AI in Sustainable Energy and Environment

Journal homepage: aisesjournal.com

Online

Retrofitting Neural Networks with Nature Inspired Optimization Algorithms, Improving Energy Efficiency in Steel Production

Sarah A. Alabbas

Education Directorate of Najaf province, 54001-Najaf, Iraq

Article Info

Received 7 January 2025

Received in Revised form 6 February 2025

Accepted 4 May 2025

Published online 7 May 2025

DOI:

Keywords

Energy Consumption,

Steel Production,

Artificial Neural Network,

Metaheuristic

Abstract

Given the huge growth in urban innovation over the past 10 years, smart cities require rational and workable solutions for transportation, building infrastructure, environmental conditions, and human enjoyment. This paper presents and explores data-mining-based predicted energy consumption models for a small-scale, intelligent steel company in South Korea. Devices built on the Internet of Things (IoT) are used to collect and process energy use data in order to forecast. Among the data used are leading and following currents, carbon dioxide emissions, load kinds, reactive power, and power factor. The Leagues Championship Algorithm (LCA), Evaporation-rate Water Cycle Algorithm (ERWCA), Multiverse Optimization Algorithm (MVO), Cuckoo Optimization Algorithm (COA), and Stochastic Fractal Search (SFS). The following metrics are used to evaluate the models' predictive power: root mean square error, mean absolute error (MAE), and coefficient of variation (R2) (RMSE). With the greatest R2 values (0.99800 during testing and 0.99815 during training), the Multilayer Perceptron (MLP) arrangement augmented by ERWCA performs exceptionally well and demonstrates a strong correlation between the predicted and actual energy consumption. Moreover, ERWCA has the lowest RMSE, a measure of the least number of prediction errors (2.09627 in the testing phase and 2.03778 in the training phase). Comparably low RMSE values (2.59167 during testing and 2.50700 during training) and excellent performance in terms of R2 values (0.99695 during testing and 0.99720 during training) also suggest that SFS could be used to improve MLP models for accurate energy consumption forecasts in smart city industrial buildings. These findings show how reliable and accurate the SFS and ERWCA energy use estimations are.

1 Introduction

Rising economic growth and population expansion in emerging nations have resulted in an increase in global energy demand. It is projected that worldwide energy consumption would increase by around 25% by 2040, notwithstanding increases in efficiency over the last several decades [1]. According to the International Energy Agency (IEA) [2], demand is expected to rise sharply in emerging countries between 2017 and 2040, especially in Asia and Africa. According to the scenario being studied, there will be a 10% rise in CO_2 emissions connected to energy. 32.6 to 35.9 gigatons, respectively. Forecasts indicate that CO_2 emissions in developed nations will decline by 23 percent and rise by 27 percent in emerging economies. Energy use has a substantial impact on the environment. It has been shown that the Earth's ongoing warming is mostly caused by CO_2 emissions [3, 4]. Moreover, future projections suggest that this upward trend will not abate. Therefore, the main objective of international political, economic, and environmental study at the moment is lowering CO_2 emissions. To save energy

and improve environmental protection, new energy-efficiency regulations are being implemented. For example, the present energy plan of the European Commission requires every member state to carry out a set of measures aimed at achieving a minimum of 20 percent energy efficiency [5]. Though many firms contribute to these emissions, the building and building construction industries account for around 40% of global CO_2 emissions and 36% of total energy use [6]. Research has demonstrated that a number of building subsectors, such as lighting, water heating, space heating and cooling, and others, have the potential to save energy in an efficient setting. Therefore, developing prediction models for energy use is more important in order to decide how energy limitations should be implemented. The forecasting problem may often be divided into three categories: load forecasting for the short, medium, and long periods, depending on the prediction horizon. Medium-term forecasting spans periods of one hour to one week, whereas short-term forecasting covers periods of one month to one year. Finally, a prediction horizon beyond a year signifies a prognosis for the long term. Most works concentrate on the short-term prediction horizon because of its higher accuracy than other horizons. These days, sensors are put in an increasing number of buildings to evaluate many aspects of the building's performance, including the electric energy usage that has been recorded [1]. "Smart buildings" are the term used to describe these types of buildings. Because these measurements are reusable, they are a valuable source of historical data that can be organized into time series and used to predict the future energy requirements of the structures [1].

Effective energy consumption prediction models using hybrid ANN meta-heuristic algorithms are essential for cost-effective energy optimization and consumption reduction in smart city industrial buildings [7]. These models employ machine learning and artificial intelligence techniques to estimate and control energy usage for non-residential buildings, with a focus on improving prediction accuracy [8]. Artificial neural networks (ANNs) have been shown to be an excellent way to control building energy use, especially when data collected over long periods of time is used [9]. Moreover, it has been proposed that deep learning approaches be used to solve low accuracy and overfitting issues, particularly when working with large datasets [10]. By utilizing data-driven machine learning techniques and optimization tactics, these models might be further enhanced to boost prediction reliability and accuracy [11]. Generally speaking, creating trustworthy energy consumption projection models is necessary to ensure that industrial buildings in smart cities satisfy sustainability and energy efficiency requirements.

Machine learning techniques are increasingly being used to predict energy usage in industrial cities, particularly in smart buildings that have sensors monitoring every element of the building's functioning, including the use of electric energy [1]. These measurements provide useful historical data that may be used to anticipate the buildings' future energy usage and build time series [1]. Energy consumption projection models are essential for smart city industrial buildings to achieve sustainability and energy efficiency standards [8]. Machine learning algorithms employ historical data to estimate and manage the amount of energy consumed by non-residential buildings, with a focus on improving forecast accuracy [10]. Moreover, it has been proposed that deep learning approaches be used to solve low accuracy and overfitting issues, particularly when working with large datasets [11]. By utilizing data-driven machine learning techniques and optimization tactics, these models might be further enhanced to boost prediction reliability and accuracy [12]. In general, to reduce costs and increase energy efficiency for smart city industrial buildings, precise models for estimating energy use must be developed [7].

Recent studies have focused on the use of machine learning to predict energy usage in various industrial scenarios. As an illustration, V E, et al. [13] recommended and looked into predicted energy consumption models for a sophisticated small-scale steel mill in South Korea that are based on data mining. Energy consumption data is collected by IoT-based systems and used for forecasting. Reactive power, carbon dioxide emissions, leading and following current power factor, and different loads are the data sets that are employed. The five statistical methods listed below are employed to predict energy consumption: Classification, generic linear regression, and regression trees The radial basis kernel, CUBIST, and K nearest neighbors make form a support vector machine. V E, et al. [13] shown that the best outcomes with the lowest error levels are produced by the CUBIST model. This model may be used to create structural designs that are energy-efficient, which will help with policy formation and energy consumption optimization in smart cities.

Chahbi, et al. [14] described a brand-new machine learning (ML) technique for figuring exactly how much energy commercial buildings use. It draws attention to a trade-off between the interpretability and performance of ML models, which are crucial factors to take into account while creating ML-based energy prediction. The case study that predicts energy consumption in the steel industry demonstrates the applicability of the recommended approach. Chahbi, et al. [14] shown that the permutation feature importance helps experts grasp the model's findings and that the Random Forest (RF) model produces the best prediction results for the steel industry. To ascertain the essential components, Ye, et al. [15] employed a data-driven random forest (RF) based approach to explore the relationship between building energy usage and building-block-level building-oriented characteristics. The dataset of Taipei City comprised 24,764 buildings in 881 city blocks. The RF model outperforms other machine learning models, including logistic regression, k-nearest neighborhood, support vector machines, and decision tree models, in terms of prediction accuracy for the classification problem. Seven of the 59 building-oriented variables that are significant at the city block level include building gross floor space, building density, building construction year, and the proportion of commercial buildings in the block. Sathishkumar, et al. [16] investigated and assessed the methods the steel industry uses to predict energy use. Among the data utilized are the load type, CO_2 emissions, and variables for leading and trailing reactive power and current. Four statistical models—Random Forest (a), Gradient Boosting Machine (GBM) (b), Radial Kernel Support Vector Machine (SVM RBF) (c), and Linear Regression (LR)—are trained and evaluated on the test set (RF). When all predictors are employed, the best model RF could produce an RMSE score of 7.33 in the test set.

The need to optimize energy usage and reduce costs in industrial contexts motivates the employment of metaheuristic algorithms to anticipate energy use [17]. These methods may be applied to optimize the gray prediction model's parameters; the Grey Wolf Optimizer (GWO) algorithm provides advantages in terms of global optimal solution attainment, stability, and convergence speed. Examining how each incentive mechanism affects prosumers, customers, and energy providers is the goal, with a focus on how well the systems under study function in relation to them other [17]. Because the research also examines the relationship between prosumer numbers and energy injection and consumption, we can more accurately predict and optimize energy use overall [17]. This study presents a novel approach to predicting energy consumption in the steel manufacturing sector by integrating five metaheuristic optimization algorithms—COA, MVO, LCA, ERWCA, and SFS—with a multilayer perceptron (MLP) model. The significance of this work lies in its comparative analysis of population sizes and their effect on model performance, an area not previously explored in such depth. The hybrid frameworks were evaluated through robust metrics (RMSE and R^2) and a unified scoring system, offering a comprehensive methodology for selecting optimal modeling strategies in industrial energy forecasting. The insights provided have direct implications for improving energy efficiency in one of the most energy-intensive industries.

2 Materials and methods

Figure 1 displays the research flowchart for this investigation. The modeling approach of hybrid optimization algorithms with neural networks combines the benefits of both optimization algorithms and neural networks to handle complex problems. There may be differences in the modeling process depending on the specific hybridization methodology, the optimization strategy used, and the specifics of the problem at hand. The effectiveness of the hybrid model depends on how well the optimization algorithm and neural network handle the complex nature of the optimization problem. Combining the learning and pattern recognition capabilities of neural networks with the optimization capabilities of metaheuristic algorithms is the aim of hybrid models. General procedures for modeling hybrid optimization algorithms with neural networks include problem definition (defining the optimization problem that needs to be addressed) and neural network architecture design (choosing the number of layers, number of neurons in each layer, activation functions, and overall structure). The neural network will determine and map the relationship between the input features and the desired output.), hybridization technique (common strategies include using the optimization algorithm for weight optimization in the neural network, initializing the neural network weights using the optimization algorithm, or combining the solutions

obtained by the optimization algorithm with the neural network predictions.); combining neural networks with optimization algorithms (involves integrating the optimization algorithm within the neural network's training process or using the optimization algorithm to guide the search space exploration for optimal neural network weights.), creating the hybrid model (which entails employing the selected optimization technique to optimize the neural network weights). Validating and fine-tuning (i.e., assessing the hybrid model's performance on an alternative dataset not utilized for training) and searching the solution space for weights that minimize a cost function—a measure of the discrepancy between expected and actual results—may be steps in the optimization process. In light of the validation results, modify the model's parameters (e.g., neural network architecture, optimization algorithm settings) to boost generalization. Evaluation and experimentation: To determine how effectively the hybrid model generalizes to new data, use a test dataset. assess the performance metrics that are relevant to the specific problem at hand (e.g., accuracy, mean squared error).

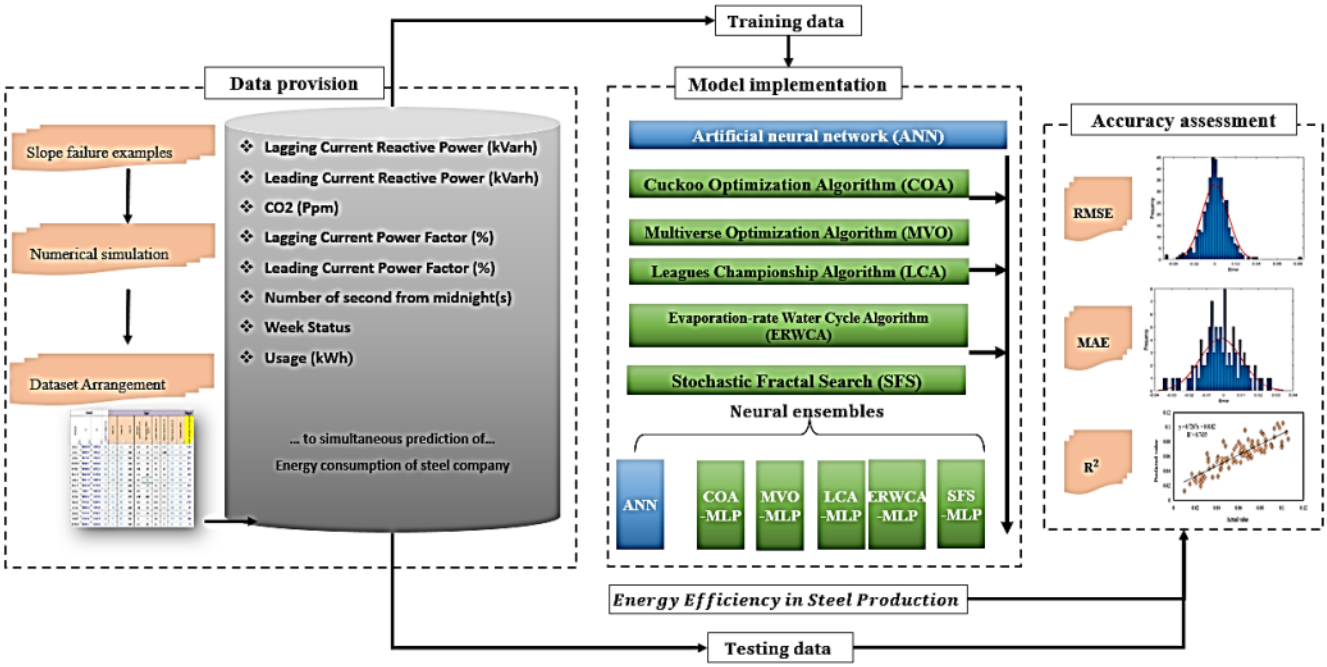


Figure 1. A synopsis of the modeling process

2.1 Multi-layer perceptron (MLP)

The energy was estimated using a two-layered feedforward neural network from the Matlab ANN Toolbox. The ANN network was trained using the Levenberg-Marquardt approach from the Matlab ANN Toolbox. Three layers make up an artificial neural network (ANN): an output layer with a linear output function, a hidden layer with a sigmoid activation function, and an input layer. The accuracy of the forecasts was increased by using random initialization. Nonlinear data can be handled using the buried layer's sigmoid transfer function. After compressing the input, which ranges from plus to negative infinity, the result is between 0 and 1 [18]. The sigmoid's activation function is shown in equation (1):

$$f(x) = 1/(1 + exp^{-x}) \quad (1)$$

While the input neurons tracked the data as it changed, the output neurons calculated the energy used. It was demonstrated that increasing the number of hidden neurons from 1 to 10 produced the optimal model structure. Training and test data sets were created using thirty percent, or seventy percent, of the whole data set. The network learns which weights are the most economical during training. The optimum model iteration for the data was found using a cost function method. In order to prevent

overfitting, the training was discontinued when the error reduction failed six times in a row.

McCulloch and Pitts [19] were the ones who originally put up the idea of an ANN. A number of ANNs have been suggested by academics for usage in a variety of application scenarios due to their potential for non-linear parameter mapping [20, 21]. Compared to other forms of artificial neural networks, the Multi-Layer Perceptron (MLP) tool is the most commonly used due to its flexible design, large representational capacity, and plenty of data samples [22]. Because of their backpropagation training, the MLPs are also known as generic approximators or feedforward neural tools [23]. They possess a type of processing unit called "neurons," which allows them to predict nearly any input-output method. Figure 2 depicts the whole MLP structure used in this study. This structure consists of three distinct layers: an output layer, a hidden layer, and an input layer. The neurons in this layer are tightly coupled to the neurons in any adjacent layers [24].

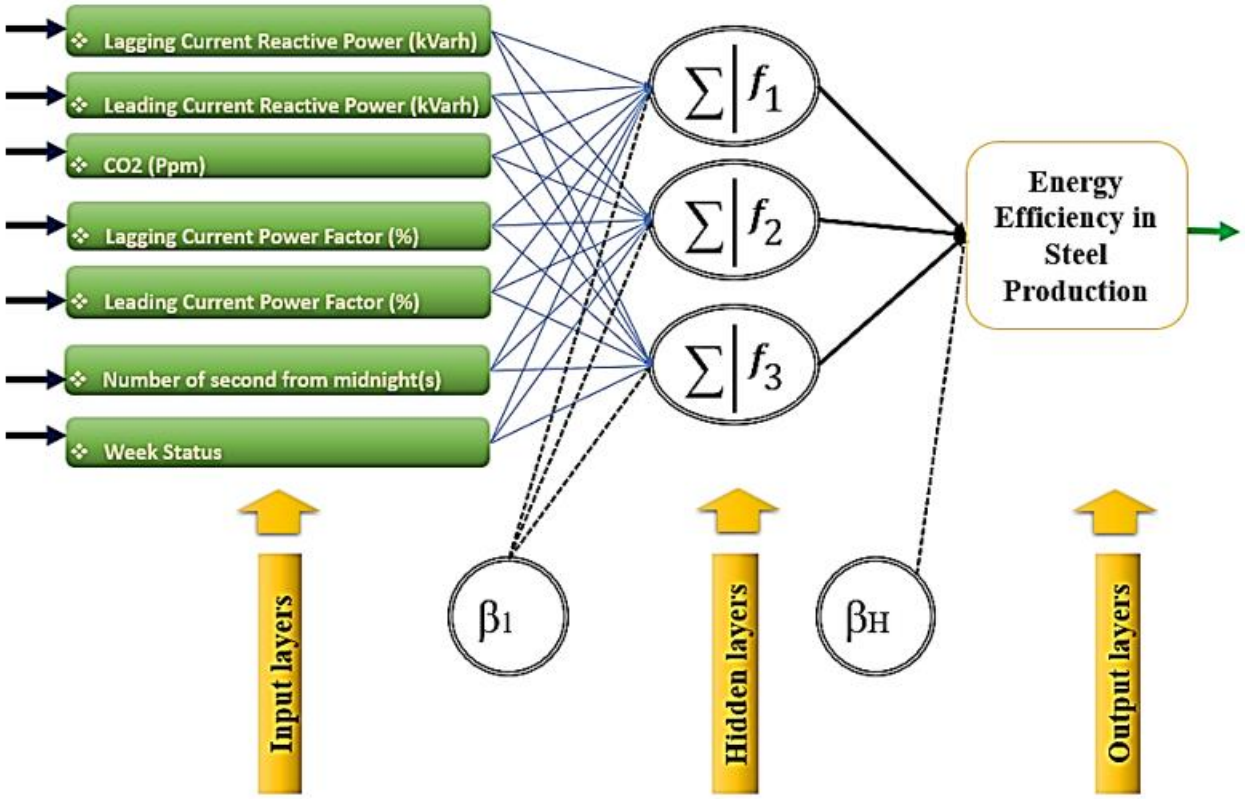


Figure 2: Diagram of the MLP algorithm

The assessment of energy use, which was finally forecast using the best predictive network, was the primary result of this investigation. For both training and testing, four statistical indicators were computed for the models. Equations (2–5) define the coefficient of determination (R^2), mean absolute error, mean squared error (MSE), and root mean square error (RMSE). (MAE). The models' accuracy was assessed using these statistical parameters. For example, R^2 was used to assess the model's robustness and RMSE was used to assess the model's accuracy. There are three values for y: a mean value of $\bar{y}$; a predicted value of y_k; and a measured value of y_k. There are n samples total.

$$RMSE = \sqrt{\left(\sum_{k=1}^{n}(\widehat{y_k} - y_k)^2\right)/n} \quad (2)$$

$$R^2 = 1 - \left(\frac{(\sum_{k=1}^{n}(\widehat{y_k} - y_k)^2)}{(\sum_{k=1}^{n}(y_k - \bar{y})^2)}\right) \quad (3)$$

$$MSE = \left(\sum_{k=1}^{n}(\widehat{y_k} - y_k)^2\right)/n \quad (4)$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}(\widehat{y_k} - y_k) \quad (5)$$

2.2 Cuckoo Optimization Algorithm (COA)

The population of cuckoos used by COA is the same as that used by earlier evolutionary algorithms. They carry eggs to the nests of their hosts. Some of the eggs may develop into adult cuckoos if they closely resemble the eggs of the host bird. Nevertheless, some are found by the host birds, who eliminate them. The nests may or may not be appropriate based on how the eggs develop. The amount of profit depends on how many eggs survive in a given area. The criterion that COA optimizes will be the place where eggs survive. In order to increase the likelihood that their eggs will survive, cuckoos search for the best place to deposit them. Upon hatching into adult cuckoos, the eggs form many communities. Every civilization has a characteristic place to reside, hence the ideal home for all civilizations is wherever cuckoos end up in other cultures. They then relocate to the ideal location. They'll make their home wherever the best circumstances exist. The ideal habitat, or egg-laying radius, is calculated using the number of eggs deposited by each cuckoo and its distance from the objective. The cuckoo then deposits eggs in impromptu nests by using her egg-laying region. This process keeps going until the majority of cuckoo populations gather in the most advantageous spot [25, 26].

