Partial Discharge Localization in 3-D With a Multi-DNN Model Based on a Virtual Measurement Method

ATTUL MUNIM ZAKI1,2, YUE HU1, (Member, IEEE), AND XIUCHEN JIANG1

1Department of Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
2Department of Electrical Engineering, University of Azad Jammu and Kashmir, Muzaffarabad 13100, Pakistan

Corresponding author: Yue Hu (yuehu@sjtu.edu.cn)

This work was supported in part by the NSFC under Grant 51777122, and in part by the Natural Science Foundation of Shanghai, China, under Grant 17ZR1414400.

ABSTRACT Time difference of Arrival (TDOA)-based localization method, although used widely, calls for a fast and accurate solution owing to its time inefficiency and sensitivity to time delay estimation. In order to speed up the solution for nonlinear TDOA equations, while guaranteeing the location accuracy, this paper presents a hybrid approach namely multi-deep neural network model based on a virtual measurement method (MDNNM-VMM). Data consisting of multiple time difference values, resulting from a virtual measurement method (VMM), are fed to a pretrained multi-deep neural network model (MDNNM). These ‘n’ number of virtually generated sequences of time delays are obtained from a single set of TDOA equations, while conforming with a uniform distribution. The multi-DNN model using these data, outputs the required partial discharge (PD) coordinates that help determine accurate PD location. While applying a measurement error of 24 ns, the average error values $\Delta r$, $\Delta \theta$, $\Delta \phi$, and $\Delta d$ for the proposed method, compared to a multi-DNN method, see a significant percentage decrease of 32 $\%$, 24 $\%$, 39 $\%$, and 44 $\%$, respectively. Additionally, varying simulated error, different array designs, and certain other parameters are studied to make the PD localization process more efficient and multifaceted.

INDEX TERMS Deep neural network (DNN), partial discharge (PD) localization, time difference of arrival (TDOA), ultra-high frequency (UHF), virtual measurement method (VMM).

I. INTRODUCTION

Insulation condition monitoring in substations through partial discharge (PD) detection has drawn great attention since the last decade [1], [2], as PD signals contain an ample amount of information regarding the insulation status of electrical equipment [3], [4]. PD is not merely an indicator to degrading insulation level, rather it warns about an alarming condition that could henceforth lead to a complete breakdown. Therefore, detecting PDS is a must to carry out insulation diagnosis; and, localizing them could help avoid complete insulation failures [5], [6].

One of the important problems with PD detection is its source localization, which is necessary for a fast and accurate fault detection [7]. Ultra-high frequency (UHF) sensor array-based methods have been widely applied owing to the powerful characteristics of UHF electromagnetic waves, namely stable transmitting speed and anti-interference. At the heart of these methods lies the concept of measuring distance amongst sensors and PD source(s). Based on angle and time, or strength information of UHF signals, UHF methods can either be based on Angle of Arrival (AOA), Time of Arrival (TOA), Time Difference of Arrival (TDOA), or Received Signal Strength Indicator (RSSI) [8]–[10]. TDOA-based method has already received more attention than other methods due to its comparatively high accuracy. However, TDOA-based method has the curse of solving non-linear equations through iterative methods [11], [12], which in addition to being time consuming is computationally inefficient at times (as the solution could be non-optimal) [7].

To avoid these shortcomings, a shift toward data-driven methods was noticed. For this purpose, a number of methods were put forward that were based on shallow architectures,
like Artificial Neural Networks (ANNs) and Support Vector Machines (SVMs). Mas’ud [1], in addition to encompassing the benefits of ANNs, details a number of well-known ANN-based architectures that have been put forward by scholars for the PD detection over the past years. These include Modular Neural Networks (MNNs), Ensemble Neural Networks (ENN), Probabilistic Neural Networks (PNN), and Radial Basis Function Networks (RBFNs). Multiple RBFNs has also been successfully applied by Nan Zhou [13]. Despite giving appreciable levels of accuracy, ANNs in fact lack when selecting the feature vector(s) and during the explanation of results. These shortcomings and limitations associated with ANNs [1] called for some new and improved approaches. This created a room for Deep Neural Networks (DNNs), as they had already proven to be efficient in various domains, and were far better than ANNs [13]–[15].

Reference [7] has presented a fast and efficient approach based on single and multiple DNNs, including both 2-D and 3-D cases, that is by far the most promising and comprehensive attempt; as it includes various array designs, parameters related to array size, and different simulated error types added to the time delay sequences. The basis of this approach is to bypass the computational work required for solving nonlinear equations while considering massive data. PD sources are randomly generated initially as test cases and distance between each sensor of the antenna array and every PD source is calculated by considering the time difference amongst the electromagnetic signals reaching different sensors, with the prior knowledge that speed of electromagnetic waves approximately equals the speed of light. These time delay sequences are thereafter fed as input to a multiple DNN-model, and the output of model for the obtained PD source locations are the predicted coordinates (r, theta, and phi). Average error between the test and predicted PD locations showed excellent performance, especially for the Y-shaped array. However, there is a problem with this approach, that it depends on availability of measured time delay data during its testing phase for its implementation. This demands repetitive measurements that are definitely time consuming. This weakness has been overcome by [16], while extending and improving the ideas given by [17] and [18] that were limited to 2-D analysis. Reference [16] fulfilled the demand of massive data through a Data Enhancement Method (DEM) based on statistical simulations. It also guaranteed a prompt solution for PD-detection problem unlike [17] and [18], by considering a DNN model for both 2-D and 3-D cases. Furthermore, the criticality of accurate time delay estimation, as highlighted by [19], was efficiently dealt with. Thus, [16] was a hybrid approach ensuring a fast and accurate solution. Despite introducing a new idea, it was a specific case having limitations due to considering merely a single PD source; therefore, it required generalization for its true applicability.

