Global Research Review in Business and Economics [GRRBE]

ISSN (Online) 2454-3217, ISSN (Print) 2395-4671
| Open-Access | Volume 8, Issue 02,
Pages 37-50 ||2022||

Artificial Intelligence Based Voltage Stability Assessment of Smart Micro-grid

[1]Achshah R M, Dr. [2]T. Ruban Deva Prakash,

[1]*(Artificial Intelligence and Data Science, SRM Easwari Engineering College, India)*

[2]*(Principal, Government College of Engineering Munnar, India)*

ABSTRACT

Due to limited fossil fuel resources and environmental contamination, solar and wind energy will inevitably penetrate the electric system. Natural resources such as wind and solar influence the planning and operation of the electric grid, introducing uncertainties to the network, thus the classic voltage stability analysis, which is a deterministic problem, may not work for micro-grid. The intermittency of these VESs and the difficulty of constraint handling for a real-time voltage stability assessment (RT-VSA) problem are the key crucial problems of this dynamic micro-grid with such variable energy sources (VES). Long calculations and several iterations are used in traditional a approach, which increases computation time. However, for an efficient smart micro-grid scheme for building voltage compensating devices that maintain constant voltage without disruptions, quick assessments based on load and renewable energy predictions are necessary. As a result, in this paper, an alternate method is proposed that takes much less time to compute and provides reliable results for real-time applications.

The goal of this project is to create a fast and accurate AI-based system for on-line voltage prediction so that compensators meant to preserve voltage stability can be activated. Three AI models based on decision tree, random forest, and KNN algorithms are built, and their performance in real-time voltage prediction is evaluated to determine which method is best for a typical micro-grid. The proposed artificial intelligence (AI) algorithms are learned utilizing training data provided by a python program written using the Newton-Raphson method's principles. Voltage compensators based on phasor measurement units are less effective and more expensive. The proposed AI models forecast the value of solar radiation, wind speed, real and reactive power load, and output the magnitude and angle of voltage to compensators in advance. A micro-grid system working in tandem with the central grid is modelled as a three-bus system, with one infinite generator bus representing the central grid that is treated as slack bus and the other two load buses, but with solar and wind integration that is modelled for real-time generation prediction. According to the findings, the decision tree technique might be used to analyze the on-line voltage stability of a practical micro-grid.

Keywords - Voltage Stability Assessment (VSA), Artificial Intelligence (AI), Random Forest, Decision Tree, K Nearest Neighbor (KNN), Newton-Raphson Load Flow (NRLF), Variable Energy Source (VES) integrated Micro-grid, Real Time Generation Prediction of VES

1. INTRODUCTION

Traditionally, central systems have met electrical demands. However, in recent years, tiny towns' energy demands have been handled locally by distributed generators such as wind and solar, with no reliance on the central grid. Smart Micro-grids are a type of grid. When demand cannot be satisfied locally, they can be engineered to work in tandem with central grids. They are primarily designed to meet the energy needs of the local people and to operate independently as self-contained power units. The smart micro-grid that we are considering in our research works in parallel with the main grid. Modifying the IEEE 3-bus test system is used to mimic the micro-grid. In this system, the core grid is described as an endless generator bus, which is also regarded as a slack bus. As a result, the voltage magnitude for this bus is specified, which remains at its rated value, and the voltage angle is set to zero. The second bus is a load bus that is powered by wind energy and is located in an industrial region. The projected load at the bus is subtracted from the forecasted generation from the wind electric generator. The third bus is also a load bus with solar panels and is located in a residential neighbourhood. The load requirement at this bus is anticipated load minus solar plant generation forecasted generation. Figure 1 depicts the test system. The Micro-grid components are: Variable energy resources (VERs) such as PV or wind energy, storage devices such as batteries, and finally the loads. Smart micro-grids use software and intelligent controls to manage electricity flow in networks. Voltage stability refers to a power system's ability to maintain consistent, acceptable voltages across all of its buses.

FACTS-based compensators are used to maintain constant voltage when voltages are below or over permitted limits. Time domain approaches, static methods, and sensitivity methods are examples of traditional voltage stability assessment methodologies. Voltage stability has recently become a critical issue in smart micro-grids as a result of renewable energy integration. The load and renewable generation are now dynamically changing. Solar and wind generators are currently ineffective in maintaining reactive power balance. Controlling micro-grids that alter hourly based on the situation necessitates an effective monitoring system [1,2].A supervisory control and data acquisition (SCADA) or energy management system is now used to monitor a power system. This has technological restrictions and is unable to give data that is synced. The phasor measurement unit (PMU) was developed recently and is beneficial for fast and accurate measurements that deliver data synchronised by a global positioning system (GPS). This data can be delivered in near-real time [3,4]. As a result, all of the snapshot data point to the same time, allowing for the representation of a dynamic state across a large area of the power system, allowing for a variety of stability monitoring and control applications. As a result, real-time voltage stability monitoring research has centred on establishing novel indices for detecting voltage instability.

A precise and efficient real-time online voltage stability monitoring system is required to prevent blackouts such as voltage collapse. In recent decades, accurate real-time measurement of voltage stability in micro grids has been a hot topic of research. Following the introduction of phasor measurement units (PMU), there has been a lot of discussion about phasor measuring approaches for real-time voltage stability in the last decade. The basic idea behind these methods is to locate the system's Thevenin equivalents and then use the equivalent circuits to calculate the voltage stability margin. For online monitoring of voltage stability margins, some approaches incorporate the use of Artificial Neural Networks (ANN). When compared to other ways, these are extremely quick. It has been demonstrated that if phase angles and voltage magnitude can be collected in real-time using phasor measurement units (PMU), then voltage stability margins can be obtained in real-time and voltage stability control methods may be initiated. Because of its simplicity, estimating a Thévenin equivalent (TE) impedance is useful for monitoring voltage stability in real time. The following is the basic concept of such a method: (i) obtain the voltage and current from local phasor measurement; then calculate the TE impedance of the remaining systems; (ii) voltage instability occurs in the power system when the measured load impedance equals the TE impedance (i.e., the maximum power delivered point); and (iii) convert the load observation and TE impedance to the system margin to monitor the power system's voltage stability. To estimate the TE impedance, a virtual voltage source and an impedance approach were devised [5]. In addition, the TE impedance and determining the voltage stability margin (VSM) using a synchronised phasor were introduced [6].

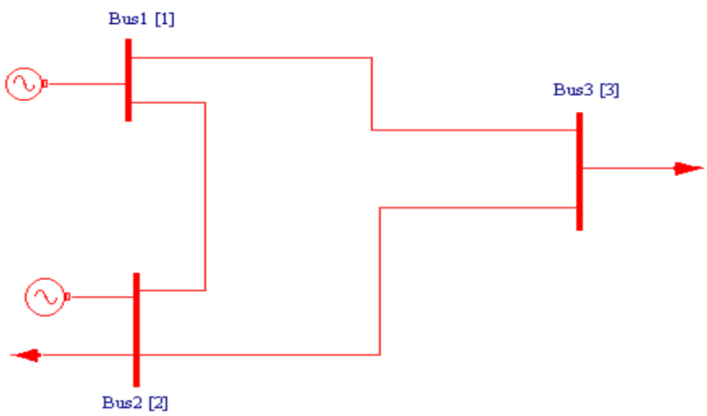

Fig. 1 Three bus smart micro-grid used as test system

Although these studies have motivated further improvements, the dynamic response of the system to practical applications must be studied further. The micro-grid systems undergo continuous variation, and the TE impedance from a particular bus cannot be fully represented by a constant value. Therefore, load modelling should be performed to capture the practical load dynamics of the power system but in micro-grid generation is also varying in nature. In addition, the network should be reconfigured using part of the calculating parameter to avoid computational burden in real time [7]. Addressing this issue, a real-time voltage stability assessment method for the Korean Power System Based on Estimation of Thévenin Equivalent Impedance is proposed in 2019 [8].

