Energy Engineering

Tech Science Press

https://doi.org/10.32604/ee.2026.078087

ARTICLE

A Novel Binary Classification Neural Network Optimized by the Mosquito Mating Swarm Optimization Algorithm for Predicting Microgrid Operational Modes

Jesús Águila-León[1], Carlos Vargas-Salgado[2,*], Dácil Díaz-Bello[2] and Fabián Lara-Vargas[3]

[1]Department of Water and Energy Studies, Universidad de Guadalajara, Guadalajara, Mexico
[2]University Institute of Energetic Engineering, Universitat Politècnica de València, València, Spain
[3]Programa de Ingeniería Electrónica, Grupo de Investigación ITEM, Universidad Pontificia Bolivariana, Montería, Colombia

*Corresponding Author: Carlos Vargas-Salgado. Email: carvarsa@upvnet.upv.es

Received: 23 December 2025; Accepted: 13 February 2026; Published: 18 June 2026

ABSTRACT: Integrating renewable energy sources presents technical challenges due to their variable nature, particularly in predicting and managing microgrid operational modes. Accurate identification of grid states—interconnected or islanded—is essential for maintaining stability and optimizing performance under fluctuating environmental conditions to meet energy demand. This work proposes a bio-inspired, optimized binary classification model based on Multi-Layer Perceptron Artificial Neural Networks (MLP-ANN), with the architecture and hyper-parameters tuned using the novel Mosquito Mating Swarm Optimization (MMSO) algorithm, inspired by mosquito mating behavior and swarm dynamics. The model employs an MLP-ANN with a variable number of hidden layers and neurons per layer, configured to maximize classification accuracy by dynamically adjusting parameters, including the learning rate and regularization coefficients. Training utilizes k-fold cross-validation on experimental microgrid data. The MMSO approach is benchmarked against Particle Swarm Optimization (PSO), Genetic Algorithm (GA), and Grey Wolf Optimizer (GWO) to validate its effectiveness. Results show that the MMSO-optimized MLP-ANN achieved an 86.34% recall, 98.96% precision, and 92.29% accuracy, while minimizing the Mean Squared Error to 0.0206. The MMSO-optimized MLP-ANN model achieved competitive classification performance compared to the other algorithms evaluated; although no statistically significant differences in recall were observed among the optimizers (p = 0.22), the MMSO achieved the lowest MSE (0.0206). The MMSO was the only algorithm capable of discovering a four-layer architecture hidden within the same search space, evidencing superior exploration of deeper architectural regions of the solution space. These findings demonstrate the model's capacity to predict microgrid operational modes under variable conditions, highlighting the potential of integrating bio-inspired algorithms with neural networks for energy management systems. This approach could enhance the efficiency and reliability of integrating renewable energy sources into dynamic energy systems.

KEYWORDS: Microgrids; binary classification; deep learning artificial neural network; mosquito mating swarm optimizer; operational mode prediction

1 Introduction

The increasing penetration of renewable energy sources (RES) in the form of hybrid electric microgrids (MGS) poses new challenges in the management and stability of electricity systems. MG is defined as distributed generation systems comprising renewable energy generation technologies that can operate connected to the grid or independently in isolated mode, ensuring local supply even in remote rural areas [1].

This ability to disconnect and operate independently of the grid provides resilience against central power grid failures, allowing critical loads to be maintained by islanding during contingencies [2]. However, control in hybrid MGs, with a mix of renewable energies, storage, and conventional generation, faces challenges not only of the variability and intermittency in the generation of renewable energy sources but also entail a challenge in control and management to achieve the effective coordination of all systems, which makes it challenging to predict and manage the operation modes optimally [3]. Suppose operating mode transitions are not detected or managed on time. In that case, the MG can experience power quality issues and even disruptions to the continuous supply of energy, which can become critical for certain loads. For example, a late switch to island mode after a fault can cause voltage and frequency instability and even damage equipment in critical facilities within a community [4]. Therefore, it is crucial to have systems that accurately predict the operating mode, anticipate changes, and apply control strategies that maintain the microgrid's stability

Although AI-based approaches have demonstrated significant advantages in terms of accuracy and adaptability, challenges remain. For example, traditional ANN models require manual hyperparameter configurations, such as learning rate or regularization, which can be tedious and suboptimal in dynamic environments. In addition, the high dimensionality of the MG operation data, combined with fluctuations in environmental variables such as solar irradiance or wind speed, requires algorithms capable of balancing exploration and exploitation in complex search spaces. This underscores the need for bio-inspired optimizers that automate parameter adjustment, thereby improving predictive robustness across varying conditions.

Traditionally, the detection of changes in the state of MGs (e.g., island detection) has been performed using passive or active methods based on local measurements of electrical parameters. However, in recent years, there has been a growing interest in artificial intelligence (AI) approaches for this task [5]. Various machine learning methods have been investigated, including artificial neural networks (ANNs), support vector machines (SVMs), decision trees, and fuzzy logic, among others, to classify operational mode events for electric MGs [6–8]. These intelligent methods can overcome some shortcomings of traditional techniques for operation modes detection schemes, for example, by avoiding the need to set thresholds manually and by improving detection speed and efficiency with less human intervention and better adaptability to changing environmental conditions [9].

In particular, ANN model-based techniques can learn nonlinear features of complex systems from experimentally measured data, such as voltages, frequencies, powers, and so on, thus identifying subtle variations in system parameters that would indicate the optimal time for an operating mode change [9] For example, in [10], it was shown that ANN approaches achieve high accuracy in island detection even with multiple distributed sources. However, a major challenge of these intelligent methods is the high computational load and algorithmic complexity [11]. Their sensitivity to the hyperparameter configuration of ANN models. Recent studies in Deep Learning have shown that architectures incorporating attention mechanisms and recurrent structures, such as multi-LSTM networks, that achieve 99.3% accuracy in islanding detection [12] and Attention-LSTM models that reach 98.4% with enhanced capabilities for extracting temporal features [13], can outperform conventional MLP networks in sequential classification tasks. Other studies have demonstrated that transformer-based architectures with domain adaptation capabilities exhibit transferability across domains in classification problems, achieving over 82.04% accuracy under domain shift conditions [14]. However, these deep learning approaches typically require extensive, standardized datasets with expert labeling [15], as well as significant computational resources that may be impractical for real-time deployment on embedded controllers for MG systems. In contrast, MLP architecture offers advantages in interpretability, computational efficiency, and rapid convergence with limited-sized tabular data, motivating their optimization through bio-inspired algorithms as a viable alternative for resource-constrained

applications. The effectiveness of an ANN model depends on its structure, specifically the number of layers and neurons, and on appropriately tuned training parameters, such as the learning rate and regularization coefficients, which must be appropriately tuned to achieve robust performance. Manually determining these hyperparameters can be challenging, especially in complex problems due to the large number of possible parameter combinations. In this context, automatic hyperparameter optimization techniques have emerged using bio-inspired optimization metaheuristic algorithms to improve the accuracy of AI models based on ANN [16], metaheuristic optimization efficiently explores the space of solutions, configurations of the ANN in this case, avoiding being trapped in local optima and considering interdependencies between hyperparameters, resulting in more precise and generalizable models as shown in [17]. Recent literature has demonstrated the successful application of metaheuristic algorithms for hyperparameter optimization and meta-optimization, including Particle Swarm Optimization (PSO), Genetic Algorithm (GA), and, more recently, the Grey Wolf Optimizer (GWO) [18,19]. For instance, the GWO algorithm was applied to ANN models to predict wind speed for RES applications in India [20] and achieved a competitive performance due to its hierarchical hunting strategy, inspired by wolf packs, which enabled effective exploration and exploitation. Nevertheless, despite the reported advancements, there is still an opportunity to explore other bio-inspired metaheuristics that could offer enhanced performance in ANN improvement. Hybrid optimization algorithms that integrate evolutionary and swarm-intelligence characteristics can achieve faster convergence and higher accuracy in complex optimization scenarios [21]. In this regard, quantum swarm intelligence algorithms are promising by integrating principles of quantum computing with metaheuristic optimization techniques. The Quantum Particle Swarm Optimization (QPSO) algorithm replaces the traditional PSO velocity vector with quantum probability distributions, employing the superposition principle to allow particles to explore multiple potential states [22]. Recent applications of QPSO in energy programming for multi-energy community systems with electric vehicle aggregators have shown operational cost reductions of up to 9.67% compared to traditional approaches [23]. For their part, the authors of [24] propose a game-theoretic framework combined with improved QPSO to optimize energy programming in multi-energy community systems, achieving significant reductions in operational costs through chaotic perturbation and adaptive quantum behaviors. These approaches demonstrate the potential of advanced swarm algorithms to address complex energy management problems, providing a relevant methodological precedent for the application of the MMSO proposed in this work.

Effective system operation must account for resource availability, historical behavioral patterns, and user energy demand profiles [25]. However, traditional control strategies generally lack adaptability to rapidly changing operating conditions and are unable to optimize performance in real time, resulting in reduced reliability and suboptimal energy utilization. To overcome these limitations, mathematical models play a key role in characterizing the thermal behavior of bifacial PV modules and guiding the design and operation of efficient energy systems [26].

Recent advances have integrated predictive thermal modeling with AI-driven forecasting and optimization techniques for more intelligent microgrid operation. For instance, cluster-based demand prediction using Grey Wolf Optimizer (GWO) combined with multilayer perceptrons has been proposed to improve forecasting accuracy in dynamic environments [27]. Similarly, other studies have applied genetic-algorithm-based approaches to increase microgrid profitability through optimal storage integration [28].

