



## Short-Term EV Charging Demand Forecast with Feedforward Artificial Neural Network

*Daniel Kwegyir, Francis Boafo Effah, Daniel Opoku, Peter Asigri, and Emmanuel Asuming Frimpong*

*Department of Electrical and Electronic Engineering, Kwame Nkrumah University of Science and Technology, Kumasi, Ghana.*

### ARTICLE INFORMATION

Received: May 03, 2023  
 Revised: July 14, 2023  
 Accepted: July 14, 2023  
 Available online: July 31, 2023

### KEYWORDS

electric vehicle, forecasting, artificial neural network, spider monkey optimization

### CORRESPONDENCE

Phone: +233(0) 24 121 6264  
 E-mail: [fbeffah.coe@knust.edu.gh](mailto:fbeffah.coe@knust.edu.gh)

### A B S T R A C T

The global increase in greenhouse gas emissions from automobiles has brought about the manufacture and usage of large quantities of electric vehicles (EVs). However, to ensure proper integration of EVs into the grid, there is a need to forecast the charging demand of EVs accurately. This paper presents a short-term electric vehicle charging demand forecast using a feedforward artificial neural network optimized with a modified local leader phase spider monkey optimization (MLLP-SMO) algorithm, a proposed variant of spider monkey optimization. A proportionate fitness selection is employed to improve the update process of the local leader phase of the spider monkey optimization. The proposed algorithm trains a feedforward neural network to forecast electric vehicle charging demand. The effectiveness of the proposed forecasting model was tested and validated with electric vehicle public charging data from the United Kingdom Power Networks Low Carbon London Project. The model's performance was compared to a feedforward neural network trained with particle swarm optimization, genetic algorithm, classical spider monkey optimization, and two conventional forecasting models, multi-linear regression and Monte Carlo simulation. The performance of the proposed forecasting model was assessed using the mean absolute percentage error of forecast and forecasting accuracy. The model produced a forecast accuracy and mean absolute percentage error of 99.88% and 3.384%, respectively. The results show that MLLP-SMO as a trainer predicted better than the other forecasting models and met industry standard forecast accuracy.

### INTRODUCTION

Greenhouse gas emissions into the atmosphere have recently become a significant global concern. Emissions from automobiles have been identified to constitute a considerable amount of total greenhouse emissions into the atmosphere. It accounts for almost a quarter with an exact value of 23% of global energy-related emissions [1]. In the United States (U.S.), greenhouse gas (GHG) emissions from automobiles account for about 29% of the total U.S. greenhouse gas emissions, making it the most significant contributor to U.S. GHG emissions [2]. Again, in Ghana, automobile emissions account for about 5% of GHG emissions [3]. To address this issue, there has been a shift in recent times towards the production and usage of electric vehicles, electric trains, etc., due to their zero or minimal emissions of greenhouse gases.

The usage of electric vehicles (EVs) has grown significantly over the years due to the implementation of policies and the provision of incentives by governments worldwide. The market share of EVs is expected to reach 10.8 million units by 2026 [4]. This will

increase the demand on the grid significantly and present other power quality issues, such as voltage imbalance, harmonic injections, voltage drops, etc., to the distribution [5], [6]. In order to accommodate the increase in demand from EV charging on the grid and ensure the effective operation of the power system in terms of cost and quality of power supply, there is a need to forecast EV charging demand. Demand forecasting is critical to reliable and secure power system operations since it helps to match generation to load. Forecasting of electricity demand is done for short-term, mid-term, and long-term periods. Short-term demand forecasting predicts demand for an hour to a week. It is needed for unit commitment and optimization roles for the grid. Its accuracy helps to maintain the system stability and minimization of energy loss [7].

Electric vehicle charging demand, unlike conventional load, is different. It is influenced by driving and travel patterns, charging time, traffic flow, etc. This stochastic nature of the EV charging demand factor forces the use of advanced forecasting techniques that can decipher major patterns in the usage behaviour of EV users with high accuracy to forecast EV charging demand. Demand forecasting accuracy is needed to ensure full utilization of network capability, reduce cost, and ensure system robustness.

Conventionally, EV demand has been forecasted using moving average (MA), exponential smoothing (ES), linear regression (LR), logistic regression, autoregressive integrated moving average (ARIMA), and Monte Carlo simulations. These methods produce inaccurate forecasts due to their inability to capture the highly stochastic EV user behaviour and forecast charging demand. Furthermore, most of these methods are linear and cannot incorporate multivariate EV user behaviour to forecast EV charging demand.

Machine learning methods such as artificial neural networks, ensemble learning methods, support vector machines, and clustering methods are currently employed to forecast EV charging demand. However, these methods are also prone to forecasting errors due to overfitting neural networks, aggregated error in ensemble learning, and difficulty handling higher dimensional non-numerical attributes in clustering algorithms. It is, therefore, necessary to employ more accurate and optimized models and tools to forecast EV demand and user behaviour to reduce the costs associated with forecasting errors.

Several methods have been proposed to forecast EV demand in the literature. The authors in [8] employed hierarchical clustering and linear regression methods to forecast the EV demand for charging stations in different districts. The method in [9] forecasted the day-ahead household EV demand using two-layer hybrid stacking ensemble learning, combining different machine learning algorithms. [10] proposed a mathematical framework of an equivalent time-variant storage model for EV demand aggregation without assuming expected future driving patterns and exact departure and arrival times for forecasting EV demand using a simple ARIMA model. A short-term linear prediction technique is proposed in [7] to forecast EV demand for light and heavy-duty electric vehicles to improve the efficiency of the EVs in terms of energy usage. The method in [11] explored the efficiency of machine learning techniques such as artificial neural networks (ANN) and support vector machines to forecast EV future demand. The authors in [12] proposed an artificial intelligence-based forecast using neural networks to forecast the daily load profile of individual loads and a fleet of randomly plugged-in PEVs in addition to upstream transformer loading.