Recently there has been considerable interest in intelligent methods based on artificial neural network (ANN), fuzzy logic and genetic algorithm to voltage stability assessment problem. ANN, with the ability to provide non-linear input/output mapping, parallel processing, learning and generalization have the potential to make them ideally suited for estimating VSI's of a power system without solving the governing power system equations [9].Artificial Neural Network (ANN) is used for assessment of voltage stability or to confirm secure and insecure mode of the power system. The input data of neural network are yield from the Newton-Raphson (NR) load flow analysis [10].

Despite the fact that more research and advances are being made in the area of voltage stability evaluation, no work has been done to address the dynamic character of smart micro-grids as a result of the penetration of wind and solar power plants. The concept of using artificial intelligence techniques to predict voltage magnitudes and angles in advance based on forecasts of solar radiation, wind velocity, and load pattern has not yet gained traction, owing to the fact that it is a relatively new technology. As a result, the focus of this research is on developing artificial intelligence models based on random forest, decision tree, and KNN algorithms. The Newton-Raphson (NRLF) approach is used to create a software programme in Python based on load flow. For each load condition, this is an iterative approach for calculating complex voltage at load buses. To calculate the generated power from such VES, a function in terms of solar radiation and wind velocity is defined [11]. The developed program's voltage output, which corresponds to various loading conditions, wind velocity, and solar radiation, is utilised to train AI models. A comparison of several AI models is offered, which will be helpful in deciding which strategy is optimal for a typical network.

2. OVERVIEW

Voltage deviation at load busses is one of the most severe power quality disturbances to be dealt with by the industrial sector. Voltage stability analysis is performed to access the capacity of a micro-grid to maintain steady acceptable voltages at all buses irrespective continuous load variation. Due to penetration of solar and wind power plants the generation also varies continuously, which adds to voltage stability issue. Intelligent power electronic compensators are used now a day to maintain constant voltage profile at load bus. These compensators need load bus voltage on real time basis as input to generate compensating voltage. Traditional compensating techniques use algorithms which are iterative in nature and consume enormous computation time. Hence they cannot be used for real time estimation of bus voltage for producing compensating voltage. Recently bus voltage inputs are given to compensators from Phasor Measurement Units which are costly and there is a time delay in injecting the compensating voltage by compensator. In order to overcome the above issues, this work proposes a AI based approach which predicts the bus voltages in advance and feeds to the compensator. Micro-grid has VES whose generations vary continuously depending on solar radiation and wind velocity. Hence the inputs to the AI model are solar radiation, wind velocity and load. A trained AI model forecast the solar radiation, wind velocity and load in advance using current load, solar radiation and wind velocity which is given as input to the proposed AI model for predicting load bus voltages.

OBJECTIVES

The main objective of this work is to develop AI models capable of giving accurate predictions of complex voltage at all load busses of a smart micro-grid in order to reduce computational time and cost. For this purpose several objectives are set to appreciate the contributions of this work and they are listed as follows:

- **Functional need** - performing real time voltage stability analysis for correcting bus voltage using intelligent controllers.
- **Minimizing count of sensors** - for cost reduction and enhancing the robustness of the system.
- **Simplicity of algorithms employed** - for digital implementation that doesn't require high speed analog to digital converters and high-performance microprocessors.
- **Eliminate iterations** - for fast dynamic response and real time estimation
- **Optimizing processing time of algorithm** - to cope up with varying power system loads and varying generations from wind and solar power plants.

METHODOLOGIES

The methodologies adopted for achieving the four main objectives are as follows:

- Define a network model to represent a smart micro-grid having renewable penetration.
- Collecting practical data of solar radiation and wind velocity in Kerala from weather station.
- Create load flow model and solve for the modeled micro-grid 3-bus test system using the basics of Newton-Raphson method by developing python program.

- Generate and store the data of bus voltage for various loads, solar radiation pattern and wind velocities using the developed software program.
- Create and test AI models using decision tree, random forest and KNN algorithms using training data for predicting bus voltage magnitude and angle.
- Compare the results from AI models and select the best method for this typical micro-grid system.
- Plot the graph of bus voltage profile for the load busses and perform real time voltage stability analysis which is needed by power electronic compensators to maintain voltage profile.

3. MODELLING OF MICRO-GRID

In India, the transmission sector of Kerala State Electricity Board Ltd is geographically divided in to two zones – the North Zone headquartered at Kozhikode and the South Zone headquartered at Thiruvananthapuram. Based on the details collected from system operation wing located at Kalamassery, the 220KV substations in Kalamassery region is modelled as Micro-grid. The micro-grid under consideration is designed to operate in tandem with central grid as the solar and wind electric generation is not sufficient to meet the load demand. It is assumed that central grid can supply any power demanded by the micro-grid at specified voltage of 1.04pu. The central grid is modelled as generator bus (PV bus) with infinite generation capacity denoted as Bus-1. This bus is also treated as slack bus (reference bus) which supplies the loss in the system and voltage angle of other buses are measured with respect to this bus. Hence magnitude of voltage is 1.04 pu and angle is zero. The other two nodes of micro-grid are modelled as load bus (PQ bus) that's real and reactive powers are known and voltage magnitude and angle are unknown. Bus-2 has a wind electric generator, but the generation capacity is less than demand under normal loading condition, hence justified as load bus. Also Bus-2 corresponds to industrial area in Ernakulum. Bus-3 has solar power plant with lesser generation capacity compared to demand under normal load, hence modelling as load bus is justified.Bus-3 is located in residential area of Kochi. The real and reactive power load varies continuously throughout the day and hence bus voltages of Bus-2 & 3 also vary accordingly. Along with this there is continuous variation of power generation from solar and wind power plants depending on solar radiation and wind velocity. Hence solar and wind power generation are modelled as follows:

The yield power of a wind turbine relies upon wind speed and can be numerically communicated through equation (1)

$$P_w(v) = \begin{cases} 0 & if \quad v < 2m/s \,\&\, v > 16m/s \\ 0.7\left(\frac{v-2}{7}\right) & if \quad 2m/s < v < 9m/s \\ 0.7 & if \quad 9m/s < v < 16m/s \end{cases} \tag{1}$$

The cut-in speed is 2m/s, cut-out speed is 16m/s and rated speed is 9m/s. The difference between rated speed and cut-in speed is 7m/s.The rated power of wind electric generator is 0.7pu. The actual power generated P_w can be calculated by substituting the current value of wind speed v in equation (1).
For the solar power, the energy conservation model given in equation (2)

$$P_s(G) = \begin{cases} 0.5\left(\frac{G^2}{96000}\right) & if \quad 0 < G < 120W/m^2 \\ 0.5\left(\frac{G}{800}\right) & if \quad G > 120W/m^2 \end{cases} \tag{2}$$

Solar radiation at standard temperature and pressure is 800W/m^2 and solar irradiation is 120W/m^2 whose product is 96000. The rated power of solar power plant is 0.5pu.The actual power generated P_s can be calculated by substituting the current value of solar radiation G in equation (2).Power injected in bus-2 is (P_w-P_{L2}) and power drawn from bus-3 is (P_s-P_{L3}). The micro-grid network is modelled using bus admittance matrix (Y_{bus}) as given in equation (3). The line data are given in table-1.

