number of initialized planets of the universe is divided into P subgroups. The position of the universe in each group moves according to the evolutionary worlds of the original MVO. When the number of iterations reaches a preset value, the cosmos in different groups exchange information to strengthen their cooperation, which will help improve the diversity of planets and avoid stagnation in search. The communication mechanism between different groups (P is the number of groups that is set to a constant) is triggered over setting specific period R times. The position of the universe is mathematically modeled as follows.

$$u_{ij} = u_{ij} + \left(u_* - u_{ij}\right) \times r_5 \tag{6}$$

where, r_5 is a number generated randomly in [0, 1]. u_* is selected from the combined value among groups; it is calculated as follows.

$$u_* = \begin{cases} u_*^1, & r_6 \le 0.25 \\ \left(u_*^1 + u_*^2\right)/2, & 0.25 < r_6 \le 0.5 \\ \left(u_*^1 + u_*^2 + u_*^3\right)/3, & 0.5 < r_6 \le 0.75 \\ \left(u_*^1 + u_*^2 + u_*^3 + u_*^4\right)/4, & 0.75 < r_6 \le 1 \end{cases} \tag{7}$$

where r_6 is a number randomly generated in [0, 1], and u_* is figured out as the best-obtained universe in the g_{th} group in different groups. The main framework of the proposed IMVO is illustrated as a pseudo-code of the proposed approach is described in Algorithm 1. In which j is the current iteration, *Maxiter* is the maximal number of pre-defined iteration, and R is the fixed iterations to communicate among groups.

Algorithm 1 The Pseudo Code of IMVO

Input:

1. **Initialization:** the universe space is set by S, with the N universes are initialized randomly, and the population is divided evenly into P groups that are set to *4*, the dimension D *is* the number of objects of each universe u, the objective function $f(u)$, the current iteration j is set to 1, the maximal number of generation *MaxIter*, R is a triggered iteration of the communication periods.

Iteration:

2. **while** $j < MaxIter$ do
3. **For** $g_{th} = 1$ to P do
4. The universe expansion rate calculated in the $P(g_{th})$, where $P(g_{th})$ is denoted the g_{th} group and sorted by the best planet obtained so far.
5. **for** $i = 1$ to N/P do
6. Select the white hole to change each u_{ij} in the roulette wheel mechanism applied by Eq. (2).
7. According to Eq. (3) that mutate u_{ij} as the wormhole to obtain the best universe.
8. **end for**
9. **end for**
10. **if** $j = R$
11. According to Eq. (7), the communication scheme applied to update u_{ij}
12. **end if**
13. $j = j + 1$
14. **end while**

Output:

The global best universe u_{gbest}, and $f\left(u_{gbest}\right)$ obtained the best expansion rate value

B. EVALUATING IMVO's EXPERIMENTAL RESULTS

In this subsection, the proposed IMVO algorithm is evaluated under the selected functions of the CEC2013 test suite [38]. For sufficient testing of functions' diversity, the twelve benchmark functions from the test suite are utilized to validate the performance of the proposed scheme. The details presetting iterating generations and searching space boundaries are listed in Table 1.

The obtained results of the proposed scheme of IMVO algorithm are compared with the original MVO [14], GWO [13], PSO [10], and PPSO [11] algorithms, respectively. The initial parameters for the algorithms are set as follows. The population size of the compared algorithms is randomly initialized the same number of solutions that N_p set to 80. The dimensions of each solution are set to 30. The boundary of about -100 to 100 of the search spaces is the range of each aspect. The maximum number of iterations is set to 100. The parameter of communicating iteration R is set to 10, and a number of the groups P set to 4. The run times for each testing function is set to 21 due to the meta-heuristic algorithms are randomness.

To comprehensively evaluate the performance of the proposed IVMO algorithm, the best value, mean, and standard deviation of the function values are recorded as the outcomes of the algorithm over 20 runs. The detailed results of the best, mean, and std. of the experimental data are presented in Tables 2 and 3.

Tables 2 and 3 depict the comparison results of the proposed IMVO with the MVO, GWO, PSO, and PPSO algorithms for twelve selected functions of the CEC 2013 test suite. From the perspective of the best, mean, and std., values compared with MVO, the proposed IMVO algorithm achieves seven better results, four worse results, and one similar result, respectively. Doing the same observing the results with GWO, PSO, and PPSO algorithms in the Tables, the number of the "*Wins*" of the IMVO obtained is more than the figure of the "*Loses*" and "*Approx.*" These experimental results confirm that IMVO performs better than the original MVO, and the compared algorithms, which means that our multi-group communication mechanism is sufficient. It is clearly seen that the proposed IMVO is outperforming the comparative algorithms.

Figure 1 plots the comparison convergence curves of the best scores of optimization algorithms for the first four selected functions, e.g., f_1, f_2, f_3, and f_8. It can be seen that the fitness error obtained by the IMVO offers the smallest, which indicates that the optimization result of IMVO is better than the MVO, GWO, PSO, and PPSO algorithms.

IV. HYBRID NN-IMVO FOR DATA FAILURE IDENTIFICATION

In this section, we present a proposed scheme of minimum error in training FNN for data failure identification in CHs of clustering WSN based on the IMVO. A combination of weights and biases of training FNN is optimized based on

TABLE 1. Twelve selected test benchmark function functions.

No.	Test functions	Ranges	Dimen-sions	Itera-tions
1	$f_1(x) = \sum_{i=1}^{n}\sum_{k=1}^{i} x_i$	±100	30	100
2	$f_2(x) = \sum_{i=1}^{N} x_i^2$	±100	30	100
3	$f_3(x) = \sum_{i=1}^{D} \frac{x_i^2}{4000} - \prod_{i=1}^{D} cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	±32	30	100
4	$f_4(x) = \sum_{i=1}^{D} (10^6)^{\frac{i-1}{D-1}} x_i^2$	±100	30	100
5	$f_5(x) = \sum_{i=1}^{D} [10 + x_i^2 - 10cos2\pi x_i$	±100	30	100
6	$f_6(x) = 418.983n - \sum_{i=1}^{N} x_i \times sin\left(\sqrt{\lvert x_i \rvert}\right)$	±100	30	100
7	$f_7(x) = 418.983n - \sum_{i=1}^{N} x_i \times \sin\left(\sqrt{\lvert x_i \rvert}\right) + f_5^*$	±100	30	100
8	$f_8(x) = 20 + \mathrm{e} - 20\exp\left(-\frac{1}{5}\sqrt{(\frac{1}{D}\sum_{i=1}^{D}\cos(i,j)}\right) - \exp(\frac{1}{\mathrm{D}}\sum_{i=1}^{D} x_i^2)$	±100	30	100
9	$f_9(x) = random[0,1) + \sum_{i=1}^{N} i \times x_i^4$	±100	30	100
10	$f_{10}(x) = \sum_{i=1}^{D} (0.1(\sin(\pi x_i))^2$	±100	30	100
11	$f_{11}(x) = \sum_{i=1}^{n-1} (100 \times \left(x_{i-1} - x_i^2\right)^2 + (1 - x_i)^2$	±100	30	100
12	$f_{12}(x) = \frac{1}{D}\sum_{i=1}^{D} (x_i^4 - 16 * x_i^2 + 5x_i)$	±100	30	100

the proposed IMVO. The scheme of optimization training feedforward NN is implemented with IMVO (namely NN-IMVO) by essential elements that need to be configured out in the following subsections.