Motivated by the above, this work proposes a model based on a Multilayer Perceptron (MLP) neural network optimized using a new generation of bioinspired algorithms, specifically the Mosquito Mating Swarm Optimization (MMSO). The objective is to improve the accuracy of predicting operating modes in hybrid microgrids, especially in maximizing the capacity to detect critical events, which involves achieving a high recall to avoid dangerous false negatives. The MMSO algorithm is chosen due to its balanced

exploration-exploitation capabilities and dynamic adaptation. The proposed MMSO is an evolutionary algorithm inspired by the swarming and mating behaviour of mosquitoes in nature, which integrates random "zigzag" flight movement stages with genetic operators of crossing and mutation stages to search for optimal solutions in complex spaces, combining the strengths of Particle Swarm Optimization (PSO) and Genetic Algorithm (GA). Unlike PSO, which depends solely on the movement of particles based on velocity and influenced by the best personal and global positions with fixed rules, and GA, which operates with discrete crossover and mutation operations, the MMSO integrates both paradigms into a two-stage optimization strategy. Conventional Swarm Intelligence (SI) algorithms govern particle movement through mathematical equations; in MMSO, a dynamic zigzag movement is incorporated to enhance exploration in complex search spaces. The zigzag mechanism shares theoretical foundations with Lévy flight strategies, as shown in recent hybrid optimization algorithms such as the Hybrid adaptive Wolf-Particle swarm optimization (HAWPSO) algorithm [29], where the authors demonstrated that nonlinear perturbations significantly enhance escape from local minima. Additionally, MMSO employs self-adjusting coefficients (α, β, γ) that are dynamically balanced for exploration and exploitation, similar to strategies that have proven effective for tuning ANN architectures [30]. The reproductive stage of MMSO, which involves crossover and mutation among the elite mosquitoes of each generation, adds mechanisms of genetic diversity absent in purely swarm-based approaches.

From a methodological standpoint, the present study employs an advanced approach that begins by defining the ANN optimization process through MMSO, including the number of hidden layers, the number of neurons per layer, the learning rate, and the L2 regularization coefficients. The optimization algorithm iteratively generates multiple candidate configurations, which are evaluated through k-fold cross-validation. Each ANN candidate is trained and validated using experimental data collected from a hybrid microgrid located at the Renewable Energy Laboratory of the Polytechnic University of Valencia. These data include electrical variables such as renewable generation power, battery state of charge, grid power exchange, voltage, and frequency at the point of common coupling (PCC). The recall metric, which indicates the model's ability to accurately detect critical operational modes, is the primary performance criterion guiding the MMSO optimization process. The novelty of this study arises not only from the introduction and application of the MMSO algorithm but also from its comprehensive search approach, exploring ANN structural complexity and training parameters simultaneously applied to real-world renewable energy challenges such as MG operation mode estimations using data from an experimental MG located in the Renewable Energy Laboratory (LabDER) of the Universitat Politècnica de València (UPV), Spain.

The rest of the paper is structured as follows: Section 2 details the methodology, explaining the architecture of the proposed ANN model, the implementation of the MMSO algorithm, and the experimental data used for model training and validation. In Section 3, the results are presented and discussed, evaluating the predictive performance of the optimized ANN relative to alternative optimization approaches and providing insights into the implications for microgrid management. Finally, Section 4 presents the main conclusions and future research opportunities emerging from this study

2 Methodology

This section outlines the methodology employed for developing a predictor of the operational state of MG based on an MLP ANN, whose structure is optimized by the novel bio-inspired algorithm, Mosquito Mating Swarm Optimization (MMSO). The methodological process comprises three main stages: (1) selection and design of the MLP-type ANN as a classifier, (2) detailed implementation of the MMSO algorithm to optimize the architecture and hyperparameters of the MLP ANN, and (3) performance

evaluation of the proposed optimized MLP ANN model. The Fig. 1 illustrates the general methodology for implementing the proposed model and evaluating it.

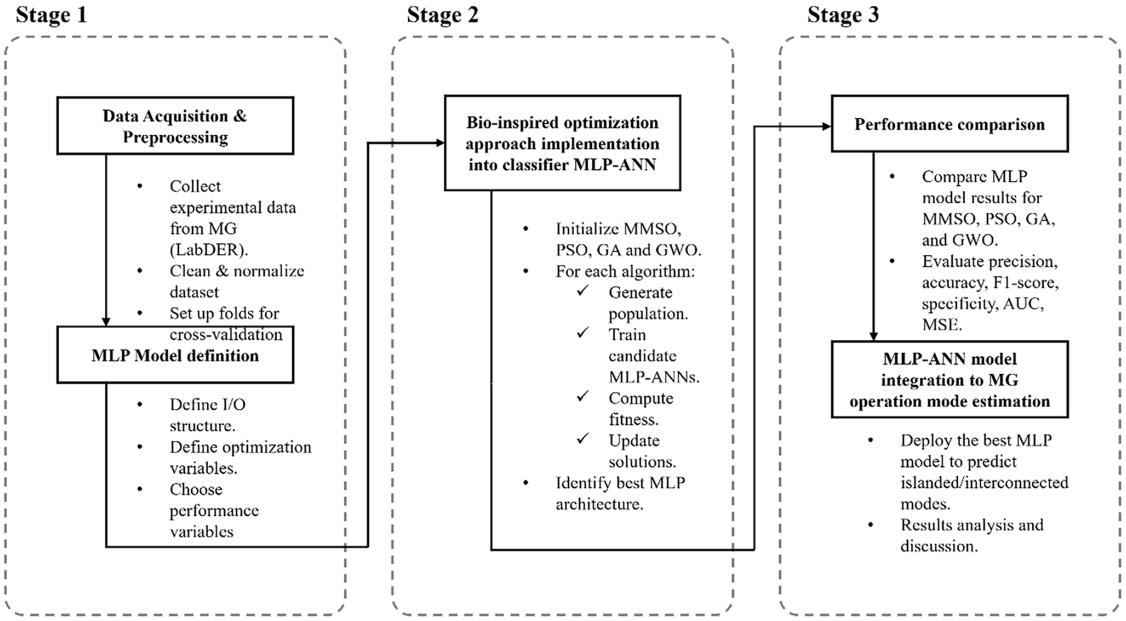

Figure 1: The overall methodology for the proposed optimized MLP-ANN model for MG operation mode estimation.

2.1 ANN MLP

An MLP (Multi-Layer Perceptron) ANN is a machine learning model that consists of a set of neurons organized into layers: an input layer, one or more hidden layers, and an output layer. Each neuron applies a nonlinear transformation to a weighted linear combination of its inputs, as defined in Eq. (1).

$$y_j = f\left(\sum_{i=1}^{n} w_{i,j}x_i + b_j\right) \tag{1}$$

where y_j is the output of the j neuron, x_i are the input of the neurons, $w_{i,j}$ are the neuron weights of connection, b_j is the neuron bias and f is the activation function.

In this work, the implemented MLP ANN employs the sigmoidal tangent activation function in the hidden layers defined by Eq. (2).

$$f(x) = \tan sig(x) = \frac{2}{1+e^{-2x}} - 1 \tag{2}$$

For the output layer of the MLP, a sigmoid logistic function is used to achieve the classification of the operating mode of the MG in binary mode (0, 1) according to Eq. (3).

$$f(x) = \log sig(x) = \frac{1}{1+e^{-x}} \tag{3}$$

An output value of 0 indicates the operation mode connected to the power grid, indicating normal interconnection with the main power grid, while an output value of 1 indicates the island operation mode, in which the MG operates autonomously, disconnected from the main power grid. For the ANN training stage, the MLP aims to adjust the weights of the neurons in such a way that the difference between the values of

the output layer and the targets of the training dataset is as minimal as possible; for this, the mean squared error (MSE) is used as the objective function of this stage, as defined in Eq. (4).

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{4}$$

where in Eq. (4), y_i is the target output value, $\hat{y}_i$ is the predicted output value, and N is total number of samples used during the training stage of the ANN. An important aspect in the training of ANN models is the implementation of techniques that avoid overfitting; for this, in the proposed model, the L2 type regularization is implemented for the function of the MSE, according to Eq. (5).

$$j(w) = MSE + \lambda \sum_{i} w_i^2 \tag{5}$$

Being λ in Eq. (5), the L2 regularization coefficient is implemented and automatically adjusted during the optimization process by the MMSO algorithm for the optimized ANN.

2.2 MMSO Algorithm

The MMSO algorithm is a bio-inspired algorithm that combines evolutionary strategies, such as mutation and crossing, with collective swarm behaviors, simulating mosquitoes' mating and flight patterns in nature. Its goal is to comprehensively explore the search space to optimize complex structures and parameters such as those of artificial neural networks.