TABLE I
LINE DATA OF THE TEST SYSTEM

Start Bus	End Bus	Series Admittance	Half Line Charging Admittance
1	2	1.47-j5.88	j0.015
1	3	2.94-j11.77	j0.07
2	3	2.75-j9.17	j0.04

$$\begin{pmatrix} I_1 \\ I_2 \\ I_3 \end{pmatrix} = \begin{pmatrix} (4.41 - j17.57) & (-1.47 + j5.88) & (-2.94 + j11.77) \\ (-1.47 + j5.88) & (4.22 - j15) & (-2.75 + j9.57) \\ (-2.94 + j11.77) & (-2.75 + j9.57) & (5.69 - j20.83) \end{pmatrix} \begin{pmatrix} V_1 \\ V_2 \\ V_3 \end{pmatrix} \quad (3)$$

4. FORMULATION OF POWER FLOW AND POWER MISMATCH EQUATIONS FOR TEST MICRO-GRID

The real power (P_{ical}) and reactive power (Q_{ical}) injected in i^{th} bus can be calculated using the fundamental equation (4) for the 3-bus system.

$$(P_{ical} - jQ_{ical}) = V_i^* I_i = V_i^* \sum_{j=1}^{3} Y_{ij} V_j \quad (4)$$

$$V_i = |V_i|\angle\delta_i\,;V_j = |V_j|\angle\delta_j\,;\ Y_{ij} = |Y_{ij}|\angle\theta_{ij} \quad (5)$$

Substituting (5) in (4),

$$(P_{ical} - jQ_{ical}) = \sum_{j=1}^{3} |V_i||Y_{ij}||V_j|\angle(\delta_j + \theta_{ij} - \delta_i) \quad (6)$$

Separating real and imaginary parts in (6),

$$P_{ical} = \sum_{j=1}^{3} |V_i||Y_{ij}||V_j| \cos(\delta_j + \theta_{ij} - \delta_i) \quad (7)$$

$$Q_{ical} = -\sum_{j=1}^{3} |V_i||Y_{ij}||V_j| \sin(\delta_j + \theta_{ij} - \delta_i) \quad (8)$$

The power mismatch equation for the test system is given in equation (9).

$$\begin{pmatrix} \Delta P_2 \\ \Delta P_3 \\ \Delta Q_2 \\ \Delta Q_3 \end{pmatrix} = \begin{pmatrix} \frac{\partial P_2 \partial P_2\ \partial P_2\ \ \partial P_2}{\partial \delta_2 \partial \delta_3 \partial |V_2| \partial |V_3|} \\ \frac{\partial P_3 \partial P_3\ \partial P_3\ \ \partial P_3}{\partial \delta_2 \partial \delta_3 \partial |V_2| \partial |V_3|} \\ \frac{\partial Q_2 \partial Q_2\ \partial Q_2\ \ \partial Q_2}{\partial \delta_2 \partial \delta_3 \partial |V_2| \partial |V_2|} \\ \frac{\partial Q_2 \partial Q_2\ \partial Q_2\ \ \partial Q_2}{\partial \delta_2 \partial \delta_3 \partial |V_3| \partial |V_3|} \end{pmatrix} \begin{pmatrix} \Delta \delta_2 \\ \Delta \delta_3 \\ \Delta |V_2| \\ \Delta |V_3| \end{pmatrix} \quad (9)$$

The 4x4 matrix of the power mismatch equation is called Jacobian matrix. Elements of Jacobian matrix can be calculated using equations (10) to (17)

$$\frac{\partial P_i}{\partial \delta_i} = \sum_{\substack{j=1 \\ j \neq i}}^{3} |V_i||Y_{ij}||V_j| \sin(\delta_j + \theta_{ij} - \delta_i) \quad (10)$$

$$\frac{\partial P_i}{\partial \delta_j} = -|V_i||Y_{ij}||V_j| \sin(\delta_j + \theta_{ij} - \delta_i) \quad (11)$$

$$\frac{\partial Q_i}{\partial \delta_i} = \sum_{\substack{j=1 \\ j \neq i}}^{3} |V_i||Y_{ij}||V_j| \cos(\delta_j + \theta_{ij} - \delta_i) \quad (12)$$

$$\frac{\partial Q_i}{\partial \delta_j} = -|V_i||Y_{ij}||V_j| \cos(\delta_j + \theta_{ij} - \delta_i) \quad (13)$$

$$\frac{\partial P_i}{\partial |V_i|} = 2|V_i||Y_{ii}| \cos\theta_{ii} + \sum_{\substack{j=1 \\ j \neq i}}^{3} |Y_{ij}||V_j| \cos(\delta_j + \theta_{ij} - \delta_i) \quad (14)$$

$$\frac{\partial P_i}{\partial |V_j|} = |Y_{ij}||V_j| \cos(\delta_j + \theta_{ij} - \delta_i) \quad (15)$$

$$\frac{\partial Q_i}{\partial |V_i|} = -2|V_i||Y_{ii}| \sin\theta_{ii} - \sum_{\substack{j=1 \\ j \neq i}}^{3} |Y_{ij}||V_j| \sin(\delta_j + \theta_{ij} - \delta_i) \quad (16)$$

$$\frac{\partial Q_i}{\partial |V_j|} = -|Y_{ij}||V_j| \sin(\delta_j + \theta_{ij} - \delta_i) \quad (17)$$

$$\Delta P_i = P_{isp} - P_{ical} \quad (18)$$

$$\Delta Q_i = Q_{isp} - Q_{ical} \quad (19)$$

$$Updated\ \delta_i = \delta_i + \Delta\delta_i \quad (20)$$

$$Updated\ |V_i| = |V_i| + \Delta|V_i| \quad (21)$$

GENERATION OF TRAINING DATA

A software program is developed in python to produce bus voltages of load buses as output for the given load, solar radiation and wind speed data. The program is developed using the basics of Newton-Raphson load flow analysis. The algorithm is given below:

1) Assume magnitude of voltage at bus 2 & 3 as 1pu and angle as zero.
2) Input real and reactive power load at bus-2 and bus-3 (P_{L2}, P_{L3}, Q_{2sp}& Q_{3sp}).
3) Input the actual solar radiation and wind speed (G & v)
4) Calculate P_w and P_s using equations (1) and (2)
5) Calculate P_{2sp} and P_{3sp} using $P_{2sp}=(P_w-P_{L2})$ & $P_{3sp}=(P_s-P_{L3})$
6) Calculate P_{2cal}, P_{3cal}, Q_{2cal} and Q_{3cal} using equation (7) and (8).
7) Calculate ΔP_2, ΔP_3, ΔQ_3 and ΔQ_3 using equation (18) and (19).
8) If magnitude of all ΔP and ΔQ are less than error tolerance value 0.001, go to step -13.
9) Calculate elements of Jacobian matrix using equations (10) to (17) and find its inverse.
10) Calculate $\Delta\delta_2$, $\Delta\delta_3$, $\Delta|V_2|$ and $\Delta|V_3|$ using equation (9).
11) Update voltage magnitude and its angle using equations (20) and (21).
12) Go to step-5
13) Print the value of $|V_2|$, $|V_3|$, $\delta_2$and δ_3.

Generate and store the data of bus voltage for various loads, solar radiation pattern and wind velocities using the developed software program which is used as training data for AI models. Flow chart of the program is given in figure-2.