In N_{var} dimension optimization issue, a habitat is an assembly of $1 \times N_{var}$, displaying the current habitat of the cuckoo. The following is an explanation of this array:

$$Habitat = [X_1, X_2, \dots, X_{N_{var}}] \quad (6)$$

It seems that the value of each variable, $X_1, X_2, \dots, X_{N_{var}}$ is a number in floating points. By analyzing the profit function f_p at a habitat of $(X_1, X_2, \dots, X_{N_{var}})$ one may determine the profit of a certain habitat. Therefore

$$Profit = f_p(habitat) = f_p(X_1, X_2, \dots, X_{N_{var}}) \quad (7)$$

It is evident that COA is a group of algorithms designed to optimize a profit function. The following profit function may be easily optimized to employ COA in cost-minimization scenarios:

$$Profit = -Cost(habitat) = -f_c(X_1, X_2, \dots, X_{N_{var}}) \quad (8)$$

A candidate habitat matrix of dimensions $N_{pop} \times N_{var}$ is created to start the optimization process. After then, it's expected that each of these initial cuckoo houses would contain a certain number of eggs that were created at random. Those figures represent the lowest and upper boundaries of the total number of eggs attributable to each cuckoo at specific iterations, as each cuckoo typically lays between five and twenty eggs. The "Egg Laying Radius (ELR)," which is inversely related to the total number of eggs the cuckoo has deposited thus far and the varying limits of var_{hi} and var_{low} is another attribute of authentic cuckoos. ELR is defined as follows:

$$ELR = \alpha \times \frac{Number\ of\ cuckoos\ eggs}{Total\ number\ of\ eggs} \times (var_{hi} - var_{low}) \quad (9)$$

According to [25], α is an integer that is meant to control the ELR's maximum value. At first, every cuckoo in her ELR scatters its eggs into many host bird nests.

2.3 Multi-verse Optimization (MVO)

According to Mirjalili, et al. [27], Wormholes, white holes, and black holes are the three main pillars of the multiverse theory in physics, and they are the mathematical models that are created using the MVO approach. Let each variable in the optimization problem reflect one of the following universes with respect to laws. A variation in the rate of inflation affects some but not all items in the universe through wormholes that lead to the ideal state. In universes with higher inflation rates, objects are more likely to pass via white holes, and in those with lower inflation rates, through black holes. Higher inflation rates are linked to white holes, whereas lower inflation rates are linked to black holes. The MVO algorithm is described as follows in brief:

Step 1: Set the universe's initial values, as well as the maximum repetitions, maximum iterations, interval variable [*lb, ub*], and universe location.

Step 2: To locate a white hole based on the inflation rate of the universe, use a roulette wheel selection approach.

$$x_i^j = \begin{cases} x_k^j & r1 < NI(U_i) \\ x_i^j & r1 \geq NI(U_i) \end{cases} \quad (10)$$

where $r1$ is a randomly generated number from the interval [0, 1]; U_i is the *i*-th universe; x_i^j is the *i*-th universe's *j*-th parameter; x_k^j is the *k*-th universe's *j*-th parameter selected by the roulette process; and $NI(U_i)$ is the universe's normative inflation rate.

Step 3. Time for a wormhole existence probability (WEP) calculation, a travel distance rate (TDR) update, and a boundary check.

$$WEP = min + l \cdot \left(\frac{max - min}{L}\right) \quad (11)$$

$$TDR = 1 - \frac{l^{\frac{1}{p}}}{L^{\frac{1}{p}}} \quad (12)$$

The numbers l for the current iteration, L for the maximum number of repetitions, and p for the accuracy of the exploitation stand for the highest and lowest WEP values, respectively. In the MVO model, low WEP and high TDR encourage exploration and the avoidance of local optima, whereas high WEP and low TDR enhance exploitation [28].

Step 4: Find the current inflation rate in the universe. The cosmos shifts if the rate of inflation rises over its present value. In all other circumstances, the cosmos seems to continue existing.

Step 5: Update the position of the universe as provided by Equation (13).

$$x_i^j = \begin{cases} X_j + TDR\left((ub_j - lb_j)r4 + lb_j\right) & r3 < 0.5 \\ X_j - TDR\left((ub_j - lb_j)r4 + lb_j\right) & r3 \geq 0.5 \quad r2 < WEP \\ x_i^j & r2 \geq WEP(i) \end{cases} \quad (13)$$

(13)

where *r2*, *r3*, and *r4* are random values chosen from the range [0, 1]; ub_j is the *j*-th variable's upper bound; and lb_j is its lower bound. Where X_j is the *j*-th parameter of the best universe at that instant.

Step 6: criteria for termination. If the prerequisites for termination are met, the required output is produced. If not, an extra iteration is performed and Step 2 of the procedure is followed.

2.4 League Championship Algorithm (LCA)

Similar to other evolutionary algorithms, the LCA operates on a population of people [29]. As a result, during the initialization stage, a league (population) of L (the league size) teams (solutions) is formed, and their playing characteristics (fitness values) are evaluated. Every team will have *n* players if we analyze a function with n variables, where n is the number of variables. For now, the setups that work well for the teams make advantage of the starting settings. The competition is the next stage. According to the league schedule, the clubs play each other in pairs for $S \times (L - 1)$ weeks, where S is the number of seasons and t is the week. Regarding the results of the games or matches between teams *I* and *J*, there is no tie. Wins and losses are shown for each outcome. The performance of each squad determines this. Every team designs a new configuration during the recuperation time based on what performed well in the play of the previous week and what is currently its finest formation. The selecting process in LCA is voracious. It swaps out the current configuration for the best one with a more powerful and efficient one. Stated otherwise, if the new configuration is now the best choice for the team, it should be considered the fittest one (i.e., the best response found thus far for the *i*-th member of the population). Upon meeting the halting criterion, the algorithm terminates.

A few terms that we used in our explanation of the LCA technique need to be defined and thoroughly explained. Creating the league schedule and figuring out if the team is winning or losing are two of these concepts. Further information on these ideas is provided in the sections that follow.

2.4.1 Generating a league schedule

Creating a schedule containing every game for every season is the first step in creating the illusion of a championship setting, complete with teams vying for supremacy. Throughout the season, each team plays each other once in a round-robin style. Since L/2 matches would be played in parallel during each of the (L - 1) weeks, if there are L (an even number of teams), there will be L (L - 1)/2 matches (if L is an odd number, there would be L weeks with (L - 1)/2 matches and one team would play no games during any given week). After that, the championship lasts for S more seasons [29].

2.4.2 Determining winner/loser

Each squad participates in the LCA and plays against other squads; no team may win or lose a game. After a game, a team's result is determined stochastically using the playing strength criterion, as long as the likelihood of a team winning is commensurate with

its fit level. According to Kashan [29], the degree of fit is determined by the distance with an ideal reference point and is associated with the team's playing strength.

2.5 Evaporation Rate Water Cycle Algorithm (ERWCA)

Sadollah, et al. [30] introduced a novel search strategy called the evaporation rate-water cycle algorithm (ER-WCA). This approach modifies the WCA technique as originally proposed [30]. Two instances of how nature influences the WCA algorithm are the water cycle and water flowing toward the ocean. During the hydrological cycle, water from streams evaporates and is used by plants for photosynthesis. Once the vapor enters the atmosphere, it condenses as clouds.

Depending on the weather, water re-enters the earth in a variety of states. According to this system, rivers are excellent persons, whilst other water flows are referred to as streams. In the event when *K* represents the issue's magnitude, the potential streams are $x_1, x_2, \dots, x_k$. The initial population is created at random, as seen below:

$$Total\ population = \begin{bmatrix} Sea \\ River_1 \\ River_2 \\ \vdots \\ Stream_{K_{sr+1}} \\ Stream_{K_{sr+2}} \\ \vdots \\ Stream_{K_{pop}} \end{bmatrix} = \begin{bmatrix} x_1^1 & x_2^1 & \dots & x_k^1 \\ x_1^2 & x_2^2 & \dots & x_k^2 \\ \vdots & \vdots & \vdots & \vdots \\ x_1^{K_{pop}} & x_2^{K_{pop}} & \cdots & x_N^{K_{pop}} \end{bmatrix} \quad (14)$$

where the swarm size is indicated by K_{pop}. The intensity of flow for each approach is computed using Equation 14:

$$Cost_i = f\left(x_1^i, x_2^i, \dots, x_K^i\right) \quad I = 1, 2, \dots, K_{pop} \quad (15)$$

Rivers and oceans, K_{sr} are chosen from the most accomplished individuals. The residual population that may flow into rivers or the sea is shown by the symbol K_{Streams}. The amount of water drawn from the sea or river varies depending on the strength of the flow. The approximate distribution of streams to each river and the sea is shown by the bellows:

$$C_n = Cost_n - Cost_{K_{sr+1}} \quad n = 1, 2, \dots, K_{sr} \quad (16)$$

$$NS_n = round\{\left|\frac{C_n}{\sum_{n=1}^{K_{sr}} C_n} \times K_{Streams}\right| \quad (17)$$

The number of streams flowing toward a certain river or sea is indicated by the symbols NS_n. The fitness function is created to distribute streams proportionally between rivers and the sea since more streams flow into the sea. In the natural world, certain streams unite to create new rivers.

Figure 3 illustrates the path a stream travels in the direction of a river when there is only one sea andK_{sr-1} rivers among a population of K_{pop} people. Additional information on the proposed methodology may be found in related papers [31].

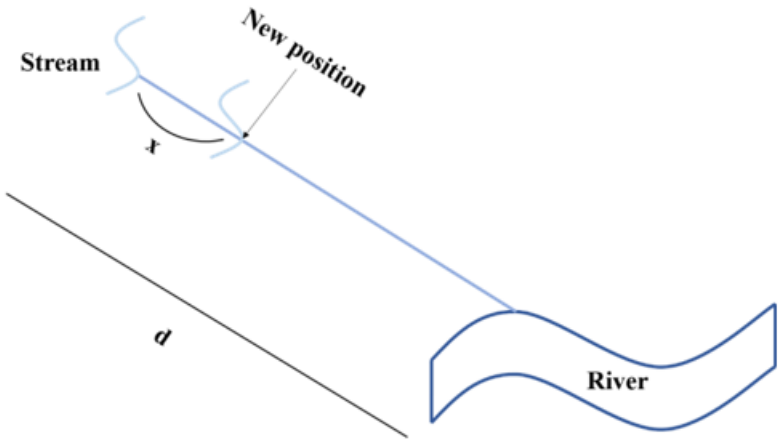


Figure 3. The direction in which a stream flows toward a particular river [30].

2.6 Stochastic Fractal Search (SFS)

An innovative swarm-based strategy for determining the ideal PID controller parameter values is the stochastic fractal search method [32]. SFS aims to increase the convergence rate by promoting information sharing across all group points, in contrast to its initial iteration, fractal search. To find the least value of the cost function, the method, which primarily makes use of the mathematical underpinnings of fractal theory, passes through the diffusing and upgrading stages. The former process, which manages the search space, is comparable to the fractal search in that it generates new particles around each particle's current position using the Gaussian walk statistical approach. In the next stage, two statistical techniques are used only for SFS as an upgrading technique to improve issue domain research. This phase serves to maintain the algorithm's exploratory property by adjusting a point's position based on the positions of other group points. It also makes information transfer between particles easier.

The first statistical method impacts each particle vector index, whereas the second statistical approach affects all points.

1. The first statistical process

The first step involves ranking particles based on their function values of fitness using the following equation:

$$P_{a_i} = \frac{rank(P_i)}{NP} \tag{18}$$

The number of particles in the group is denoted by NP in this instance, and the rank of the i-th particle is indicated by $rank(P_i)$. For every point P_i in the system, if $P_{a_i} < \varepsilon$, the j-th component of P_i is upgraded using Equation (19); if not, the corresponding component remains unchanged.

$$P_i'(j) = P_m(j) - \varepsilon\big(P_n(j) - P_i(j)\big) \tag{19}$$

The newly updated position of P_i in this equation is denoted by P_i'. Random points from the group are P_m and P_n whereas ε is an accidental integer in the interval [0, 1].

2. The second statistical process

Generally speaking, this technique aims to improve group variety over the original procedure by altering the conditions of the chosen point while accounting for the placements of other points. This is accomplished by evaluating the requirement $P_i < \varepsilon$, and ranking all of the points earned in the first phase using Eq (18). According to Eq. (20), if the condition is met, the existing state of P_i' is improved. If not, it will persist into the next generation unchanged.

$$P''_i = P_i' - \varepsilon(P_m' - P_{best}) \quad if \quad \varepsilon \le 0.5 \tag{20}$$

$$P''_i = P_i' - \varepsilon(P_m' - P_n') \quad if \quad \varepsilon > 0.5 \tag{21}$$

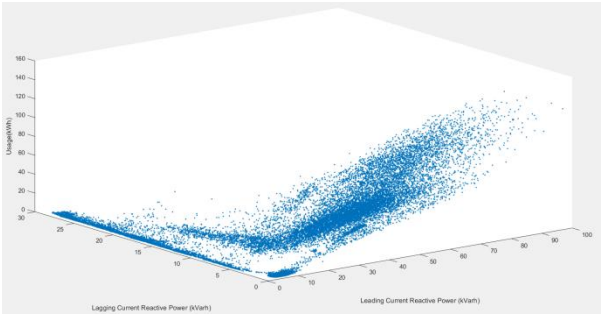
In SFS, the variables P_m' and P_n' represent two randomly selected points from the initial process, and P''_i denotes the modified state of P_i'. Additionally, P_{best} indicates the position of the best point, and the accidental number ε is between 0 and 1. After every statistical process, SFS uses a greedy selection mechanism to compare the fitness values of the old and improved solutions and selects the option that is more fit.

3 Established database

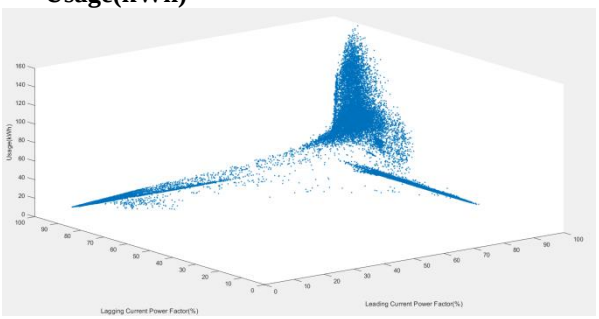
Data on energy use that is gathered every fifteen minutes is used in this investigation. After that, prediction models are created using machine learning techniques. The data was supplied by the DAEWOO Steel Co. Ltd. in Gwangyang, South Korea. It produces iron and steel plates in addition to a range of coils. Energy use data is kept in a cloud-based system. Statistics on the sector's energy use are available on the Korea Electric Power Corporation website (pccs.kepco.- go.kr). These statistics include computations and representations of daily, monthly, and annual data. Smart meters are used to measure the energy consumption of machine equipment used in the steel industry. These meters also collect and store additional energy use data in a cloud-based system. After demand was forecasted via data analytics techniques, energy consumption rules were improved and modified. The primary focus of this inquiry is the energy data (in Kwh) that is recorded for the industry every 15 minutes. The 15-minute reporting interval was used to track abrupt variations in energy consumption. R was used to analyze each and every piece of data [33]. The data covers the entire year (12 months). Most studies on energy consumption used meteorological criteria to forecast

energy use based on the influence of weather components [34]. Because of the open architecture and lack of heating and cooling systems in the steel sector, weather had minimal effect on the patterns of energy use in this research. The output variables and inputs by the output of the dataset are shown in Figure 4. Additional details like the day of the week, the period or weekday position, and the number of seconds till midnight for each day are provided by the date/time variable (NSM). The process of developing new features from pre-existing variables is known as feature engineering. Increasing the effectiveness of statistical learning algorithms is the aim of this.