The first generation of mosquitoes initialization is described by Eq. (6), for each mosquito i, where x_i^0 is its initial position for the upper UB and lower bounds LB related to each optimization variable.

$$x_i^0 = LB + (UB - LB) \cdot rand\,(1, nvars) \tag{6}$$

After the mosquito population has been created and distributed in the search space, the position of the mosquitoes is updated, and the movement of the particles along the search space is carried out by Eq. (7), which also incorporates the zigzag motion of the mosquitoes' flight.

$$x_i^{t+1} = x_i^t + \alpha \cdot \left(x_{best} - x_i^t\right) + \beta \cdot \left(x_{best} - x_i^t\right) + Z\left(x_i^t\right) \tag{7}$$

In Eq. (7), x_i^{t+1} is the updated position of mosquito i for the iteration $t+1$; x_i^t is the current mosquito position; coefficients α and β are global and personal influence factors which dynamically adjusted during algorithm iterations; and $Z\left(x_i^t\right)$ is the mosquito zigzag fight pattern function, defined by next Eq. (8).

$$Z\left(x_i^t\right) = zigzagFactor \cdot \sin(2\pi \cdot r \cdot x_i^t) \tag{8}$$

The $zigzagFactor$ term is a coefficient determining the amplitude of the zigzag movement during mosquito flight. This coefficient decreases along with iterations. The zigzag flight pattern implemented in the MMSO serves a dual purpose in the optimization process. First, it introduces controlled stochastic perturbations to prevent premature convergence to local optima, analogous to the Lévy flight mechanisms employed in next-generation hybrid optimizers [30], where nonlinear perturbations have been shown to significantly enhance the escape of local optima in multimodal search spaces. The sinusoidal formulation implemented by Eq. (8) generates nonlinear displacement vectors, allowing exploration of neighborhoods of solutions that would be inaccessible with purely linear position updates, such as those employed in the traditional PSO algorithm. Second, the zigzagFactor decreasing over the iterations, implicitly defined as

$zigzagFactor(t) = zigzagMax \cdot (1 - t/T_{\max})$, implements an adaptive balance between exploration and exploitation that translates into perturbations of greater magnitude in the initial iterations to promote a wide coverage of the search space, while perturbations when they progressively decrease in later iterations facilitate a refined exploitation in the vicinity of promising solutions.

MMSO equations incorporate the alpha α, beta β, and gamma γ dynamic coefficients. A high value of α intensifies mosquitoes movement to the best overall solution, starting with a low value promoting exploration and gradually increasing to encourage exploitation as the algorithm converges to the best global solution. The β coefficient is the attraction to the best personal position of each mosquito. Each mosquito "remembers" its best position to guide itself in the next movements. A high value of β increments the influence of personal best position on the mosquito's flight stage, promoting the exploration and helping to avoid local optimal. The γ coefficient is implemented in the MMSO for the crossover and mutation operations in the mating stage of the mosquitos. The variation during iterations of the α, β and γ coefficients is defined by Eqs. (9)–(11).

$$\alpha = \alpha_{initial} + \left(\alpha_{final} - \alpha_{initial}\right) \cdot \frac{t}{T_{\max}} \tag{9}$$

$$\beta = \beta_{initial} + \left(\beta_{final} - \beta_{initial}\right) \cdot \frac{t}{T_{\max}} \tag{10}$$

$$\gamma = \gamma_{initial} + \left(\gamma_{final} - \gamma_{initial}\right) \cdot \frac{t}{T_{\max}} \tag{11}$$

In the crossover stage, the crossover and mutation of the selected mosquitoes is done by the function of Eq. (12).

$$offspring_{starting}(i) = \begin{cases} parent_1[i] & if\ i \le c \\ parent_2[i] & if\ i > c \end{cases} \tag{12}$$

In Eq. (12), the c term represents the crossing point for the mosquitoes' genetic material. The crossing point c is an interval between $1 \le c <$ the number of variables of the optimization problem. The offsprings generation depends on two stages, first crossing between parents, and the second in the mutation of genes. The Eq. (13) defines how the final offsprings of mosquitos are obtained for each generation after the mutation, and the Eq. (14) defines the integration of the dynamic γ coefficient into the mutation stage, n is a random number between 0 and 1.

$$offspring_{final}(i) = offspring_{starting}(i) + mutation(i) \tag{13}$$

$$mutation(i) = \gamma \cdot random(n) \tag{14}$$

Based on the above mentioned equations, the pseudocode implementation of the MMSO algorithm is shown in Algorithm 1.

The following section details the integration of the MMSO with the MLP-type ANN for its optimization in the context of the problem of estimating optimal modes of operation for MG.

Algorithm 1: Pseudocode for the MMSO proposed optimization algorithm

1. Initialization:
   1.1. Randomly distribute mosquitoes within the search space boundaries. Evaluate their fitness, identify the best global solution, and initialize α, β, and γ coefficients.
2. Iterative Process—FOR each iteration t = 1 to maxIter:
   2.1. Update Parameters: Adjust α, β, and γ dynamically based on iteration progress.
   2.2. Update Non-Reproductive Mosquitoes:
   FOR each non-reproductive mosquito:
   - Update position influenced by the best global and a random mosquito (modulated by α, β).
   - Apply zigzag movement for exploration and correct boundary violations.
   - Evaluate the new fitness value.

   2.3. Reproduction and Mutation:
   - Select reproductive mosquitoes based on fitness.
   - Perform crossbreeding and mutate offspring (regulated by γ).
   - Replace worst mosquitoes with better offspring if applicable.

   2.4. Update Global Best: Update the best global solution if a superior one is found.
   2.5. Go to 2 if stop criteria are not fulfilled; if so, go to 3.
3. Output Results: Return the best position found and fitness value.

2.3 Optimized MMSO MLP-ANN Proposal

In the proposed approach for this paper, an integrated optimization framework is developed for combining a Multi-Layer Perceptron (MLP) neural network with the novel Mosquito Mating Swarm Optimization (MMSO) algorithm to predict the operational modes of microgrids under different conditions. The framework is designed to optimize simultaneously the architecture and the training hyperparameters of the related MLP, enhancing its predictive performance. To reinforce the validity of the proposed MMSO-based strategy, the same optimization procedure has been implemented using three widely adopted metaheuristics, PSO, GA, and GWO, for validation and comparison. The following Fig. 2 shows the overall architecture of the proposed optimized MLP-ANN model.

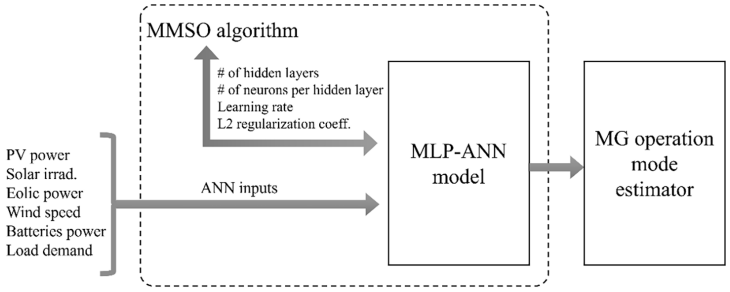

Figure 2: Diagram of the MLP-ANN optimized model using the MMSO algorithm.

As shown in Fig. 2 above, the model proposed in this work is based on an MLP-ANN model of binary classifier type, which aims to establish recommendations of mode of operation for a hybrid MG. The MLP-ANN is optimized in its structure and hyperparameters by the MMSO within an internal optimization loop, which at the same time is fed with the inputs of the model, which are the power of the solar PV array,

the solar irradiance, the power generated by the wind, the wind speed, the power in the batteries and the energy demand in the MG. Once the data was obtained, it was preprocessed. Data preprocessing included: (1) elimination of physically implausible outliers, (2) linear interpolation for missing values in gaps smaller than five samples, excluding segments with larger gaps, and (3) min-max normalization to the range [0, 1] for all input variables. The resulting validated dataset comprises 5605 instances. Based on this structure, the optimization problem and its parts are defined.

The optimization variables vector X represents each candidate solution encoded, representing the ANN configuration and structure. The optimization vector is defined by Eq. (15).

$$X = [L,\ n_1,\ n_2, \ldots,\ n_L,\ \eta,\ \lambda] \tag{15}$$

where L is the number of hidden layers of the MLP ANN, n_j is the number of neurons in the j^{th} hidden layer for $j = 1, \ldots, L$; η is the learning rate, and, λ is the L2 regularization coefficient. The vector X unifies the optimization both architecture and the training parameters of the MLP ANN for the optimization through the MMSO algorithm.

The proposed MMSO MLP-ANN model's primary objective is to maximize the ANN's recall in predicting critical operational modes. For a given candidate solution X, the corresponding ANN is trained and evaluated on experimental data using k-fold cross-validation. The training objective of the ANN is based on minimizing the MSE augmented with L2 regularization. The cost function for the ANN is defined in Eq. (16).

$$j(w) = MSE + \lambda \sum_i w_i^2 \tag{16}$$

where $j(w)$ is the objective function integrated into the MMSO, MSE is the mean squared error, λ is the regularization L2 coefficient and w_i are the ANN weights. The recall metric was obtained to evaluate each candidate optimal solution, as defined in Eq. (17).

$$Recall = \frac{TP}{TP + FN} \tag{17}$$

With TP representing the true positives and FN the false negatives. The overall objective function of the ANN optimization process is therefore defined in the next Eq. (18).

$$F_{\min} \rightarrow F\left(\overline{X}\right) = 1 - Recall \tag{18}$$

The minimization of $F\left(\overline{X}\right)$ corresponds to the maximization of the recall. This formulation is integrated into the MMSO, which is structured to minimize the objective function according to the optimization process of the MMSO algorithm described in previous Eqs. (6)–(14).

The ANN is trained using these parameters on the entire experimental dataset obtained from the Renewable Energy Laboratory (LabDER) at the Polytechnic University of Valencia. The final model is evaluated on unseen data using metrics such as recall, precision, accuracy, and MSE, ensuring that the integrated approach yields a robust predictor of microgrid operational modes. In order to contextualize the performance of the proposed method, all the optimization algorithms (MMSO, PSO, GA, and GWO) were run under identical dataset partitions and parameter bounds. This uniform experimental setup systematically compares their respective convergence behaviors and classification metrics.