Most of the afore-mentioned PD detection approaches either focus on accuracy or deal merely with speed. This work, on the other hand, gives a wide-ranging and multifaceted solution for a fast and quick PD location and detection problem. In addition to considering aspects related to array size and shape etc., the proposed method presents a generalized methodology that combines the pros of previous methods and avoids their cons, while comparing ‘N’ PD test points against ‘N’ PD predicted points. This method, unlike the conventional or contemporary methods, does not require massive datasets for predicting PD sources. Only a single TDOA equations’ set is required to predict each PD source efficiently and accurately. Neither repetitive measurements nor computational burden is needed; therefore, this method is both time-efficient and cost-effective.

In order to validate this method, after getting a TDOA equations’ set for each randomly generated PD source, firstly an ‘n’ number of similar sets of TDOA equations are obtained statistically, through a Virtual Measurement Method (VMM). Secondly, these datasets of time difference equations are fed to a multi-DNN model (MDNNM) that consequently predicts the PD location coordinates (r, theta, phi) for all the PDs. Accuracy is ensured by VMM, while a fast solution is guaranteed by MDNNM.

After the introduction, section 2 explains the theoretical aspects of 2-D and 3-D models for the PD localization, including the details related to the VMM. Section 3 shows how VMM helps fulfill the data demand of MDNNM. Subsequently, section 4 analyzes the obtained results and gives a comparative discussion with some previous works; and finally, the work ends with a conclusion section.

II. LOCALIZATION METHOD

Two major goals while working with PD localization problems are their precision and speed [6], [9]. Therefore, to ensure an accurate detection of PD source, a statistical simulation-based VMM is proposed in this section; whereas, for speedy detection of PD source a Multi DNN model is introduced in the next section. VMM-based mathematical model used in this paper has been introduced previously by [9], which itself is an improved version of extended 2-D model presented by [10].

The objective of proposing this Multi DNN Model based on Virtual Measurement Method (MDNNM-VMM) is to suffice the process of detecting PDs in the real-world scenario. Obtaining a real-world dataset for PD measurements, especially with regards to those occurring in the substations, is not always an easy task; that is why, publicly available standard datasets could hardly be found.

In addition to being time consuming, the process is costly as well. Repetitive measurements are generally needed to get the exact PD locations. These repetitive measurements generate bigdata, which requires costly arrangements for its storage and processing. Likewise, is the case for already available practical datasets, which require denoising and/or pre-processing prior to becoming useful for the PD detection. Our approach chiefly provides an alternate solution in this scenario, as it just requires a single set of TDOA equations; that is, a single time measurement for the time difference values amongst array sensors for each PD. Furthermore,
in order to take into account, the complete range of possible TDOA values, statistically generated datasets obey a standard distribution (whether normal or uniform). Moreover, different standard deviations, with different ranges, are chosen so as to better replicate the system-related errors and possible time delay errors.

Adding different values for the standard deviations also serves as an indirect way to knowing the strength of the proposed method, as we can figure out that minimum value of Tm which can locate PD source(s) stably. Availability of accurate data is not enough in getting a fast solution; and that is why, some recent researchers [17] working on statistical methods also leveraged iterative methods to locate PD locations. Iterative methods are well-known to be time-inefficient. At this stage, our approach uses a pretrained MDNNM that ensures a quick localization of PD(s). This way the proposed approach ensures cost-effectiveness and time-efficiency.

A. MATHEMATICAL MODELLING FOR PD LOCALIZATION

TDOA measurements are believed to be a simple and most widely accepted means of evaluating the PD location by straightforwardly solving certain simultaneous nonlinear equations. Four sensors are used to get the exact solution for 3-D case; that is, 6 TDOA measurements. Modelling the 3-D case involves complexity, as it considers the arrangement types of antenna array [7].

Fig. 1 shows basic 2-D model for the PD localization. S1-S4 denote four UHF sensors; while, P represents the PD source with rectangular coordinates (x, y), and phasor coordinates (r, θ). Proceeding in the anticlockwise direction, the coordinates for S1, S2, S3, and S4 are (l, w), (-l, w), (-l, -w), and (l, -w), respectively. di is the distance between P and Si; whereas, the corresponding real time of arrival from P to Si is ti, tij0 (as described in Fig. 2) equals tij − tij0, and finally c stands for the speed of light.