First, a definition of the objective function is modeled based on the error measure of the NN, and the IMVO is applied to evaluate the fitness of solutions.

Second, identifying data correctness for the CHs in WSN is constructed by optimizing the parameters of the weights, biases of the gradient descent in NN for the solutions.

Finally, the results are presented and discussed to further confirmation of the proposed scheme performance.

A. APPLYING IMVO TO TRAINING NEURAL NETWORKS

The feed-forward neural networks (NN) have been employed to solve complex real-world problems [39]. The NN can generalize, adapting, and learning the large amount of data there-fore; it has become the dominant tool for the classification of complex benchmark problems [40]. NNs have a processing unit that consists of neurons as neural take biological inspiration from the brain of a human. The network linked like a brain of the human with the interconnected neurons that are robust to the output of other connections that have changeable measures [41]. The achieved models of NNs are the following facets. In essence, the NNs are flexible enough in that they can adjust themselves in the absence of

TABLE 2. Comparison of testing results of the IMVO with the MVO and GWO algorithms for twelve selected functions of the CEC 2013 test suite.

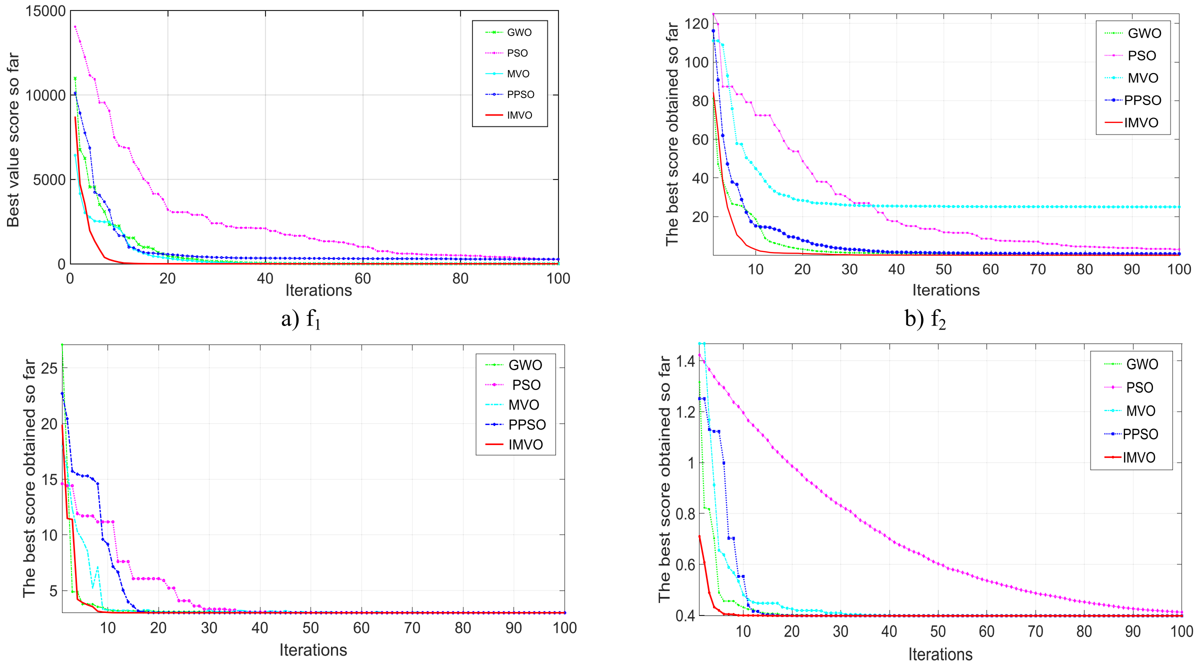
No. Tests	IMVO			MVO			GWO		
	Best	Mean	Std.	Best	Mean	Std.	Best	Mean	Std.
1	2.33E-08	1.04E-07	1.37E-07	1.22E-06	1.06E-05	1.31E-06	1.31E-02	1.12E+02	7.54E+01
2	3.27E-09	2.24E-08	2.13E-08	3.26E-07	1.57E-06	6.63E-07	2.31E-03	1.79E+01	5.59E+01
3	2.05E+01	2.38E+02	2.96E+02	9.76E+01	1.08E+03	1.06E+03	3.01E+03	1.05E+04	4.51E+03
4	2.01E-03	4.64E-03	1.39E-03	9.97E-04	1.52E+01	2.97E+01	1.31E-02	2.62E+01	1.63E+01
5	2.43E-03	1.01E+01	1.36E+01	1.05E+00	1.01E+01	9.63E+00	1.38E+00	2.19E+01	2.64E+01
6	6.21E-01	2.12E+01	2.58E+01	6.32E-01	1.51E+01	1.11E+01	1.26E-01	1.36E+01	1.57E+01
7	2.01E+01	2.04E+01	1.04E-01	2.03E+01	2.04E+01	7.00E-02	2.03E+01	2.04E+01	6.47E-02
8	8.01E-01	3.62E+00	1.56E+00	3.29E+00	5.19E+00	1.43E+00	1.86E+00	3.67E+00	9.60E-01
9	1.18E-01	2.86E-01	1.57E-01	1.16E-01	1.40E+00	4.99E+00	4.61E-01	8.93E+00	7.86E+00
10	8.95E+00	2.00E+01	6.94E+00	9.95E+00	2.59E+01	1.26E+01	5.02E+00	1.53E+01	8.57E+00
11	4.97E+00	1.80E+01	9.26E+00	7.12E+00	2.15E+01	7.89E+00	5.99E+00	2.01E+01	9.37E+00
12	5.61E+00	3.09E+01	1.22E+01	1.72E+01	4.15E+01	7.09E+00	8.96E+00	2.04E+01	7.03E+00
Win	compared	compared	compared	7	8	8	8	8	9
Lose	compared	compared	compared	4	3	4	3	2	3
Approx.	compared	compared	compared	1	1	0	1	2	0

TABLE 3. Comparison testing results of the IMVO with PSO and PPSO algorithms for twelve selected functions of the CEC 2013 test suite.