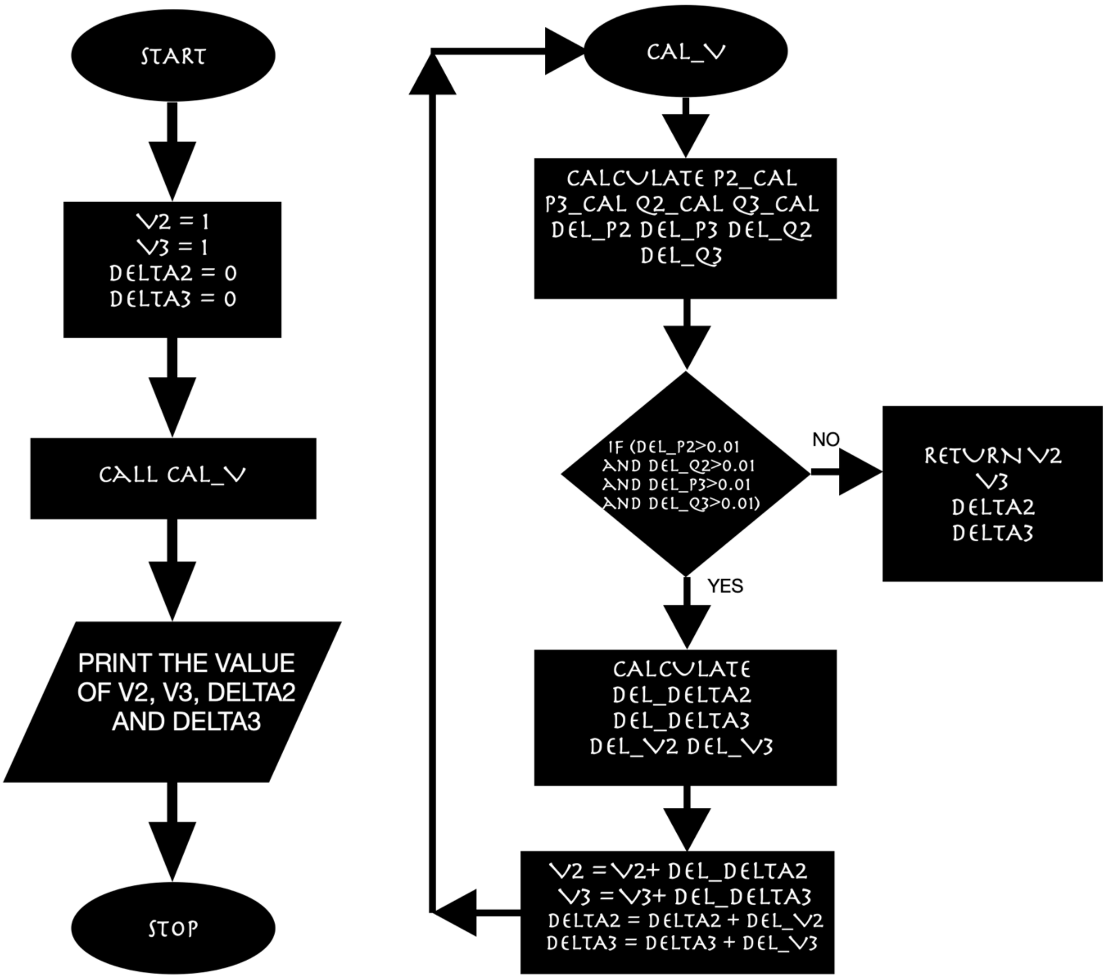

Fig. 2 Flowchart for generation of training data

SAMPLE TRAINING DATA

The average daily load curve for several months pertaining to industrial and residential area of central Kerala is obtained from Kerala state load despatch centre, Kalamassery and is used as input for generating training data. Training data is obtained for variation in real and reactive power from 30% to full load as per load curve pattern. Statistics of wind speed for Fort Kochi under different months is obtained from WINDY.APP is used input the wind speed. The solar radiation pattern in Kochi is obtained from TuTiempo.net. Sample training data for a given day generated from developed python program is given in table-2. S is wind speed in m/s, G is solar radiation in W/m^2, P_{L2} and P_{L3} are real power load at bus-2 & 3, Q_2 and Q_3 are reactive power load at bus-2 & 3, P_s and P_w are the generation from solar and wind, V_2 and V_3 are magnitude of voltage atbus-2 & 3 and DEL_2 and DEL_3 are angle of voltage atbus-2 & 3. The variation of various quantities verses time is shown in figure-3.

TABLE III
SAMPLE TRAINING DATA FOR SOLAR, WIND AND LOAD PATTERN OF A DAY

Time (hrs)	S (m/s)	G (W/m2)	P_L2 (pu)	P_L3 (pu)	Q2 (pu)	Q3 (pu)	V2 (pu)	V3 (pu)	DEL_2 (rad)	DEL_3 (rad)	Ps (pu)	Pw (pu)
0	3	20	0.2	0.42	0.13	0.35	1.019	1.016	-1.142	-1.421	0	0.1
2	4	20	0.25	0.38	0.18	0.31	1	1	0	0	0	0.2
4	6	100	0.2	0.4	0.13	0.33	1.027	1.021	0.406	-0.589	0.05	0.4
6	8	400	0.8	0.5	0.73	0.43	0.962	0.988	-0.476	-0.692	0.25	0.6
8	7	600	1	0.7	0.93	0.63	0.929	0.963	-1.693	-1.284	0.375	0.5
10	10	800	0.95	0.8	0.88	0.73	0.935	0.961	-0.477	-0.625	0.5	0.7
12	6	1000	0.9	0.75	0.83	0.68	0.937	0.967	-1.385	-0.614	0.625	0.4
14	10	600	0.93	0.7	0.86	0.63	0.941	0.968	-0.508	-0.766	0.375	0.7
16	11	300	0.9	0.8	0.83	0.73	0.938	0.959	-0.943	-1.603	0.188	0.7
18	13	40	0.95	1	0.88	0.93	0.92	0.937	-1.793	-2.786	0.008	0.7
20	10	20	0.2	0.98	0.13	0.91	1.005	0.976	0.818	-1.593	0.002	0.7
22	8	20	0.15	0.7	0.08	0.63	1.022	1	0.954	-1.02	0.002	0.6
24	3	20	0.2	0.42	0.13	0.35	1.019	1.016	-1.142	-1.421	0	0.1

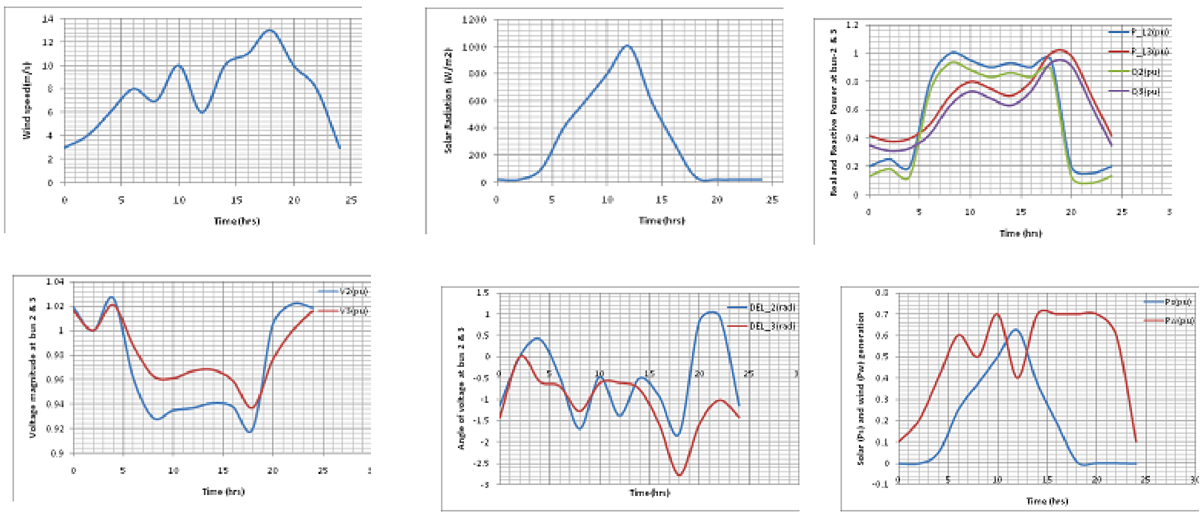

Fig. 3 Variation S, G, P_{L2}, P_{L3}, Q_2, Q_3, V_2, V_3, DEL_2, DEL_3, P_s, P_w Vs. time