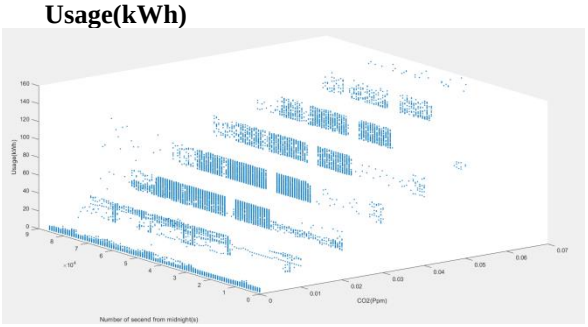
The dataset was randomly divided into 70% training and 30% testing subsets, a common convention in machine learning to ensure that the model generalizes well and avoids overfitting. This ratio allows the model sufficient exposure to data for learning patterns while reserving enough unseen data for robust performance evaluation. The independent variables were selected based on their demonstrated relevance in prior literature and their direct influence on energy consumption in steel production. Variables such as [list the actual variables] reflect operational, environmental, and process-related factors that contribute significantly to energy usage. These variables were also selected due to their availability and consistency in the public dataset obtained from the South Korean industrial report. Although the authors are based in Iraq, the dataset used in this study originates from publicly available South Korean industrial energy reports. The data were retrieved and validated from official sources, ensuring their reliability and relevance. The study does not involve direct experimentation or confidential national data; it is solely based on open-access datasets intended for research and policy analysis. Proper citation and verification of the data source have been ensured.



a) Leading/Lagging Current Power Factor/ Usage(kWh)



b) Leading/Lagging Current Power Factor/ Usage(kWh)



c) CO_2 (Ppm)/of second from midnight(s)/(kWh)

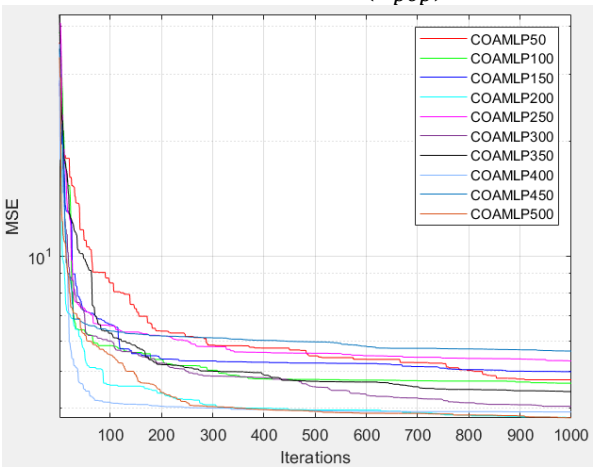
Figure 4: Details on the inputs and outputs.

4 Results and discussion

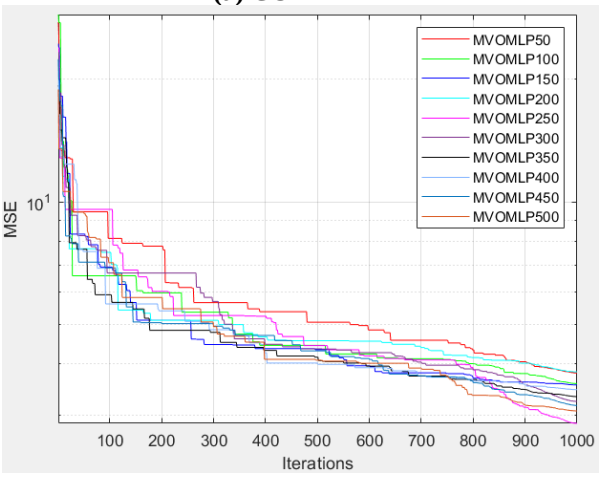
To find the ideal configuration, a variety of networks with varying numbers of layers and kinds of neurons have been built. The accuracy of the models is also impacted when a standard MLP's layer and neuron counts are changed. Using a feed-forward back-propagation approach and an average of five hidden units determined by the RMSE and R^2 metrics, the optimal network was built. Early optimization results are the starting point for many optimization approaches. The model with the highest score provides the best prediction network. Interestingly, the model's predicted accuracy was used to decide ranking. For instance, the chosen model receives a higher score when the RMSE decreases. The score and the R^2 rise in unison. For this reason, the

following sections make use of the results of these networks. The MSE fluctuations for each technique are shown in Figure 5.

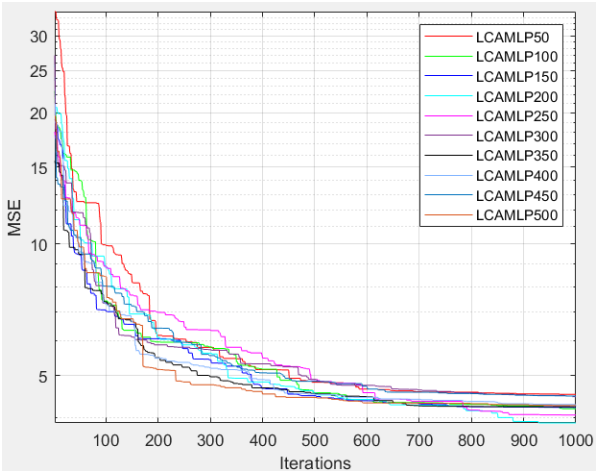
The first optimization discovery step will lay the groundwork for further stages of optimization methodologies. Consequently, these networks' outputs are used in the parts that follow. As the mean square error goes down, structures are more consistently correct (MSE). Regression and classification issues may be more accurately solved with the help of the anticipated values of the suggested model. Figure 5 shows the mean square error (MSE) variations across several cycles of the energy consumption prediction system estimations for the combined COA, MVO, LCA, ERWCA, and SFS designs. COA, MVO, LCA, ERWCA, and SFS have decided that the best options are 200, 250, 200, 250, and 400 based on this data (N_{pop}).



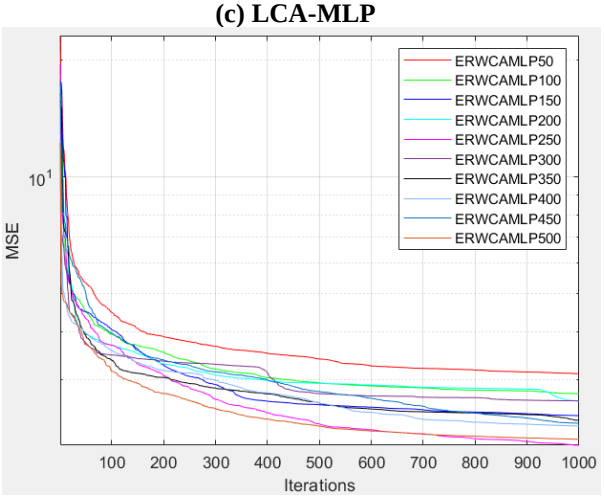
(a) COA-MLP



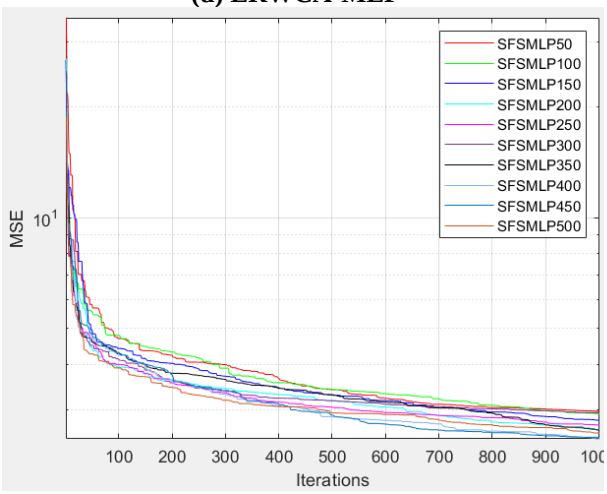
(b) MVO-MLP



(c) LCA-MLP



(d) ERWCA-MLP



(e) SFS-MLP

Figure 5. MSE variance amongst techniques.

4.1 Assessing Statistical Accuracy

An item or person is assigned a score by a scoring system according to their qualities or performance. Different ranking strategies may be required for different goals and circumstances. A common way is the total score rank methodology, which involves adding up the points for every item or individual and assigning a score based on their cumulative score. An alternative method (a term not frequently employed) is the color-scoring ranking system. That may, however, be a reference to a color-coded, level-based, classified rating system. For each grading level, the best hybrid designs are identified using R^2 and RMSE (Table 1-5). The best hybrid strategy for energy consumption in industrial cities uses swarm populations of 200, 250, 200, 250, and 400 to train and assess the results of predictive modeling (i.e., How well the program was able to forecast energy usage). It also shows how closely the outcomes of step two match those of phase one. Tables 1–5 display the network findings for the various COA–MLP, MVO–MLP, LCA–MLP, ERWCA–MLP, and SFS–MLP models.

Tables 1 through 5 present the outcomes of utilizing various combinations of meta-heuristic algorithms on Multilayer Perceptron (MLP) models, namely COA–MLP, MVO–MLP, LCA–MLP, ERWCA–MLP, and SFS–MLP, in order to predict energy consumption in a small-scale steel firm in South Korea. Each table provides a comprehensive picture of the models' performance across a range of population sizes by displaying metrics such as R-squared (R^2) and Root Mean Squared Error (RMSE) over the training and testing datasets. Techniques for ranking and grading offer a comparative assessment of how well the algorithms produce accurate estimates of energy usage. The designs that scored the highest overall for ERWCA-MLP and SFS-MLP, respectively, had 400 and 250 populations, indicating that they are more predictive. These results provide valuable guidance on how to enhance meta-heuristic algorithms to more precisely predict energy use in the context of small-scale steel enterprises, which in turn supports the development of smart city infrastructure and sustainable industrial practices.

The COA-MLP model demonstrated its optimal performance when configured with a population size of 200. During training, the model achieved a remarkably low Root Mean Square Error (RMSE) of 3.76348, indicating a high precision in predicting energy efficiency. The corresponding R^2 value for training was 0.99369, reflecting a strong correlation between the predicted and actual values. When evaluated on the testing dataset, this configuration continued to showcase excellence with an RMSE of 3.85332 and an R^2 of 0.99324. These results suggest that a population size of 200 led to a well-tuned COA-MLP model, effectively capturing the underlying patterns in the data and generalizing to unseen instances. Conversely, the worst-performing configuration for COA-MLP was observed when the population size was set to 450. Despite having a larger population, this configuration exhibited challenges in accurately modeling the energy efficiency of the steel production process. During training, the model struggled, yielding a higher RMSE of 5.62769 and a lower R^2 of 0.98583, signifying a less effective fit to the training data. This trend persisted when applied to the testing dataset, where the RMSE increased to 5.68331, and the R^2 decreased to 0.98524. These results indicate a diminished ability of the COA-MLP model to generalize to new, unseen data, possibly due to overfitting or lack of convergence with the larger population size. The population size of 200 emerged as the most suitable choice for the COA-MLP model, providing a balance between complexity and generalization. Conversely, the larger population size of 450 led to a suboptimal model performance, emphasizing the importance of parameter tuning in meta-heuristic algorithms like COA when applied to MLP structures.

Table 1. The network results for various COA-MLP pairings.

Population size	Training dataset		Testing dataset		Scoring				Total Score	Rank
	RMSE	R^2	RMSE	R^2	Training		Testing			
50	4.74548	0.98994	4.85438	0.98925	4	4	4	4	16	7
100	4.65071	0.99034	4.69975	0.98993	5	5	5	5	20	6
150	4.98577	0.98889	5.08978	0.98818	3	3	3	3	12	8

200	3.76348	0.99369	3.85332	0.99324	10	10	10	10	40	1
250	5.31614	0.98736	5.33573	0.98700	2	2	2	2	8	9
300	4.04138	0.99272	4.11705	0.99228	7	7	7	7	28	4
350	4.40944	0.99132	4.48229	0.99084	6	6	6	6	24	5
400	3.90806	0.99319	3.96893	0.99283	8	8	8	8	32	3
450	5.62769	0.98583	5.68331	0.98524	1	1	1	1	4	10
500	3.77750	0.99364	3.86867	0.99319	9	9	9	9	36	2

The MVO-MLP model demonstrated its superior performance when configured with a population size of 250. During training, this configuration achieved an impressive RMSE of 2.86040, indicating a high level of precision in predicting energy efficiency. The corresponding R^2 value for training was 0.99636, reflecting a strong correlation between the predicted and actual values during the training phase. When evaluated on the testing dataset, this configuration continued to exhibit excellent performance with an RMSE of 2.94209 and an R^2 of 0.99607. These results suggest that a population size of 250 led to a well-optimized MVO-MLP model, effectively capturing the underlying patterns in the data and generalizing to new instances. Contrary to the initial assessment, it appears that a population size of 200 is considered the worst-performing configuration for MVO-MLP. The RMSE for training was 3.91689, and the R^2 was 0.99302. When applied to the testing dataset, the RMSE increased to 3.91689, and the R^2 decreased to 0.99302. These results indicate that a population size of 200 led to a suboptimal MVO-MLP model performance, potentially due to convergence issues or insufficient exploration of the solution space.

Table 2. The network results for various MVO-MLP pairings.

Population size	Training dataset		Testing dataset		Scoring				Total Score	Rank
	RMSE	R^2	RMSE	R^2	Training		Testing			
50	3.80074	0.99356	3.87873	0.99315	2	2	2	2	8	9
100	3.58083	0.99429	3.63533	0.99399	3	3	3	3	12	8
150	3.54821	0.99439	3.61757	0.99405	4	4	4	4	16	7
200	3.82472	0.99348	3.91689	0.99302	1	1	1	1	4	10
250	2.86040	0.99636	2.94209	0.99607	10	10	10	10	40	1
300	3.23848	0.99533	3.32938	0.99496	7	7	7	7	28	4
350	3.32624	0.99507	3.39047	0.99477	6	6	6	6	24	5
400	3.46003	0.99467	3.52017	0.99436	5	5	5	5	20	6
450	3.16521	0.99554	3.23368	0.99525	8	8	8	8	32	3
500	3.06791	0.99581	3.15861	0.99546	9	9	9	9	36	2

The LCA-MLP model performed exceptionally well with a population size of 200. During training, this configuration achieved an RMSE of 3.88871 and an R^2 of 0.99326, indicating a high degree of accuracy in predicting energy efficiency. When evaluated on the testing dataset, the model's performance remained robust with an RMSE of 3.97548 and an R^2 of 0.99281. These results suggest that a population size of 200 led to a well-optimized LCA-MLP model, demonstrating its ability to generalize effectively to new data. Contrarily, a population size of 50 appears to be the least effective for the LCA-MLP model. The model trained with a population size of 50 exhibited higher errors, with an RMSE of 4.51894 and an R^2 of 0.99089 during the training phase. These issues persisted when the model was tested on new data, resulting in an RMSE of 4.59644 and an R^2 of 0.99037. These outcomes indicate that a smaller population size of 50 led to suboptimal training and testing performance for the LCA-MLP model. The model may have struggled to capture the complexity of the underlying patterns in the data, resulting in reduced predictive accuracy.

Table 3. The network results for various LCA-MLP pairings.

Population size	Training dataset		Testing dataset		Scoring				Total Score	Rank
	RMSE	R^2	RMSE	R^2	Training		Testing			
50	4.51894	0.99089	4.59644	0.99037	1	1	1	1	4	10
100	4.19568	0.99215	4.27880	0.99166	8	8	8	8	32	3
150	4.21899	0.99206	4.35842	0.99135	7	7	6	6	26	4
200	3.88871	0.99326	3.97548	0.99281	10	10	10	10	40	1
250	4.05630	0.99266	4.18441	0.99203	9	9	9	9	36	2
300	4.47348	0.99107	4.56618	0.99050	3	3	3	3	12	8
350	4.23930	0.99198	4.29066	0.99161	6	6	7	7	26	4
400	4.25251	0.99193	4.40533	0.99116	5	5	5	5	20	6
450	4.48317	0.99103	4.57893	0.99044	2	2	2	2	8	9
500	4.27031	0.99186	4.41622	0.99111	4	4	4	4	16	7

The ERWCA-MLP model achieved its peak performance with a population size of 250. During training, this configuration demonstrated an impressive RMSE of 2.03778 and an R^2 of 0.99815, showcasing high accuracy in predicting energy efficiency. When evaluated on the testing dataset, the model's excellence persisted with an RMSE of 2.09627 and an R^2 of 0.99800. These results indicate that a population size of 250 led to a well-optimized ERWCA-MLP model, displaying strong generalization capabilities. In contrast, a population size of 50 appears to be the least effective for the ERWCA-MLP model. The model trained with this smaller population size exhibited higher errors, with an RMSE of 3.11111 and an R^2 of 0.99569 during the training phase. These issues persisted when the model was tested on new data, resulting in an RMSE of 3.19425 and an R^2 of 0.99536. These outcomes suggest that a smaller population size of 50 led to suboptimal training and testing performance for the ERWCA-MLP model. The model may have struggled to capture the complexity of the underlying patterns in the data, resulting in reduced predictive accuracy.

Table 4. The network results for various ERWCA-MLP pairings.

Population size	Training dataset		Testing dataset		Scoring				Total Score	Rank
	RMSE	R^2	RMSE	R^2	Training		Testing			
50	3.11111	0.99569	3.19425	0.99536	1	1	1	1	4	10
100	2.76610	0.99659	2.85663	0.99629	2	2	2	2	8	9
150	2.42653	0.99738	2.48403	0.99720	5	5	5	5	20	6
200	2.64471	0.99689	2.70127	0.99668	4	4	4	4	16	7
250	2.03778	0.99815	2.09627	0.99800	10	10	10	10	40	1
300	2.65120	0.99687	2.76177	0.99653	3	3	3	3	12	8
350	2.36080	0.99752	2.37750	0.99743	6	6	8	8	28	4
400	2.28214	0.99768	2.38845	0.99741	8	8	7	7	30	3
450	2.32085	0.99760	2.41863	0.99734	7	7	6	6	26	5
500	2.10771	0.99802	2.21501	0.99777	9	9	9	9	36	2

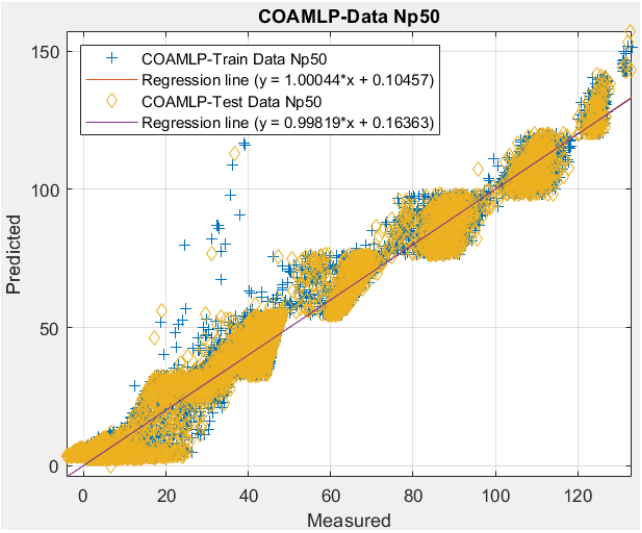
The SFS-MLP model demonstrated its optimal performance with a population size of 400. During the training phase, this configuration achieved a remarkable RMSE of 2.50700 and an R^2 of 0.99720, highlighting its ability to accurately predict energy efficiency based on the given features. Upon evaluation on the testing dataset, the excellence of the model persisted with an RMSE of 2.59167 and an R^2 of 0.99695. These results indicate that a population size of 400 led to a well-tuned SFS-MLP model, showcasing strong generalization capabilities. Conversely, a population size of 50

appears to be the least effective for the SFS-MLP model. The model trained with this smaller population size exhibited higher errors, with an RMSE of 2.97527 and an R^2 of 0.99606 during the training phase. These issues persisted when the model was tested on new data, resulting in an RMSE of 3.04438 and an R^2 of 0.99579. These outcomes suggest that a smaller population size of 50 led to suboptimal training and testing performance for the SFS-MLP model. The model may have struggled to adequately capture the underlying patterns in the data, resulting in reduced predictive accuracy.