For accurate PD detection in 3-D case, 6 TDOA measurements (obtained from 4 sensors) are considered.

3-D modelling involves a certain level of complexity [6], as different array designs are used in addition to uniform circular array (UCA). The rectangular coordinates for 3-D case are given in Table 1, and (1) corresponds to the corresponding equations.

\[d_1 - d_2 = c(t_{120} + \sigma_{e12} + \sigma'_{e12})\]
\[d_2 - d_3 = c(t_{230} + \sigma_{e23} + \sigma'_{e23})\]

Two additional parameters in (1), representing standard deviations \(\sigma_{eij}\) and \(\sigma'_{eij}\) respectively, denote the deviations caused by ‘error of TDOA measurement system’ and ‘supposed additional error for TDOA’. It is a known fact that there is a difference between actual and measured values; thus, \(\sigma_{eij}\) is inserted in the equations. As \(e_i\) represents the measurement time error associated with P and Si, so measured time difference between time of arrival associated with (P, Si) and \((P, S_j)\) is given as \(t_{ij0} + e_{ij}\), instead of simply \(t_{ij0}\). Moreover, here: \(e_{ij} = e_{ij} - e_{ij}\). The measured value (that is, time difference between a pair of sensors) when multiplied with the speed of light gives the distance between P and relevant sensors (because UHF signals have approximately same speed as that of light). The calculated distance difference equations for 2-D case, when solved through iterative methods, output the PD location angles. These are plotted as angle histograms, with their highest peak regarded as the PD source location. PD is located on the hyperbola with the corresponding coordinates.
of two sensors as focuses. Selecting different sensors results in different hyperbolas, and their intersection point is theoretically the PD location. In hyperbolic equations (2), the major axial length and short axial length of hyperbola are respectively $2a_{ij}$ and $2b_{ij}$; while, the focal distance of hyperbola (decided by considering the shape and size of UHF array) is given as $c_{ij}$. The former system-related deviation $\sigma_{ij}$ in (1), caused by $e_{ij}$, gets affected by factors such as: measuring accuracy of arrival time, system’s sampling rate, and sensing ability of UHF sensors. The latter deviation $\sigma'_{ij}$ caused by $e'_{ij}$, which is related to difference in the arrival time, helps in finding out its consequent effects on the PD location.

$$\begin{align*}
y^2 - \frac{(x-I)^2}{a_{14}^2} - \frac{(y-w)^2}{b_{12}^2} &= 1, \quad x^2 - \frac{(y-w)^2}{b_{12}^2} = 1 \\
y^2 - \frac{(x+I)^2}{a_{23}^2} - \frac{(y+w)^2}{b_{23}^2} &= 1, \quad x^2 - \frac{(y+w)^2}{b_{23}^2} = 1 \\
a_{ij} = \frac{|d_i - d_j|}{2} = \frac{c(t_{ij} + \sigma_{ij} + \sigma'_{ij})}{2}, & \quad b_{ij} = \sqrt{c_{ij}^2 - a_{ij}^2} (2)
\end{align*}$$

$e_{ij}$ (causing a deviation in the measurements) has to have an upper bound $T_s$, such that $e_{ij}$ follows a specific Gaussian distribution on $[-T_s, T_s]$, as the naturally occurring distribution is a Gaussian distribution. Although unknown in an actual measurement, but as per assumption, the theoretical value for time difference should be located in the range $[-T_s, T_s]$ (see Fig. 2).

It is noteworthy that a single set of time delay equations obeys the normal distribution. On the other hand, the latter deviation is considered to be a uniform distribution, as supposed by [10], and proved to be stable (especially for $\theta$ coordinate) by [9] also. To obtain a desired number of virtual measurements for the VMM, in addition to studying the impact of ‘error in time difference of arrival’ on the PD location, an error following a uniform distribution ($e'_{ij} \sim U(-T_m, T_m)$) is deliberately added to the measured $t_{ij0} + e_{ij}$. The upper bound of this purposely included simulated error is $T_m$. The theoretical value for time difference must be located in the range $(t_{ij} + e_{ij} - T_m, t_{ij} + e_{ij} + T_m)$, provided it is chosen properly. Following this pattern, for each amongst the ‘N’ PD sources generated randomly, an error-based time sequences’ set is fed as input to the Multi-DNN model, which finally outputs the set of PD source coordinates.

### B. 3-D SENSOR ARRAY CONFIGURATIONS

Designs for 3-D array configurations have previously been detailed by [20], while skipping the uniform linear array (ULA) due to its dramatically changing accuracy [7].