No. Tests	IMVO			PPSO			PSO		
	Best	Mean	Std.	Best	Mean	Std.	Best.	Mean.	Std.
1	4.33E-08	5.23E-07	5.21E-07	5.35E-06	5.18E-07	5.67E-07	6.24E-02	4.18E+02	2.56E+02
2	3.25E-09	2.26E-08	2.25E-08	3.25E-07	1.59E-06	6.67E-07	2.38E-03	1.69E+01	5.45E+01
3	2.09E+01	2.37E+02	2.95E+02	9.79E+01	1.09E+03	1.06E+03	3.01E+03	1.05E+04	4.51E+03
4	2.01E-03	4.64E-03	1.39E-03	9.97E-04	1.49E+01	2.97E+01	1.29E-02	2.62E+01	1.63E+01
5	2.43E-03	1.01E+01	1.36E+01	1.05E+00	1.01E+01	9.63E+00	1.38E+00	2.19E+01	2.64E+01
6	6.21E-01	2.12E+01	2.58E+01	6.32E-01	1.51E+01	1.11E+01	1.26E-01	1.36E+01	1.57E+01
7	8.01E-01	3.62E+00	1.56E+00	3.29E+00	5.19E+00	1.43E+00	1.86E+00	3.67E+00	9.60E-01
8	1.18E-01	2.86E-01	1.57E-01	1.16E-01	1.40E+00	4.99E+00	4.61E-01	8.93E+00	7.86E+00
9	4.97E+00	1.80E+01	9.26E+00	8.96E+00	2.15E+01	7.89E+00	5.99E+00	2.01E+01	9.37E+00
10	5.62E+00	3.01E+01	1.22E+01	1.69E+01	3.65E+01	8.19E+00	8.92E+00	2.01E+01	7.23E+00
11	4.74E+02	9.63E+02	3.10E+02	3.05E+02	1.02E+03	3.72E+02	1.32E+02	5.22E+02	2.26E+02
12	2.28E+02	5.92E+02	2.48E+02	5.25E+02	8.08E+02	2.56E+02	1.69E+02	3.63E+02	2.26E+02
Win	compared	compared	compared	6	5	6	7	8	9
Lose	compared	compared	compared	4	4	4	4	2	3
Approx.	compared	compared	compared	2	3	2	1	2	0

distributional and functional form for the hidden layer model, and the approximate any complex non-linear function with the hidden dimensions [42]. However, the achieved model existed a drawback during the weight optimization process of training the classifier, such as dropping stuck in local minima [43].

In this study, we apply the IMVO to optimize the training of the feed-forward neural network to avoid the drawbacks. The scheme of optimization training network is implemented with NN-IMVO that adopted the NN to identify collecting data failure as classify real-world problems.



a) f_1 b) f_2 c) f_3 d) f_8

FIGURE 1. **Comparison of the optimum-obtained values of the IMVO with the other algorithms, e.g., PSO, MVO, PPSO, and GWO, for the first four selected functions. Subfigures: a, b, c, and d are the convergence curves for four selected functions, e.g., f_1, f_2, f_3, and f_8.**

The essential elements steps of implementing NN-IMVO are configured out as follows. The core objective is to train the feed-forward neural network to reach the optimized weights for enhancing the performance of the classifier. A combination of weights and biases of training FNN is optimized based on the IMVO. The classical technique used for the training of intelligent classifier is backpropagation, which is an improved version of the gradient descent method [44]. A mean squared error (MSE) of the training sample is computed as follows.

$$E_j = \sum_{i=1}^{n} \left(o_i^j - y_i^j\right)^2 \tag{8}$$

where E_j is a mean squared error as training error at j^{th} training sample, y_i^j denoted as the desired output of the ith input variable x_i, the n is a number of the training sample, o_i^j is the final output at the training sample j^{th} that can be calculated as follows.

$$o_j = \sum_{j=1}^{m} \omega_{jk} \cdot f(h_k) - \beta_j \quad j = 1, 2, \ldots, m, \tag{9}$$

where ω_{jk} is the connecting weights from the i^{th} node in the input layer, f is the outcome of the hidden node that can be computed mathematically in each iteration of training as follows.

$$f\,(h_k) = \frac{1}{\left(1 + exp^{-\left(\sum_{i=1}^{n} \omega_{ij} \cdot x_i - \beta_j\right)}\right)}, \quad j = 1, 2, .., k, \tag{10}$$

where h_k is the k^{th} figured node out of the hidden layer that is calculated as the following summary: h_k is set to $\sum_{i=1}^{n} \omega_{jk} \cdot x_i - \beta_k$, with β_k is the threshold as the bias of j^{th} hidden node, and x_i denotes the i^{th} input. The fitness function is modeled based on learning error E with the number of training samples.

$$Fitnes\,(X_i) = MinimizeE(X_i) = \sum_{j=1}^{q} \frac{E_j}{q} \tag{11a}$$

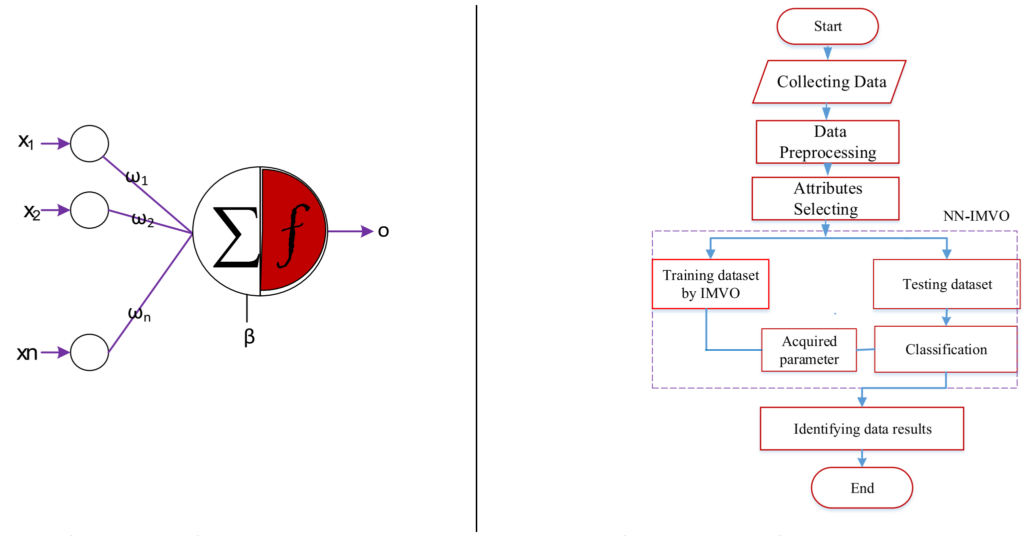
$$s.t.\ \varepsilon_n \geq P_n, \quad \forall v_i \epsilon V; \tag{11b}$$