The studies above failed to utilize optimized machine learning models to forecast EV aggregated demand, resulting in lower forecasting accuracies and higher errors. Again, they failed to capture user behaviour from different charging locations, presenting another high level of EV charging demand variability to perform forecasts. This study, therefore, presents a short-term forecast of EV charging demand considering EV user behaviour at different charging stations using a feedforward neural network optimized with the proposed modified local leader phase spider monkey optimization (MLLP-SMO). The subsequent sections of this paper provide a detailed description of the proposed approach, results and discussion, and conclusion.

## METHOD

A greedy approach is applied to update a spider monkey's ( $SM_{ij}$ ) position during the local leader phase of the SMO algorithm. In this case, the monkey's fitness  $SM_{ij}$  at a new position is only

accepted if it is greater than the fitness in the current position. This approach, however, has a disadvantage in that  $SM_{ij}$  with low fitness located near the global solution may be overlooked and not given the opportunity to update. As a result, the algorithm may move in a non-optimal direction, missing the true solution. From (1), the position of a  $SM_{ij}$  is heavily influenced by  $\beta$  and the impact of a random spider ( $SM_{rj}$ ).  $\beta$  is a uniformly distributed random number within the range of (-1, 1), whereas  $\alpha$  is a random number generated within the range (0, 1).

$$SM_{newij} = SM_{ij} + \alpha \times (LL_{kj} - SM_{ij}) + \beta \times (SM_{rj} - SM_{ij}) \quad (1)$$

The SMO algorithm's local leader phase updates  $SM_{ij}$  positions using a greedy approach, which may not provide equal opportunities to all  $SM_{ijs}$ . Consequently, some  $SM_{ijs}$  with high fitness or those  $SM_{ijs}$  chosen by  $\alpha > pr$  may fail to reach the global optima in a given iteration. As a result, the algorithm may produce sub-optimal results. The perturbation rate  $pr$  is typically between 0.1 and 0.8.

To improve the SMO algorithm, every  $SM_{ijs}$  chosen by  $\alpha > pr$  for updating is given the opportunity to move to a better position in the local leader phase based on the fitness of their old position. Using fitness proportionate selection and the total number of  $SM_{ijs}$  in the search space ( $Y$ ), the number of opportunities given to each  $SM_{ij}$  to update is determined. The number of opportunities given to each  $SM_{ij}$  for the next iteration is defined using (2).

$$No. \text{ of chances of } SM_{ij} = \frac{fit(SM_{ij(OLD)})}{\sum_i^N fit(SM_{ij(OLD)})} \times Y \quad (2)$$

Where  $fit(SM_{ij(OLD)})$  is the fitness of  $SM_{ij}$  in its old position, and  $N$  is the number of  $SM_{ijs}$ . The proposed MLLP algorithm improves the SMO by providing multiple chances for each  $SM_{ij}$  chosen by the local leader to update its current position based on its fitness. The number of chances for each  $SM_{ij}$  is determined by (2), which considers the total number of  $SM_{ijs}$  in ( $Y$ ) and the fitness of the  $SM_{ij}$ 's previous position. If an update does not improve fitness after a specified number of chances, the  $SM_{ij}$  remains in its original position. The following is the pseudocode for the position update in the MLLP algorithm:

```

1  for each member  $SM_{ij} \in k^{th}$  group
2    for each  $j \in \{1, 2, \dots, D\}$  do
3      if  $\alpha(0,1) > pr$  then
4         $f_{ij} = SM_{ij} + \alpha(0,1) \times (LL_{kj} - SM_{ij})$ 
5         $g_{ij} = (SM_{rj} - SM_{ij})$ 
6        while (chances to update has not elapsed), do
7           $SM_{chance(ij)} = f_{ij} + \beta(-1,1) \times g_{ij}$ 
8          if  $fit(SM_{chance(ij)}) > fit(SM_{ij})$ 
9             $SM_{new(ij)} = SM_{chance(ij)}$ 
10         break
11        else if (chances elapsed)
12           $SM_{new(ij)} = SM_{ij}$ 
13        break
14    end if

```

15            **end while**  
 16            **end if**  
 17        **end for**  
 18        **end for**