5. DATA PRE-PROCESSING AND EDA

Data pre-processing is the important step as the performance of the algorithm depends on the quality of our data on which the algorithm is trained. In data pre-processing firstly we need to handle missing data. Missing Data occurs when no values is provided for one or more items. Missing Data is a very big problem in a real-life scenarios. However, for our training data there are no missing values. Next we generally remove duplicates as duplicates are an extreme case of non-random sampling, and they can make are model bias. Sometimes duplicates might lead to overfitting. Nevertheless, for model to train we need significant amount of data therefore

we are not removing the duplicates here. Next vital step is to check the outliers. Some algorithms are very sensitive to outliers.
If the value is greater than upper limit or lower than lower limit then the value is considered to be outliers. In our data there is one outliers in wind velocity (S), no outliers in solar radiation (G), real power of bus 2 (PL_2) and reactive power of bus 2 (Q2) and few outliers in real power of bus 3 (PL_3) and reactive power of bus 3 (Q3) as you can see in figure 4.

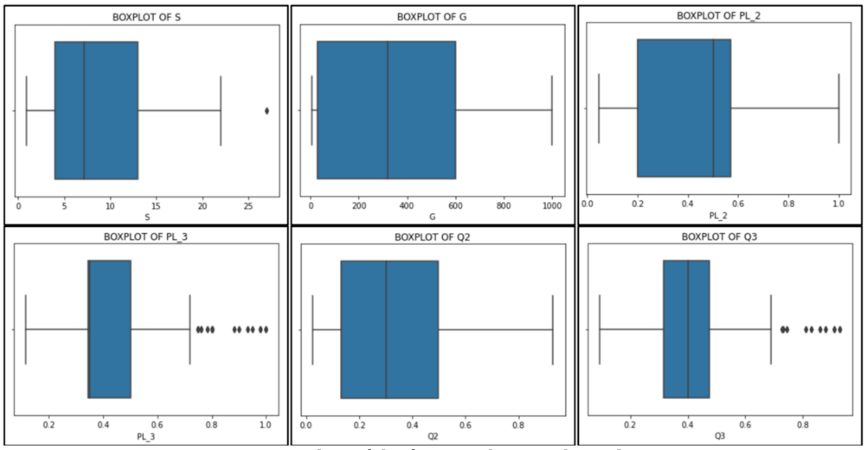

Fig. 4 Box plots of the features showing the outliers

To decide on what algorithm to use for our dataset we need to visualise and learn the distributions of our data. This processes of analysing the dataset and visualising them is called exploratory data analysis (EDA). Analysis the distribution of features is done using the distribution plot. Distribution of S (wind velocity), G (solar radiation), PL_2(real power) and Q2(reactive power) are normal while the distribution of PL_3 and Q3 are positively skewed as seen in figure 5. We know that PL_3 and Q3 are positively skewed because the tail of distribution is towards the right (positive) side.

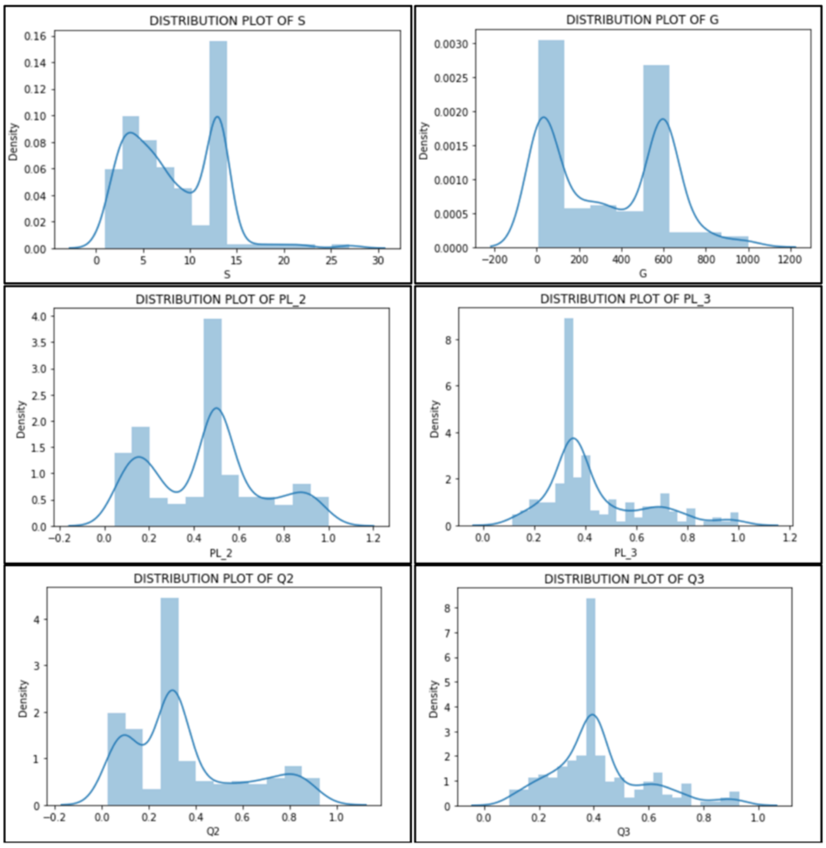

Fig. 5 Distribution plots of the features

Best fit line is drawn between a feature and the target to understand the relationship between them and to find if the target is dependent on this feature. For our dataset there is good relationship as seen figure 6. The line plot is drawn for the real power and reactive power only as the wind velocity and solar radiation doesn't directly have impact on targets.