Table 1 lists the coordinates $(x, y, z)$, for each amongst four sensors, within a single array design. Simple formulas related to conversion can be used to switch from rectangular to phasor coordinates or vice versa. All antennas are arranged on a spherical surface, which has a radius R. Moreover, antennas are linearly and vertically polarized, and omnidirectional with identical response.

| Case | $t_{120}$ | $t_{140}$ | $\theta$ range | Quadrant |
|------|-----------|-----------|---------------|----------|
| 1    | -ve       | -ve       | $\theta \in (0^\circ, 90^\circ)$ | 1st      |
| 2    | -ve       | +ve       | $\theta \in (90^\circ, 180^\circ)$ | 2nd      |
| 3    | +ve       | +ve       | $\theta \in (180^\circ, 270^\circ)$ | 3rd      |
| 4    | +ve       | -ve       | $\theta \in (270^\circ, 360^\circ)$ | 4th      |

The designs shown by Table 1 are: uniform circular array ($D_1$), Y-shaped array ($D_2$), and Novel Right Triangle Pyramid-shaped array ($D_3$) [7].

### C. VIRTUAL MEASUREMENT METHOD

Owing to nonlinearities involved while solving TDOA equations, the solution process becomes time inefficient. To overcome this situation, DNN-based techniques are incorporated that are better known for their capability of solving nonlinearities efficiently. But the problem with DNN-based algorithms during their testing phase is that they are sensitive to the quality of input data. If the data provided (evaluated time delay datasets) are not estimated properly, that would lead to false prediction of PDs.

An appreciable effort regarding forecasting of stock price index using DNN has been given by [21], but it depends mainly on the past data. This creates a room for a statistical simulations-based method known as VMM, which fulfils the quality data demand of DNN model(s) with merely a single set of TDOA equations’ set, subsequently making the processing time efficient. Similar efforts based on statistical data evaluation could be seen in [22]–[24], where the authors discuss multi-line outage identification, incipient building faults, and short-circuit diagnosis, respectively.

The data being fed to the DNN model is really critical for the PD localization problem, because it directly affects the output prediction (as illustrated by Fig. 4). Therefore, adding varying amount of additional simulated error, and monitoring the prediction process beforehand, helps realize the stability and robustness of the MDNNM-VMM in a real scenario. The detailed implementation of MDNNM-VMM algorithm is shown in Fig. 3.

Another worth mentioning aspect in Fig. 3 is the assignment of initial value, that is, choosing the values for $t_{ij0}$, which being positive/ negative, decide about the quadrant of PD source. For $D_1$ (UCA), this is explained in Table 2; whereas, the other designs also possess similar characteristics.

### III. DEEP NEURAL NETWORK METHODS FOR THE PD LOCALIZATION

#### A. DIFFERENCE BETWEEN SINGLE AND MULTI-DNN MODELS

Once, when a sequence consisting of six time-delays is generated through VMM, it is fed as input to a pre-trained DNN model that outputs three coordinates $(r, \theta, \Phi)$; and
A. U. M. Zaki et al.: Partial Discharge Localization in 3-D With a Multi-DNN Model

FIGURE 3. Algorithm flowchart for the PD location with MDNNM-VMM.

this happens with four antennas. The model presented in this section is different from the single-DNN based model [16], as it is based on multi DNNs. Despite following [7], this model helps overcome the shortcomings possessed by that DNN-based approach. For satisfying the ‘immense-data-availability’ requirement, instead of averaging the repetitive measurements to get minimum error-based sets of TDOA equations, this method makes the output of VMM method as its input. This way, by overcoming the shortcomings and benefitting from the strengths, the proposed model optimizes the performance of previous methods [7], [16], and [17].

Moreover, the single-DNN model was used just with a single PD source, where the theoretical coordinates were initially known. Furthermore, after getting a single TDOA set of time delays, and generating the desired number of statistical simulations through a DEM, data were fed directly as input to the single pre-trained DNN model. The output was plotted as histograms and kernel density estimation (KDE) plots, separately, for all three coordinates; where, the most frequently occurring peak was regarded as the PD position.

Finally, all three obtained coordinates were compared against the set theoretical coordinates, and error was found for different values of simulated Tm. That method worked well, but was confined to just a single PD. This motivated the authors of the present work to bring an improvement by generating ‘N’ PD points randomly, instead of merely a single point. The scheme for training and testing these ‘N’ randomly generated PDs (with three coordinates each, as the case is 3-D) is given in the forthcoming subsection.

B. TRAINING AND TESTING PHASE STRATEGY

This sophisticated scheme has been borrowed from [7]. A careful approach is needed for all this process because, as per Fig. 4, the input to Multi-DNN model directly affects the output, which is actually the prediction result of PD location. Thus, the more properly the training and testing of model is carried out, the more efficiently this pre-trained-pre-classified-Multi-DNN model would predict the PD location. The scheme follows through as:

1) Randomly generate a training set ‘ytrain’, containing m training samples and three coordinates (r, θ, φ), such that the size of the set is [m,3]. Similarly, generate another set for testing ‘ytest’, containing n testing samples and three coordinates (r, θ, φ), such that the size of this set is [n,3]. Declare both these sets as an output for the training and testing of multi-DNN model, respectively.

2) Transform both the above sets from phasor (r, θ, φ) to rectangular coordinates (x, y, z) with the help of simple 3-D conversion formulas.