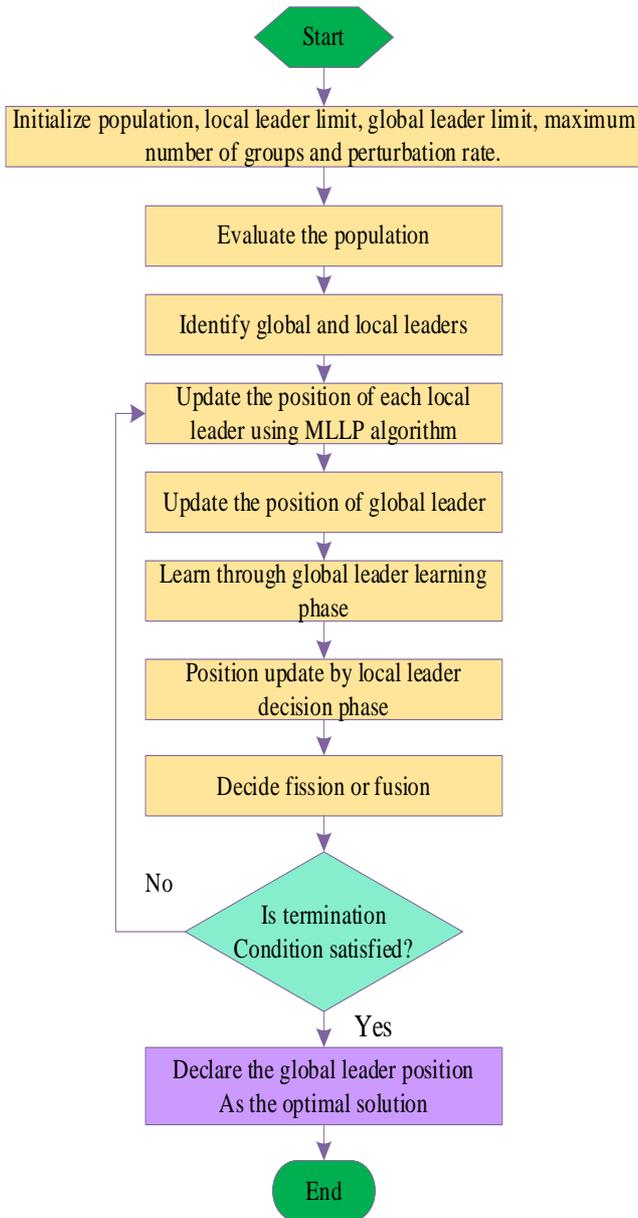


Figure 1. Implementation of the MLLP-SMO Algorithm

To update the position of a spider monkey,  $SM_{ij}$ , using the MLLP algorithm, a random number  $\alpha$  is generated between 0 and 1. If  $\alpha$  is greater than the perturbation rate ( $pr$ ), the  $SM_{ij}$  is chosen to update its position in the  $J^{th}$  dimension. The effect of the local leader's position ( $LL_{kj}$ ) on the  $SM_{ij}$  is checked using the parameter ( $f_{ij}$ ). The impact of a random spider ( $SM_{rj}$ ) on  $SM_{ij}$  is also checked using  $g_{ij}$ . The  $SM_{ij}$  is then given chances to update its position in the search space, with the number of chances defined by (2). Each position generated under the chances given ( $SM_{chances(ij)}$ ) is checked for fitness and compared to the fitness of the spider monkey's ( $SM_{ij}$ ) old position. If a better position is found, the spider monkey updates its position to the new position,  $SM_{new(ij)}$ . During the position update of the spider monkeys in the search space using MLLP

algorithm, the update is completed regardless of whether the maximum number of chances given to each spider monkey is reached. If a spider monkey cannot find a better position after the maximum number of chances, its position remains unchanged. The update of each spider monkey in the search space is influenced by the local leader's position in the current dimension and the influence of a random spider in another dimension. The number of chances given to each spider monkey is balanced using (2) to maintain a stochastic nature in the local leader phase. The implementation of the MLLP-SMO algorithm is depicted in Figure 1.

### Proposed MLLP-SMO Trainer

This work employs an MLLP-SMO algorithm to optimally select weights and biases for a feed-forward neural network to forecast electric vehicle demand. MLLP-SMO is a modified version of spider monkey optimization (SMO) with improved exploration and exploitation ability to avoid local optima entrapment problem of stochastic optimization algorithm and also with improved convergence. The MLLP-SMO has been benchmarked on a machine learning dataset and verified against algorithms such as genetic algorithm (GA), grey wolf optimizer (GWO), particle swarm optimization (PSO) and spider monkey optimization (SMO) to train a feed-forward multi-layer perceptron in [13], [14]. MLLP-SMO performed better than all these algorithms regarding mean square error (MSE) and classification accuracy. Hence it is used as a trainer for feedforward neural networks to forecast electric vehicle demand.

Training of neural networks deals with continuously mapping input datasets to the output datasets to find the optimal set of weights and biases within a minimum number of iterations. The training process aims to increase classification accuracy by reducing classification errors. The performance of ANN is purely based on the synaptic weights [15]. Generally, in classification problems, error functions such as MSE, the sum of squared error (SSE), root mean square error (RMSE), etc., are used to evaluate the performance of the training process. In this work, the mean square error function is implemented. The objective of training multi-layer perceptron (MLP) with MLLP-SMO is to minimize the MSE of each training iteration defined according to (3).

$$\text{(Minimize)} \quad MSE(W, O) = \sum_{j=1}^n \frac{\sum_{i=1}^m (A_i^j - F_i^j)^2}{n} \quad (3)$$

where  $n$  is the total number of training samples,  $m$  is the output sample,  $A_i^j$  is the actual output of the  $i$  input data point from the  $j$  training sample and  $F_i^j$  is the desired output of the  $i$  input data point from the  $j$  training sample. The weights ( $w$ ) and biases ( $O$ ) supplied to the MLLP-SMO as variables to be optimized. Figure 2 shows the architecture of the proposed MLLP-SMO-FNN Trainer.

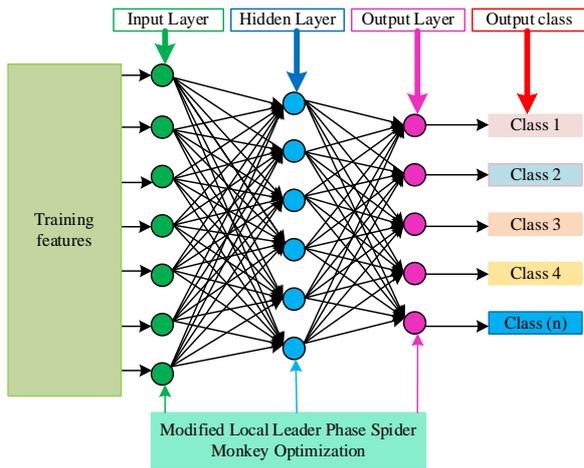


Figure 2. Proposed MLLP-FNN Trainer

From Figure 2, the MLLP-SMO algorithm supplies weights and biases to the MLLP and receives the errors. The MLLP-SMO iteratively alters the weights and biases to reach optimal values to perform the classification test. The output of the MLP using the optimal weights and bias values is used to calculate the MSE value. The processes involved are summarized in Figure 3.