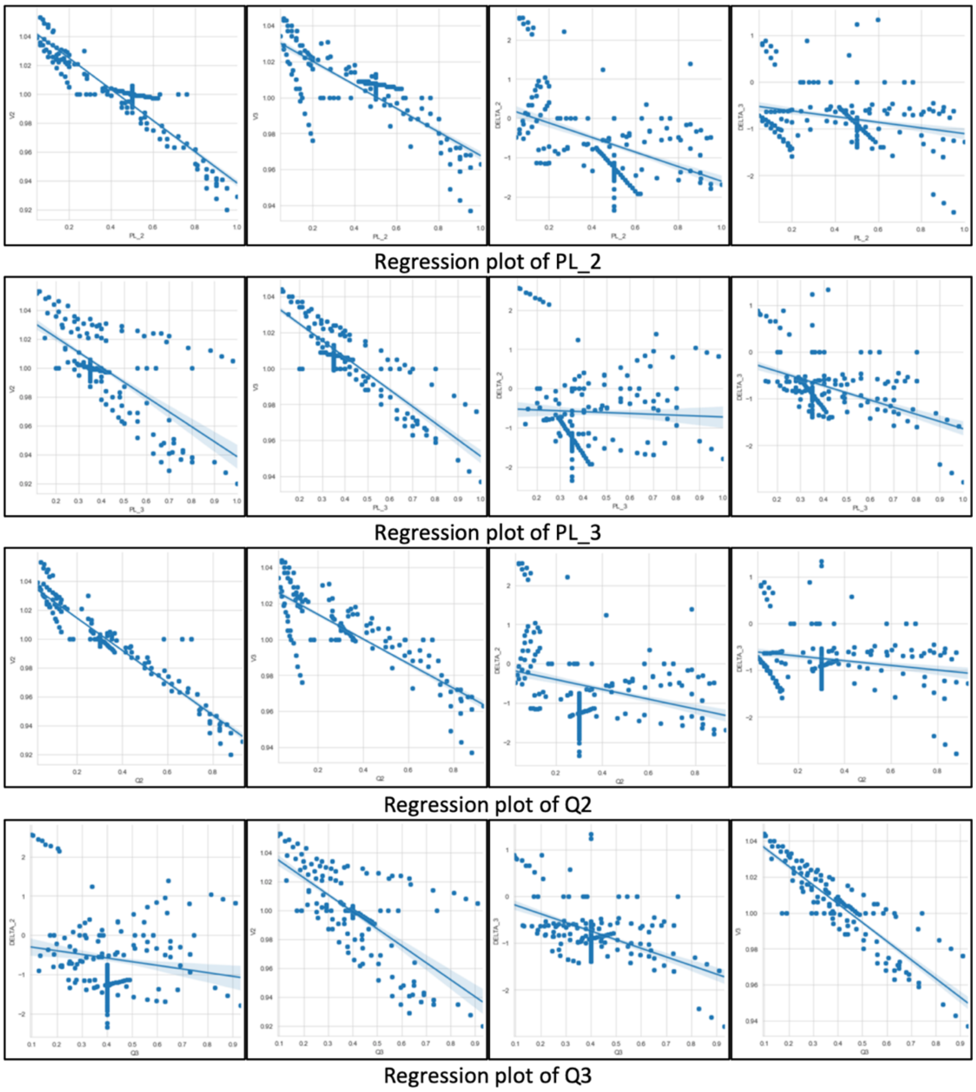

Fig. 6 LM plot of feature with targets

Finally, the correlation with the target and among the feature has to be analysed. The correlation with the target seems pretty good for all features but there some multicollinearity between PL_2 and Q2, and PL_3 and Q3. In order to visualise the correlation, heat map is drawn in figure 7 and correlation table is shown in table 3. As we can see the data contains outliers, duplicates, multicollinearity and not normally distributed data therefore Decision tree, Random forest and K nearest neighbour are the best suitable algorithm for this problem.

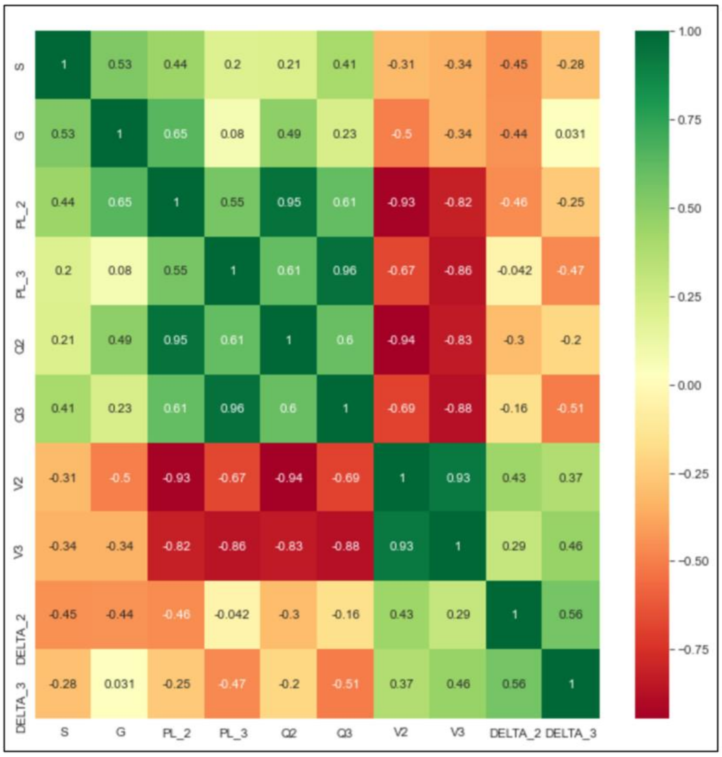

Fig. 7 Heat map of the features and targets

TABLE IIIII
CORRELATION TABLE OF FEATURES AND TARGETS

	S	G	PL_2	PL_3	Q2	Q3	V2	V3	DELTA_2	DELTA_3
S	1.000000	0.532254	0.440508	0.195727	0.209306	0.405835	-0.310219	-0.340491	-0.447685	-0.284786
G	0.532254	1.000000	0.648628	0.079731	0.489575	0.226864	-0.503130	-0.339679	-0.441854	0.031249
PL_2	0.440508	0.648628	1.000000	0.554022	0.951897	0.606505	-0.934919	-0.815404	-0.463606	-0.246466
PL_3	0.195727	0.079731	0.554022	1.000000	0.611930	0.959569	-0.672900	-0.863057	-0.042188	-0.468175
Q2	0.209306	0.489575	0.951897	0.611930	1.000000	0.604840	-0.944906	-0.832591	-0.304549	-0.196350
Q3	0.405835	0.226864	0.606505	0.959569	0.604840	1.000000	-0.691798	-0.879141	-0.157874	-0.507075
V2	-0.310219	-0.503130	-0.934919	-0.672900	-0.944906	-0.691798	1.000000	0.928860	0.430506	0.374450
V3	-0.340491	-0.339679	-0.815404	-0.863057	-0.832591	-0.879141	0.928860	1.000000	0.293875	0.459685
DELTA_2	-0.447685	-0.441854	-0.463606	-0.042188	-0.304549	-0.157874	0.430506	0.293875	1.000000	0.563034
DELTA_3	-0.284786	0.031249	-0.246466	-0.468175	-0.196350	-0.507075	0.374450	0.459685	0.563034	1.000000

6. DECISION TREE ALGORITHM

Decision Trees are a type of supervised learning technique that can be used to solve classification and regression problems. Internal nodes represent dataset attributes, branches represent decision rules, and each leaf node provides the conclusion in this tree-structured classifier. The Decision Node and the Leaf Node are the two nodes of a Decision tree. Leaf nodes are the output of those decisions and do not contain any more branches, whereas Decision nodes are used to make any decision and have several branches. The decisions or tests are made based on the characteristics of the given dataset. It's termed a decision tree because, like a tree, it starts with the root node and grows by branching out to form a tree. The procedure for determining the class of a given dataset in a decision tree starts at the root node of the tree. This algorithm checks the values of the root attribute with the values of the record (actual dataset) attribute and then follows the branch and jumps to the next node based on the comparison. The algorithm compares the attribute value with the other sub-nodes and moves on to the next node. It continues the process until it reaches the leaf node of the tree. The procedure is given below:

1. Begin the tree with the root node, says S, which contains the complete dataset.
2. Find the best attribute in the dataset using Attribute Selection Measure (ASM).
3. Divide the S into subsets that contains possible values for the best attributes.
4. Generate the decision tree node, which contains the best attribute.
5. Recursively make new decision trees using the subsets of the dataset created in step -3. Continue this process until a stage is reached where you cannot further classify the nodes and called the final node as a leaf node.