3) Calculate the distance ‘di’ between ‘rectangular coordinates of PD source (x, y, z)’ and ‘rectangular coordinates of each amongst the four antennas Sxi, yi, zi’; where: i = 1, 2, 3, and 4. Coordinates for the four antennas are given in the forthcoming subsection.

FIGURE 4. Comparison between pre-trained single-DNN and multi-DNN models.
sensors are taken in compliance with Table 1 for any of the chosen antenna array designs.

4) While using the calculated ‘d’, its corresponding time ‘t’ is calculated by the relation \( t = \frac{d}{c} \), where, ‘c’ is for the speed of light that approximates the UHF signals (as UHF signals have got approximately same speed as that of light).

5) Then, calculate the corresponding time delay sequence ‘\( t_{ij} \)’ between four antennas using the relation \( t_{ij} = t_i - t_j \). This results in two sequences, separately for the train and test data sets- that are regarded as the input for the training and testing of DNN-based model, respectively. The training set has ‘m’ number of samples and six time-difference values, such that x-train input has the size \([m,6]\), that is, x-train \((t_{12}, t_{43}, t_{14}, t_{23}, t_{24}, t_{13})\). Similarly, the testing set has ‘n’ number of samples and six time-difference values, such that x-test input has the size \([n,6]\), that is, x-test \((t_{12}, t_{43}, t_{14}, t_{23}, t_{24}, t_{13})\). Generally, ratio for training samples versus testing samples is 70/30%.

6) Add random simulated error to the TDOA sequence, and generate the desired dataset in accordance with the procedure described for the VMM in section II.

C. CLASSIFIER

The classifier, shown in Fig. 4, has been added between the error carrying time delay sequences and multi-DNNs in order to reduce the error between test (randomly generated) coordinates and predicted coordinates. It is done by following the data characteristics of time delay sequence as shown by Table 2; the basis of which relies on the idea that \( t_{ij} \) being positive or negative decides the quadrant of PD source. Therefore, while taking into account estimated \( \theta \) range initially, the sequences for \( D_1 \), and \( D_2-D_3 \) can respectively be classified as 4 and 3. \( \theta \) range is realized on basis of a trained classifier. The number of DNNs for multi-DNN based model is same as the output size of the classifier. Firstly, each neural network (NN) gets trained with sample datasets having specific \( \theta \) range; and afterward, for the prediction purpose, the most suitable DNN amongst multi-NNs is used. In this manner, the optimal result obtained for PD source has less error as compared to non-classified single-DNN model used by [16].

IV. RESULTS AND ANALYSIS

This section shows the performance of the MDNNM-VMM by carrying out various simulations and comparing the obtained results with some of the latest research works [7] discussing MDNN. Furthermore, it also compares the performance of different array designs by inserting varying values for the simulated error. It is to be noted that the coordinates of PD sources for all simulations were randomly generated with some limits: \( r \in (0m, 60m) \), \( \theta \in (0^\circ, 360^\circ) \), \( \varphi \in (0^\circ, 90^\circ) \). Additionally, \( R \) (that is the radius of sphere introduced in Table 1) for all the array designs is set as 18 m in order to get a smooth comparison.

For the comparison of coordinates, average results based on (3) are used. The set of coordinates for each randomly generated PD source is \((r_{test}, \theta_{test}, \Phi_{test})\), whereas, the set for the predicted coordinates by the pre-classified pretrained multi-DNN model is \((r_{predicted}, \theta_{predicted}, \Phi_{predicted})\).

The deviation between these two sets, with respect to the number of test samples, is called the mean absolute error (MAE). This error, which is based on (3), determines how efficient the proposed model works for different array designs and varying simulated errors.

\[
\begin{align*}
\Delta r &= r_{predicted} - r_{test} \\
\Delta \theta &= \theta_{predicted} - \theta_{test} \\
\Delta \Phi &= \Phi_{predicted} - \Phi_{test} \\
\Delta d &= d(S_{predicted} - S_{test}) \text{OR} \\
\Delta d &= \sqrt{(x_{predicted} - x_{test})^2 + (y_{predicted} - y_{test})^2 + (z_{predicted} - z_{test})^2} \quad (3)
\end{align*}
\]

A. EFFECT OF MEASUREMENT ERROR ON LOCATION ACCURACY

In order to monitor the robustness and efficiency of the MDNNM-VMM, this section fixes the system error \( T_s \) at 2 ns, and varies the simulated error up to a large value of 24 ns. Afterward, the results are compared against the results of [7]. The test case is set at \( N = 200 \) for the simulation studies. Error is applied to two time delays, that are, \( t_{12} \) and \( t_{43} \).

1) EFFECT OF VARYING SIMULATED ERROR \( T_m \) ON INDIVIDUAL PD COORDINATES

Different values of simulated error are considered here. Prior to deeply pondering on the individual PD coordinate behavior (see Figs. 5(a)-5(c)), while comparing the test and predicted PD points, we observe that coordinate \( \theta \) behaves quite linearly compared to \( r \) and \( \Phi \), even for large error values. Less variations being noted, from a small error value to large error values, depict the robustness of coordinate \( \theta \). On the other hand, other two coordinates go on spreading apart as the error values increase; and, certain outliers are also seen for the \( \Phi \) coordinate. For an error value of 2 ns, both these coordinates show a matching linear trend quite similar to that of \( \theta \); but for increasing error values like 10 ns and 24 ns, there is a significant mismatch amongst the test and predicted points. Therefore, it can be concluded that \( \theta \) coordinate has better prediction accuracy compared to the other coordinates.