$$C\,(O) = 1; \tag{11c}$$

$$D\,(v_i, v_0) \in \{1, 2\}\,; \tag{11d}$$

$$X \in [\boldsymbol{x}_L, \boldsymbol{x}_U], \tag{11e}$$

where q is the number of training samples, X_L, and X_U are the lower and upper bounds for constraints, respectively, space X is a vector as $X_1, X_2, .., X_u \in X \in R^d$. ε_n and P_n are the least residual power, and the totals consumed energy for the WSN [4]; $V(v_1, v_2, ..v_k)$ is denoted sensor nodes [6]; C is the totals network coverage, and $Oo_1, o_2, ..o_n$ is the sensed object; $D\,(v_i, v_0)$ is the network latency of CH to BS, e.g., the queue, the propagation, and the transmission delay. Eq.(11a) is an objective function for the optimization, Eq.(11b) is the constraint of the limited least remaining power for guaranteeing the sensor's regular duties, Eq.(11c) is the restriction of the limited network coverage, and Eq.(11d) is the restriction of for guaranteeing the network latency condition. In the experiment, boundaries for the weights and biases are set to arrange $[-3, 3]$. The solution for the training of feed-forward neural networks is a combination of weights and biases that often is expressed suitably with the matrix. Thus, the array is used to encode for the IMVO agent mapping to the optimization solution. Figure 3 shows an example of a



a) A structure of the neural network

b) The applied IMVO to FNN training processing

FIGURE 2. **The applied IMVO to optimizing weights and bias of FNN training for collected data failure identification. Subfigures a) and b) are the structure of the neural network and the FNN training processing based on applying IMVO, respectively.**

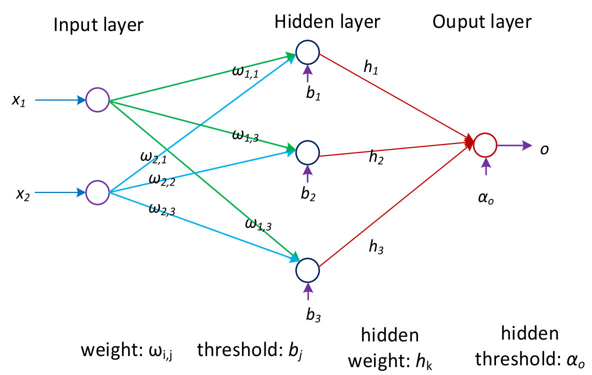


FIGURE 3. **An example of a neural network structure of 2-3-1.**

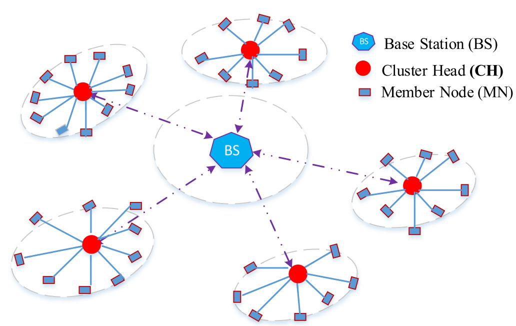


FIGURE 4. **A structure of the hierarchy of WSN clusters.**

neural network structure of 2-3-1; the universe of the IMVO is expressed for parameters of the feed-forward neural network as follows.

$$Universe(:, :, i) = [w_1, \beta_1, w_2', \beta_2] \tag{12}$$

where $[w_1, \beta_1, w_2', \beta_2]$ are the variables of the neuron that described as follows.

$$w_1 = \begin{bmatrix} \omega_{1,1} & \omega_{2,1} \\ \omega_{1,2} & \omega_{2,2} \\ \omega_{1,3} & \omega_{2,3} \end{bmatrix}, \beta_1 = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}, w_2' = \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix},$$
$$\beta_2 = [a_0] \tag{13}$$

The set of the network weights after forming the neural network can be approximated to the necessary results based on the objective function of minimum error. It means the weights are adjusted under met some error conditions by applying the NN-IMVO algorithm for the training net. The next subsection presents implementing NN-IMVO solutions to identify collecting failure data in CHs of the hierarchy WSN. The initial parameter setting for simulation is structured as; the population size is 80; the dimensions of each solution are set to 30, $maxiter = 200$; R is set to 15, and the groups' number P set to 4. Doing the same initial parameters setting for the comparative algorithms, e.g., PSO, GWO and MVO: $c_1 = c_2 = 1{:}45; = 2, C = 2; D = 30;\ maxiter = 200$.

B. FAILURE IDENTIFICATION FOR WSN COLLECTED DATA

As mentioned in the section of the introduction, WSN has many practical applications in the fields of healthcare, vehicle monitoring, search and rescue, and military defense, etc., for decades [25]–[27]. Necessary relevant environment information, e.g., temperature, humidity, pressure, sunlight, sound, wind speed, and air and water pollution levels, can be recorded and measured by the WSN's sensing nodes [34], [45]. An effective way of clustering networks is by dividing the sensor nodes into several clusters that directly affect the total net's power consumption [46]. The cluster WSN includes member-nodes (MNs) and cluster-headers (CH) that selected among MNs [47], [48]. The function of CH is not only to capture information from MNs but