The biggest challenge that emerges while developing a Decision tree is how to choose the best attribute for the root node and sub-nodes. So, there is a technique called Attribute Selection Measure, or ASM, that can be used to overcome such situations. We can easily determine the best property for the tree's nodes using this measurement. Information gain and the Gini index are two often used ASM approaches. The assessment of changes in entropy after segmenting a dataset based on an attribute is known as information gain. It determines how much data a feature offers about a class. We split the node and built the decision tree based on the value of information gained. The Gini index is a measure of impurity or purity used in the CART (Classification and Regression Tree) technique to create a decision tree. In comparison to a high Gini index, an attribute with a low Gini index should be favoured. Over fitting is more likely with a large tree, yet a small tree may not capture all of the key properties of the dataset. Pruning is a strategy for reducing the size of the learning tree without reducing accuracy.

7. RANDOM FOREST ALGORITHM

Random Forest is a well-known machine learning algorithm that uses the supervised learning method. In machine learning, it can be utilised for both classification and regression tasks. It is based on ensemble learning, which is a method of integrating several classifiers to solve a complex problem and increase the model's performance. Random Forest is a classifier that combines a number of decision trees on different subsets of a dataset and averages the results to increase the dataset's predicted accuracy. Instead than relying on a single decision tree, the random forest collects the predictions from each tree and forecasts the final output based on the majority votes of predictions. Random Forest works in two-phase first is to create the random forest by combining N decision tree, and second is to make predictions for each tree created in the first phase. The procedure is given below:

1. Select random K data points from the training set.
2. Build the decision trees associated with the selected data points (Subsets).
3. Choose the number N for decision trees that you want to build.
4. Repeat Step 1 & 2.

For new data points, find the predictions of each decision tree, and assign the new data points to the category that wins the majority votes.

8. K-NEAREST NEIGHBOR (KNN) ALGORITHM

The K-Nearest Neighbour algorithm is based on the Supervised Learning technique and is one of the most basic Machine Learning algorithms. The K-NN method assumes that the new case/data and existing cases are similar and places the new case in the category that is most similar to the existing categories. The K-NN method stores all available data and classifies a new data point based on its similarity to the existing data. This means that new data can be quickly sorted into a well-defined category using the K-NN method. The K-NN approach can be used for both regression and classification, but it is more commonly utilised for classification tasks. The K-NN algorithm is a non-parametric algorithm, which means it makes no assumptions about the underlying data. It's also known as a lazy learner algorithm since it doesn't learn from the training set right away; instead, it saves the dataset and performs an action on it when it comes time to classify it. During the training phase, the KNN algorithm simply stores the dataset, and when it receives new data, it classifies it into a category that is quite similar to the new data. The procedure is given below:

1. Select the number K of the neighbors
2. Calculate the Euclidean distance of K number of neighbors
3. Take the K nearest neighbors as per the calculated Euclidean distance.
4. Among these k neighbors, count the number of the data points in each category.
5. Assign the new data points to that category for which the number of the neighbor is maximum.
6. The model is ready.

9. RESULTS AND DISCUSSIONS

After EDA and data pre-processing, we need to train the selected algorithms with the training data. Now, since the algorithm is trained, we can give data the model will make predictions for us. One set of data, one day's data, is given as input to the Newton Raphson Method, Decision Tree, Random Forest and K- Nearest Neighbor. The input data is show in table 4. The output from the Newton Raphson Method, Decision Tree Model, Random Forest Model and K- Nearest Neighbor is recorded in table 5. By comparing the outputs with Newton Raphson Method, we can see Decision Tree and Random Forest Models predicted correctly but K- Nearest Neighbor is little inaccurate.

TABLE IVV
INPUT DATA FOR THE MODELS

Day 1	Input Data					
Time(hrs)	S(m/s)	G(W/m2)	P_L2(pu)	P_L3(pu)	Q2(pu)	Q3(pu)
0	2.85	19	0.19	0.399	0.12	0.329
2	3.8	19	0.2375	0.361	0.1675	0.291
4	5.7	95	0.19	0.38	0.12	0.31
6	7.6	380	0.76	0.475	0.69	0.405
8	6.65	570	0.95	0.665	0.88	0.595
10	9.5	760	0.9025	0.76	0.8325	0.69
12	5.7	950	0.855	0.7125	0.785	0.6425
14	9.5	570	0.8835	0.665	0.8135	0.595
16	10.45	285	0.855	0.76	0.785	0.69
18	12.35	38	0.9025	0.95	0.8325	0.88
20	9.5	19	0.19	0.931	0.12	0.861
22	7.6	19	0.1425	0.665	0.0725	0.595
24	2.85	19	0.19	0.399	0.12	0.329

TABLE V

OUTPUT FROM ALL THE MODELS

Day 1	Newton Raphson Method				Decision Tree				Random Forest				K- Nearest Neighbor			
Time(hrs)	V2(pu)	V3(pu)	DEL_2(rad)	DEL_3(rad)	V2(pu)	V3(pu)	DEL_2(rad)	DEL_3(rad)	V2(pu)	V3(pu)	DEL_2(rad)	DEL_3(rad)	V2(pu)	V3(pu)	DEL_2(rad)	DEL_3(rad)
0	1.021	1.018	-1.145	-1.384	1.021	1.018	-1.145	-1.384	1.021	1.018	-1.145	-1.384	1.021	1.018	-1.145	-1.384
2	1	1	0	0	1	1	0	0	1	1	0	0	1.0042	1.0036	-0.229	-0.2768
4	1.029	1.023	0.32	-0.601	1.029	1.023	0.32	-0.601	1.029	1.023	0.32	-0.601	1.0292	1.0232	0.3028	-0.6034
6	0.966	0.991	-0.52	-0.704	0.966	0.991	-0.52	-0.704	0.966	0.991	-0.52	-0.704	0.9652	0.9904	-0.5112	-0.7016
8	0.935	0.968	-1.669	-1.254	0.935	0.968	-1.669	-1.254	0.935	0.968	-1.669	-1.254	0.9374	0.9688	-1.4006	-1.1386
10	0.942	0.966	-0.293	-0.531	0.942	0.966	-0.293	-0.531	0.942	0.966	-0.293	-0.531	0.9534	0.9738	-0.47	-0.5802
12	0.942	0.971	1.389	-0.618	0.942	0.971	1.389	-0.618	0.942	0.971	1.389	-0.618	0.9422	0.971	0.2814	-0.6202
14	0.947	0.972	-0.327	-0.677	0.947	0.972	-0.327	-0.677	0.947	0.972	-0.327	-0.677	0.9446	0.9712	-0.5954	-0.7924
16	0.943	0.963	-0.762	-1.472	0.943	0.963	-0.762	-1.472	0.943	0.963	-0.762	-1.472	0.9518	0.9716	-0.7418	-1.3094
18	0.928	0.943	-1.544	-2.585	0.928	0.943	-1.544	-2.585	0.928	0.943	-1.544	-2.585	0.9278	0.943	-1.551	-2.5884
20	1.008	0.98	0.929	-1.453	1.008	0.98	0.929	-1.453	1.008	0.98	0.929	-1.453	1.0096	0.982	0.9738	-1.397
22	1.0238	1.002	0.845	-1.012	1.0238	1.002	0.845	-1.012	1.0238	1.002	0.845	-1.012	1.02388	1.0032	0.7946	-1.0048
24	1.021	1.018	-1.145	-1.384	1.021	1.018	-1.145	-1.384	1.021	1.018	-1.145	-1.384	1.021	1.018	-1.145	-1.384

To get a clear review of the performance of the algorithms, we have calculated the R2 score and Mean absolute percentage error. R-squared (R2) is a statistical measure that quantifies the proportion of variation explained by an independent variable or variables in a regression model for a dependent variable. R-squared reveals how much the variation of one variable explains the variance of the second variable, whereas correlation explains the strength of the relationship between an independent and dependent variable. So, if a model's R2 is 0.50, the model's inputs can explain nearly half of the observed variation. Decision Tree has a score of 0.99, Random forest has score of 0.99 and KNN has a score of 0.83. A forecast system's accuracy is measured by the mean absolute percentage error (MAPE). It is determined as the average absolute percent error for each time period minus actual values divided by real values, and it is expressed as a percentage. MAPE of decision tree is 2.70, random forest is 13310078105615.7 and KNN is 53661166042813.9. To visualise a comparison chart is plotted in figure 8.