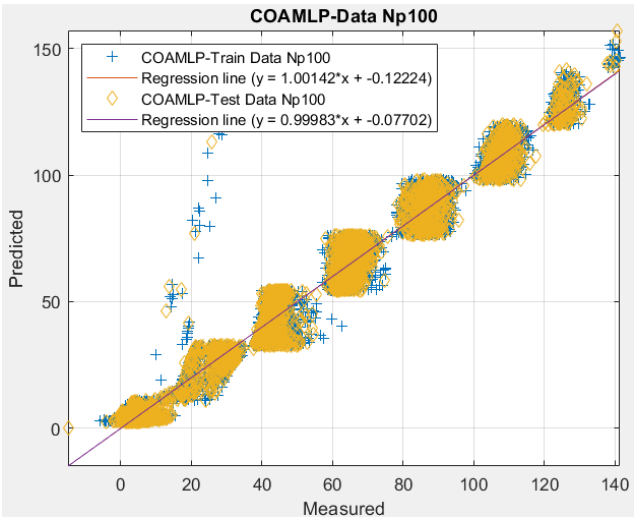
Table 5. The network results for various SFS-MLP pairings.

Population size	Training dataset		Testing dataset		Scoring				Total Score	Rank
	RMSE	R^2	RMSE	R^2	Training		Testing			
50	2.97527	0.99606	3.04438	0.99579	1	1	1	1	4	10
100	2.92716	0.99619	3.00053	0.99591	3	3	3	3	12	8
150	2.81352	0.99648	2.89390	0.99619	4	4	4	4	16	7
200	2.63799	0.99690	2.72825	0.99662	7	7	6	6	26	4
250	2.73470	0.99667	2.81791	0.99639	5	5	5	5	20	6
300	2.95066	0.99612	3.00906	0.99588	2	2	2	2	8	9
350	2.64454	0.99689	2.69327	0.99670	6	6	7	7	26	4
400	2.50700	0.99720	2.59167	0.99695	10	10	10	10	40	1
450	2.53216	0.99715	2.60059	0.99693	9	9	9	9	36	2
500	2.59696	0.99700	2.67468	0.99675	8	8	8	8	32	3

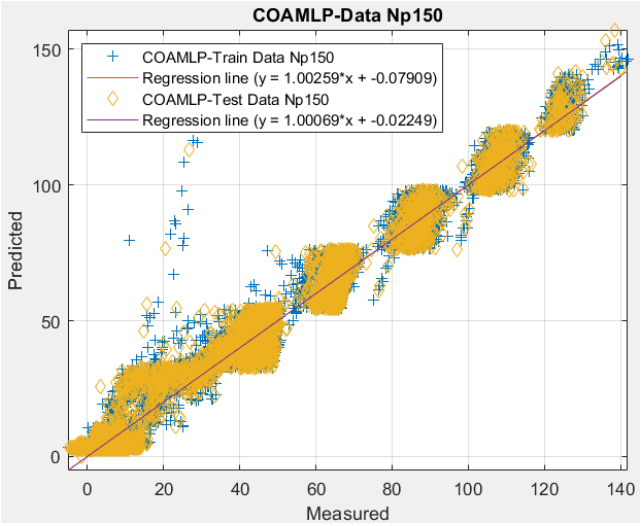
The performance of the model is evaluated in the second stage by comparing the anticipated values of the hybrid design with the actual data. This study uses the R^2 statistic, which is a commonly used metric for assessing the efficacy of hybrid designs. Researchers may get more understanding by comparing projected values with actual data to assess the model's accuracy and capacity to capture dataset variation. An important factor influencing a binary classifier system's diagnostic performance is the discriminating threshold selection. Figure 6-10 provides a graphical picture of how changing this threshold affects the model's capacity to discriminate between positive and negative categories. Higher R^2 values suggest that the model performs better when it comes to group differentiation. The structural R^2 plots of the hybrid are best-fit. These charts show how well the model performs in relation to several distinguishing factors. Using the suggested hybrid COA-MLP, MVO-MLP, LCA-MLP, ERWCA-MLP, and SFS-MLP models as a basis, the first and most important stage is to identify the optimal prediction model. During the iteration phase, ideal population sizes—200, 250, 200, 250, and 400—are selected, highlighting the necessity of fine-tuning these parameters to improve prediction accuracy. This all-inclusive method evaluates the hybrid models' prediction power and helps choose the best design for enhanced diagnostic performance based on the particular dataset and application. Iteration makes models more helpful for estimating steel energy usage by ensuring that they are selected and improved for best outcomes.



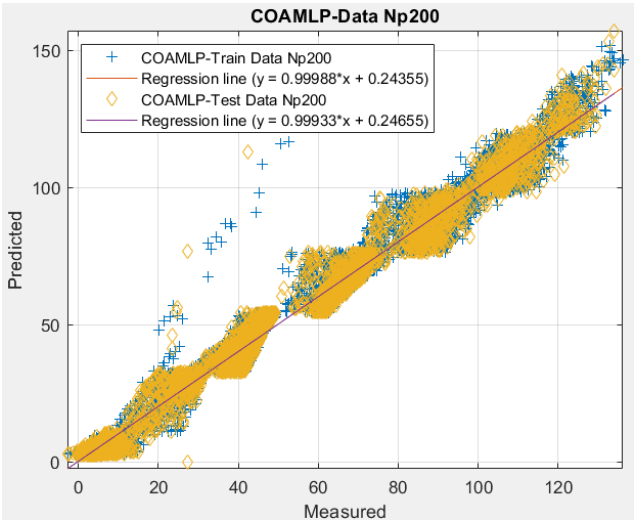
(a) COAMLP-N_p=50



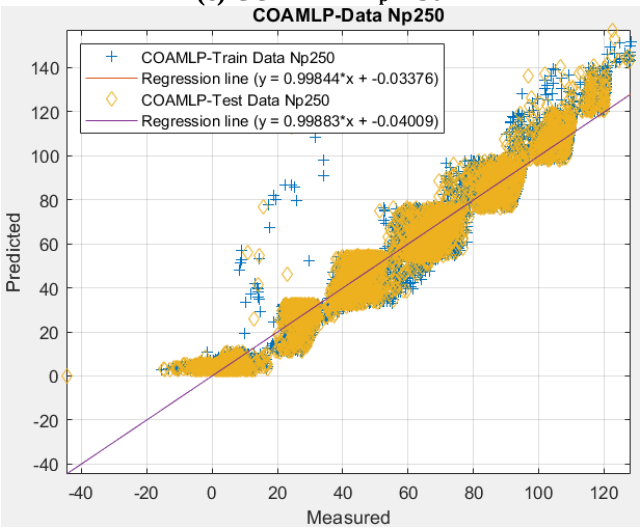
(b) COAMLP-N_p=100



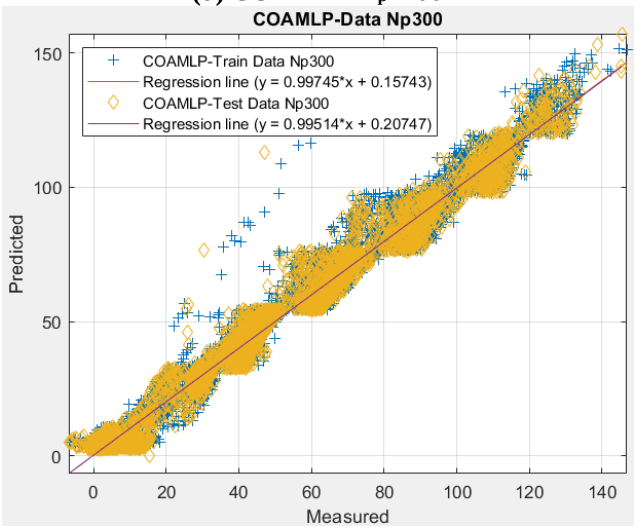
(c) COAMLP-N_p=150



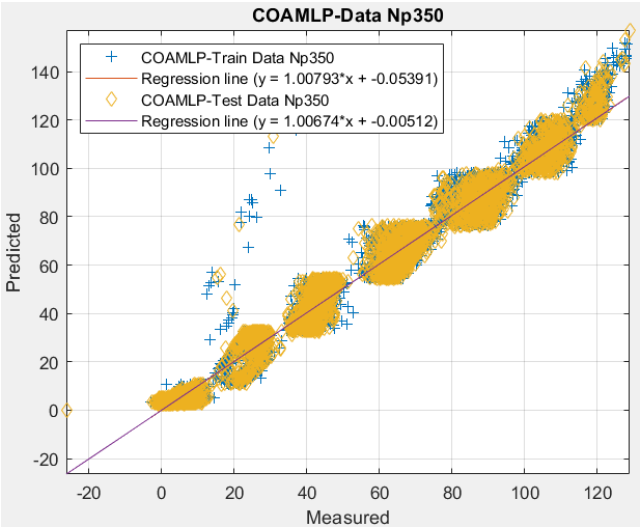
(d) COAMLP-N_p=200



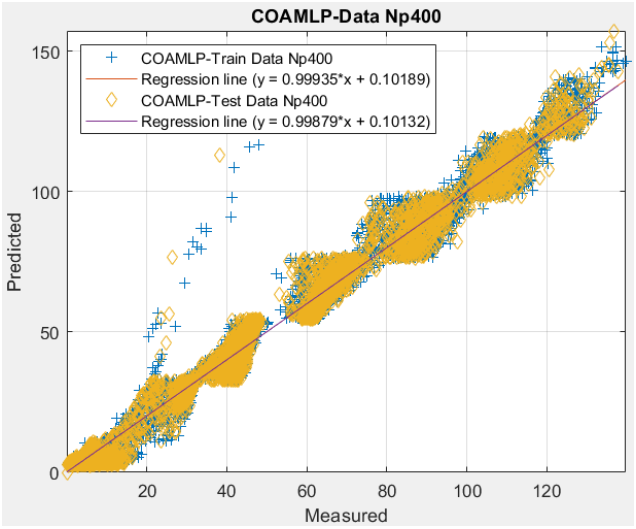
(e) COAMLP-N_p=250



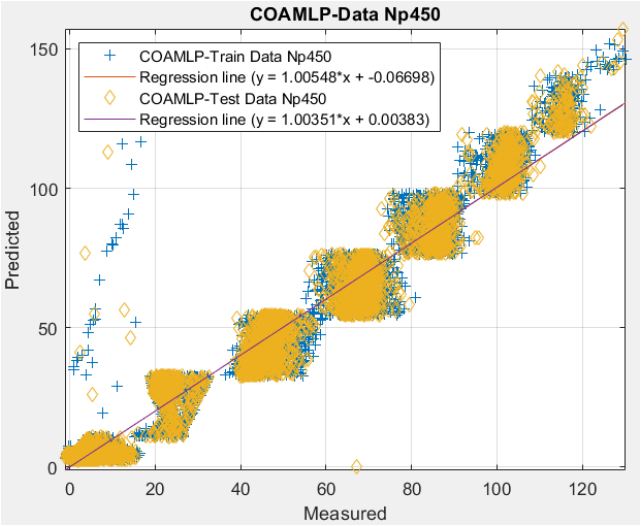
(f) COAMLP-N_p=300



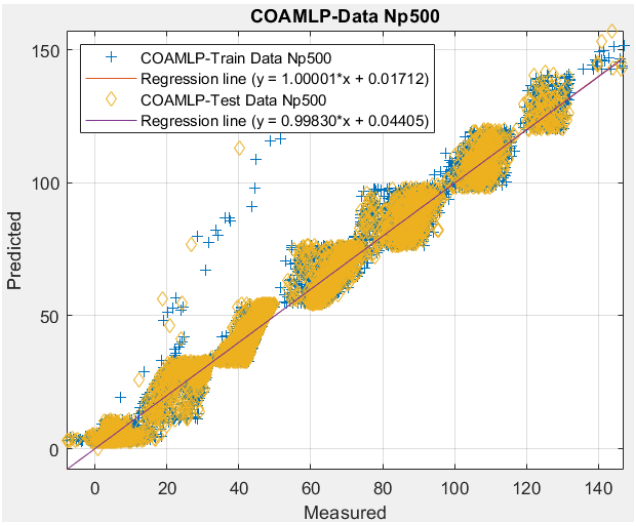
(g) COAMLP-N_p=350



(h) COAMLP-N_p=400

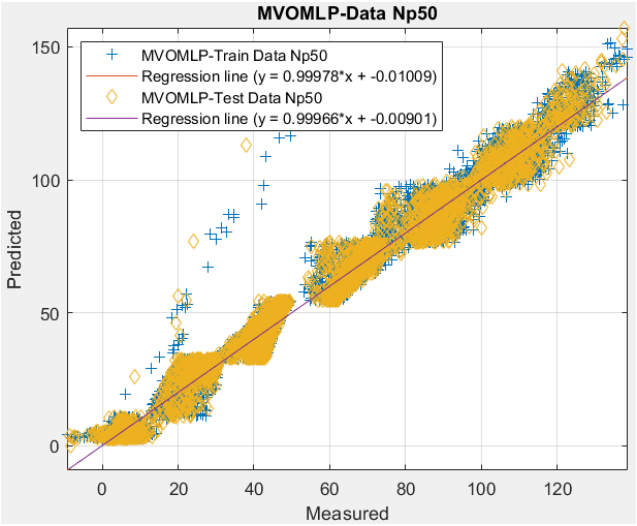


(i) COAMLP-N_p=450

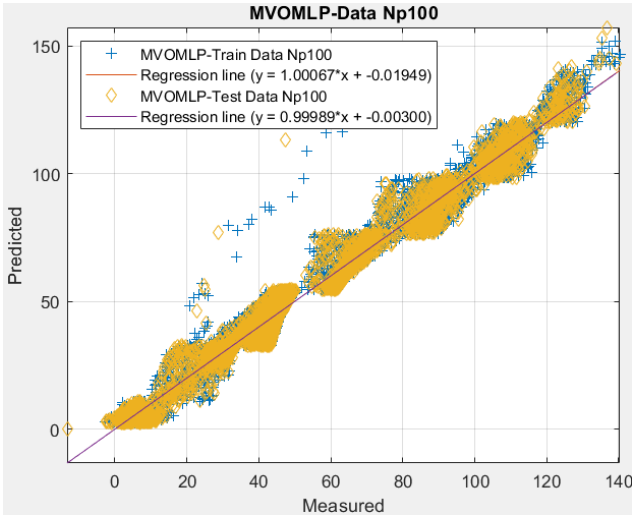


(j) COAMLP-N_p=500

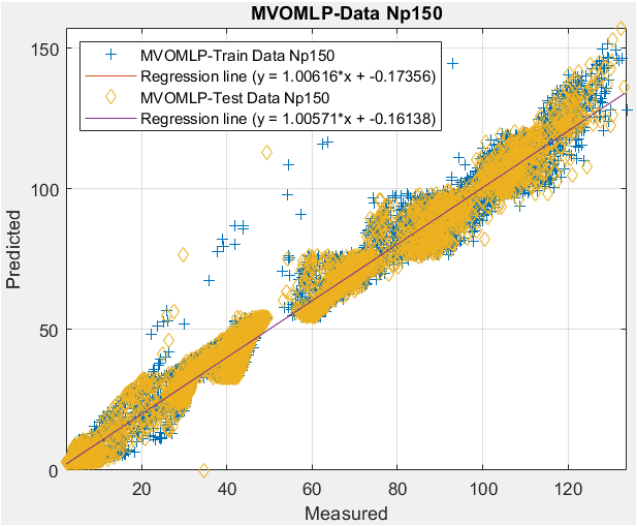
Figure 6. The accuracy findings for best-fit architectures for the COA-MLP model.