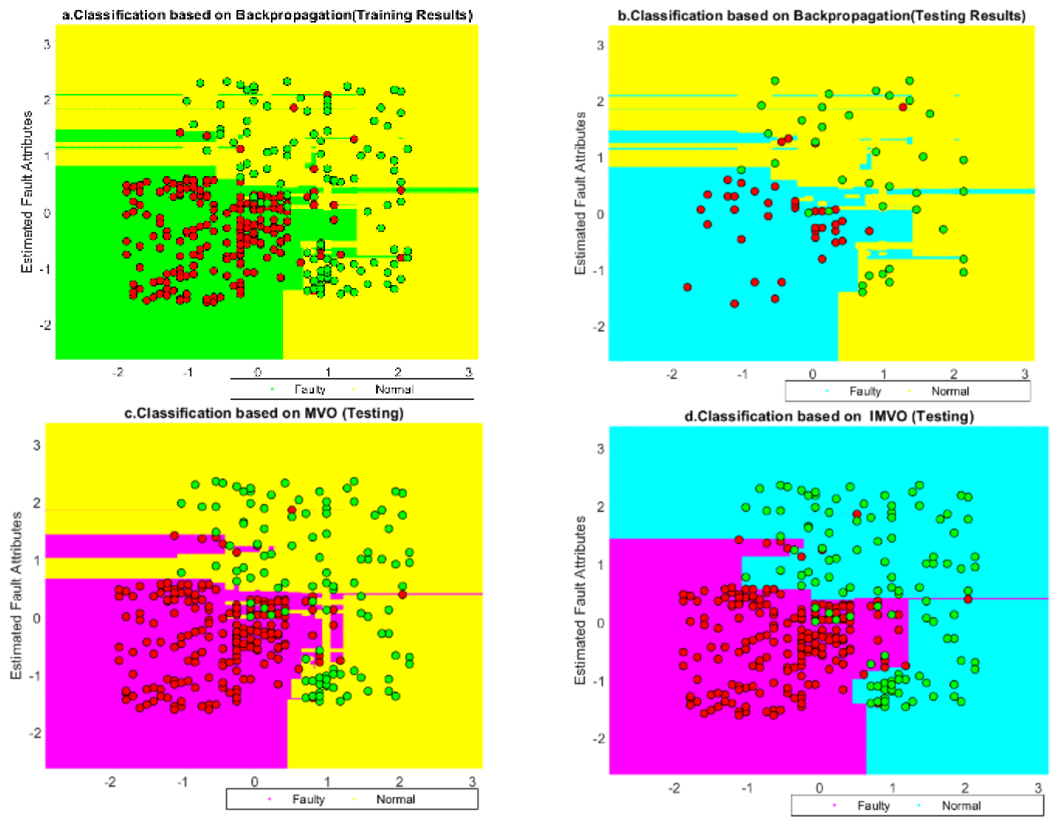


FIGURE 5. A visualization results of the backpropagation approach for the training and testing set, and IMVO, and MVO methods for the testing set of identification correctness data; a) and b) are the classification based on backpropagation for training and testing; c) and d) are the classification based on MVO and IMVO respectively for testing.

also to aggregate the data collected and forward it to the BS. Figure 4 shows the structure of the WSN clusters.

The purpose of the proposed scheme is to set a decision function in CH, which can identify any failure of collecting data from the MNs in real-time to accurately synthesize data, forward it to the BS in WSN. Data collected may contain noise or faults due to various reasons by sensor nodes from multiple sources that are used to the input of the system. The defects data may cause data classification errors. The solution of the NN-IMVO for filtering the faults data is an essential task of a successful WSN application. Its decision function is to identify data correctly in CH that can be combined with expert knowledge and optimal classification. Notice some WSN-applications could have the equipped CHs with unlimited power supply due to CH consume more energy.

We simulated scenarios of the WSN as follows: some set faulty nodes of the WSN, i.e., humidity, temperature the sensed data that can be lost on traffic congestion conditions. Conditions of the healthy network and/or some faulty nodes in the system are generated randomly. Training of multi-layer FNN is performed by using the backpropagation algorithm (BPA) [21], the beta distribution PSO (NN-PSO) [33], and the MVO [22]. The ratio for the testing and training is set to 20 and 80, respectively.

The experimental results of the proposed NN-IMVO are compared against the other previous methods. Figure 5 shows a comparison of visualization obtained results of the IMVO and MVO methods for the testing set of identification correctness data, and the backpropagation approach for the training and testing collected data set. Noticed obviously, the proposed scheme of NN-IMVO provides more accurate identification for CHs than its competitors.

C. ANALYSIS AND DISCUSSION

Assumed a network of N nodes deployed in specific $M \times M$ area randomly, several N nodes is set 100, 200, 300, respectively; M is a used area length is configured to 200, 300, 400m, respectively. BS equipped with an unlimited power recourse supply in the network that managed system and aggregated data from CHs.

Simulation is compiled in Matlab for the proposed scheme and utility the kits of sensing nodes in the Lab. Matlab 2018b on Windows 10, 64 operating system on Intel®Core^{TM} i7-8665U, and 8Gb of RAM of Lenovo T470p laptop are used to execute in the simulation. The scheduling network operation assumed like periods of the transmission packets.

Some scenario's data packets are initialized randomly. For example, collecting data packets for dataset include attributes, e.g., node ID, Packet-ID, ClusterHeads (0 or 1), Sensing information (moisture, temperature, network status, gas, and lights that depends on node sensor), noise estimated,

TABLE 4. The results of the proposed scheme of NN-IMVO for some selected nodes in training collected data over some fault types.

Node ID	Data-loss fault		Gain fault		Drift fault		Hardware fault	
	DR	FR	DR	FR	DR	FR	DR	FR
001	82%	31%	78%	31%	77%	39%	80%	35%
002	83%	31%	79%	32%	80%	41%	73%	36%
003	80%	38%	75%	30%	82%	42%	73%	33%
009	76%	39%	80%	32%	83%	44%	79%	33%
037	82%	31%	75%	30%	76%	43%	78%	32%
051	79%	39%	77%	31%	76%	32%	75%	33%
052	86%	32%	76%	31%	86%	44%	78%	33%
074	83%	31%	79%	32%	86%	42%	81%	32%
072	90%	31%	82%	33%	80%	45%	73%	33%
170	89%	35%	75%	30%	85%	44%	77%	31%
AVG	**83%**	**37%**	**78%**	**31%**	**81%**	**43%**	**77%**	**33%**

TABLE 5. Comparison of the proposed scheme with the PSO and MVO for optimization of the FNN training of collected data in WSN with different hidden nodes.