To illustrate in detail the integration of MMSO into MLP-ANN processing, Algorithm 2 below presents the pseudocode for MLP-ANN integration.

Algorithm 2: Pseudocode for the MMSO-MLP integration

INPUT: Dataset (inputs, outputs), search bounds (lb, ub), population size N, maximum iterations Max_iter
OUTPUT: Optimal MLP configuration, trained network, performance metrics

1. INITIALIZATION
   - 1.1. Define optimization vector X = [L, n_1, n_2, ..., n_L, η, λ]
     where: L = number of hidden layers
     n_j = neurons in layer j (j = 1, ..., L)
     η = learning rate
     λ = L2 regularization coefficient
   - 1.2. Initialize N mosquitoes randomly: x_i^0 = lb + (ub − lb)·rand(1, dim)
   - 1.3. Set initial coefficients: α_0, β_0, γ_0, $zigzagFactor_0$
2. FITNESS EVALUATION (for each mosquito i)
   - 2.1. Define solution vector → MLP architecture [n_1, ..., n_L]
   - 2.2. Build MLP network: patternnet([n_1, ..., n_L], trainscg)
   - 2.3. Configure ANN: tansig (hidden layers), logsig (output), L2 = λ
   - 2.4. Evaluate fitness using stratified 5-fold cross-validation:
     fitness(i) = mean(Recall across k folds)
   - 2.5. Update global best if fitness(i) > best_fitness
3. MMSO ITERATIVE PROCESS (for t = 1 to Max_iter)
   - 3.1. Update dynamic coefficients α, β, γ according to Eqs. (9)–(11)
   - 3.2. NON-REPRODUCTIVE PHASE (Exploration):
     FOR each non-reproductive mosquito i:
     - Compute zigzag movement: $Z(x_i^t)$ according to Eq. (8)
     - Update position: x_i^{t+1} according to Eq. (7)
     - Apply boundary constraints
     - Evaluate fitness (Go to Step 2)
   - 3.3. REPRODUCTIVE PHASE (Exploitation):
     - Select top 75% mosquitoes by fitness as reproductive
     - FOR each reproductive pair ($parent_1$, $parent_2$):
       - Crossover according to Eq. (12)
       - Mutation according to Eqs. (13) and (14)
       - Replace worst mosquito if offspring is better
   - 3.4. Update global best solution
   - 3.5. Store convergence: Convergence_curve(t) = best_fitness
4. FINAL TRAINING
   - 4.1. Build MLP with optimal configuration from X_best
   - 4.2. Train on complete dataset with k-fold cross-validation
   - 4.3. Compute metrics: Recall, Precision, F1, Specificity, Accuracy, AUC, MSE
   - 4.4. Generate confusion matrix and ROC curve
5. RETURN optimal MLP architecture, hyperparameters, trained network

Additionally, Fig. 3 presents the flowchart corresponding to the MMSO-MLP optimization framework, illustrating the logical sequence of operations and the iterative cycles of the algorithm.

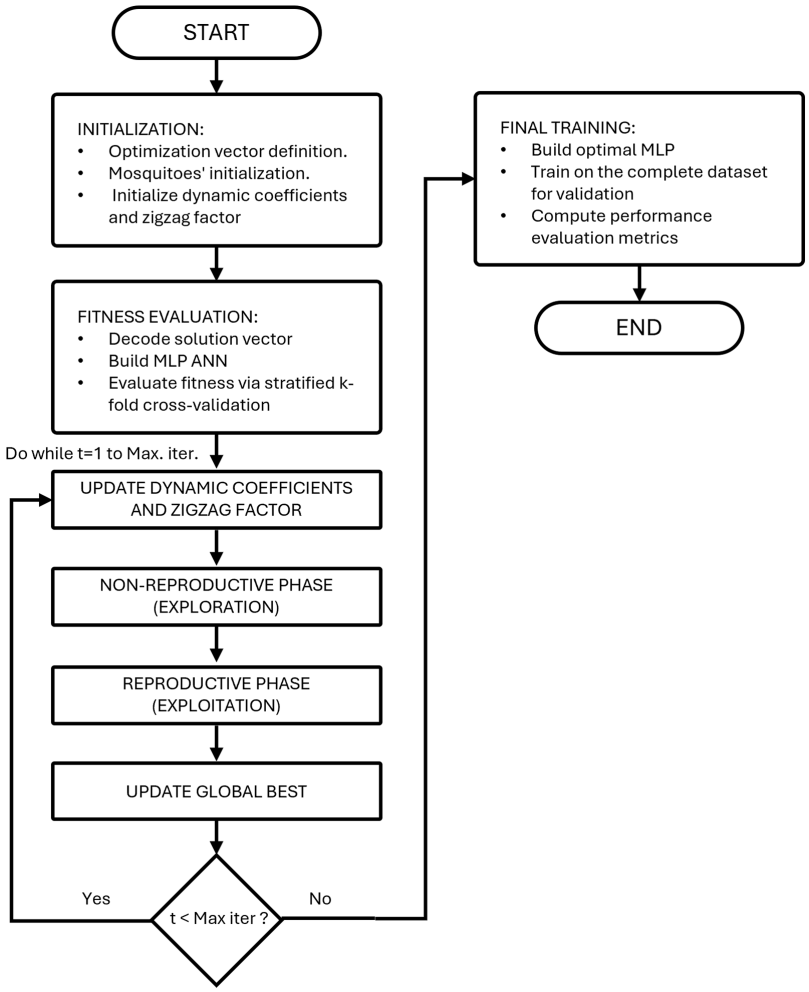

Figure 3: Flowchart of the MMSO-MLP optimization framework showing the integration of the mosquito mating swarm optimization algorithm with the MLP training process.

3 Results and Discussion

This section provides an in-depth analysis of the proposed MLP-ANN classifier optimized by MMSO. It compares it to three different bio-inspired algorithms, PSO, GA, and GWO, to predict the operational mode for a hybrid microgrid.

3.1 MLP Configurations Identified by the Bio-Inspired Optimization Algorithms

The proposed MLP-ANN was optimized using four different bio-inspired algorithms to identify the best optimizer for the application. Each algorithm explored a predefined solution space and evaluated different MLP-ANN architectures and hyperparameters. During optimization, each method iteratively generated candidate solutions, trained and validated the corresponding MLP, and updated the search according to its specific evolutionary or swarm-based rules. The Table 1 shows the preset search space and configurations of the implemented algorithms to optimize the proposed MLP-ANN.

Table 1: Configuration parameter for the MLP-ANN optimization.

Parameter	Value
Hidden layers	0–10
Neurons per hidden layer	0–300
Learning rate	0.0001–2
L2	0.01–0.05
Search agents (MMSO, PSO, GA, GWO)	1–100

As mentioned, the proposed approach of an MLP-ANN optimized by the new proposed MMSO algorithm was validated against PSO, GA, and GWO to compare performances. After performing the simulations and training of the MLP-ANN using all the algorithms, the following Table 2 was obtained, which summarizes the results of the optimizations of the hyperparameters and architecture of the MLP-ANN to obtain the best recall in the prediction of the operating state of the experimental MG.

Table 2: Best MLP-ANN configurations obtained for each optimization algorithm.

Algorithm	Hidden Layers and Neurons Per Layer	Learning Rate	L2	Computing Time (min)	MSE
MMSO	[8, 173, 106, 54]	0.0001	0.0100	60	0.020618
GWO	[261, 231]	1.1661	0.0377	57	0.023577
PSO	[300]	1.2542	0.0145	42	0.028834
GA	[12, 95]	1.1823	0.0435	43	0.021806

The above Table 2 compiles the resulting optimal configurations after the optimization process of each of the algorithms, illustrating that the MMSO algorithm converged on a four-layer architecture with the architecture of [8, 173, 106, 54], paired with a relatively small learning rate (0.0001) and a moderate L2 value (0.0100). These settings suggest a preference for a more intricate representation, potentially enabling the model to capture subtle nonlinearities in the MG dataset. The PSO algorithm, in contrast, concentrated on a single wide hidden layer of 300 neurons with a larger learning rate (1.2542), indicating a more straightforward structure but faster parameter updates. The GA favored a two-hidden-layer design of [12, 95] and a similarly high learning rate of 1.1823, while GWO also settled on two hidden layers but with larger neuron counts [261, 231] and a learning rate of 1.1661. The total computing time ranged from 42 to 60 min, with PSO finishing the fastest but simultaneously reporting the highest mean-squared error (MSE). The proposed MMSO algorithm, though not the quickest, obtained the smallest MSE (0.020618), providing an early indication that deeper networks might yield more precise decision boundaries. The analysis of trainable parameters reveals differences in model complexity. The MMSO algorithm's four-layer architecture contains 25,890 trainable parameters, compared to 62,581 for GWO (two layers), 2401 for PSO (single layer), and 1415 for GA (two layers). Notably, the GWO's architecture, despite having only two hidden layers, requires the highest number of parameters due to its wide layer configuration (261 and 231 neurons), whereas MMSO achieves superior MSE performance with a deeper but narrower topology. This suggests that architectural depth, rather than sheer parameter count, may be more effective for capturing the nonlinear dynamics inherent in microgrid operational data.

3.2 Cross-Validation Performance and Statistical Analysis

A stratified 5-fold cross-validation procedure for evaluating each optimized network on multiple independent partitions to reduce variance and to offer a more reliable picture of generalization for each MLP-ANN compared. The Table 3 displays the average metrics of recall, precision, F1-score, specificity, accuracy, and area under the ROC curve (AUC) and their standard deviations computed across the five k-folds. These metrics reflect each bio-inspired optimization algorithm's ability to correctly distinguish the "islanded" and "interconnected" classes for the MG operation mode, primarily emphasizing recall for capturing critical islanding events.