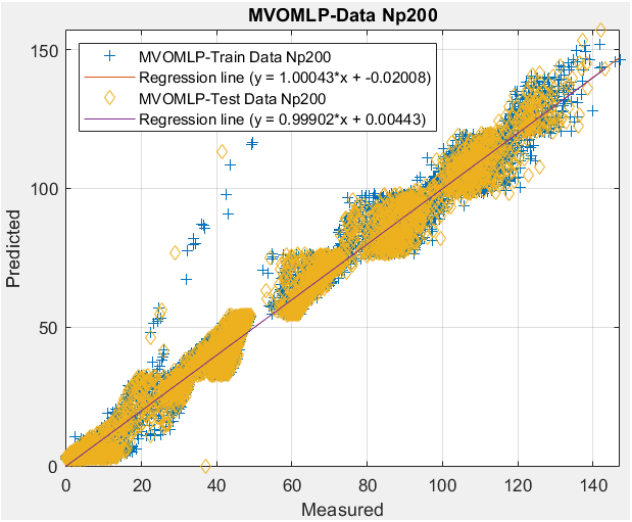
(a) MVOMLP-N_p=50



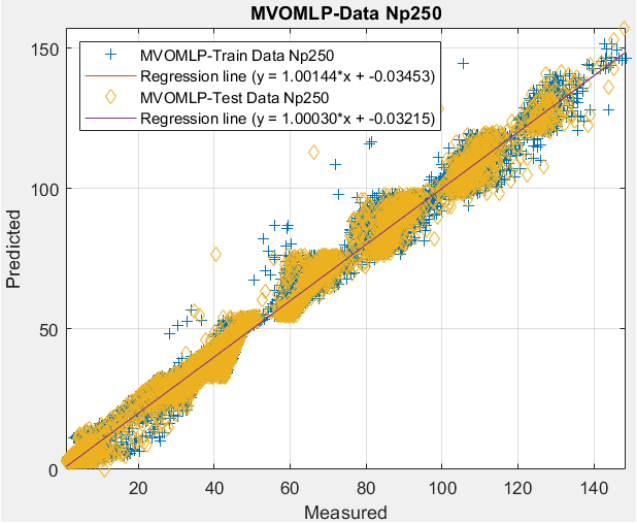
(b) MVOMLP -N_p=100



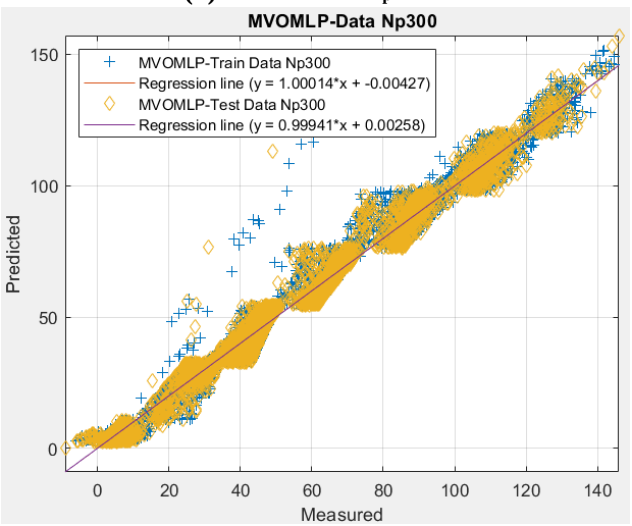
(c) MVOMLP -N_p=150



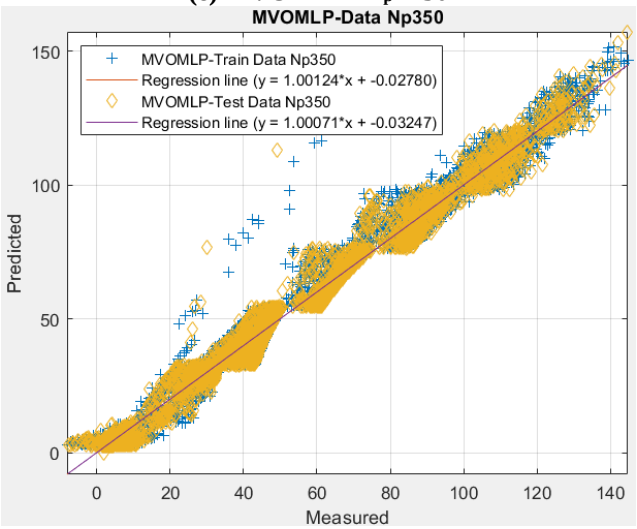
(d) MVOMLP -N_p=200



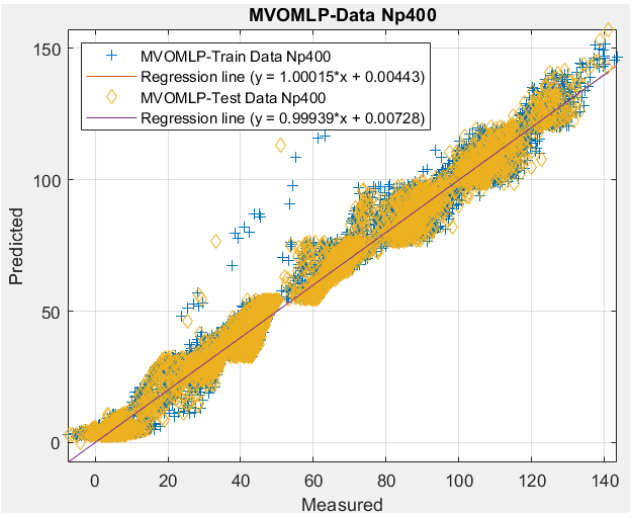
(e) MVOMLP -N_p=250



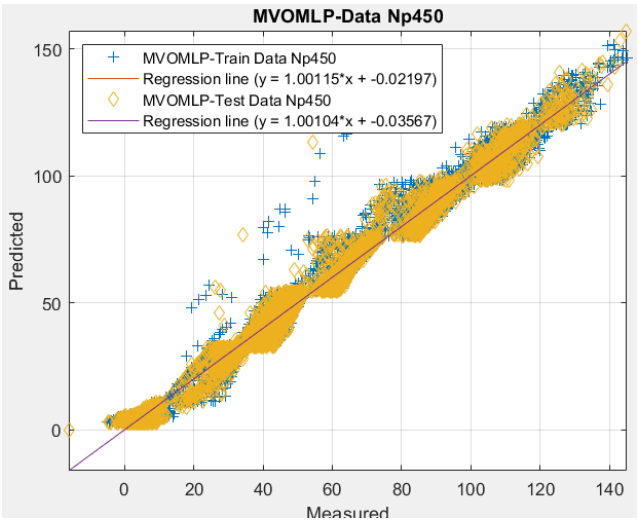
(f) MVOMLP -N_p=300

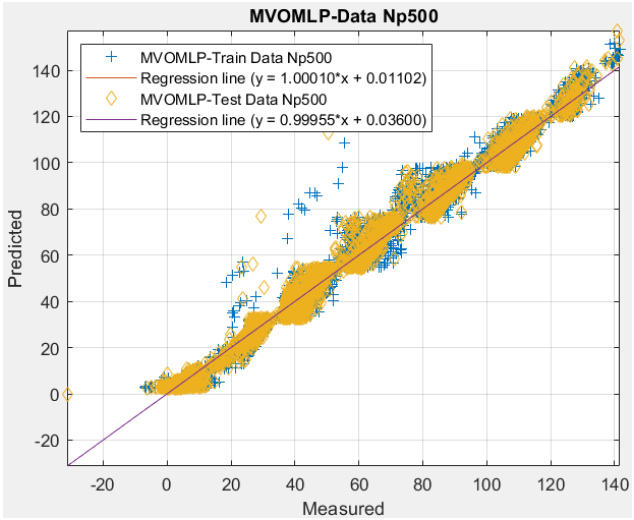


(g) MVOMLP -N_p=350



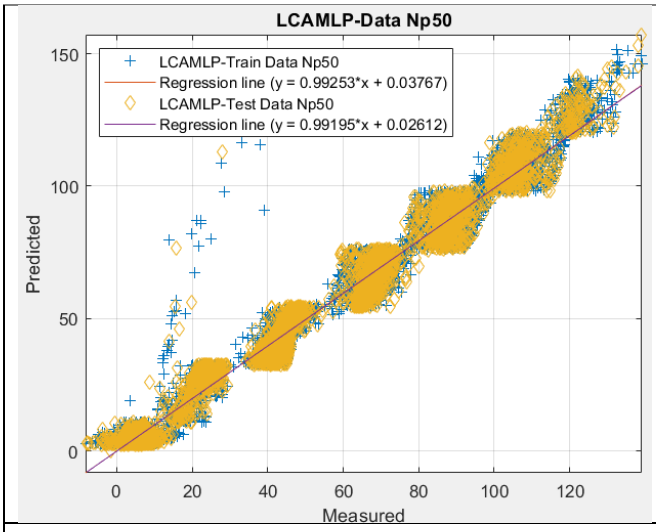
(h) MVOMLP -N_p=400



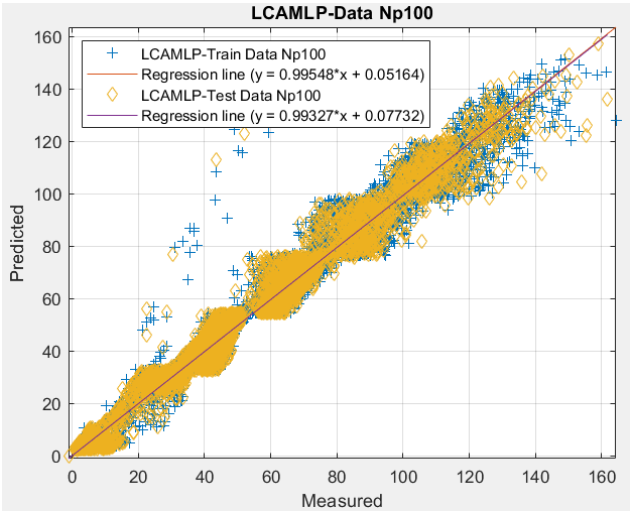


(i) MVOMLP -N_p=450 **(j) MVOMLP -N_p=500**

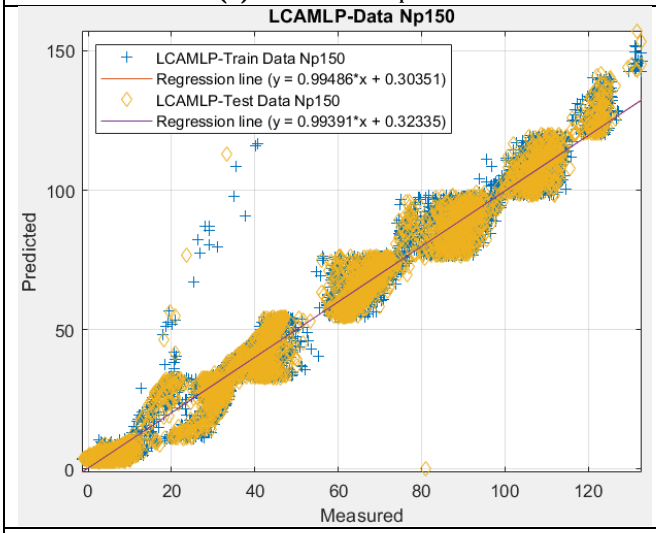
Figure 7. The accuracy findings for best-fit architectures for the MVO-MLP model.



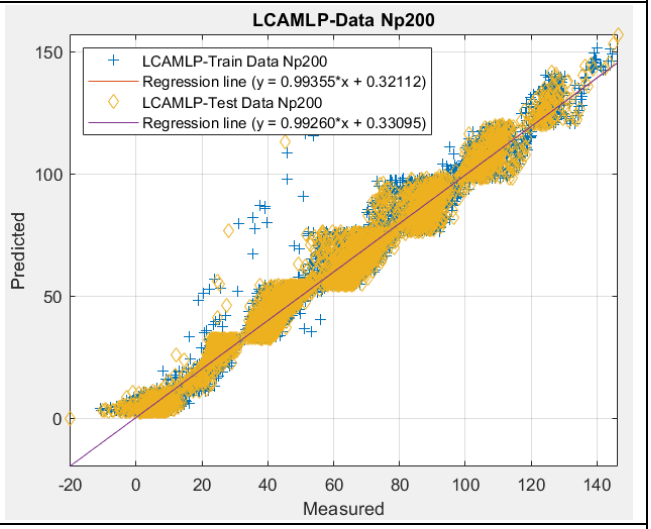
(a) LCAMLP-N_p=50



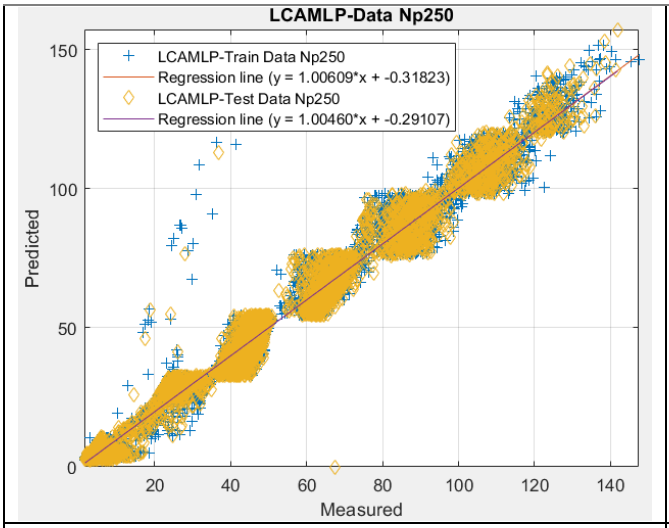
(b) LCAMLP -N_p=100



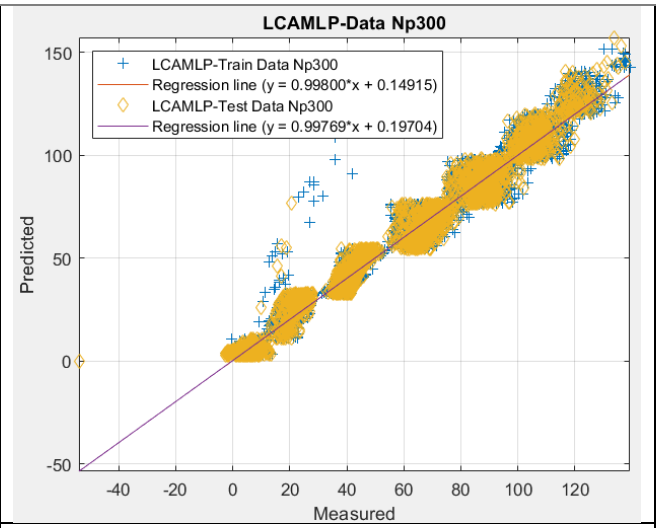
(c) LCAMLP -N_p=150



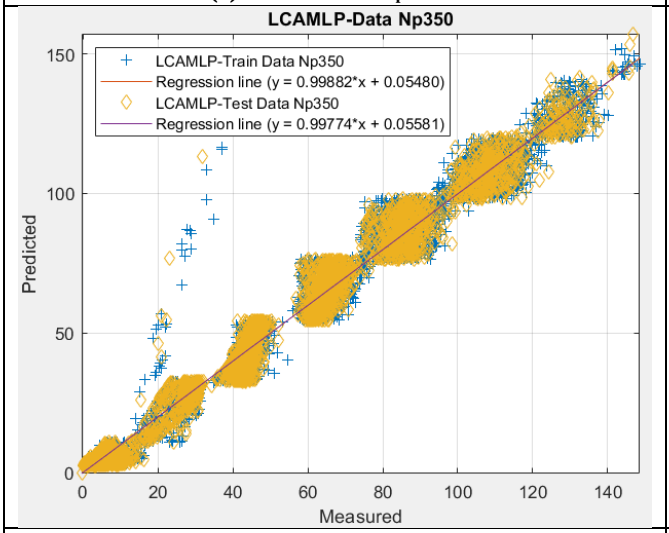
(d) LCAMLP -N_p=200



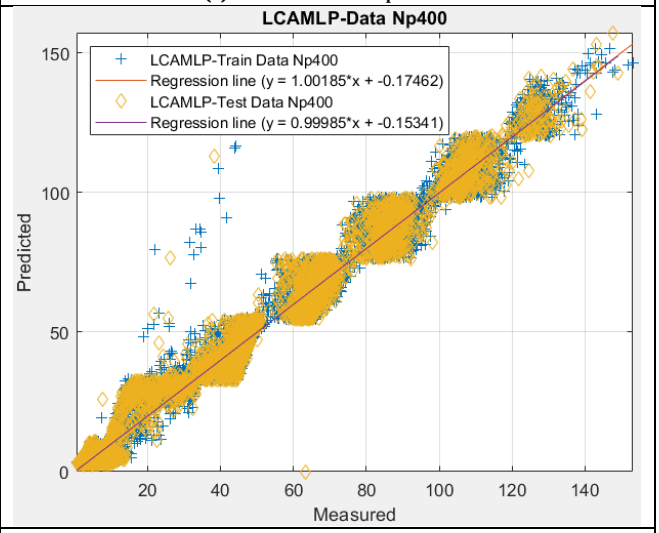
(e) LCAMLP -N_p=250



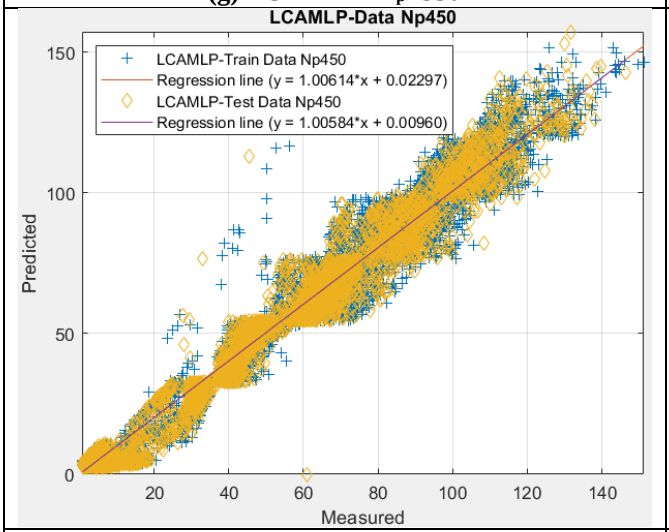
(f) LCAMLP -N_p=300



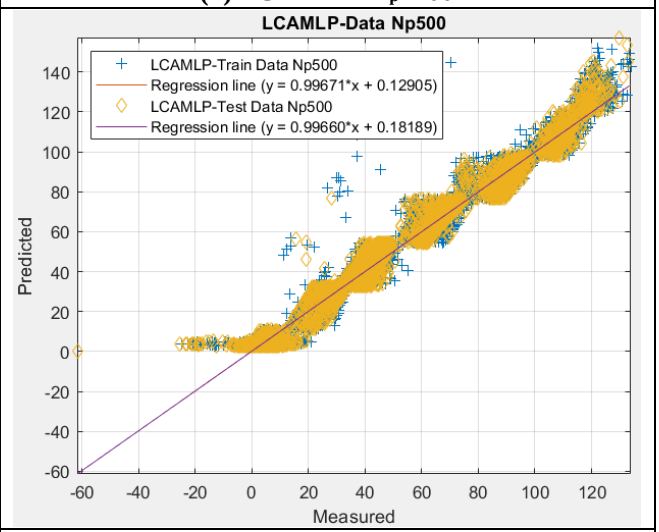
(g) LCAMLP -N_p=350



(h) LCAMLP -N_p=400

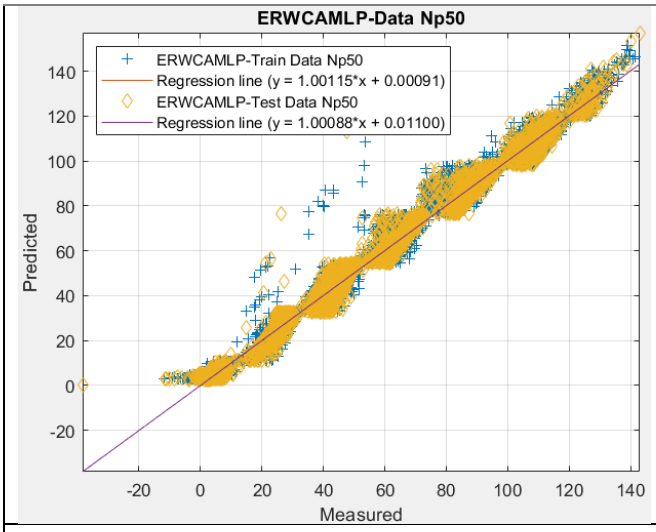


(i) LCAMLP -N_p=450

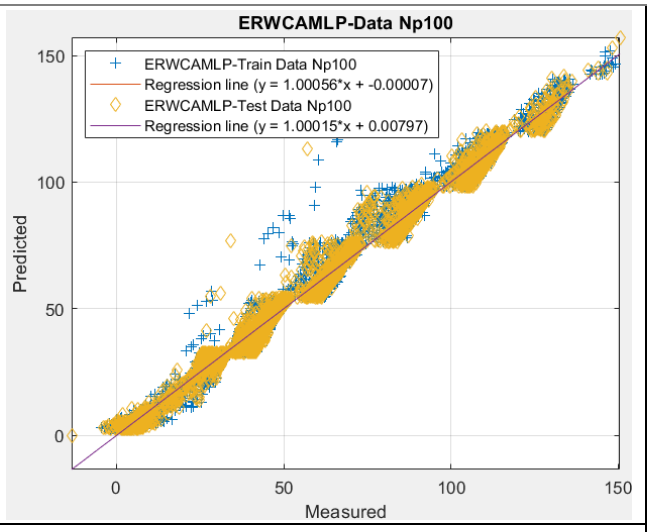


(j) LCAMLP -N_p=500

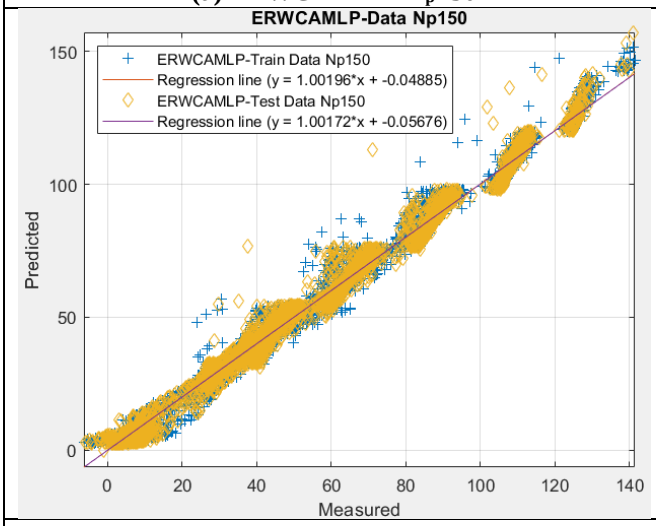
Figure 8. The accuracy findings for best-fit architectures for the LCA-MLP model.