Fig. 6 gives a deeper insight of individual PD coordinates by illustrating the averaged error results between the test and predicted coordinate values. X-axis represents the number of PD sources that have been predicted by the pre-trained MDNNM; whereas, Y-axis shows the averaged error between the test and predicted PD coordinates. Therefore, the first value for the error would be against the first test PD, and so on. The spike(s) in these illustrations would show a positive error value, provided the difference between the test and predicted coordinates is positive; otherwise, the spike(s) would be on the negative side.
It can be seen in Figs. 6(a)-6(c) that the error value $\Delta r$ for $r$ coordinate, when $T_m = 2$ ns, fluctuates between $\pm 2.5$ m mostly with some peaks going beyond $\pm 5$ m; whereas, when $T_m$ is increased to 10 ns, $\Delta r$ varies in between $\pm 5$ m for most cases with some peaks surpassing $\pm 10$ m. Likewise, for $T_m = 24$ ns, $\Delta r$ has nearly same fluctuation pattern as compared to the 10 ns case, but with a greater number of spikes going beyond $\pm 10$ m. This behavior of $\Delta r$ is comprehensible
TABLE 3. Mean absolute error for the PD coordinates.

| Tm (ns) | ΔrMDNM | rMDNN | θMDNM | θrMDNN | ΦMDNM | ΦrMDNN | dMDNM | dMDNN |
|---------|---------|-------|--------|--------|--------|--------|--------|--------|
| 0       | 2.47    | 2.89  | 4.77   | 4.87   | 4.55   | 4.41   | 2.28   | 2.46   |
| 2       | 2.53    | 3.12  | 4.71   | 5.00   | 4.57   | 4.54   | 2.36   | 2.65   |
| 4       | 3.30    | 5.93  | 4.82   | 5.70   | 4.10   | 4.85   | 2.80   | 3.29   |
| 6       | 3.39    | 4.83  | 5.51   | 6.11   | 4.41   | 5.52   | 2.91   | 4.05   |
| 8       | 4.09    | 5.92  | 6.11   | 6.55   | 5.70   | 6.95   | 3.68   | 5.06   |
| 10      | 4.54    | 6.73  | 7.64   | 7.35   | 6.00   | 7.85   | 3.99   | 5.91   |
| 12      | 4.81    | 7.67  | 6.14   | 7.78   | 5.76   | 8.84   | 4.12   | 7.02   |
| 14      | 4.89    | 8.43  | 7.50   | 7.74   | 5.69   | 9.00   | 4.47   | 7.71   |
| 16      | 5.68    | 9.18  | 6.55   | 8.73   | 5.03   | 10.34  | 4.90   | 8.63   |
| 18      | 6.24    | 9.78  | 7.67   | 8.70   | 6.35   | 10.48  | 5.51   | 9.52   |
| 20      | 6.86    | 10.19 | 8.67   | 9.25   | 6.55   | 11.36  | 5.79   | 9.99   |
| 22      | 7.25    | 10.48 | 7.21   | 9.32   | 7.49   | 12.55  | 6.43   | 11.01  |
| 24      | 7.45    | 11.01 | 7.52   | 8.96   | 8.04   | 13.08  | 6.75   | 12.01  |

When we see the absolute error values for Δr in Table 3, that are 2.53, 4.54, and 7.49, respectively, for Tm values 2 ns, 10 ns, and 24 ns. It is noteworthy that the greater number of spikes for Tm = 24 ns case has brought a significant variation to the Δr value owing to averaging the results with respect to the number of test cases. If the Tm value is small, it indicates that we are simulating a condition where time delay error is small; whereas, when this value is increased, it means time delay error is large. Less time delay error means lesser interference with other sources and/or lesser noise added to the UHF signal approaching the array sensors. Now, when the number of spikes increases, it specifies that the pre-trained MDNNM is not performing very well. It is because of the fact that large time delay error has affected the quality of input data that have to be fed to the MDNNM. This can be understood the other way around, that is, for a certain time delay error, what is the strength of the MDNNM. Here, once again, our proposed model excels any practical PD measurement scheme, because it takes much time and excessive measurements for a practical methodology to know the strength of any scheme; whereas, testing a few random values for Tm can be enough to know the strength of our method.