Table 3: Average k-fold performance metrics for the MLP-ANN.

Algorithm	Recall (%)	Precision (%)	F1-Score (%)	Specificity (%)	Accuracy (%)	AUC (%)
MMSO	85.90 ± 1.68	98.23 ± 1.05	91.64 ± 1.10	98.26 ± 1.06	91.72 ± 1.04	99.91 ± 0.12
PSO	85.93 ± 1.21	98.57 ± 0.65	91.81 ± 0.94	98.60 ± 0.64	91.90 ± 0.90	99.95 ± 0.05
GA	86.00 ± 1.37	99.53 ± 0.33	92.27 ± 0.87	99.55 ± 0.32	92.38 ± 0.80	99.99 ± 0.01
GWO	86.40 ± 1.63	99.11 ± 0.60	92.32 ± 1.03	99.13 ± 0.59	92.40 ± 0.97	99.96 ± 0.03

As shown in Table 3, the four optimization algorithms all achieved recall values between the 85.9% and 86.4% range, reflecting their shared capacity to detect islanded conditions with low false negative rates. Although GA and GWO show slight numerical improvements in recall (86.0% and 86.4%), the differences remain within approximately 1% of the MMSO and PSO results. A similar pattern emerges in precision, specificity, and accuracy, where GA and GWO surpass 99% in precision and specificity, while MMSO and PSO remain close behind at around 98%. The AUC values exceed 99.9% for all methods, confirming that each optimized MLP correctly separates the two operational modes across various decision thresholds.

A repeated-measures one-way ANOVA was conducted on the recall metric, using the individual fold-level data for all four algorithms, but the resulting p-value exceeded 0.05 ($p = 0.22$). This lack of statistical significance implies that the cross-validation differences between algorithms cannot be conclusively distinguished within the present dataset. Post-hoc Tukey analyses further confirmed overlapping 95% confidence intervals. Hence, despite the slight numerical variations, all four strategies robustly solve the classification problem. However, the deeper network arrived at by MMSO emerges as notable since it systematically explores more multi-layer topologies. While its recall does not exceed that of GA or GWO in this dataset, it appears well-suited to more complex or larger training scenarios, given its exceptionally low MSE during optimization. These findings emphasize that MMSO and PSO represent robust metaheuristics for discovering near-optimal MLP architectures. The final choice may thus be influenced by secondary factors, such as ease of parameter tuning, convergence speed, or system complexity, which MMSO appears to handle better than the other optimization algorithms. The Fig. 4 illustrates the recall performance with 95% confidence intervals for each optimization algorithm, providing a visual representation of this statistical equivalence.

After determining each algorithm's best hyperparameters, the MLP was retrained on the entire dataset of 5605 instances to generate final predictions. The following Table 4 shows the key classification metrics recall, precision, F1-score, specificity, accuracy, and AUC, along with the MSE for the proposed optimized MLP-ANN comparison for all optimization algorithms evaluated.

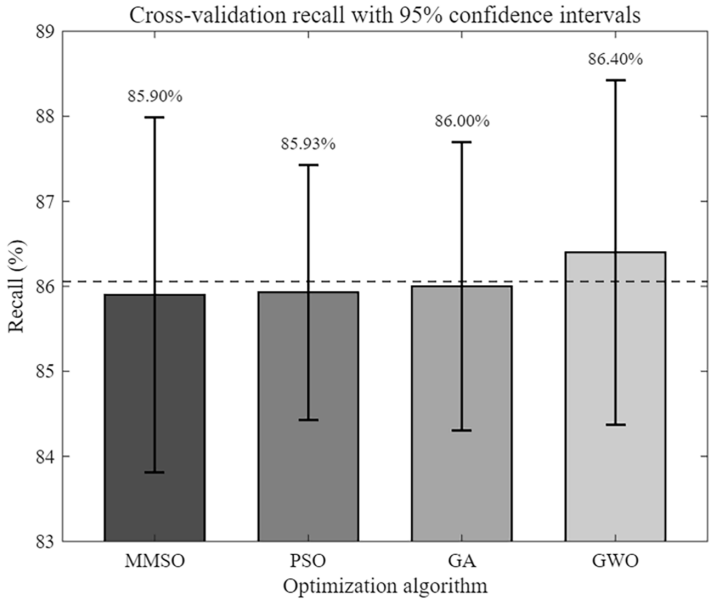

Figure 4: Cross-validation recall performance with 95% confidence intervals for each optimization algorithm.

Table 4: Key classification metrics for the proposed optimized MLP-ANN.

Algorithm	Recall (%)	Precision (%)	F1-Score (%)	Specificity (%)	Accuracy (%)	AUC (%)	MSE
MMSO	86.34	98.96	92.22	98.98	92.29	99.99	0.020618
PSO	86.07	98.80	91.99	98.83	92.08	99.96	0.028834
GA	86.44	99.73	92.61	99.73	92.70	99.99	0.021806
GWO	86.47	99.30	92.44	99.32	92.52	99.98	0.023577

The results shown in Table 4 are consistent with the cross-validation findings, with all four methods maintaining recall in the mid-86% range, accuracy near or above 92%, and extremely high AUC values. It is notable how MMSO registers the lowest MSE (0.020618), underscoring that its deeper architecture may offer a tighter fit to the global data distribution, even if its recall is not markedly distinct from its competitors. The precision achieved for the GA was 99.73%, and GWO achieved 99.30%, slightly surpassing the ~98% range of MMSO and PSO, indicating fewer false positives. Fig. 5 presents a comparative visualization of the MSE achieved by each optimization algorithm, ordered from lowest to highest error.

The MSE comparison in Fig. 5 above reveals differences in model fit across optimization algorithms. MMSO achieved the lowest MSE (0.0206), representing a 28.5% reduction compared to PSO (0.0288) and a 12.5% improvement over GWO (0.0236). This indicates that the MMSO-optimized architecture provides a tighter fit to the underlying data distribution, resulting in predictions closer to the actual target values. While GA achieved a competitive MSE (0.0218), the MMSO result demonstrates that its balanced exploration-exploitation strategy enables the discovery of network configurations that minimize prediction error more effectively than the benchmark algorithms.

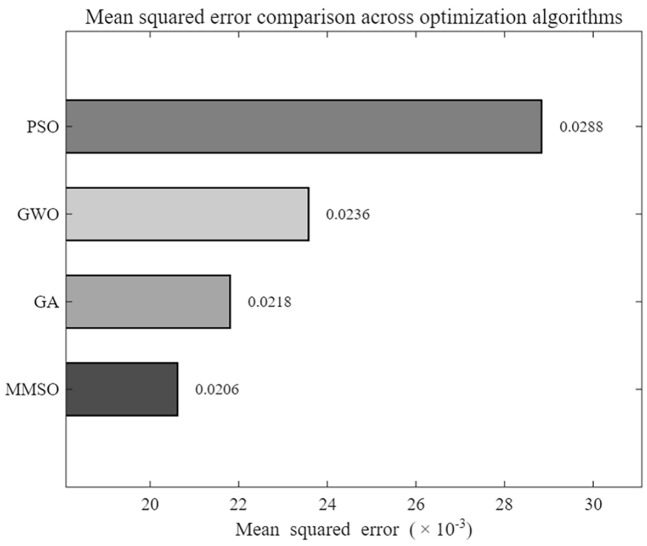

Figure 5: Mean squared error comparison across optimization algorithms.

The Fig. 6 illustrates the confusion matrices of MMSO, PSO, GA, and GWO for the entire dataset, emphasizing the low ratio of false negatives across all approaches. This is particularly important since failing to detect a legitimate island mode can compromise the resilience of a microgrid, potentially leading to adverse voltage or frequency events.

The consistently high precision achieved by the optimized MLP-ANN confirms that false alarms remain minimal, an essential consideration if unnecessary isolation disrupts the power supply or imposes additional switching losses. The strong alignment between the fold-by-fold statistics and the overall confusion matrices affirms that none of the algorithms overfit a particular subset of data and that all four trained MLPs achieve stable performance across the microgrid's operational range.