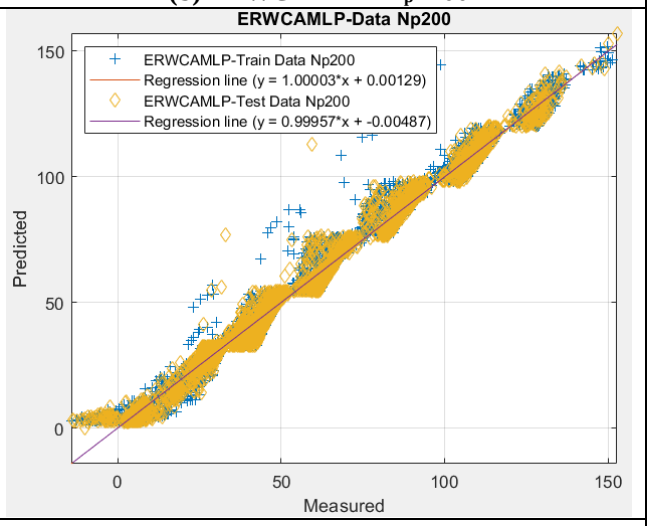
(a) ERWCAMLP -N_p=50



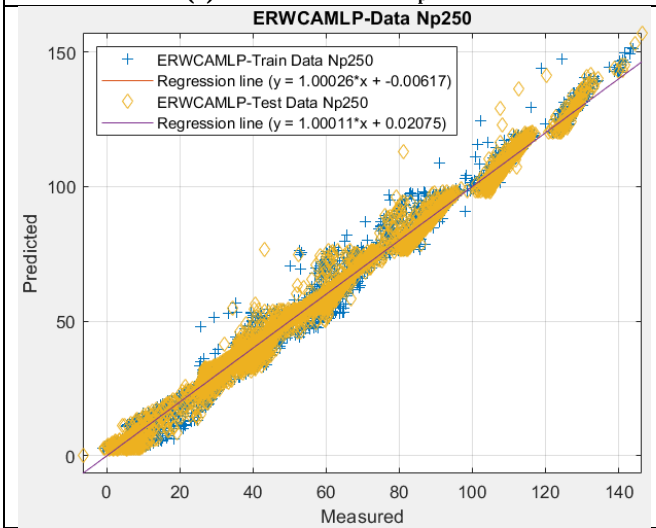
(b) ERWCAMLP -N_p=100



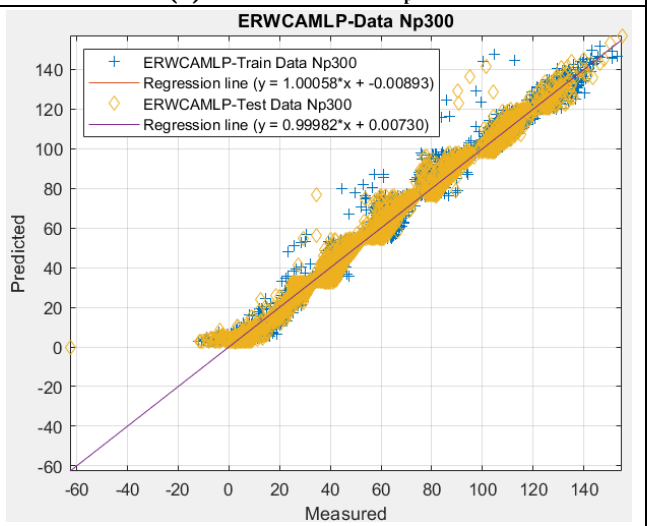
(c) ERWCAMLP -N_p=150



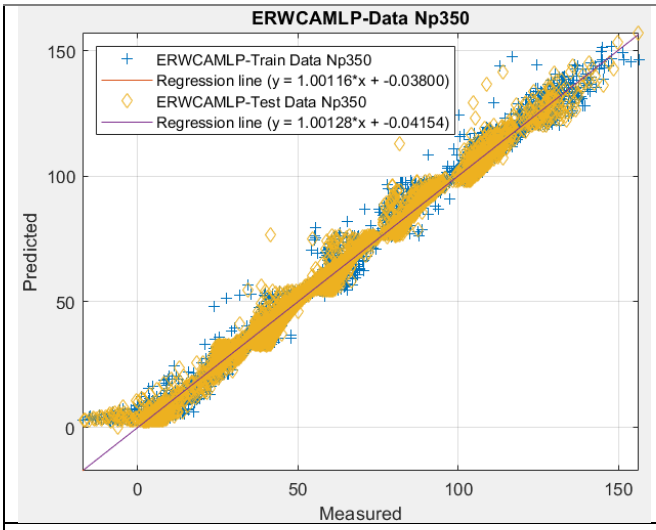
(d) ERWCAMLP -N_p=200



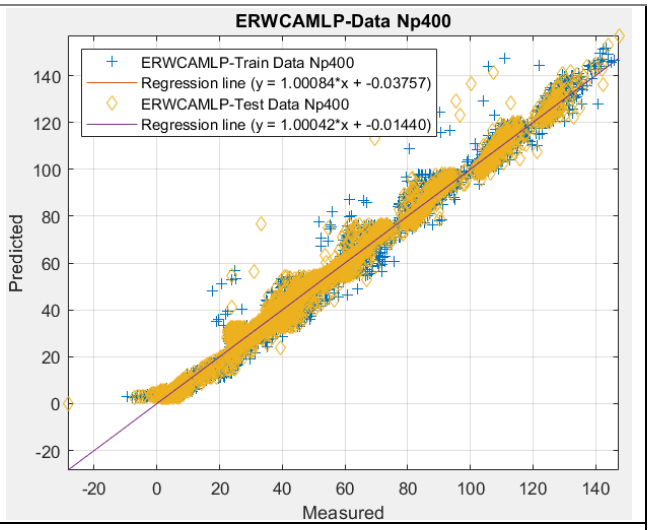
(e) ERWCAMLP -N_p=250



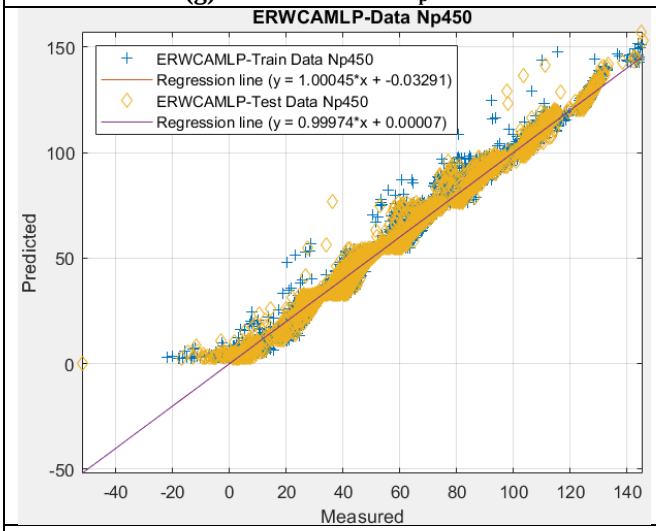
(f) ERWCAMLP -N_p=300



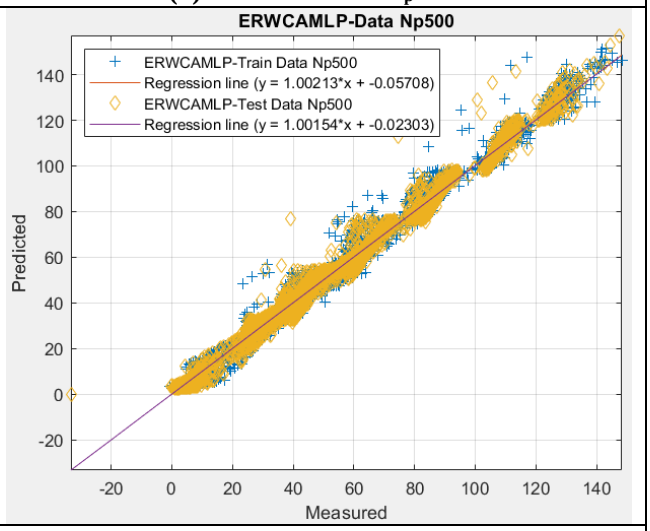
(g) ERWCAMLP -N_p=350



(h) ERWCAMLP -N_p=400

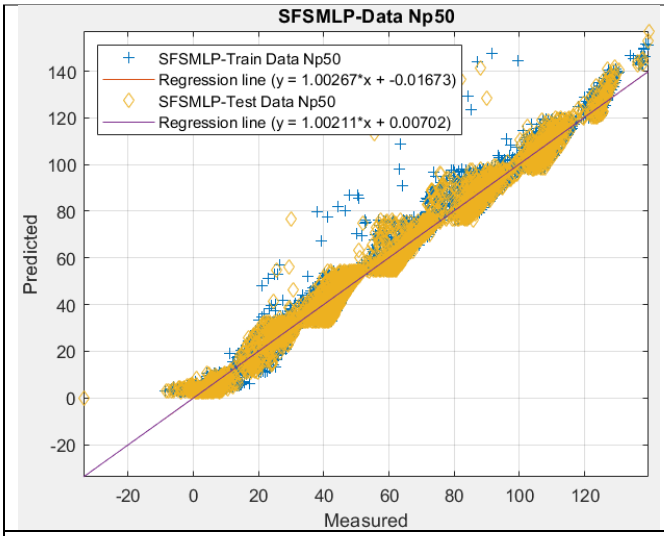


(i) ERWCAMLP -N_p=450

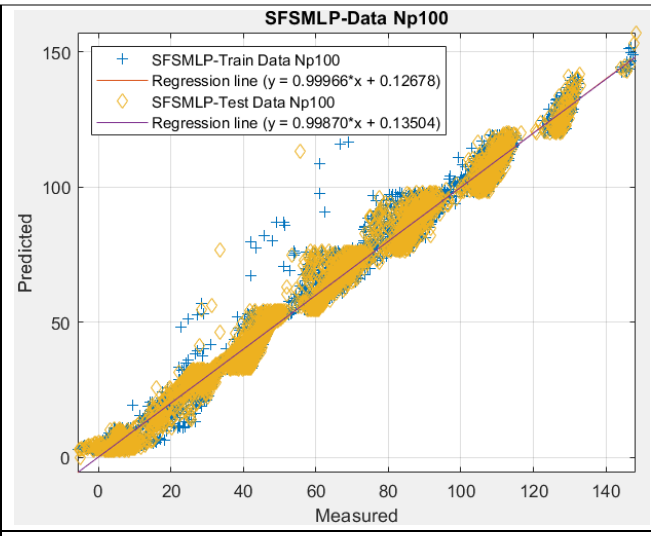


(j) ERWCAMLP -N_p=500

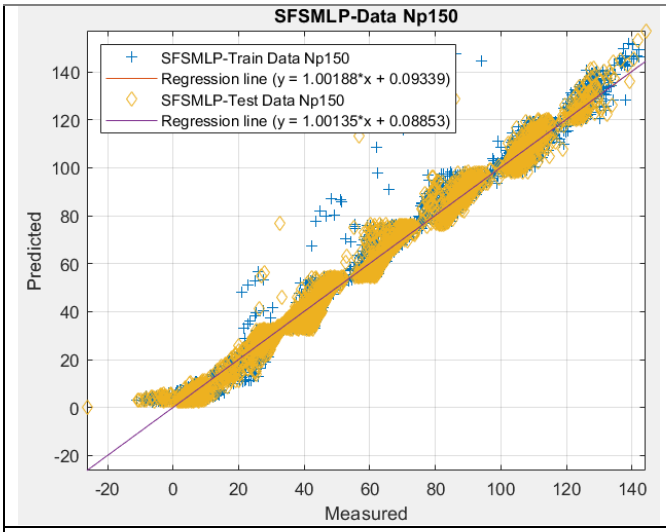
Figure 9. The accuracy findings for best-fit architectures for the ERWCA-MLP model.