**Proposed historical feature extraction algorithm for different charging stations**

Forecasting EV charging demand requires well-structured historical data with relevant features such as EV state of charge (SOC), distance travelled, charging start and end time, charging duration, charging power, etc. Due to variability in the charging behaviour of EV users in different charging regions, charging data from different charging stations are also different in terms of EV user behaviour. Forecasting EV charging demand for a period using historical data requires efficient extraction and combination of historical charging demand data from different charging stations. Figure 4 shows the proposed algorithm for extracting and aggregating EV charging data from different charging stations for forecasting.

In Figure 4, the parameters  $D$ ,  $N$ , and  $M$  represent the number of days in the year, the total number of charging stations and the total number of electric vehicles arriving at the public charging station. The aggregated demand for an hour and a day are determined with (4) and (5).

$$P_{ev}(hour) = \sum_{hour=1}^{24} \sum_{i=1}^M P_{ev}(i) \tag{4}$$

$$P_{ev}(daily) = \sum_{day=1}^D \sum_{hour=1}^{24} P_{ev}(hour) \tag{5}$$

where  $i$  represents electric vehicles arriving at a charging station in a day,  $M$  is the total number of EVs arriving at each hour, and  $P_{ev}$  is the charged power of the  $i$ th EV.

The initial state of charge (SOC), the time required for a full charge, the duration of charge, and the initial distance travelled before the charge are determined with (6)-(9) [12]. The algorithm provides an EV charging demand database of EV user behaviour where attributes can be extracted for forecasting.

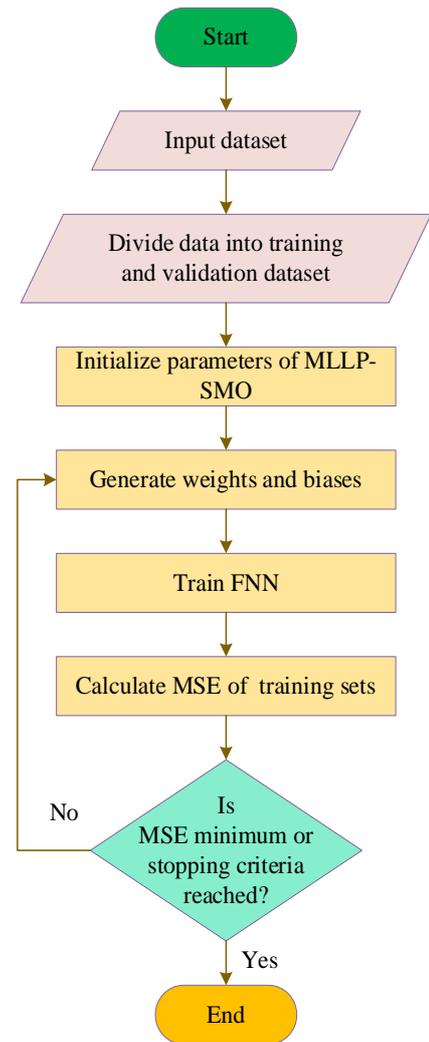


Figure 3. Implementation of MLP Training with MLLP-SMO

$$SOC_{initial} = 100 \left( 1 - \frac{D_{hr}}{\gamma_{ev} \times C_{bat}} \right) \tag{6}$$

$$T_{charge} = 100(1 - SOC) \times \frac{C_{bat}}{P_r \times \gamma_{ev}} \tag{7}$$

$$T_{duration} = T_{end} - T_{start} \tag{8}$$

$$D_{initial} = \frac{D_r(P_r - P_c)}{P_r} \tag{9}$$

where  $D_{hr}$  is the hourly/daily travelled distance in km,  $\gamma_{ev}$  is the driving efficiency of EV type,  $C_{bat}$  is the battery capacity of EV,  $P_r$  is the rated power of EV type,  $\gamma_{ev}$  is EV type efficiency,  $T_{end}$  is charging end time,  $T_{start}$  is charging start time,  $D_r$  is EV model rated distance, and  $P_c$  is the charged power.

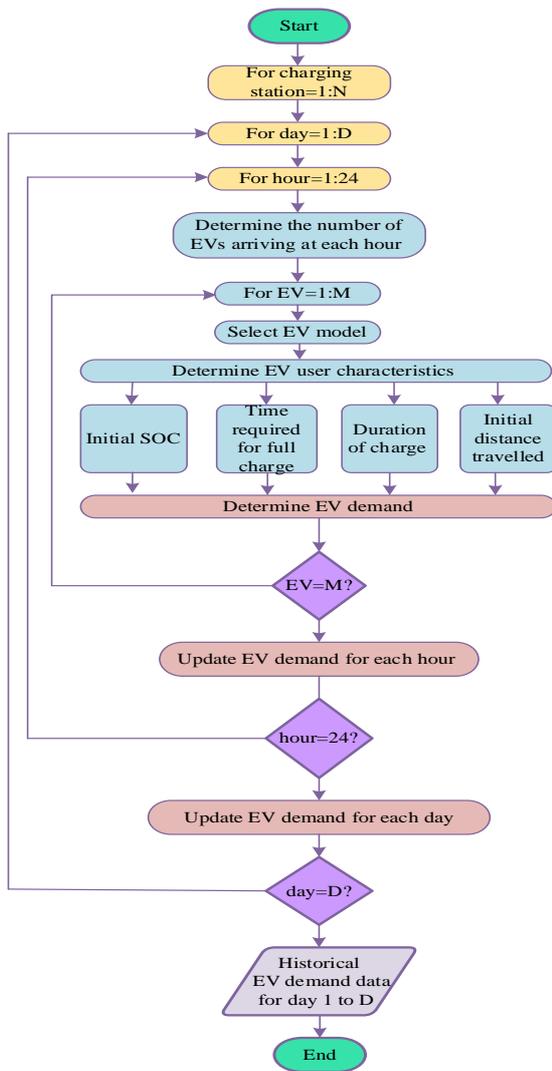


Figure 4. EV Demand Aggregator Algorithm

Details of the processes involved in the extraction are outlined below.