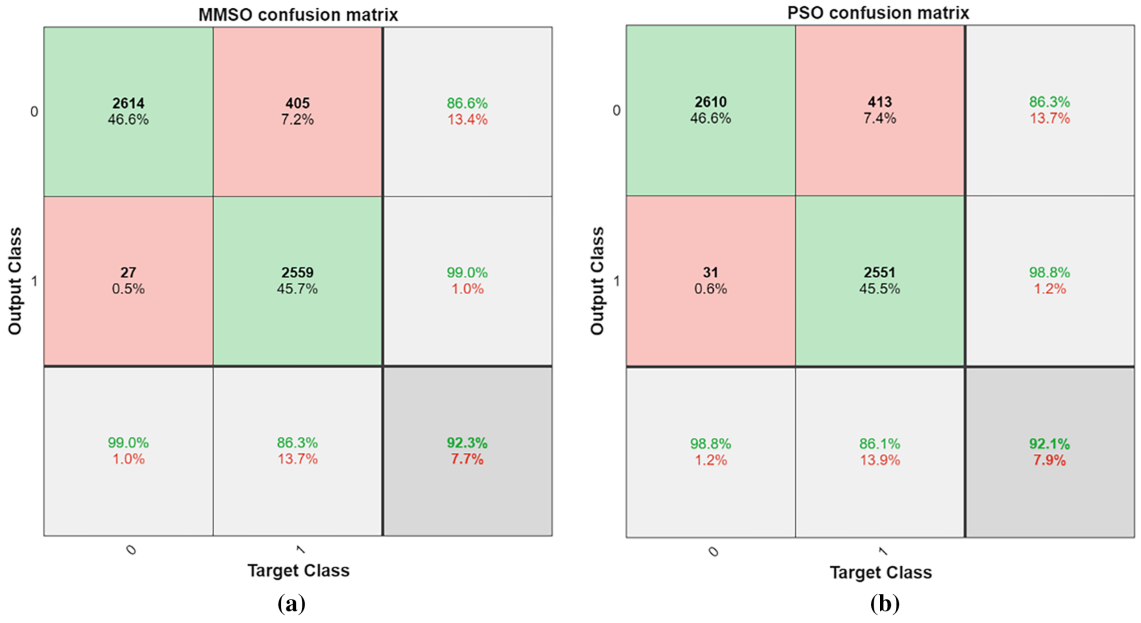

Figure 6: (Continued)

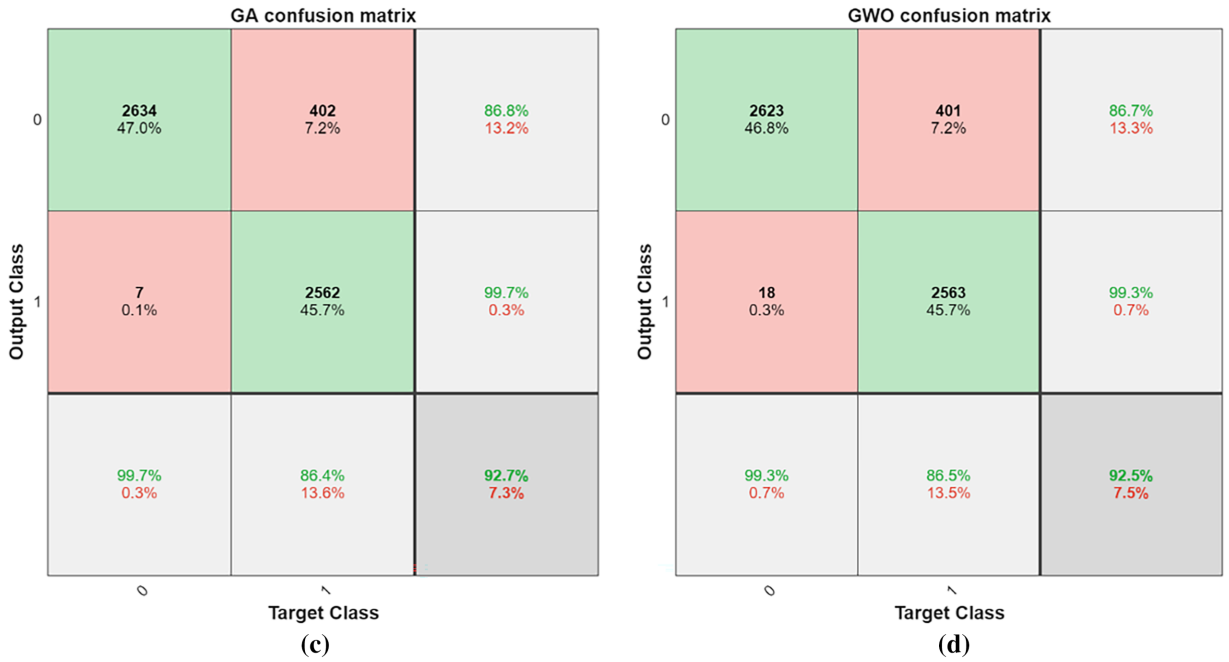

Figure 6: Confusion matrix from (**a**) MMSO, (**b**) PSO, (**c**) GA, and (**d**) GWO obtained after the MLP-ANN training.

3.3 Microgrid Operation Mode Estimation

The mid-80% recall documented in this study suggests that all four bio-inspired optimization techniques can effectively detect islanding events in a timely manner, thus reducing the risk of equipment damage or supply interruptions for critical loads. In addition, precision near or above 98% translates to minimal false positives that might otherwise lead to spurious islanding decisions. While GA and GWO algorithms achieve slightly higher recall and precision on the final dataset, the deeper MLP discovered by MMSO stands out for attaining the lowest MSE, hinting at a more thorough exploration of the optimal possible parameter spaces. This outcome aligns with the strategic design of MMSO, which combines "zigzag" exploration, swarm-based adaptation, and crossover–mutation steps to navigate the search space with a balance of exploration and exploitation. Although these advantages do not manifest as a substantial difference in recall or AUC in the present dataset, they hold promise for more demanding or expansive microgrid environments. Indeed, a deeper and more expressive network architecture may prove advantageous if the system evolves to include additional generation units or the operating conditions become increasingly heterogeneous. The relationship between architectural depth and prediction error is illustrated in Fig. 7, where bubble size represents the total number of neurons in hidden layers

Fig. 7 reveals an inverse relationship between network depth and MSE for the evaluated algorithms. MMSO uniquely discovered a four-layer architecture with 341 total neurons, achieving the lowest MSE, while PSO converged to a single-layer network with 300 neurons but yielded the highest error. Interestingly, GWO produced the largest network (492 neurons across two layers). This finding highlights MMSO's capability to explore deeper topological configurations that other algorithms did not reach, providing empirical evidence that its zigzag exploration mechanism and dynamic coefficient adjustment facilitate the discovery of more expressive architectures for complex classification tasks. The Fig. 8 shows the results of a practical case application for estimating the optimal mode of operation of the experimental MG located in the Renewable Energy Laboratory of the Polytechnic University of Valencia, Spain.

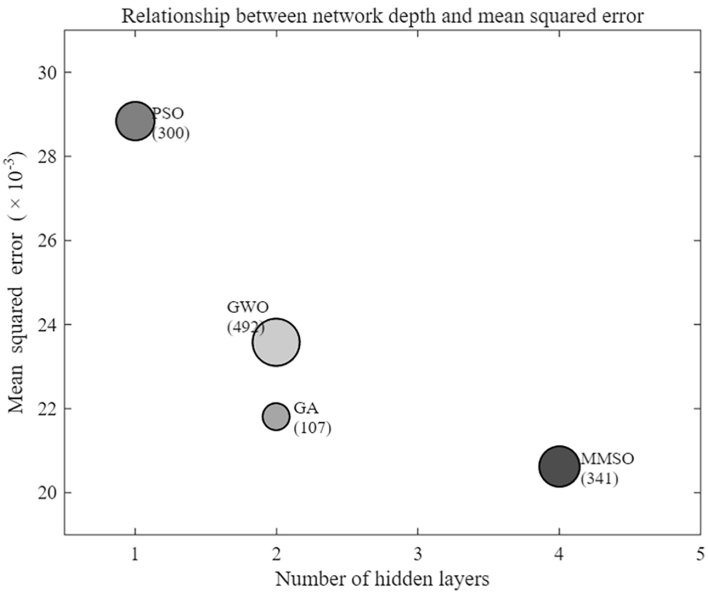

Figure 7: Relationship between network architectural depth number of hidden layers and mean squared error.

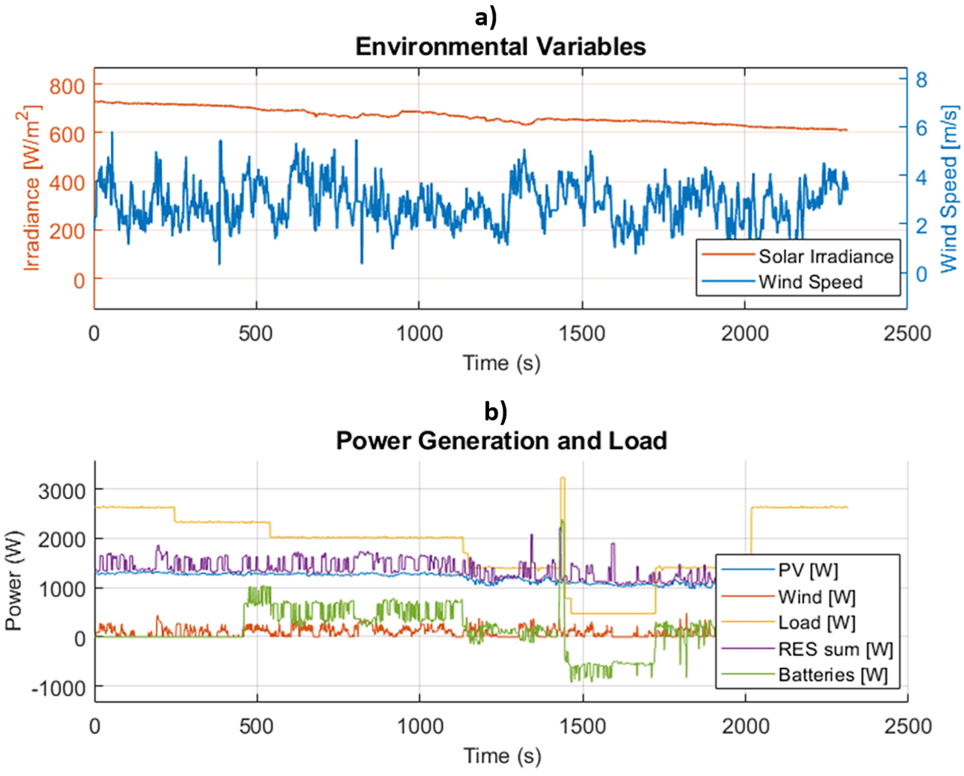

Figure 8: (Continued)