Figs. 6(d)-6(f) show the variation trend for Δθ. It could be seen in Fig. 6(d) that for Tm = 2 ns, Δθ alters within ±5° range, while some spikes go even beyond ±10°; while, when Tm = 10 ns, Δθ remains approximately within ±10° range alongside rarely occurring spikes approaching ±20°. The trend for Tm = 24 ns is similar to that of previous case, but a few spikes now touch the limits of ±30°. The absolute error values from Table 3 make this behavior clear, as for Tm values 2 ns, 10 ns, and 24 ns, the respective error values are obtained as 4.78°, 6.74°, and 7.52°. A very small difference between absolute error values for Tm = 10 ns and Tm = 24 ns, unlike that for Δr again, shows that θ coordinate has a stable nature compared to that of r coordinate. The behavior of Δθ, compared to that of Δr, depicts that Δθ coordinate is less affected by the increase in time delay error. Less number of spikes means there are a few cases for which the predictions have gone wrong/ less accurate, unlike the case for Δr.

Figs. 6(g)-6(i) show how coordinate ΔΦ changes w.r.t changing Tm values. Its behavior can also be monitored corresponding to the values provided by Table 3. Its performance remains in-between that of the other two coordinates.

Once all coordinates are discussed individually, now Fig. 7 shows their respective variation, along with the corresponding distance coordinate, when Tm is varied up to 24 ns. It is seen that for Tm = 24 ns and N = 200 PDs, the absolute error values Δr, Δθ, ΔΦ, and Δd are, respectively, below 8 m, 8°, 9°, and 7 m. There is a significant percentage decrease in error compared to some recent researches [7] that would be further discussed in the upcoming subsection. It can be seen in Fig. 7 that Δr shows less variations, for increasing Tm values, as compared to Δθ. This can be understood by monitoring the trend of ΔΦ, which shows abnormal behavior for Tm values surpassing 10 ns. Owing to this behavior, Δθ is affected much more than Δr; which is quite obvious, as strangely changing ΔΦ would verily affect Δθ directly. On the other hand, radial distance would not be affected directly. However, the positive aspect being noticed is that the finally computed factor Δd remains stable no matter how Δr, Δθ, and ΔΦ alter. This figure has been explored more in the upcoming section, where for each given Tm value, the corresponding error value has been compared with the error values given by [7].

2) COMPARATIVE ANALYSIS OF MDNNM-VMM AND MDNN

For further analysis of the proposed method, results obtained are compared against those of [7]. For the comprehensibility of figures, our method would be regarded as VMM; while, that of reference [7] would be called as DNN.

While applying a measurement error of 24 ns, the average error values Δr, Δθ, ΔΦ and Δd for the proposed method, compared to the multi-DNN method, respectively see an approximate percentage decrease of 32 %, 24 %, 39 %, and 44 %. Percentage change is obtained by (4); while, absolute error values for both methods being compared can be found in Table 3. The corresponding graphical comparison for all

FIGURE 7. Effect of varying Tm on PD coordinates.
the coordinates can be visualized from Fig. 8 to Fig. 11, where $T_m$ values range from 0-24 ns.

Fig. 8 shows that for a value of $T_m = 0$ (supposedly no simulated error), the absolute error $\Delta r$ between the test and predicted cases for the proposed method is 2.47 m. This error value, without considering the simulated error, is because of the system-related error $T_s$ that has been taken as 2 ns throughout. This has already been discussed in section II (under subsection A). Further, as we go on increasing the $T_m$, $\Delta r$ value slowly increases compared to that of [7]; but this difference approximately doubles when the $T_m$ approaches 14 ns. This shows that how the performance of proposed method is far less affected by the increasing time delay errors. Moreover, no rapid increase in the error values depict that the proposed method is also stable.

Fig. 9, while comparing the $\Delta \theta$ coordinate, shows that the proposed method performs quite better as compared to [7]; however, the difference is not so significant as compared to that of $\Delta r$ (see Fig. 8) and $\Delta \Phi$ (see Fig. 9) coordinates. It means that MDNN [7] was also good at predicting, provided $\theta$ coordinate is considered only.

Fig. 10, unlike Fig. 9, demonstrates that the performance of [7] compared to our approach was poor for $\theta$ coordinate. As the value of $T_m$ reached 16 ns, the error value $\Delta \Phi$ doubled as compared to our approach. This large error value resulted in a bad overall performance for the MDNN [7] method, which is quite apparent from Fig. 11.

While comparing all the coordinates individually for varying simulated error values, our proposed method shows greater prediction accuracy, which results in a little mismatch between the test and predicted PD points. Thus, Fig. 11 demonstrates that how the proposed method brings the mismatch half way down from 12 m to 6 m approximately for even a large simulated error value of 24 ns. It is also observed that, unlike MDNN [7], the proposed method does not rapidly undergo a large deviation (after $T_m$ value 10 ns) between the test and predicted cases, rather it alters slowly; and this fact also endorses the stability and efficiency of this method.

\[
\text{Percentage change} = \frac{\text{AverageError}_{\text{MultiDNN}} - \text{AverageError}_{\text{MDNN-MVMM}}}{|\text{AverageError}_{\text{MultiDNN}}|}
\]  