- Step 1:** Initialize the charging stations with the history of their charging events.
- Step 2:** For each day within a year, determine the number of electric vehicles (EVs) arriving at each hour at all charging stations.
- Step 3:** Select the vehicle model and its ratings for each of the EVs arriving at the charging station. The EV models used in this work are shown in Table 1.
- Step 4:** For the selected EV model, determine the initial SOC and initial distance travelled before arrival at the charging station, the time required for a full charge, and the duration of charge. In this work, these are calculated using (6), (9), (7) and (8), respectively.
- Step 5:** Using the results in step 4, determine the charging demand of EVs arriving at the charging station for an hour, day and month of the year. Again, hourly and daily demand is aggregated in this work with (4) and (5).
- Step 4:** Update the historical database with the results from Step 5. Update the EV charging demand for each hour, day, and month of the year.

## Proposed forecasting framework with MLLP-MSO

The proposed short-term EV demand forecast model is outlined in the steps below;

- Step 1:** Data mining is performed to extract knowledge from the EV feature extraction database. Mining processes such as feature extraction, data normalization and filtering are performed. This reveals hidden characteristics such as data patterns, associations, variations, etc. Features such as State of Charge (SOC), distance travelled, charging duration, day of charge, month of charge, etc., are extracted.
- Step 2:** The extracted data is divided into training, validation and testing (forecasting) data. The data is afterwards prepared in neural network training format.
- Step 3:** The training, validation and testing data are normalized. This helps the neural network to learn the relationships between the input features rather than the magnitudes of data points. In this work, min-max normalization in (10) is employed, where each data point is scaled according to the maximum and minimum data points in the datasets. In (10),  $\min(X)$  and  $\max(X)$  are minimum and maximum data points in the dataset  $X_i$  with  $x(i)$  being the datapoint to be normalized.

$$x' = \frac{x(i) - \min(X)}{\max(X) - \min(X)} \quad (10)$$

- Step 4:** Training and validation data are supplied to the feedforward neural network for the training process to begin.
- Step 5:** Parameters, i.e., population size (i.e., number of spider monkeys), local leader limit, global leader limit, maximum group, and perturbation rate of MLLP-SMO, are initialized.
- Step 6:** The neural network is trained by MLLP-SMO by supplying weights and biases on training datasets to minimize the mean square error (MSE) between each training and target data. Finally, the optimized neural network model with optimal weights and biases is outputted when stopping criteria or minimum MSE is reached to perform a forecast for the future.
- Step 7:** Testing data is supplied to the optimized model to perform a forecast for a preferred period. The mean absolute percentage error (MAPE) between the forecasted EV charging demand and actual demand is determined.

In this work, the results of the forecasts are compared to three other algorithms as trainers; particle swarm optimization (PSO), genetic algorithm (GA) and original spider monkey optimization (SMO) in terms of mean absolute percentage error (MAPE) and r-correlation between the actual and forecasted demand during testing in (11). The industry standard for MAPE is a maximum of 5% [16]. It is also compared to conventional forecasting methods such as multi-linear regression and Monte Carlo simulation.

$$MAPE = \frac{\sum_{i=1}^N \left| \frac{A_i - F_i}{A_i} \right|}{N} \quad (11)$$

where  $A_t$  is the actual demand,  $F_t$  is the forecasted demand, and  $N$  is the total number of forecasts made in the period.

### Case study datasets

The EV demand forecasting framework is tested and validated with public charging station data from the United Kingdom (UK) Power Networks Low Carbon London (LCL) Project [17]. This innovation project was set up to investigate the impact of a wide range of low-carbon technologies on London's electricity distribution network. The public trial charging datasets consist of four (4) charging stations with charging events data. The data has a charging history from 2012 to 2014. The data has the following attributes; charging event, EV user ID, charging start time, charging end time, charging start date, charging end date, charging energy and charging price. The four charging stations provided 25,440 charging events from 2012 to 2014.

### Data pre-processing and feature extraction

The feature extraction algorithm is applied to the case study data to build charging historical data for three years to perform the forecast. The initial state of charge (SOC) and distance travelled before arrival at the charging station, charging duration and daily charging demands from 2012 to 2014 are determined in addition to the existing features in the data. To do this, it is assumed that charging energies below 16.5 kWh are Chevrolet Volt EVs, energies above 16.5 kWh and less than 24 kWh are Nissan Leaf EVs and energies above 24 kWh are assumed to be Tesla Model S EVs. The processed data contained 2003 user IDs with the following attributes; the month of charging, week of the year, day of the month, day type (weekday or weekend), distance travelled before charge (km), initial State of Charge (SOC) before charge and charging energy. A sample generated initial SOC and initial distance before the charge, and Figure 5 and the aggregated charging demand for 2012-2014 are shown in Figure 6.

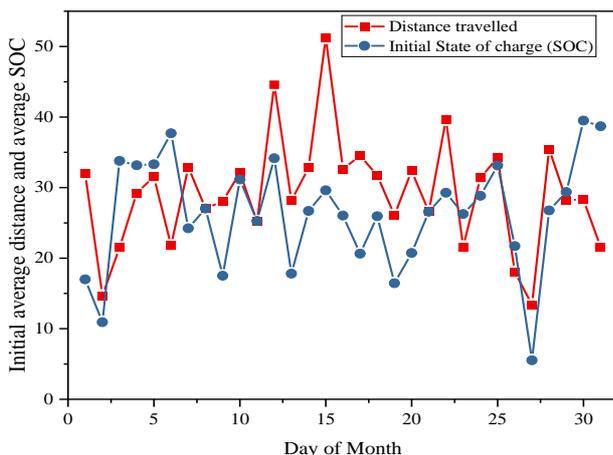


Figure 5. Generated Average Distance and Average Initial State of Charge (SOC) for January 2013