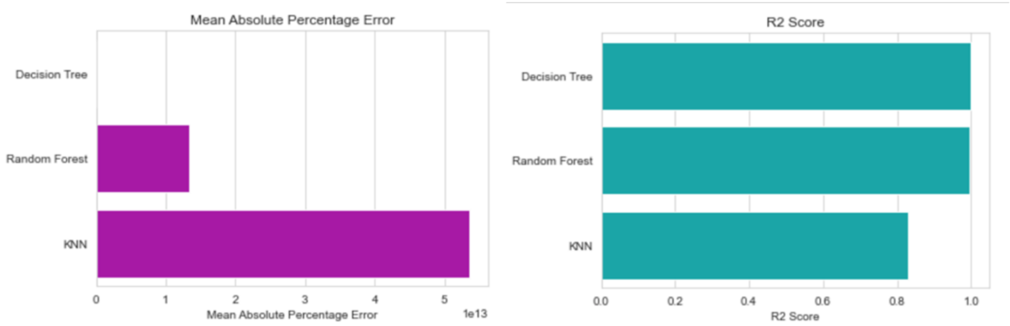

Fig. 8 Bar chart for comparison

It is clear from charts and tables that Decision tree gives the best performance. To visualise how the decision tree has predicted we can plot a regression plot. The regression plot is shown in figure 9. The plots seem very accurate so we can confirm that decision tree model for this problem.

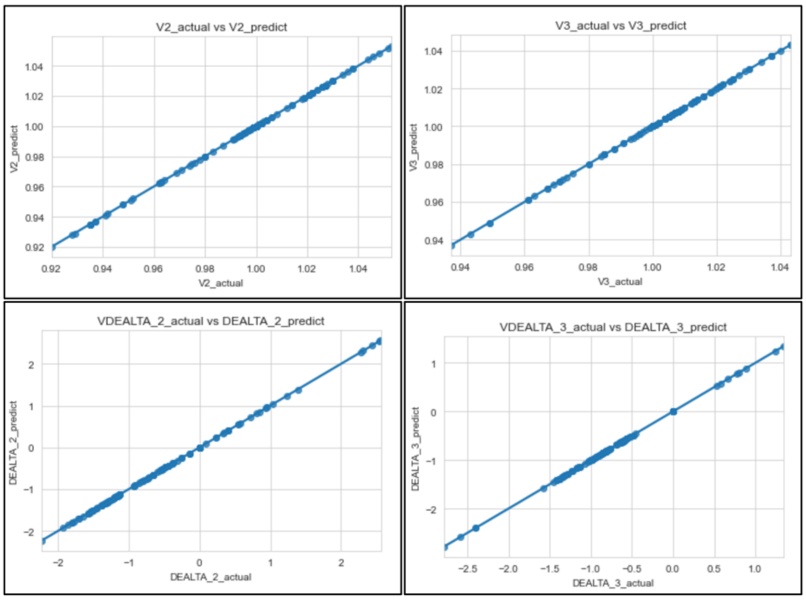

Fig. 9 Regression plot for decision tree model

10. CONCLUSION

Traditional voltage stability analysis and AI-based real-time voltage stability analysis methodologies did not account for the implications of VES integration in micro-grid in the prior studies. Furthermore, the fluctuation of available power adds to the difficulty of considering scenarios of overestimation and underestimation. As a result, the bus voltage magnitudes are predicted ahead of time in this study, and the consequences of VES intermittency are taken into account. The bus voltage magnitude and angle matching to input solar radiation, wind velocity, real power and reactive power demands are output by the suggested AI model. An updated 3-bus micro-grid system with solar and wind power plant integration is used to verify the accuracy of the suggested model. It's worth highlighting that the provided approach outperformed other similar approaches in numerical analysis. However, the work can be expanded in the future by incorporating more renewable energy sources, MPPT for solar arrays, battery storage systems, and a larger number of buses in the micro-grid system, as well as future scope in the integration of FACT devices for addressing voltage stability issues and an algorithm to forecast load, solar radiation, and wind speed.

REFERENCES

1. Van Cutsem, T. Voltage instability: Phenomena, counter measures, and analysis methods. Proc. IEEE 2000, 88, 208–227.
2. Taylor, C.; Erickson, D.; Martin, K.E.; Wilson, R. WACS-wide-area stability and voltage control system: R&D and online demonstration. Proc. IEEE 2005, 93, 892–906.

3. Sinha, A.K.; Hazarika, D. A comparative study of voltage stability indices in a power system. Int. J. Electr. Power Energy Syst. 2000, 22, 589–596.
4. Zarate, L.; Castro, C. A critical evaluation of a maximum loading point estimation method for voltage stability analysis. Electr. Power Syst. Res. 2004, 70, 195–202.
5. Mou, X.; Li, W.; Li, Z. A preliminary study on the Thevenin equivalent impedance for power systems monitoring. In Proceedings of the 2011 4th International Conference on Electric Utility Deregulation and Restructuring and Power Technologies, Weihai, Shandong, China, 6–9 July 2011; pp. 730–733.
6. Huang, L.; Xu, J.; Sun, Y.; Cui, T.; Dai, F. Online monitoring of wide-area voltage stability based on short circuit capacity. In Proceedings of the 2011 Asia-Pacific Power and Energy Engineering Conference, Wuhan, China, 25–28 March 2011; pp. 1–5.
7. Mohammadi, F.; Zheng, C. Stability Analysis of Electric Power System. In Proceedings of the 2018 4th National Conference on Technology in Electrical and Computer Engineering, Bern, Switzerland, 20–22 December 2018; pp. 1–8.
8. Yunhwan Leeand Sangwook Han, Real-Time Voltage Stability Assessment Method for the Korean Power System Based on Estimation of Thévenin Equivalent Impedance, MDPI Applied Sciences,23 April 2019, pp.1-19.
9. Yuba Raj Adhikari, Rishi Kumar Barnawal, Artificial Neural Network Based Approach for Voltage Stability Analysis of for Sustained Operation of Power System, Journal of Advanced College of Engineering and Management, Vol.6, October 2021, pp.259-267
10. Ankit kumar Sharma, Akash Saxena, Voltage Stability Assessment using Artificial Neural Network, IEEE-IEEMA International Conference “Engineer Infinite”, December 2021, pp.1-6.
11. Muhammad Arsalan Ilyas, ThamerAlquthami, Muhammad Awais, A day ahead dynamic optimal power flow with renewable energy integration in smart grid, Frontiers in Energy Research, Vol.9, August 2021.