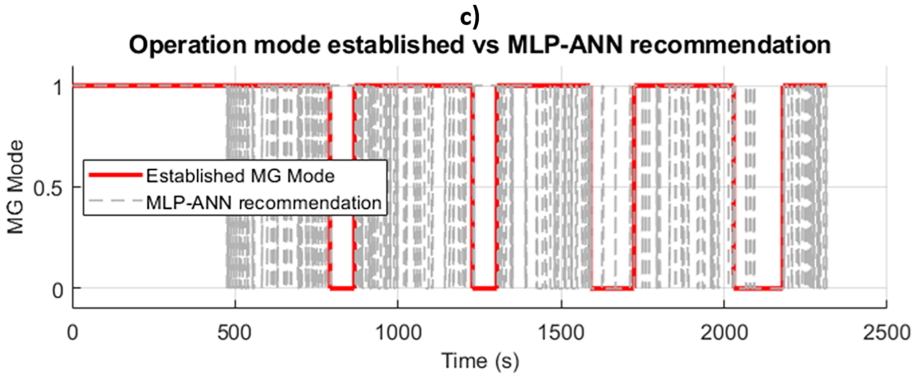

Figure 8: Plots of (**a**) environmental variables, (**b**) power generation and load from the MG and (**c**) optimized MLP-ANN model MG mode recommendation and established MG operation mode based on model recommendation.

As shown in Fig. 8, data were obtained from an experimental MG to validate the optimized MLP-ANN model proposed in this work; the model was then trained using the proposed MMSO. Panel (a) reflects the variability in environmental conditions, particularly solar irradiation and wind speed, while panel (b) illustrates the corresponding variations in power generation and load. To estimate the most appropriate mode of operation, the optimized MLP-ANN model, considering all variables, provides a recommendation, as shown in Fig. 8c by the grey line. As shown, the MLP-ANN recommendation exhibits significant differences between the connected mode (value of 1) and the isolated mode (value of 0). However, frequent changes between modes can damage the equipment involved, increase wear on switching mechanisms, and even render the process physically unviable. To address this issue, in addition to the optimized MLP-ANN model recommendation, a practical filtering layer was implemented that accounts for the minimum residence times in each mode. In this way, the red line in Fig. 8c represents the final, established MG mode. This additional management layer ensures that the microgrid does not oscillate excessively between modes, thereby balancing the predictive responsiveness of the MLP-ANN with the practical operational constraints that aim to protect the hardware and maintain stable power quality. From an operational perspective, the false negative rate of approximately 14% (corresponding to a recall of approximately 86%) warrants careful consideration. In microgrid applications, a false negative occurs when the system fails to recommend islanding when it should, potentially exposing critical loads to grid disturbances. However, the consistently high specificity (>98.8%) ensures that unnecessary islanding events, which could disrupt service and increase equipment wear, are minimized. This asymmetric error profile, which favors specificity over recall, may be appropriate for microgrids where false alarms incur higher operational costs than brief exposure to grid anomalies; however, the optimal trade-off depends on the specific application requirements.

Recent studies on power system classification have shown that the cost ratio between false negatives and false positives typically ranges from 4:1 to 5:1, reflecting asymmetric consequences where undetected critical events incur significantly higher operational and safety costs than false alarms [30]. In MG applications, false negatives occur when the system fails to recommend switching to island mode when grid conditions require it, potentially exposing critical loads to voltage and frequency disturbances, damaging sensitive equipment, and posing safety risks to maintenance personnel [3]. The consistently high specificity (>98.8%) achieved by all optimizers in this study ensures that unnecessary isolation events remain minimized, reducing wear on switching equipment and service interruptions, while the approximately 86% recall represents an acceptable trade-off for applications where false alarms generate immediate operational costs greater than brief exposure to grid anomalies. However, for critical safety installations such as hospitals or data centers,

future implementations should consider cost-sensitive learning approaches or threshold adjustments to prioritize recall over precision, potentially reaching recall levels above 98% as recommended for compliance with the IEEE 1547 standard, which requires detection within less than 2 s.

4 Conclusions

This work presented a binary classification model based on an MLP-ANN model whose architecture and hyperparameters were optimized by the novel MMSO algorithm. The primary objective was to enhance the accuracy of predicting operational modes in a hybrid microgrid, with a focus on reducing false negatives to prevent late or missed detections of island conditions. The proposed approach was trained and validated with real data from an experimental microgrid hosted at the Renewable Energy Laboratory of the Polytechnic University of Valencia, thereby providing a robust test bed with varying environmental conditions and power flow profiles.

The comparative analysis against other widely used bio-inspired optimizers (PSO, GA, and GWO) underscored that all methods achieved commendably high recall, exceeding 85%, and accuracy surpassing 92%. However, the proposed MMSO-based model exhibited the lowest MSE, with a value of approximately 0.0206, which suggests a finer-grained fit to the underlying data distribution for complex systems. Although GA and GWO attained slightly higher recall in some folds, the observed differences were statistically nonsignificant. These findings indicate that each algorithm can reliably capture essential system dynamics; however, the proposed MMSO's capacity to consistently search deeper and more complex network structures positions it as a powerful metaheuristic for scenarios that necessitate a more nuanced representation of microgrid behavior. Furthermore, the resultant MLP-ANN model demonstrated strong resilience to false positives, with a precision exceeding 98%, thereby mitigating the risk of unnecessary switching events. This work introduced a methodological framework for optimizing neural network architecture and hyperparameters using bio-inspired algorithms, showing that the search space can be effectively explored without manual intervention. The presented MMSO algorithm's ability to discover four-layer architectures, which exceeds the reach of PSO, GA, or GWO within the same bounds, underscores its enhanced exploration capabilities, especially when optimal solutions are located in deeper architectural regions.

From an operational standpoint, the MMSO-optimized MLP-ANN's practical feasibility was demonstrated by integrating an operation-mode management layer that filters a mechanism enforcing minimum residence times in both connected and islanded modes. This solution aims to balance the sensitivity of the proposed classifier, which can detect subtle fluctuations in power or environmental conditions, with the need to avoid excessive switching and maintain hardware longevity, as well as practical implementation constraints. Overall, the results suggest that a bio-inspired hybridization of neural networks with evolutionary swarm-based strategies can provide a viable foundation for advanced energy management in modern hybrid microgrid systems. The MLP-ANN model optimized using MMSO proposed in this work has characteristics that make it attractive for practical implementation in MG systems. The offline optimization, which takes about 60 min, is a one-time computational processing cost that must be performed only once to adapt the model to the system on which it will operate; in turn, the ANN architecture found by MMSO with four hidden layers [8, 173, 106, 54] allows inference in the order of milliseconds, meeting the requirements of the IEEE 1547 standard. Additionally, the training using the scale-conjugate gradient method (trainscg) showed stable convergence, and validation via k-fold cross-validation with k = 5 yielded a model with standard deviations below 1.68% for recall. However, the limitations of the proposed approach must be acknowledged, including that, since the model is based on an ANN, it must be retrained with data from each new location where it is to be implemented. Furthermore, the performance of the proposed model depends on the availability and quality of measurements, as well as on the filtering and preprocessing of such data.

Future research may extend this framework to larger, more heterogeneous systems, incorporate additional performance metrics (e.g., latency or computational overhead), and embed real-time constraints for grid-interactive MG control.

Acknowledgement: The authors would like to express their gratitude to the Institute of Energy Engineering of the Universitat Politècnica de València (Spain), the Institute of Renewable Energies, and the Department of Water and Energy Studies of the Universidad de Guadalajara (México) for their support and collaboration in the development of this research. The authors are also grateful for the funding through the PROSNII 2025 support granted by the University of Guadalajara for the author Jesús Águila León. Also, the authors want to express their gratitude to the Secretariat of Science, Humanities, Technology and Innovation SECIHTI of Mexico. During the preparation of this manuscript, the authors utilized Claude AI (Anthropic) for coding assistance and language refinement. The authors have carefully reviewed and edited all AI-assisted output and accept full responsibility for the content of this publication.

Funding Statement: This research was partially supported by the PROSNII 2025 program granted by the University of Guadalajara to Jesús Águila-León. In addition, the research was supported by the Vicerrectorado de Investigación of the Universitat Politècnica de València through the PAID-11-25 program. The authors also acknowledge the support of the Secretariat of Science, Humanities, Technology and Innovation (SECIHTI) of Mexico. The Institute of Energy Engineering at the Universitat Politècnica de València provided laboratory facilities and technical support for the experimental microgrid data collection.