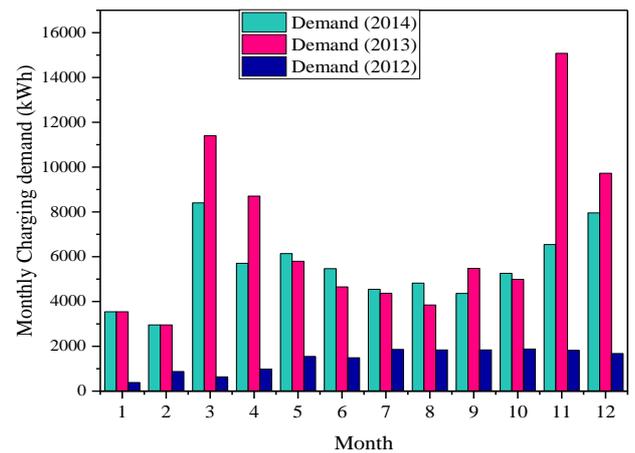


Figure 6: Aggregated EV Charging Demand

Once the historical data was built, the attributes were selected using the RankSearch algorithm in the WEKA [18] machine learning tool. This selects the best features which give a more accurate forecast. The following columns in the data were selected by the RankSearch algorithm to forecast the day-ahead charging demand: Charging month, week of the month, day of the week, day type (weekday or weekend), previous day average distance travelled, previous day average initial state of charge and previous day charging demand.

The input data is structured properly to train the MLLP-SMO FNN for forecasting with a meaningful relationship between inputs and outputs. The selected features are formatted as follows. Month: number of months (1-12)

- i. Week: number of weeks (1-7), starting Monday
- ii. Day: day of the month (1-D), where D is the total number of days in a particular month
- iii. Day type: (Weekday or weekend)
- iv. Previous day average distance travelled: Average distance travelled by all EVs on a previous day before the start of the charge.
- v. Previous day average initial state of charge (SOC): average SOC before charge of all EVs in a previous day.
- vi. Previous day demand: charging a particular month's demand for a previous day.
- vii. A day ahead demand: charging demand of the day to be forecasted.

An input matrix is further constructed for training the forecasting model. The input matrix to the forecasting model for January 2012, week one, is shown in Table 1. The data structure for the training, validation and testing is shown in Table 2. The training and validation data contained EV data from January 2012 to October 2014. The inputs nodes were 7 representing a month, week, day, day type, previous day average distance travelled, previous day average initial SOC and previous day charging demand. The output had a single node of the day ahead charging demand. For comparison, 10 hidden nodes were used for each trainer, i.e., PSO-FNN, SMO-FNN, GA-FNN and MLLP-SMO-FNN and ran for 10000 iterations. The forecasting models were simulated in MATLAB<sup>®</sup> software, and plots were done in Origin software. A daily forecast is then made for November 2014 to validate the forecasting model. The results obtained are compared

to Monte Carlo simulation and multi-regression conventional forecasting models.

Table 1. Input and Output Matrix for Week one of January 2012

Selected feature	Value			
Month	1	...	1	
Week	1	...	1	
Day	7	...	6	
Day type	2	...	2	
Previous day average distance travelled	0	$D_{\text{day}7}$	...	$D_{\text{day}5}$
Previous day initial state of charge	0	$\text{SOC}_{\text{day}7}$	...	$\text{SOC}_{\text{day}5}$
Previous day demand	0	$E_{\text{day}7}$	...	$E_{\text{day}5}$
Day ahead demand (output)	$E_{\text{day}7}$	$E_{\text{day}1}$	...	$E_{\text{day}6}$

Table 2. Data Structure for Training, Validation and Testing

Datasets	Percentage of total data (%)	Actual number of datapoints	Rows	Columns
Training	80	6083	869	7
Validation	15	1169	167	7
Forecasting	5	427	61	7

## RESULTS AND DISCUSSIONS

The results of the proposed forecasting model are presented in this section. The MLLP feedforward neural network forecast result is compared to other popular optimization algorithms: genetic algorithm (GA), particle swarm optimization (PSO) and the original spider monkey optimization (SMO) and two conventional methods: multi-linear regression and Monte Carlo simulation. The comparison is done in terms of mean absolute percentage error (MAPE), forecasting accuracy and r-correlation between actual and forecasted EV charging demand. Descriptive statistics (average forecast, maximum, and minimum forecasts) for each month are also determined and compared to that of the actual demand to check the closeness of forecasted demand to actual demand for each optimization algorithm. It should be noted that the same training and testing data were supplied to each forecasting model.

## Forecast performance of optimization algorithms for November 2014

Table 3 presents the results of various metaheuristic models for the November 2014 forecast, evaluated using three performance metrics: Mean Absolute Percentage Error (MAPE), Accuracy, and r-Correlation.

Table 3: Performance for Metaheuristic Models for November 2014 Forecast

Forecasting model	Performance metric		
	MAPE (%)	Accuracy (%)	r-Correlation (%)
MLLP-SMO-ANN	3.384	96.66	99.88
GA-ANN	4.3741	93.33	80.75
PSO ANN	6.2069	93.33	71.02
SMO-ANN	4.0145	96.66	99.52

The first model listed in the table is the proposed MLLP-SMO-ANN. It achieved a MAPE of 3.384%, indicating that, on average, its forecasts had an error of approximately 3.384% relative to the actual values. The accuracy score for the proposed method is high at 96.66%, indicating that it accurately predicted the outcomes in 96.66% of cases. Moreover, the r-Correlation score of 99.88% indicates a strong positive correlation between predicted and actual values. These results demonstrate the effectiveness of the proposed MLLP-SMO-ANN model in forecasting the November 2014 data. These results are similar to those obtained by works done in [7] and [9]. However, the MAPE, accuracy and r-Correlation obtained by the proposed MLLP-SMO-ANN showed a superiority.