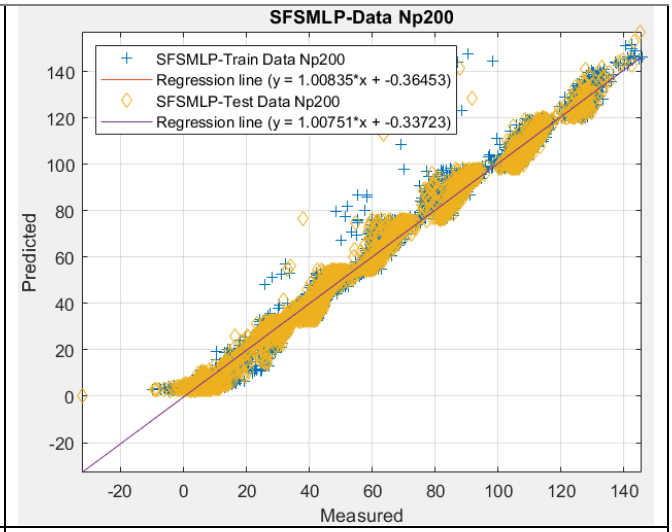
(a) SFSMLP-N_p=50



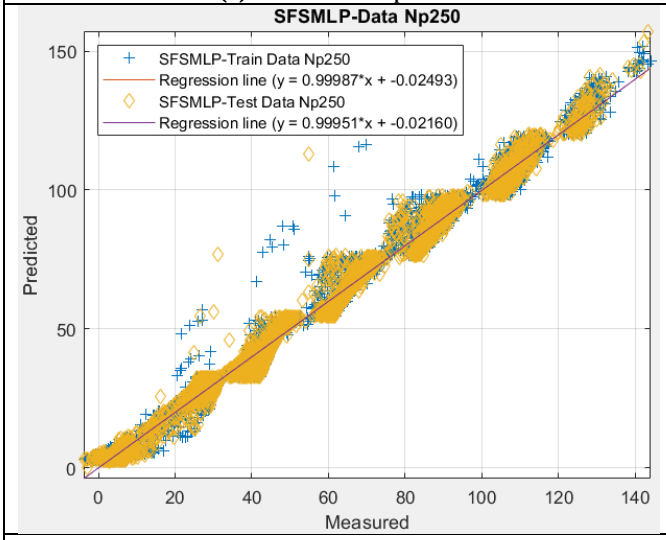
(b) SFSMLP -N_p=100



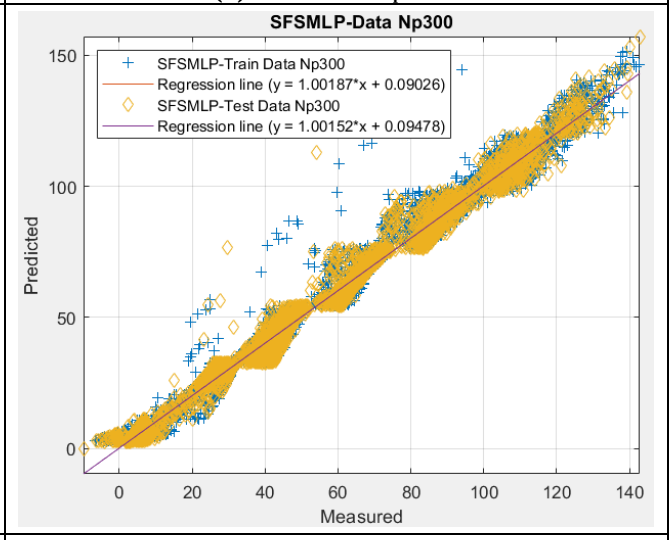
(c) SFSMLP -N_p=150



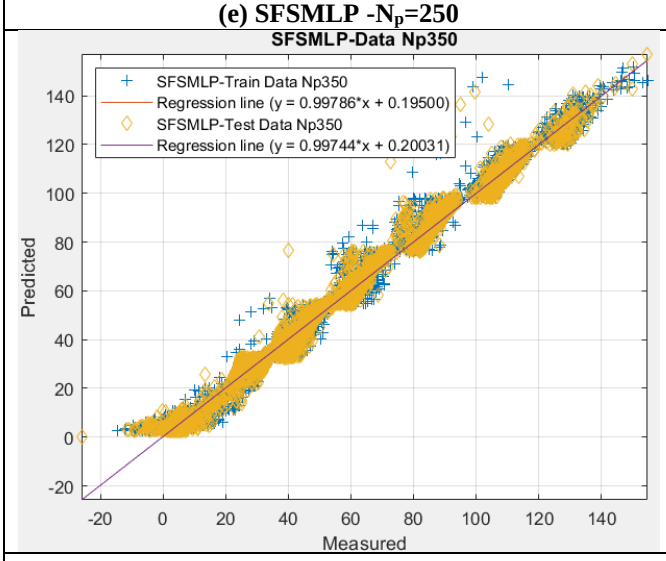
(d) SFSMLP -N_p=200



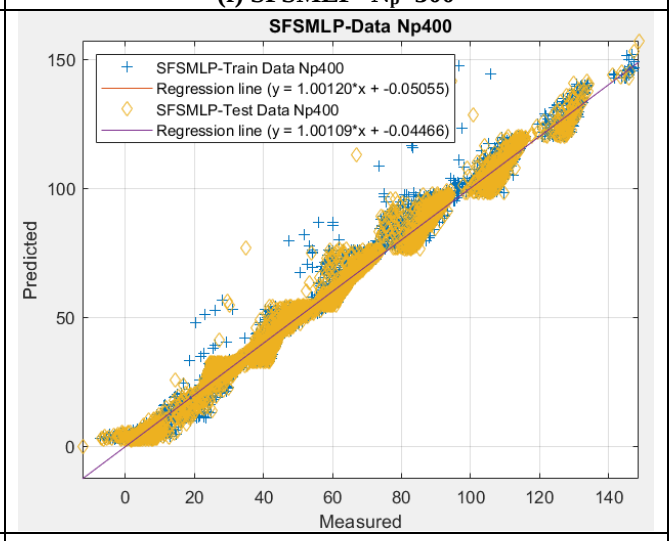
(e) SFSMLP -N_p=250



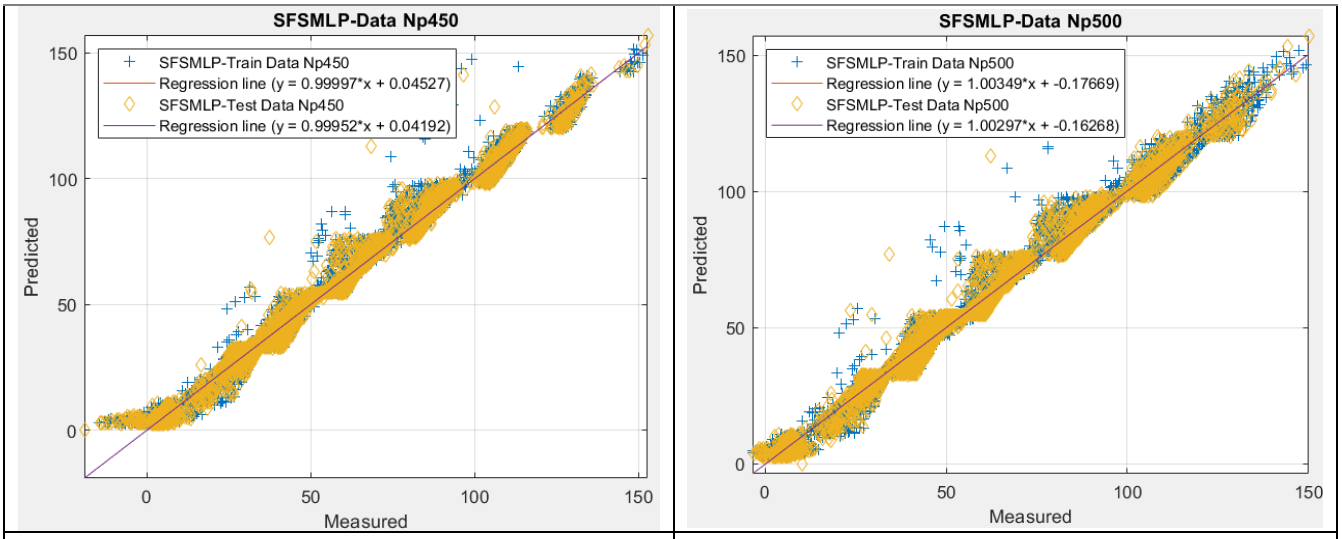
(f) SFSMLP -N_p=300



(g) SFSMLP -N_p=350



(h) SFSMLP -N_p=400

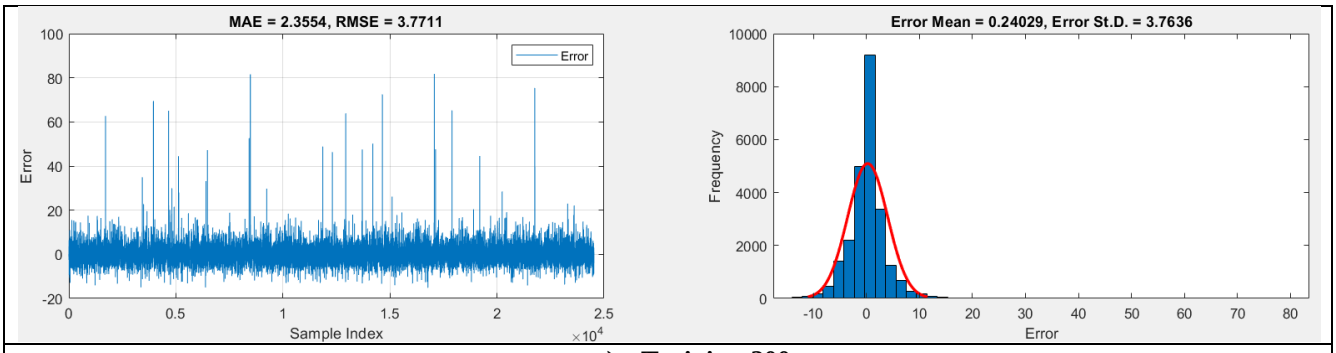


(i) SFSMLP -N_p=450 **(j) SFSMLP -N_p=500**

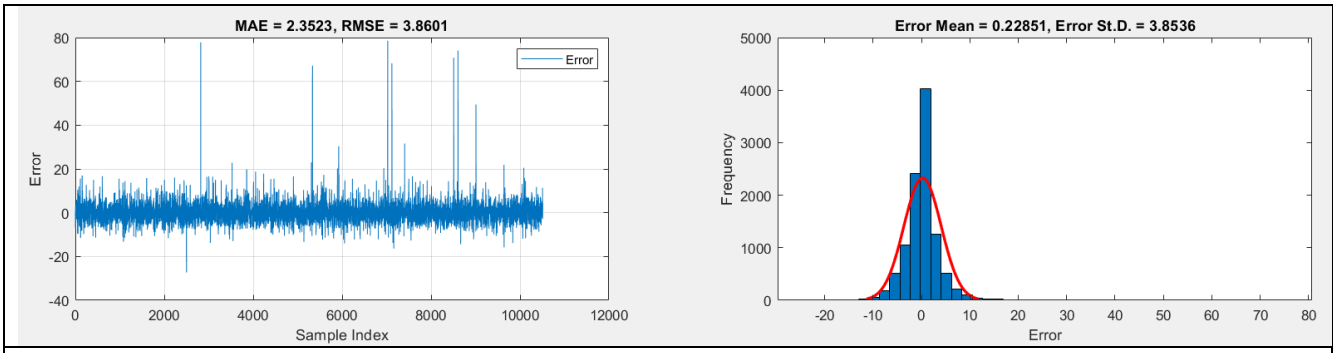
Figure 10. The accuracy findings for best-fit architectures for the SFS-MLP model.

4.2 Error analysis

Figures 11–15 display the frequency distribution of the best-fitted structures for COA–MLP, MVO–MLP, LCA–MLP, ERWCA–MLP, and SFS–MLP. The findings from the training and testing datasets show an extraordinarily high degree of agreement between the computed and observed energy consumption indicators. The consistency of the data indicates that the models are effective in accurately forecasting the energy consumption in industrial environments. The study's findings demonstrate the high degree of agreement between the estimated and observed energy usage metrics. The training and testing datasets are used to produce them. This result highlights the models' value in providing accurate estimates of industrial cities' energy usage. The methods employed by the SFS-MLP, ERWCA-MLP, LCA-MLP, COA-MLP, and MVO-MLP models collectively contribute to the reliable and accurate estimation of energy consumption. Surprisingly strong agreement between the computed and observed data suggests that these models accurately capture the underlying dynamics and patterns of energy use. This shows that the neural network models and optimization approaches utilized in this work provide a dependable method for producing accurate predictions of energy use. The results validate the utility of the proposed models and add to the growing body of knowledge about effective methods for estimating energy consumption in industrial settings.

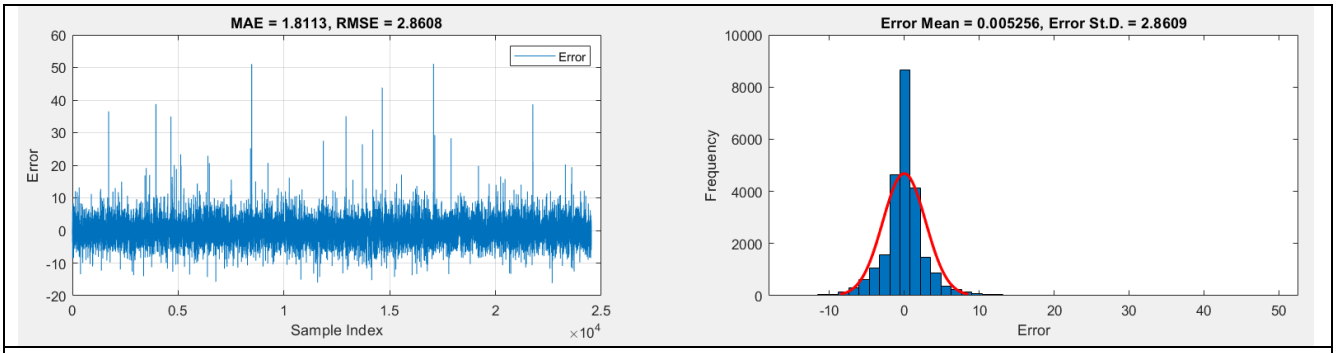


a) Training-200

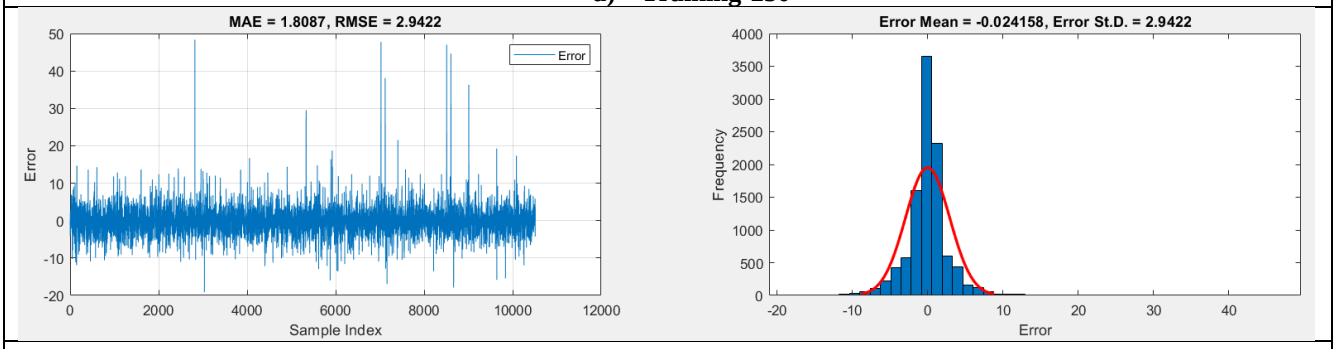


a) Testing-200

Figure 11. The ideal frequency for the COA-MLP method.

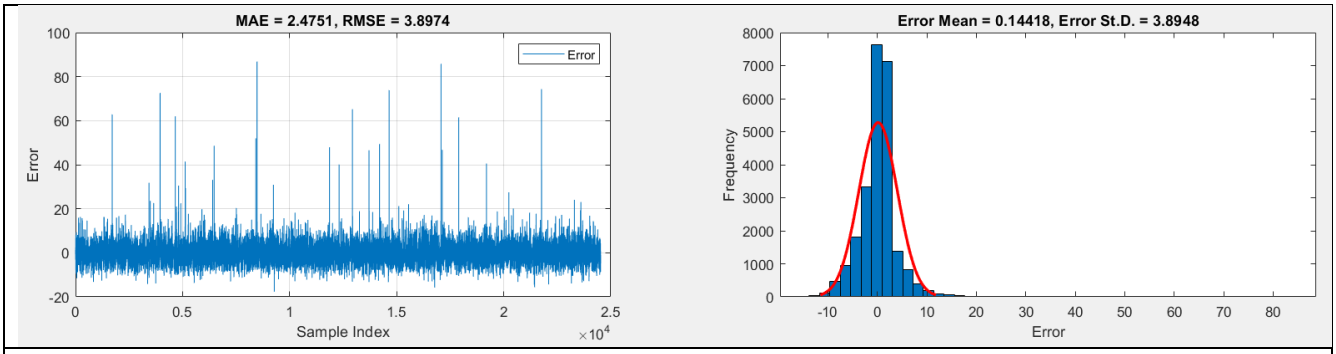


a) Training-250

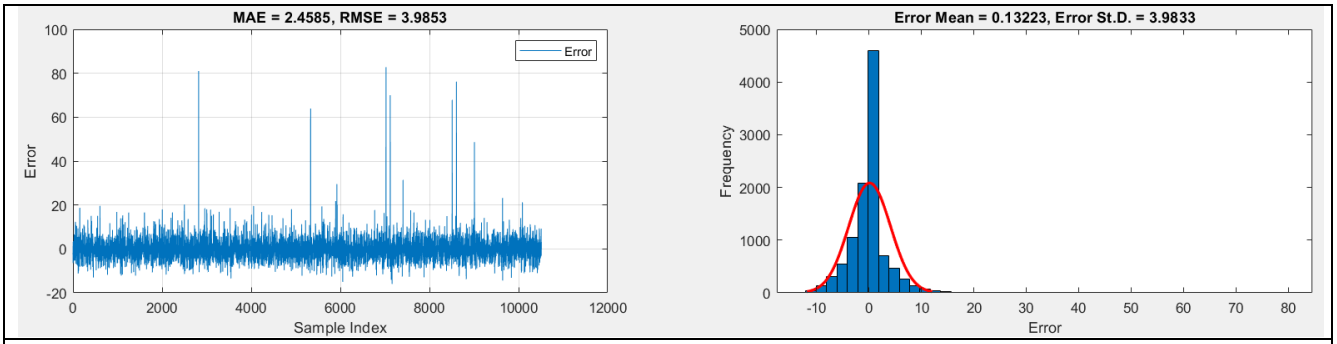


a) Testing-250

Figure 12. The ideal frequency for the MVO-MLP method.

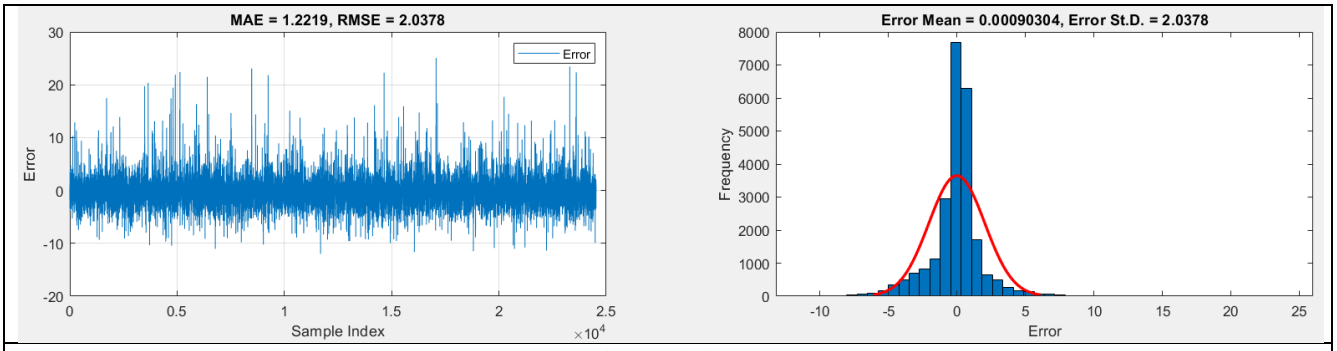


a) Training-200

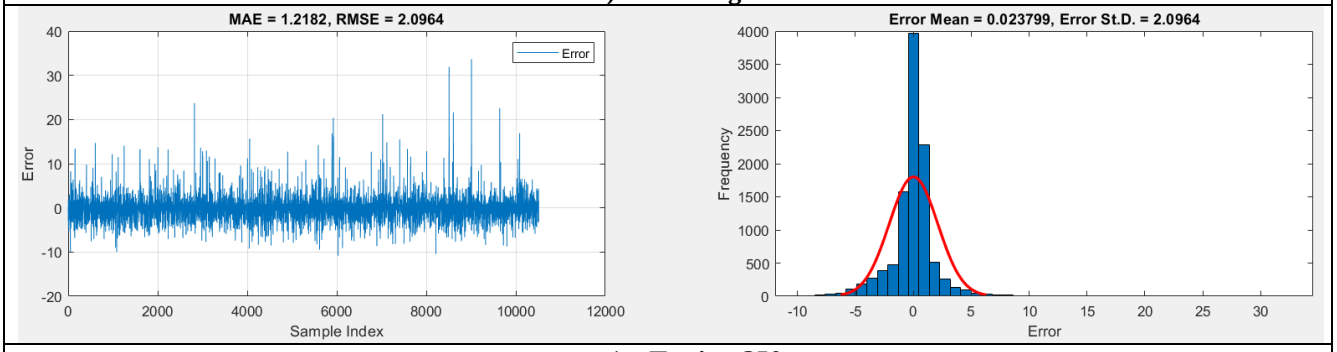


a) Testing-200

Figure 13. The ideal frequency for the LCA-MLP method.