Hidden nodes	NN-IMVO			NN-PSO			NN-MVO		
	Best	**Mean**	**Std.**	**Best**	**Mean**	**Std.**	**Best**	**Mean**	**Std.**
5	3.18E-10	1.58E-06	2.61E-02	9.00E-09	3.61E-02	2.24E-02	4.35E-02	1.80E-01	5.95E-01
6	1.23E-09	4.51E-03	1.59E-02	4.19E-12	2.12E-02	8.35E-03	4.59E-02	6.45E-02	1.42E-01
7	1.42E-11	2.71E-03	3.10E-07	3.83E-24	1.03E-02	2.77E-02	1.24E-02	1.26E-01	6.53E-03
8	1.26E-11	2.04E-05	6.29E-05	3.32E-13	5.13E-03	2.30E-02	7.13E-03	1.15E-01	7.64E-04
9	5.53E-12	7.72E-06	2.65E-05	1.31E-24	5.01E-03	2.28E-02	1.06E-02	8.04E-02	5.44E-02
10	1.55E-10	6.13E-06	2.89E-05	4.42E-09	1.29E-02	3.80E-02	1.21E-02	7.76E-02	4.31E-02
11	4.66E-10	1.83E-05	7.70E-05	2.10E-23	2.28E-02	3.92E-03	1.08E-02	6.58E-02	4.120E-04
12	1.65E-10	4.17E-02	2.28E-02	3.56E-50	3.15E-02	9.26E-03	9.28E-03	6.98E-02	4.12E-02
13	4.67E-11	4.17E-03	2.28E-02	2.29E-15	5.01E-03	2.28E-02	8.46E-08	4.76E-05	7.08E-04
14	5.17E-11	4.57E-03	7.28E-02	4.67E-11	4.17E-03	2.28E-02	6.49E-08	2.33E-06	2.27E-02
15	2.80E-10	1.68E-02	4.32E-02	2.85E-09	5.01E-03	2.28E-02	7.77E-08	9.91E-05	3.29E-10
Win	-	-	-	4	7	5	8	7	6
Lose	-	-	-	3	2	3	2	3	3
Approx.	-	-	-	4	1	3	1	1	2

and radio signal strength (Decibels (dB)). The captured random data is sent it to CHs. CHs' tasks are to aggregate the collected data, pack the data gathered, and forward it to the BS. The data is divided into two parts: one for training and the other for variety size testing: (70% and 40%), (70% and 30%), and (80% and 20%), respectively. The results of studies carried out for training and testing data with a size of 80 percent and 20 percent provided the most influential and most reliable test relative to the other measurements [35].

Two parameters of metrics for determining the false rate (*FR*) and detecting rate (*DR*) are used to validate the performance and classification accuracy of the proposed method. The metrics are described as formulating follows.

$$FR = \frac{number\ of\ non-faulty\ data\ classified\ as\ faulty}{total\ faulty\ data} \tag{14}$$

where *FR* denoted the false-positive rate; the ratio with some non-faulty data is identified and classified as faulty to the total number of incomplete data.

$$DR = \frac{number\ of\ classified\ faulty\ data}{total\ number\ of\ present\ faulty\ data} \tag{15}$$

where *DR* denoted as the detected accuracy rate. This parameter is used to clarify rightness collecting data as the ratio of the number of erroneous data identified to the total number of current incorrect data.

There are some types of failure received data such as data-loss fault, gain fault, drift fault, and hardware fault [49]. Table 4 depicts the results of the proposed scheme of NN-IMVO for some selected nodes in training collected data over some fault types.

A structure of FNN with four inputs, the number of hidden nodes (H), and three outputs denoted 4-H-3 that is used to apply NN-IMVO for data failure identification in CHs of WSN, where H can be set 4 to 15, respectively. The experimental results of the proposed method are compared with MVO and PSO, as shown in Table 5. Observably from Table 5, it can be clear that the performance of the NN-IMVO offers more accuracy than NN-PSO and NN-MVO. The highest number of wins belong to NN-IMVO. The mathematical algorithm used for NN requires not only strong discoverability but also correct usability. The results of NN-IMVO demonstrate that it has both significant developmental capacity and functional exploration capabilities due to solve the problem of traps at a local minimum and can provide fast convergence rates.

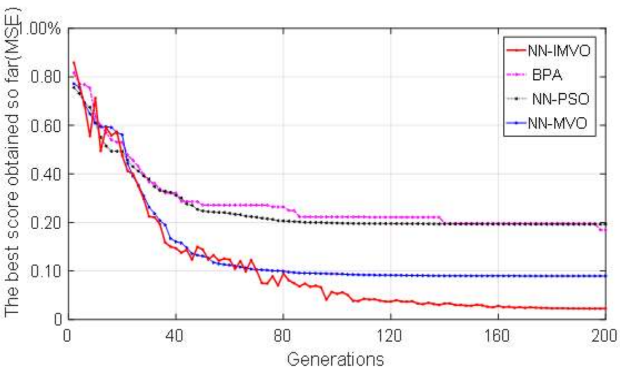


FIGURE 6. Comparison of convergence curves of the IMVO with the BPA, PSO, and MVO based on the function of averages of the MSE for all training collected data over twenty runs with H set 10, and 15 respectively.

Figure 6 shows the comparison of convergence curves of the IMVO with the BPA, PSO, and MVO based on the function of averages of the MSE for all training collected data over twenty runs with H set 10, and 15 respectively. Observed the figure, it can be seen apparently to confirm that NN-IMVO has the best convergence rate for identification failure collected data in CHs in WSN.

V. CONCLUSION

In this paper, a new training method of feed-forward neural network (FNN) for identifying failure data in cluster heads (CH) of WSN was proposed based on improving multi-verse optimizer (MVO). The MVO algorithm was improved based on enhancing diversity universes (named IMVO) for avoiding it's a drawback of the local optimal. The best parameter values of the weight and biases are obtained by taking enhanced diversity of the IMVO algorithm. The training data method of classification is applied for identification data failure in aggregating CHs then forward it to the base station (BS) in WSN by hybridizing IMVO and FNN. The experimental results are investigated on twelve selected well-known benchmark functions and the problem of identifying failure data in WSN to test the proposed method performance. Compared results with the other methods in the literature that exhibits the proposed approach provides higher detection accuracy than the competitors.

REFERENCES

[1] A. Gogna and A. Tayal, "Metaheuristics: Review and application," *J. Exp. Theor. Artif. Intell.*, vol. 25, no. 4, pp. 503–526, 2013.

[2] T.-K. Dao, J.-S. Pan, T.-S. Pan, and T.-T. Nguyen, "Optimal path planning for motion robots based on bees pollen optimization algorithm," *J. Inf. Telecommun.*, vol. 1, no. 4, pp. 351–366, 2017.