Also, the GA-ANN model yielded a slightly higher MAPE of 4.3741% compared to MLLP-SMO-ANN. Although its MAPE is slightly higher, it is still within an acceptable range for forecasting accuracy. The accuracy score of 93.33% indicates that the model's predictions were accurate in 93.33% of cases. However, the r-Correlation score of 80.75% indicates a weaker correlation between predicted and actual values than the MLLP-SMO-ANN model. This shows that the GA-ANN model was less successful in capturing the underlying patterns in the November 2014 data.

Similarly, the PSO-ANN model achieved a higher MAPE of 6.2069% than the previous two models. This higher MAPE indicates a larger average forecasting error. The accuracy score of 93.33% is the same as the GA-ANN model, suggesting a similar level of accuracy. However, the r-Correlation score of 71.02% indicates a weaker correlation, indicating that the PSO-ANN model did not capture the underlying relationships as effectively as the other models.

Finally, the SMO-ANN model achieved a MAPE of 4.0145%, an accuracy of 96.66%, and an r-Correlation of 99.52%. These results are similar to those of the MLLP-SMO-ANN model, indicating that the SMO-ANN model performed well in forecasting the November 2014 data.

### Descriptive statistics performance of optimization algorithms for November 2014

Table 4 presents the descriptive statistics for the November 2014 forecast, including the mean, maximum, and minimum forecasts for four different models: MLLP-SMO-ANN, GA-ANN, PSO-ANN, and SMO-ANN. It also includes the corresponding values for the actual demand.

Table 4: Descriptive Statistics for November 2014

Forecasting model	Mean forecast (kWh)	Maximum forecast (kWh)	Minimum forecast (kWh)
MLLP-SMO-ANN	421.79	532.23	363.14
GA-ANN	417.49	515.75	310.47
PSO-ANN	402.92	483.90	381.49
SMO-ANN	421.50	522.43	379.61
Actual demand	431.74	536.50	364.66

In the November 2014 forecast, the table shows that the PSO-ANN and GA-ANN models provided the least average forecasts of 402.92 kWh and 417.49 kWh, respectively. The MLLP-SMO-ANN model had an average forecast of only 9.95 kWh less than the actual average demand, while the SMO-ANN model's forecast was 10.24 kWh less. This indicates that the MLLP-SMO-ANN model's average forecast was closer to the actual average demand than the GA-ANN and PSO-ANN forecasts. Therefore, the MLLP-SMO-ANN model's average forecast was considered better regarding its proximity to the average demand.

Furthermore, when considering the maximum and minimum forecasts, the MLLP-SMO-ANN model outperformed the other models. The maximum forecast of MLLP-SMO-ANN was closer to the actual maximum demand, as indicated by its value of 532.23 kWh, compared to the values of other models. Similarly, the minimum forecast of MLLP-SMO-ANN (363.14 kWh) was closer to the actual minimum demand. These results support the notion that the MLLP-SMO-ANN model was a better forecasting model for the November 2014 dataset.

### Forecast performance of conventional methods

Table 5 provides each forecasting model's Mean Absolute Percentage Error (MAPE) and r-Correlation values.

Table 5: Performance for Conventional Models for November 2014 Forecast

Forecasting model	Performance metric	
	MAPE (%)	r-Correlation (%)
MLLP-SMO-ANN	3.384	99.88
Multi-linear Regression	24.99	18.02
Monte Carlo Simulation	44.04	19.49

The MLLP-SMO-ANN model achieved a remarkably low MAPE of 3.384%, indicating that, on average, its forecasts had an error of only 3.384% relative to the actual values. Additionally, the model attained a high r-Correlation score of 99.88%, indicating a strong positive correlation between the predicted and actual values. These results highlight the effectiveness of the MLLP-SMO-ANN model in accurately forecasting the November 2014 data and show the superiority of machine learning models over conventional forecasting models.

The multi-linear regression model yielded a significantly higher MAPE of 24.99% and a considerably lower r-Correlation of 18.02%. These results show that the multi-linear regression model had a higher average forecasting error and a weaker correlation with the actual values than the MLLP-SMO-ANN model. Consequently, the MLLP-SMO-ANN model outperformed the multi-linear regression model in terms of both accuracy and correlation.

Furthermore, the Monte Carlo Simulation model exhibited an even higher MAPE of 44.04% and a low r-Correlation of 19.49%. These outcomes indicate that the Monte Carlo Simulation model had the largest average forecasting error and the weakest correlation with the actual values among the three models discussed. Therefore, it can be concluded that the MLLP-SMO-ANN model significantly outperformed the Monte Carlo Simulation model regarding accuracy and correlation. The actual forecasts of the models are shown in Figure 7.

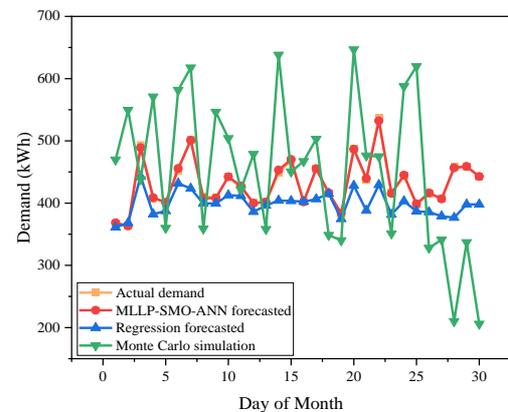


Figure 7: Charging Demand Forecast for November 2014

## CONCLUSION

This research focused on the short-term electric vehicle charging demand forecast, considering the global increase in greenhouse gas emissions and the need for accurate EV charging predictions for effective integration into the grid. The study introduced a feedforward artificial neural network trained with a modified local leader phase spider monkey optimization algorithm as a proposed variant of spider monkey optimization. The proposed MLLP-SMO model achieved an impressive average forecast accuracy of 99.88% and a low mean absolute percentage error of 3.384%. These results indicate the superior performance of the MLLP-SMO model in accurately predicting EV charging demand, surpassing the performance of the other forecasting models.