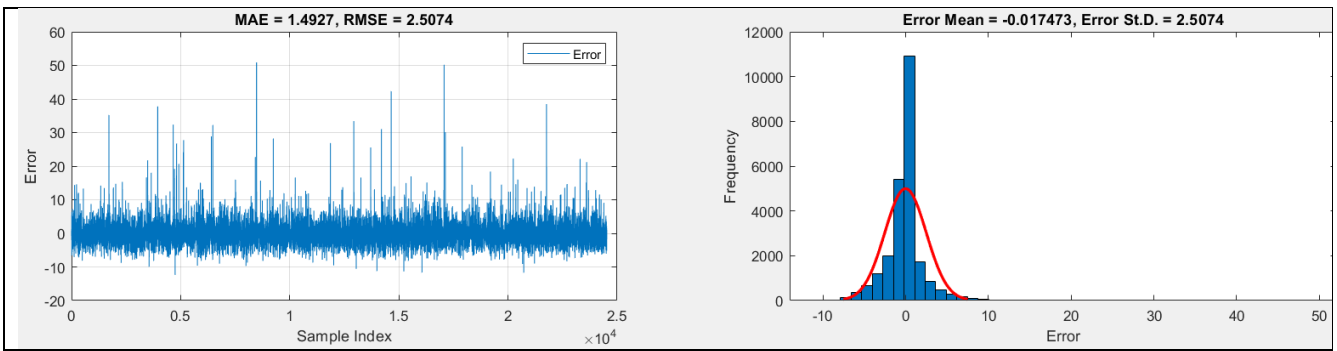


a) Training-250

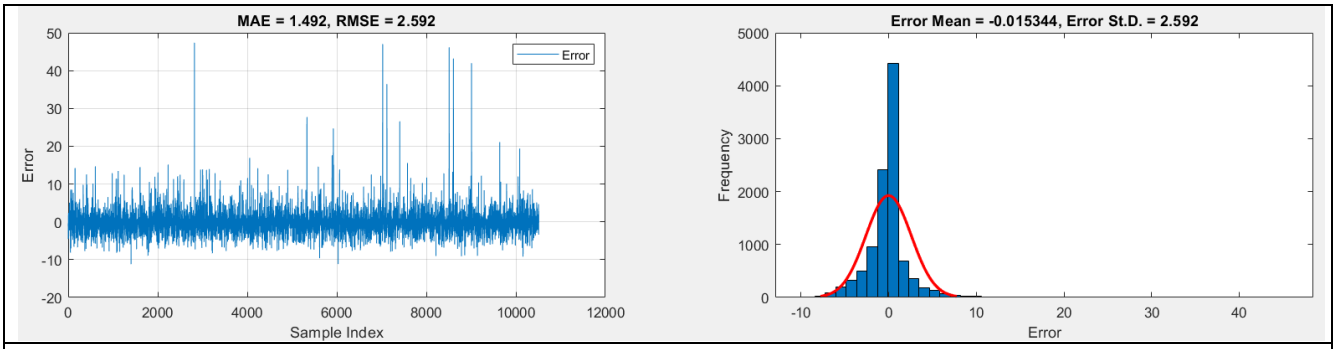


a) Testing-250

Figure 14. The ideal frequency for the ERWCA-MLP method.



a) Training-400



a) Testing-400

Figure 15. The ideal frequency for the SFS-MLP method.

4.3 Measurement of energy consumption using MLP

ERWCAMLP with a swarm size of 250 stands out with the highest R^2 scores for both training (0.99815) and testing (0.99800) datasets. This indicates a superior ability to explain the variance in the data and make accurate predictions. MVO-MLP, utilizing a swarm size of 250, achieved a notable balance between training and testing performance, exhibiting an R^2 of 0.99636 in training and 0.99607 in testing. This suggests good generalization capabilities of the model to new, unseen data. SFSMLP, configured with a population size of 400, achieved the lowest RMSE on the testing dataset (2.59167). This indicates that the model's predictions were, on average, closer to the true values during the testing phase compared to other algorithms. COAMLP and LCAMLP, both with a swarm size of 200, exhibit higher RMSE values in training compared to other algorithms. This suggests that these models may face challenges in accurately fitting the training data. The scoring metrics (based on RMSE and R^2) and the total score provide a comprehensive evaluation of each algorithm's overall performance. ERWCAMLP, with a total score of 20, secures the top rank, emphasizing its superiority in optimizing energy efficiency. This comparative analysis aids in understanding the strengths and weaknesses of each meta-heuristic algorithm in the context of energy efficiency optimization for steel production. It provides valuable insights for selecting the most effective algorithmic approach based on specific requirements and priorities.

Table 6. The COA-MLP, MVO-MLP, LCA-MLP, ERWCA-MLP, and SFS-MLP structures' network outcomes

Methods	Swarm size	Training dataset		Testing dataset		Scoring				Total Score	Rank
		RMSE	R^2	RMSE	R^2	Training		Testing			
COAMLP	200	3.76348	0.99369	3.85332	0.99324	2	2	2	2	8	4
MVOMLP	250	2.86040	0.99636	2.94209	0.99607	3	3	3	3	12	3
LCAMLP	200	3.88871	0.99326	3.97548	0.99281	1	1	1	1	4	5
ERWCAMLP	250	2.03778	0.99815	2.09627	0.99800	5	5	5	5	20	1
SFSMLP	400	2.50700	0.99720	2.59167	0.99695	4	4	4	4	16	2

4.4 Taylor Diagrams

In meteorology and climate research, the Taylor diagram—named for Karl E. Taylor—is a graphical tool used to assess how well many datasets match with a reference dataset. It is frequently used to evaluate the performance of model outputs on observational data, including climate models and numerical simulations. Visual representations of each dataset's standard deviation, correlation, and centered root mean square difference (RMSD) with respect to the reference dataset are provided. This

image can assist researchers in identifying the most sophisticated datasets since it provides a thorough overview of model performance across several domains. For evaluating and comparing models, Taylor diagrams are useful tools. They might also help in the model's development by drawing attention to regions that require work. They conduct a comprehensive analysis of the model's performance in terms of correlation, variability, and overall agreement with observational data. When it was first displayed, it was Taylor [35] provided a visual depiction of the degree to which an observation and a pattern or set of patterns are comparable. The standard deviations, the centered root-mean-square difference, and the similarity score between the two patterns are obtained from the correlation. These graphics are very helpful for analyzing multi-component complicated models or assessing how well different models perform, as the IPCC has shown [36]. The Taylor diagram in Figure 16 compares how well the model can replicate the regional distribution of the yearly average precipitation in the present datasets. We conducted a statistical analysis using four labeled models. The position of each label on the map shows how well the predicted precipitation pattern of the model matches the observed data. For the COA-MLP, MVO-MLP, LCA-MLP, ERWCA-MLP, and SFS-MLP, the pattern correlation coefficients are about 0.99.

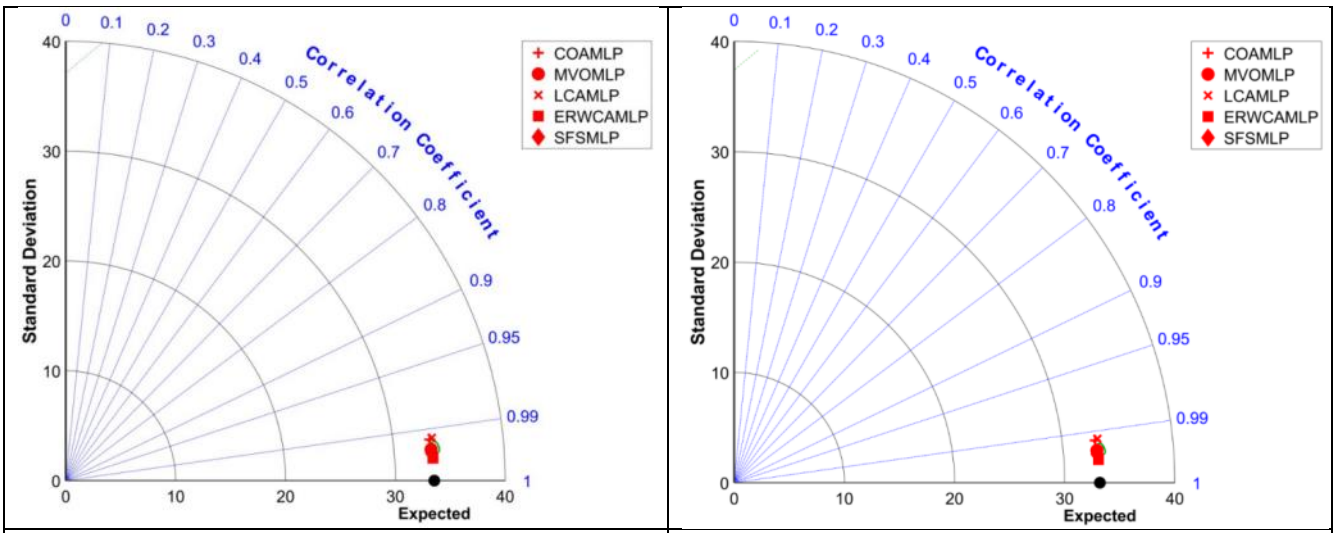


Figure 16. The Taylor diagram shows the energy consumption of the industrial metropolis.

4.5 Discussion

The pursuit of energy efficiency in steel production demands cutting-edge techniques to optimize Multi-Layer Perceptron (MLP) structures effectively. Our study delves into five distinct optimization methods, each contributing uniquely to the challenge. COA-MLP exhibited superior training and testing results with a swarm size of 200. This configuration achieved a balance between accuracy and computational efficiency, making it an optimal choice for practitioners aiming for competitive results without sacrificing computational resources. MVO-MLP emerged as a robust performer, showcasing outstanding training and testing accuracy with a swarm size of 250. This method is particularly suitable for scenarios where achieving high precision in energy efficiency predictions is a top priority. LCA-MLP, while demonstrating competitive results, achieved a balanced performance with a swarm size of 200. This method could be preferred in situations where a trade-off between accuracy and computational resources is necessary. ERWCA-MLP outshone others, securing the top rank. Its ability to navigate the solution space effectively and achieve high accuracy in both training and testing phases positions it as a top contender for applications demanding uncompromised precision. SFS-MLP demonstrated exceptional efficiency when configured with a larger swarm size (400). This suggests that, when computational resources allow, incorporating the

sequential forward selection method can significantly boost the overall performance of MLP models. The selection of an optimization method depends on the specific goals and constraints of a given project. COA-MLP and MVO-MLP offer well-balanced solutions, while ERWCA-MLP and SFS-MLP shine in scenarios prioritizing accuracy. Practitioners should carefully consider the computational resources available. While ERWCA-MLP and SFS-MLP excel in accuracy, the computational demands may vary. Project-specific constraints should guide the selection process. Our study underscores the importance of a holistic approach to optimization. Considering the trade-offs between computational efficiency and accuracy ensures that the chosen MLP structure aligns with the broader goals of enhancing energy efficiency in steel production.

5. Conclusions

This study evaluated the performance of five metaheuristic-MLP models for predicting energy-related outcomes using a range of population sizes. Among them, ERWCA-MLP achieved the best results with the lowest RMSE and highest R^2, followed closely by SFS-MLP and MVO-MLP. COA-MLP and LCA-MLP also showed reasonable predictive accuracy but ranked lower. The results highlight the importance of algorithm selection and tuning (e.g., population size) in optimizing hybrid neural models. These findings provide actionable insights for developing accurate, efficient predictive tools in energy-intensive industries.

References

[1] Divina, Federico, Miguel García Torres, Francisco A. Goméz Vela, José Luis Vázquez Noguera, A Comparative Study of Time Series Forecasting Methods for Short Term Electric Energy Consumption Prediction in Smart Buildings, Energies, 12 (2019) 1934,
[2] Agency, International Energy, World energy outlook, OECD/IEA Paris, 2009.
[3] Ledu, M, H Damon Matthews, R de Elía, Regional estimates of the transient climate response to cumulative CO2 emissions. Nat, Clim. Chang, 6 (2016) 474-478,
[4] Nawaf, Ali, Assad Mamdouh El Haj, Fard Habib Forootan, Jourdehi Babak Amani, Mahariq Ibrahim, A. AlShabi Mohammad, CO2 emission modeling of countries in Southeast of Europe by using artificial neural network, Proc.SPIE, 2022, pp. 121200F.
[5] Directive, Water Framework, Available online: https://eur-lex. europa. eu/legal-content/EN, TXT, 2000.
[6] Efficiency, Energy, Buildings The global exchange for energy efficiency policies, data and analysis: https://www. iea. org/topics/energyefficiency/buildings, Accessed, 2019.
[7] Sari, Mustika, Mohammed Ali Berawi, Teuku Yuri Zagloel, Nunik Madyaningarum, Perdana Miraj, Ardiansyah Ramadhan Pranoto, Bambang Susantono, Roy Woodhead, MACHINE LEARNING-BASED ENERGY USE PREDICTION FOR THE SMART BUILDING ENERGY MANAGEMENT SYSTEM, Journal of Information Technology in Construction, 28 (2023),
[8] Xu, Xiaodong, Wei Wang, Tianzhen Hong, Jiayu Chen, Incorporating machine learning with building network analysis to predict multi-building energy use, Energy and Buildings, 186 (2019) 80-97, https://doi.org/10.1016/j.enbuild.2019.01.002.
[9] Hong, Goopyo, Namchul Seong, Optimization of the ANN Model for Energy Consumption Prediction of Direct-Fired Absorption Chillers for a Short-Term, Buildings, 13 (2023) 2526,
[10] Hossain, Jahangir, Aida. F. A. Kadir, Ainain. N. Hanafi, Hussain Shareef, Tamer Khatib, Kyairul. A. Baharin, Mohamad. F. Sulaima, A Review on Optimal Energy Management in Commercial Buildings, Energies, 16 (2023) 1609,
[11] Kim, Jee-Heon, Nam-Chul Seong, Won-Chang Choi, Comparative Evaluation of Predicting Energy Consumption of Absorption Heat Pump with Multilayer Shallow Neural Network Training Algorithms, Buildings, 12 (2022) 13,
[12] Mounter, William, Chris Ogwumike, Huda Dawood, Nashwan Dawood, Machine Learning and Data Segmentation for Building Energy Use Prediction—A Comparative Study, Energies, 14 (2021) 5947,
[13] V E, Sathishkumar, Changsun Shin, Yongyun Cho, Efficient energy consumption prediction model for a data analytic-enabled industry building in a smart city, Building Research & Information, 49 (2021) 127-143, 10.1080/09613218.2020.1809983.
[14] Chahbi, I., N. Ben Rabah, I. Ben Tekaya, Towards an efficient and interpretable Machine Learning approach for Energy Prediction in Industrial Buildings: A case study in the Steel Industry, 2022 IEEE/ACS 19th International Conference on Computer Systems and Applications (AICCSA), 2022, pp. 1-8.
[15] Ye, Zhongnan, Kuangly Cheng, Shu-Chien Hsu, Hsi-Hsien Wei, Clara Man Cheung, Identifying critical building-oriented features in city-block-level building energy consumption: A data-driven machine learning approach, Applied Energy, 301 (2021) 117453, https://doi.org/10.1016/j.apenergy.2021.117453.
[16] Sathishkumar, VE, Myeongbae Lee, Jonghyun Lim, Yubin Kim, Changsun Shin, Jangwoo Park, Yongyun

Cho, An energy consumption prediction model for smart factory using data mining algorithms, KIPS Transactions on Software and Data Engineering, 9 (2020) 153-160,
[17] Mihaylov, Mihail, Roxana Rădulescu, Iván Razo-Zapata, Sergio Jurado, Leticia Arco, Narcís Avellana, Ann Nowé, Comparing stakeholder incentives across state-of-the-art renewable support mechanisms, Renewable Energy, 131 (2019) 689-699, https://doi.org/10.1016/j.renene.2018.07.069.
[18] Hagan, Martin T, Howard B Demuth, Orlando De Jesús, An introduction to the use of neural networks in control systems, International Journal of Robust and Nonlinear Control: IFAC ‐ Affiliated Journal, 12 (2002) 959-985, https://doi.org/10.1002/rnc.727.
[19] McCulloch, Warren S., Walter Pitts, A logical calculus of the ideas immanent in nervous activity, The bulletin of mathematical biophysics, 5 (1943) 115-133, https://doi.org/10.1007/BF02478259.
[20] Hydrology, A.T.C.o.A.o.A.N.N.i., Artificial Neural Networks in Hydrology. II: Hydrologic Applications, Journal of Hydrologic Engineering, 5 (2000) 124-137, 10.1061/(ASCE)1084-0699(2000)5:2(124).
[21] Gao, Wei, Hualong Wu, Muhammad Kamran Siddiqui, Abdul Qudair Baig, Study of biological networks using graph theory, Saudi Journal of Biological Sciences, 25 (2018) 1212-1219, https://doi.org/10.1016/j.sjbs.2017.11.022.
[22] Fan, Xinghua, Li Wang, Shasha Li, Predicting chaotic coal prices using a multi-layer perceptron network model, Resources Policy, 50 (2016) 86-92, https://doi.org/10.1016/j.resourpol.2016.08.009.
[23] Zarei, Taleb, Reza Behyad, Predicting the water production of a solar seawater greenhouse desalination unit using multi-layer perceptron model, Solar Energy, 177 (2019) 595-603, https://doi.org/10.1016/j.solener.2018.11.059.
[24] Heidari, Mahsa, Hossein Shamsi, Analog programmable neuron and case study on VLSI implementation of Multi-Layer Perceptron (MLP), Microelectronics Journal, 84 (2019) 36-47, https://doi.org/10.1016/j.mejo.2018.12.007.
[25] Rajabioun, Ramin, Cuckoo optimization algorithm, Applied soft computing, 11 (2011) 5508-5518,
[26] Bulatović, Radovan R, Stevan R Đorđević, Vladimir S Đorđević, Cuckoo search algorithm: a metaheuristic approach to solving the problem of optimum synthesis of a six-bar double dwell linkage, Mechanism and Machine Theory, 61 (2013) 1-13,
[27] Mirjalili, Seyedali, Seyed Mohammad Mirjalili, Abdolreza Hatamlou, Multi-verse optimizer: a nature-inspired algorithm for global optimization, Neural Computing and Applications, 27 (2016) 495-513,
[28] Zhao, Huiru, Xiaoyu Han, Sen Guo, DGM (1, 1) model optimized by MVO (multi-verse optimizer) for annual peak load forecasting, Neural Computing and Applications, 30 (2018) 1811-1825,
[29] Kashan, Ali Husseinzadeh, League Championship Algorithm (LCA): An algorithm for global optimization inspired by sport championships, Applied Soft Computing, 16 (2014) 171-200,
[30] Sadollah, Ali, Hadi Eskandar, Ardeshir Bahreininejad, Joong Hoon Kim, Water cycle algorithm with evaporation rate for solving constrained and unconstrained optimization problems, Applied Soft Computing, 30 (2015) 58-71,
[31] Wedyan, Ahmad, Jacqueline Whalley, Ajit Narayanan, Hydrological Cycle Algorithm for Continuous Optimization Problems, Journal of Optimization, 2017 (2017) 1-25, 10.1155/2017/3828420.
[32] Salimi, Hamid, Stochastic fractal search: a powerful metaheuristic algorithm, Knowledge-Based Systems, 75 (2015) 1-18,
[33] Team, R Developement Core, R: A language and environment for statistical computing, (No Title), (2010),
[34] Candanedo, Luis M., Véronique Feldheim, Dominique Deramaix, Data driven prediction models of energy use of appliances in a low-energy house, Energy and Buildings, 140 (2017) 81-97, https://doi.org/10.1016/j.enbuild.2017.01.083.
[35] Taylor, Karl E, Summarizing multiple aspects of model performance in a single diagram, Journal of geophysical research: atmospheres, 106 (2001) 7183-7192, https://doi.org/10.1029/2000JD900719.
[36] Change, IPCC Climate, The Physical Science Ba-sis. Contribution of Working Group 1 to the Fourth Assessment Report ofthe Intergovernmental Panel on Climate Change, Cambridge. UK, 2007.