[3] A. A. Juan, J. Faulin, S. E. Grasman, M. Rabe, and G. Figueira, "A review of simheuristics: Extending metaheuristics to deal with stochastic combinatorial optimization problems," *Oper. Res. Perspect.*, vol. 2, pp. 62–72, Dec. 2015.

[4] T.-T. Nguyen, J.-S. Pan, S.-C. Chu, J. F. Roddick, and T.-K. Dao, "Optimization localization in wireless sensor network based on multi-objective firefly algorithm," *J. Netw. Intell.*, vol. 1, no. 4, pp. 130–138, 2016.

[5] T.-T. Nguyen, J.-S. Pan, and T.-K. Dao, "A compact bat algorithm for unequal clustering in wireless sensor networks," *Appl. Sci.*, vol. 9, no. 10, p. 1973, May 2019.

[6] Y. Chang, X. Yuan, B. Li, D. Niyato, and N. Al-Dhahir, "Machine-learning-based parallel genetic algorithms for multi-objective optimization in ultra-reliable low-latency WSNs," *IEEE Access*, vol. 7, pp. 4913–4926, 2019.

[7] S. Mahdavi, M. E. Shiri, and S. Rahnamayan, "Metaheuristics in large-scale global continues optimization: A survey," *Inf. Sci.*, vol. 295, pp. 407–428, Feb. 2015.

[8] Z. Meng, J.-S. Pan, and K.-K. Tseng, "PaDE: An enhanced differential evolution algorithm with novel control parameter adaptation schemes for numerical optimization," *Knowl.-Based Syst.*, vol. 168, pp. 80–99, Mar. 2019.

[9] J. H. Holland, "Genetic algorithms," *Sci. Amer.*, vol. 267, no. 1, pp. 66–73, 1992.

[10] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. Int. Conf. Neural Netw. (ICNN)*, vol. 6, no. 4, pp. 1942–1948, 1995.

[11] S. C. Chu, J.-F. Chang, J. F. Roddick, and J.-S. Pan, "A parallel particle swarm optimization algorithm with communication strategies," *J. Inf. Sci. Eng.*, vol. 21, no. 4, pp. 809–818, 2005.

[12] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: Optimization by a colony of cooperating agents," *IEEE Trans. Syst. Man, Cybern. B, Cybern.*, vol. 26, no. 1, pp. 29–41, Feb. 1996.

[13] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014.

[14] S. Mirjalili, S. M. Mirjalili, and A. Hatamlou, "Multi-verse optimizer: A nature-inspired algorithm for global optimization," *Neural Comput. Appl.*, vol. 27, no. 2, pp. 495–513, Feb. 2016.

[15] T. T. Nguyen Y. Qiao, J. S. Pan. S. C. Chu, K. C. Chang, X. S. Xue, and T. K. Dao, "A hybridized parallel bats algorithm for combinatorial problem of traveling salesman," *J. Intell. Fuzzy Syst.*, vol. 38, no. 5, pp. 5811–5820, 2020.

[16] T.-T. Nguyen, J.-S. Pan, and T.-K. Dao, "An improved flower pollination algorithm for optimizing layouts of nodes in wireless sensor network," *IEEE Access*, vol. 7, pp. 75985–75998, 2019.

[17] T.-K. Dao, T.-S. Pan, T.-T. Nguyen, and J.-S. Pan, "Parallel bat algorithm for optimizing makespan in job shop scheduling problems," *J. Intell. Manuf.*, vol. 29, no. 2, pp. 451–462, Feb. 2018.

[18] A. Zell, *Simulation Neuronaler Netze*. Bonn, Germany: Addison-Wesley, 1994.

[19] H. Bakırcıo lu and T. Koçak, "Survey of random neural network applications," *Eur. J. Oper. Res.*, vol. 126, no. 2, pp. 319–330, Oct. 2000.

[20] L. Zhang and P. N. Suganthan, "A survey of randomized algorithms for training neural networks," *Inf. Sci.*, vols. 364–365, pp. 146–155, Oct. 2016.

[21] H. Leung and S. Haykin, "The complex backpropagation algorithm," *IEEE Trans. Signal Process.*, vol. 39, no. 9, pp. 2101–2104, Sep. 1991.

[22] H. Faris, I. Aljarah, and S. Mirjalili, "Training feedforward neural networks using multi-verse optimizer for binary classification problems," *Int. J. Speech Technol.*, vol. 45, no. 2, pp. 322–332, Sep. 2016.

[23] C. Hu, Z. Li, T. Zhou, A. Zhu, and C. Xu, "A multi-verse optimizer with levy flights for numerical optimization and its application in test scheduling for network-on-chip," *PLoS ONE*, vol. 11, no. 12, Dec. 2016, Art. no. e0167341.

[24] X. Wang, J.-S. Pan, and S.-C. Chu, "A parallel multi-verse optimizer for application in multilevel image segmentation," *IEEE Access*, vol. 8, pp. 32018–32030, 2020.

[25] C. F. García-Hernández, P. H. Ibargüengoytia-González, J. García-Hernández, and J. A. Pérez-Díaz, "Wireless sensor networks and applications: A survey," *J. Comput. Sci.*, vol. 7, no. 3, pp. 264–273, 2007.

[26] J.-S. Pan, T.-T. Nguyen, T.-K. Dao, T.-S. Pan, and S.-C. Chu, "Clustering formation in wireless sensor networks: A survey," *J. Netw. Intell.*, vol. 2, no. 4, pp. 287–309, 2017.

[27] N. Liu, J.-S. Pan, and T.-T. Nguyen, "A bi-population QUasi-affine TRansformation evolution algorithm for global optimization and its application to dynamic deployment in wireless sensor networks," *EURASIP J. Wireless Commun. Netw.*, vol. 2019, no. 1, p. 175, Dec. 2019.

[28] Y. Qiao, T.-K. Dao, J.-S. Pan, S.-C. Chu, and T.-T. Nguyen, "Diversity teams in soccer league competition algorithm for wireless sensor network deployment problem," *Symmetry*, vol. 12, no. 3, p. 445, Mar. 2020.

[29] H. El Alami and A. Najid, "MS-routing-G_i: Routing technique to minimise energy consumption and packet loss in WSNs with mobile sink," *IET Netw.*, vol. 7, no. 6, pp. 422–428, Nov. 2018.