The findings demonstrate that the MLLP-SMO model outperformed the alternative optimization algorithms and met industry standards for forecast accuracy. This indicates the effectiveness and reliability of the proposed model for short-term EV charging demand forecasting. The accurate predictions provided by the MLLP-SMO model can contribute to the successful integration of EVs into the power grid, enabling efficient management of charging infrastructure and optimization of energy resources. Overall, this research contributes to the field of EV charging demand forecasting and highlights the importance of accurate predictions for the successful implementation of electric vehicles on a larger scale. The MLLP-SMO model stands out as a promising approach for accurately forecasting EV charging demand, paving the way for efficient and sustainable integration of EVs into the power grid.

## REFERENCES

- [1] D. Yu, M. P. Adhikari, A. Guiral, A. S. Fung, F. Mohammadi, and K. Raahemifar, "The Impact of Charging Battery Electric Vehicles on the Load Profile in the Presence of Renewable Energy," *2019 IEEE Can. Conf. Electr. Comput. Eng. CCECE 2019*, pp. 3–6, 2019, doi: 10.1109/CCECE.2019.8861730.
- [2] "Carbon Pollution from Transportation | Transportation, Air Pollution, and Climate Change | US EPA." <https://www.epa.gov/transportation-air-pollution-and-climate-change/carbon-pollution-transportation> (accessed Jun. 29, 2021).
- [3] USAID, "Greenhouse gas emissions in Guatemala," *USAID Factsheet*, no. December 2009, pp. 1–2, 2017, doi: 13 / 006.
- [4] A. Gautam, A. K. Verma, and M. Srivastava, "A Novel Algorithm for Scheduling of Electric Vehicle Using Adaptive Load Forecasting with Vehicle-to-Grid Integration," *2019 8th Int. Conf. Power Syst. Transit. Towar. Sustain. Smart Flex. Grids, ICPS 2019*, pp. 6–11, 2019, doi: 10.1109/ICPS48983.2019.9067702.
- [5] A. F. Botero and M. A. Rios, "Demand Forecasting Associated with Electric Vehicle Penetration on Distribution Systems," 2012.
- [6] R. M. G. D. Ranathunga and L. A. Samaliarachchi, "Impact of Electric Vehicle Loads on the System Load Profile of Sri Lanka Impact of Electric Vehicle Loads on the System Load Profile of Sri Lanka," no. December, 2017, doi: 10.4038/engineer.v50i4.7270.
- [7] M. M. Sangdehi, K. Lakshmi Varaha Iyer, K. Mukherjee, and N. C. Kar, "Short term power demand forecasting in light- and heavy-duty electric vehicles through linear prediction method," *2012 IEEE Transp. Electr. Conf. Expo, ITEC 2012*, no. 2, pp. 4–9, 2012, doi: 10.1109/ITEC.2012.6243480.
- [8] Q. Huang *et al.*, "Forecasting of the electric vehicles' charging amount of electricity based on curves clustering," *ICNC-FSKD 2017 - 13th Int. Conf. Nat. Comput. Fuzzy Syst. Knowl. Discov.*, pp. 2424–2428, 2018, doi: 10.1109/FSKD.2017.8393153.
- [9] S. Ai, A. Chakravorty, and C. Rong, "Household EV charging demand prediction using machine and ensemble learning," *Proc. - 2nd IEEE Int. Conf. Energy Internet, ICEI 2018*, pp. 163–168, 2018, doi: 10.1109/ICEI.2018.00037.
- [10] M. Pertl, F. Carducci, M. Tabone, M. Marinelli, S. Kiliccote, and E. C. Kara, "An Equivalent Time-Variant Storage Model to Harness EV Flexibility: Forecast and Aggregation," *IEEE Trans. Ind. Informatics*, vol. 15, no. 4, pp. 1899–1910, 2019, doi: 10.1109/TII.2018.2865433.
- [11] E. S. Xydas, C. E. Marmaras, L. M. Cipcigan, A. S. Hassan, and N. Jenkins, "Electric Vehicle Load Forecasting using Data Mining Methods," pp. 1–6.
- [12] D. Panahi, S. Deilami, M. A. S. Masoum, and S. M. Islam, "Forecasting plug-in electric vehicles load profile using artificial neural networks," *2015 Australas. Univ. Power Eng. Conf. Challenges Futur. Grids, AUPEC 2015*, pp. 1–6, 2015, doi: 10.1109/AUPEC.2015.7324879.
- [13] D. Kwegyir, E. A. Frimpong, and D. Opoku, "Optimization of Feedforward Neural Network Training using Modified Local Leader Phase Spider Monkey Optimization," no. July, pp. 2157–2167, 2021.
- [14] D. Kwegyir, E. A. Frimpong, and D. Opoku, "Modified Local Leader Phase Spider Monkey Optimization Algorithm," vol. 5, no. 2, pp. 1–18, 2021.
- [15] D. Devikanniga, K. Vetrivel, and N. Badrinath, "Review of meta-heuristic optimization based artificial neural networks and its applications," *J. Phys. Conf. Ser.*, vol. 1362, no. 1, 2019, doi: 10.1088/1742-6596/1362/1/012074.
- [16] E. S. Xydas, C. E. Marmaras, L. M. Cipcigan, A. S. Hassan, and N. Jenkins, "Forecasting Electric Vehicle charging demand using Support Vector Machines," *Proc. Univ. Power Eng. Conf.*, 2013, doi: 10.1109/UPEC.2013.6714942.
- [17] "Publisher: UK Power Networks - London Datastore." [https://data.london.gov.uk/publisher/uk-power-networks?res\\_format=XLS](https://data.london.gov.uk/publisher/uk-power-networks?res_format=XLS) (accessed Jul. 01, 2021).
- [18] E. Frank, M. A. Hall, and I. H. Witten, "The WEKA workbench," *Data Min.*, pp. 553–571, 2017, doi: 10.1016/b978-0-12-804291-5.00024-6.