Author Contributions: Jesús Aguila-León: conceptualization, methodology, software development, validation, formal analysis, investigation, data curation, original draft preparation, visualization, and contributed to funding acquisition. Carlos Vargas-Salgado: conceptualization, validation, resources, data curation, review and editing, supervision, and project administration. Dácil Díaz-Bello: validation, investigation, and review and editing. Fabián Lara-Vargas: resources, review and editing, supervision, and funding acquisition. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The experimental microgrid data used in this study were collected at the Renewable Energy Laboratory (LabDER) of the Universitat Politècnica de València. The data that support the findings of this study are available from the corresponding author, Carlos Vargas-Salgado, upon reasonable request, subject to approval from the laboratory administration.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

Nomenclature

Symbol	Description
α	Global-best attraction coefficient in MMSO algorithm
β	Personal-best attraction coefficient in MMSO algorithm
γ	Crossover and mutation dynamic coefficient in MMSO algorithm
λ	L2 regularization coefficient
η	Learning rate
b_j	Bias term of the j-th neuron
c	Crossover point in genetic operations
$f(\cdot)$	Activation function
j(w)	Cost function with L2 regularization
L	Number of hidden layers in MLP
LB	Lower bound of optimization variables
n_j	Number of neurons in the j-th hidden layer

N	Total number of training/validation samples
t	Current iteration number
Tmax	Maximum number of iterations
UB	Upper bound of optimization variables
wi,j	Synaptic weight connecting i-th input to j-th neuron
X	Optimization variables vector
x_i	Input variable to the MLP model
x_i^t	Position of mosquito i at iteration t
xbest	Best global position found
y_j	Output of the j-th neuron
$\hat{y}_i$	Predicted output value
y_i	Target output value
Z(·)	Zigzag flight pattern function in MMSO
Abbreviation	**Definition**
AI	Artificial Intelligence
ANN	Artificial Neural Network
ANOVA	Analysis of Variance
AUC	Area Under the Curve
CI	Confidence Interval
CV	Coefficient of Variation
FN	False Negative
FP	False Positive
GA	Genetic Algorithm
GWO	Grey Wolf Optimizer
LabDER	Laboratorio de Energías Renovables (Renewable Energy Laboratory)
logsig	Logistic sigmoid activation function
MG	Microgrid
MLP	Multilayer Perceptron
MMSO	Mosquito Mating Swarm Optimization
MSE	Mean Squared Error
PCC	Point of Common Coupling
PSO	Particle Swarm Optimization
PV	Photovoltaic
RES	Renewable Energy Sources
ROC	Receiver Operating Characteristic
tansig	Hyperbolic tangent sigmoid activation function
TN	True Negative
TP	True Positive
UPV	Universitat Politècnica de València

References

1. Uddin M, Mo H, Dong D, Elsawah S, Zhu J, Guerrero JM. Microgrids: a review, outstanding issues and future trends. Energy Strategy Rev. 2023;49:101127. doi:10.1016/j.esr.2023.101127.
2. Bajwa AA, Mokhlis H, Mekhilef S, Mubin M. Enhancing power system resilience leveraging microgrids: a review. J Renew Sustain Energy. 2019;11(3):035503. doi:10.1063/1.5066264.
3. Hmad J, Houari A, El Moubarek Bouzid A, Saim A, Trabelsi H. A review on mode transition strategies between grid-connected and standalone operation of voltage source inverters-based microgrids. Energies. 2023;16(13):5062. doi:10.3390/en16135062.

4. Pullins S. Why microgrids are becoming an important part of the energy infrastructure. Electr J. 2019;32(5):17–21. doi:10.1016/j.tej.2019.05.003.
5. Ezzat A, Elnaghi BE, Abdelsalam AA. Microgrids islanding detection using Fourier transform and machine learning algorithm. Electr Power Syst Res. 2021;196:107224. doi:10.1016/j.epsr.2021.107224.
6. Worku MY, Hassan MA, Maraaba LS, Abido MA. Islanding detection methods for microgrids: a comprehensive review. Mathematics. 2021;9(24):3174. doi:10.3390/math9243174.
7. Baghaee HR, Mlakić D, Nikolovski S, Dragicević T. Support vector machine-based islanding and grid fault detection in active distribution networks. IEEE J Emerg Sel Top Power Electron. 2020;8(3):2385–403. doi:10.1109/JESTPE.2019.2916621.
8. Kermany SD, Joorabian M, Deilami S, Masoum MAS. Hybrid islanding detection in microgrid with multiple connection points to smart grids using fuzzy-neural network. IEEE Trans Power Syst. 2017;32(4):2640–51. doi:10.1109/TPWRS.2016.2617344.
9. Alshareef S, Talwar S, Morsi WG. A new approach based on wavelet design and machine learning for islanding detection of distributed generation. IEEE Trans Smart Grid. 2014;5(4):1575–83. doi:10.1109/TSG.2013.2296598.
10. Admasie S, Ali Bukhari SB, Gush T, Haider R, Kim CH. Intelligent islanding detection of multi-distributed generation using artificial neural network based on intrinsic mode function feature. J Mod Power Syst Clean Energy. 2020;8(3):511–20. doi:10.35833/MPCE.2019.000255.
11. Kaviani S, Sohn I. Application of complex systems topologies in artificial neural networks optimization: an overview. Expert Syst Appl. 2021;180:115073. doi:10.1016/j.eswa.2021.115073.
12. Özcanlı AK, Baysal M. A novel multi-LSTM based deep learning method for islanding detection in the microgrid. Electr Power Syst Res. 2022;202:107574. doi:10.1016/j.epsr.2021.107574.
13. Xia Y, Yu F, Xiong X, Huang Q, Zhou Q. A novel microgrid islanding detection algorithm based on a multi-feature improved LSTM. Energies. 2022;15(8):2810. doi:10.3390/en15082810.
14. Wang S. Domain adaptation using transformer models for automated detection of exterior cladding materials in street view images. Sci Rep. 2025;16(1):2696. doi:10.1038/s41598-025-32524-7.
15. Wang S. Real operational labeled data of air handling units from office, auditorium, and hospital buildings. Sci Data. 2025;12:1481. doi:10.1038/s41597-025-05825-9.
16. Mumtahina U, Alahakoon S, Wolfs P. Hyperparameter tuning of load-forecasting models using metaheuristic optimization algorithms—a systematic review. Mathematics. 2024;12(21):3353. doi:10.3390/math12213353.
17. Díaz-Bello D, Vargas-Salgado C, Gómez-Navarro T, Águila-León J. Enhancing energy efficiency and profitability in microgrids through a genetic algorithm approach, analyzing the use of storage systems. Sustain Energy Technol Assess. 2025;73:104154. doi:10.1016/j.seta.2024.104154.
18. Zúñiga-Grajeda V, Coronado-Mendoza A, Gurubel-Tun KJ. Meta-optimization of bio-inspired algorithms for antenna array design. Kybernetika. 2018:610–28. doi:10.14736/kyb-2018-3-0610.
19. Díaz-Bello D, Vargas-Salgado C, Montuori L, Alcázar-Ortega M. Cluster-based demand prediction: a hybrid approach with grey wolf optimizer and multilayer perceptron network. Clean Technol Environ Policy. 2025;27(12):8411–29. doi:10.1007/s10098-025-03299-2.
20. Cinar AC, Natarajan N. An artificial neural network optimized by grey wolf optimizer for prediction of hourly wind speed in Tamil Nadu, India. Intell Syst Appl. 2022;16:200138. doi:10.1016/j.iswa.2022.200138.
21. Tawhid MA, Ibrahim AM. An efficient hybrid swarm intelligence optimization algorithm for solving nonlinear systems and clustering problems. Soft Comput. 2023;27(13):8867–95. doi:10.1007/s00500-022-07780-8.
22. Liu X, Zhao M, Wei Z, Lu M. The energy management and economic optimization scheduling of microgrid based on Colored Petri net and Quantum-PSO algorithm. Sustain Energy Technol Assess. 2022;53:102670. doi:10.1016/j.seta.2022.102670.
23. Paul K, Jyothi B, Kumar RS, Singh AR, Bajaj M, Hemanth Kumar B, et al. Optimizing sustainable energy management in grid connected microgrids using quantum particle swarm optimization for cost and emission reduction. Sci Rep. 2025;15(1):5843. doi:10.1038/s41598-025-90040-0.

24. Xiao D, Deng W, Liu B, Huang W, Zhu Z. Game-theoretic energy scheduling for community multi-energy system with electric vehicle aggregator leveraging quantum swarm intelligence. Int J Electr Power Energy Syst. 2025;172:111200. doi:10.1016/j.ijepes.2025.111200.
25. Vargas-Salgado C, Díaz-Bello D, Alfonso-Solar D, Lara-Vargas F. Validations of HOMER and SAM tools in predicting energy flows and economic analysis for renewable systems: comparison to a real-world system result. Sustain Energy Technol Assess. 2024;69:103896. doi:10.1016/j.seta.2024.103896.
26. Aguila-Leon J, Chiñas-Palacios C, Garcia EXM, Vargas-Salgado C. A multimicrogrid energy management model implementing an evolutionary game-theoretic approach. Int Trans Electr Energy Syst. 2020;30(11):e12617. doi:10.1002/2050-7038.12617.
27. Chen K, Xie J. Hybrid adaptive Wolf-Particle swarm optimization algorithm and its application in CNN neural network hyperparameters optimization. Discov Comput. 2025;28(1):319. doi:10.1007/s10791-025-09878-7.
28. Elhani D, Megherbi AC, Zitouni A, Dornaika F, Sbaa S, Taleb-Ahmed A. Optimizing convolutional neural networks architecture using a modified particle swarm optimization for image classification. Expert Syst Appl. 2023;229:120411. doi:10.1016/j.eswa.2023.120411.
29. Zhao C, Jiao Z. Ensemble boosting method of power system transient stability assessment based on cost-sensitive learning. In: Proceedings of the 2023 6th International Conference on Electrical Engineering and Green Energy (CEEGE); 2023 Jun 6–9; Grimstad, Norway. doi:10.1109/CEEGE58447.2023.10246600.
30. Mishra M, Patnaik B, Bansal RC, Naidoo R, Naik B, Nayak J. DTCDWT-SMOTE-XGBoost-based islanding detection for distributed generation systems: an approach of class-imbalanced issue. IEEE Syst J. 2022;16(2):2008–19. doi:10.1109/JSYST.2021.3086298.