[30] T.-T. Nguyen, J.-S. Pan, and T.-K. Dao, "A novel improved bat algorithm based on hybrid parallel and compact for balancing an energy consumption problem," *Information*, vol. 10, no. 6, p. 194, Jun. 2019.

[31] T.-T. Nguyen, J.-S. Pan, T.-Y. Wu, T.-K. Dao, and T.-D. Nguyen, "Node coverage optimization strategy based on ions motion optimization," *J. Netw. Intell.*, vol. 4, no. 1, pp. 1–9, 2019.

[32] Y. Chang, X. Yuan, B. Li, D. Niyato, and N. Al-Dhahir, "A joint unsupervised learning and genetic algorithm approach for topology control in energy-efficient ultra-dense wireless sensor networks," *IEEE Commun. Lett.*, vol. 22, no. 11, pp. 2370–2373, Nov. 2018.

[33] Y. Chen, B. Yang, and J. Dong, "Evolving flexible neural networks using ant programming and PSO algorithm," in *Proc. Int. Symp. Neural Netw.*, 2004, pp. 211–216.

[34] T.-T. Nguyen, T.-K. Dao, H.-Y. Kao, M.-F. Horng, and C.-S. Shieh, "Hybrid particle swarm optimization with artificial bee colony optimization for topology control scheme in wireless sensor networks," *J. Internet Technol.*, vol. 18, no. 4, pp. 743–752, 2017.

[35] S.-C. Chu, T.-K. Dao, J.-S. Pan, and T.-T. Nguyen, "Identifying correctness data scheme for aggregating data in cluster heads of wireless sensor network based on naive Bayes classification," *EURASIP J. Wireless Commun. Netw.*, vol. 52, no. 1, pp. 52-1–52-16, Dec. 2020.

[36] T.-K. Dao, T.-T. Nguyen, J.-S. Pan, Y. Qiao, and Q.-A. Lai, "Identification failure data for cluster heads aggregation in WSN based on improving classification of SVM," *IEEE Access*, vol. 8, pp. 61070–61084, 2020.

[37] I. Aljarah, M. Mafarja, A. A. Heidari, H. Faris, and S. Mirjalili, "Multi-verse optimizer: Theory, literature review, and application in data clustering," in *Nature-Inspired Optimizers: Theories, Literature Reviews and Applications*, S. Mirjalili, J. S. Dong, and A. Lewis, Eds. Cham, Switzerland: Springer, 2020, pp. 123–141.

[38] J. J. Liang, B. Y. Qu, P. N. Suganthan, and A. G. Hernández-Díaz, "Problem definitions and evaluation criteria for the CEC 2013 special session on real-parameter optimization," Comput. Intell. Lab., Zhengzhou Univ. Zhengzhou, China, Nanyang Technol. Univ., Singapore, Tech. Rep. 201212, 2013.

[39] D. Svozil, V. Kvasnicka, and J. Pospichal, "Introduction to multi-layer feed-forward neural networks," *Chemometric Intell. Lab. Syst.*, vol. 39, no. 1, pp. 43–62, Nov. 1997.

[40] J. Ilonen, J.-K. Kamarainen, and J. Lampinen, "Differential evolution training algorithm for feed-forward neural networks," *Neural Process. Lett.*, vol. 17, no. 1, pp. 93–105, 2003.

[41] S. Ding, H. Li, C. Su, J. Yu, and F. Jin, "Evolutionary artificial neural networks: A review," *Artif. Intell. Rev.*, vol. 39, no. 3, pp. 251–260, Mar. 2013.

[42] R. Kozma, C. Alippi, Y. Choe, and F. C. Morabito, *Artificial Intelligence in the Age of Neural Networks and Brain Computing*. New York, NY, USA: Academic, 2018.

[43] J. Yang and J. Ma, "Feed-forward neural network training using sparse representation," *Expert Syst. Appl.*, vol. 116, pp. 255–264, Feb. 2019.

[44] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, pp. 533–536, 1986, doi: 10.1038/323533a0.

[45] X. Xue and J.-S. Pan, "A compact co-evolutionary algorithm for sensor ontology meta-matching," *Knowl. Inf. Syst.*, vol. 56, no. 2, pp. 335–353, Aug. 2018.

[46] H. El Alami and A. Najid, "ECH: An enhanced clustering hierarchy approach to maximize lifetime of wireless sensor networks," *IEEE Access*, vol. 7, pp. 107142–107153, 2019.

[47] J.-S. Lee and C.-L. Teng, "An enhanced hierarchical clustering approach for mobile sensor networks using fuzzy inference systems," *IEEE Internet Things J.*, vol. 4, no. 4, pp. 1095–1103, Aug. 2017.

[48] T. T. Nguyen, C. S. Shieh, M. F. Horng, T. G. Ngo, and T. K. Dao, "Unequal clustering formation based on bat algorithm for wireless sensor networks," in *Proc. Adv. Intell. Syst. Comput.*, vol. 326, 2015, pp. 667–678.

[49] J. Chen, S. Kher, and A. Somani, "Distributed fault detection of wireless sensor networks," in *Proc. Workshop Dependability Issues Wireless Ad Hoc Netw. Sensor Netw. (DIWANS)*, 2006, pp. 65–72.

THI-KIEN DAO received the Ph.D. degree in electronics and engineering from the National Kaohsiung University of Technology and Sciences, Taiwan, in 2019. She is currently a Lecturer with the College of Information Science and Engineering, Fujian University of Technology. Her current research interests include computational intelligence, data mining, and sensor networks.

JIE YU received the M.S. degree in automotive engineering and the Ph.D. degree in automotive engineering from Fuzhou University, China, in 2009 and 2020, respectively. He is currently a Lecturer with the College of Mechanical and Automotive Engineering, Fujian University of Technology, Fuzhou, China. His current research interests include swarm intelligence, automotive engineering, and data mining.

TRONG-THE NGUYEN received the Ph.D. degree in communication engineering from the National Kaohsiung University of Applied Sciences, Taiwan, in 2016. He is currently a Lecturer with the College of Information Science and Engineering, Fujian University of Technology, China, and the Department of Information technology, Haiphong University of Management and Technology, Vietnam. His current research interests include computational intelligence, signal processing, and sensor networks.

TRUONG-GIANG NGO received the Ph.D. degree in mathematical theory for informatics from the Graduate University of Science and Technology, Vietnam Academy of Science and Technology, Vietnam, in 2017. He is currently a Lecturer with the Faculty of Computer Science and Engineering, Thuyloi University, Vietnam. His current research interests include machine learning, data mining, and computational intelligence.