The discussion until now has been confined to polar coordinate system. Now, Fig. 12 collectively analyzes the behavior of $r$ and $\theta$ coordinates in cartesian coordinate system, while the time delays are subjected to different values of $T_m$. This figure shows an error vector plot between the test and predicted cases, with vectors representing the errors in-between them. As the test PDs have been randomly generated in space, so error vector plot can better visualize how far are
the predicted PDs situating. This plot gives a deep insight of the proposed method, while demonstrating the strength and stability collectively. For $T_m = 2$ ns, we see that error vectors (blue arrows) look like small dots in the central region, which means there is a small difference between the test and predicted PDs. As we move away from the center, the error vectors start getting significant. Similarly, when $T_m = 10$ ns, the error vectors (now red arrows) follow the same trend, but this time the circular spread has increased. This is due to the fact that for this case simulated error value is large, and as a consequence, some predictions go wrong far away from the test points. Likewise, is the scenario for $T_m = 24$ ns, which is demonstrated with black arrows as error vectors.

It is apparent that as we move outwards from the center, we observe error vectors becoming significant. This shows that those PD sources which are far from the antenna array would not be predicted as accurate as those in the vicinity of the array. Moreover, it is also observed that as the $T_m$ goes on increasing, the circular spread of the error vector plot also increases. However, this increase in circular spread is not found to be very significant, which again endorses the fact that the proposed method ensures stability and efficiency while making predictions, even for those cases where time delays are inclusive of large-scale errors.

**B. EFFECT OF DIFFERENT ARRAY DESIGNS ON THE LOCATION ACCURACY**

A method cannot solely meet the challenges of accurate PD detection and localization unless the best array design for its true practical applicability is determined. For this purpose, we studied the comparative behavior of sensor array designs in 3-D (as illustrated by Fig. 13), while applying varying amount of simulated error $T_m$. In order to compare the designs more meaningfully, we have plotted error distributions for all the coordinates. The mean and standard deviation values are demonstrated too; and this is done while the time delays are subjected to varying error values, that is, for $T_m$ ranging from 0-14 ns.

Starting from coordinate $r$, we see that $D_2$ not only has small absolute error as compared to the other two designs, but it also deviates less from its mean value. It means this model is efficient and stable both. On the other hand, $D_1$ has better properties compared to $D_3$; and this could easily be understood by consulting Table 4. $\Delta r$ for $D_1$ has an approximate range 5-7 m, unlike that of $D_3$, which is approximately 7-8 m. Right from the start, where $T_m$ is too small, $D_1$ and $D_3$ as compared to $D_2$ have twofold and threefold value for error. This means that these models are not efficient because a small deviation (or error in the time delay sequences) should not have this much compromising impact on the PD accuracy.

Moving on to the next coordinate $\theta$, it is observed that all designs perform quite well, and the behavior of $D_1$ and $D_2$ is nearly similar. For certain $T_m$ values, $D_1$ has smaller value of $\Delta \theta$ compared to $D_2$, but the error deviation from its mean depicts that $D_2$ experiences less deviations, which is again a sign for its robust nature.

$D_1$, which showed a considerable behavior for $\Delta \theta$, performs poorly for the $\Delta \Phi$, with error values distributed between 10-12°. Here, $D_3$ performs better than $D_2$ as well, and $D_2$ lies in between both the other models (as is generally observed). The overall behavior can be monitored while observing $\Delta d$. It is obvious that $D_2$ tops the other two designs, as its standard error distribution and mean value have small deviation. However, $D_3$ has better performance compared to $D_1$; and this
TABLE 4. Comparative analysis amongst array designs.

| Tm  | 2   | 4   | 6   | 8   | 10  | 12  | 14  |
|-----|-----|-----|-----|-----|-----|-----|-----|
| Δr1 | 4.57| 4.89| 5.84| 6.33| 6.55| 6.71| 6.81|
| Δϕ1 | 2.53| 3.30| 3.39| 4.09| 4.54| 4.81| 4.89|
| Δr2 | 6.64| 6.58| 6.79| 7.07| 7.16| 6.98| 8.03|
| Δϕ2 | 4.89| 4.94| 4.42| 7.94| 5.56| 5.35| 7.63|
| Δr1 | 4.71| 4.82| 5.51| 6.11| 6.74| 6.14| 7.30|
| Δr2 | 6.03| 5.12| 5.76| 6.18| 6.16| 6.72| 8.00|
| Δϕ1 | 10.00| 10.25| 8.26| 9.07| 11.44| 11.25| 12.61|
| Δϕ2 | 4.57| 4.10| 4.41| 5.70| 6.00| 5.76| 5.69|
| Δθ1 | 1.93| 2.88| 2.53| 4.19| 2.89| 3.55| 3.90|
| Δθ2 | 7.33| 7.56| 8.43| 9.73| 10.25| 10.89| 10.98|
| r | 2.36| 2.80| 2.91| 3.68| 3.99| 4.12| 4.47|
| d_1 | 4.50| 4.68| 4.67| 4.91| 5.07| 5.13| 5.69|

V. CONCLUSION

This paper proposes a multi-DNN model (MDNNM) based on a virtual measurement method (VMM) for the accurate and prompt detection of ‘N’ PD sources in 3-D. ‘N’ test PD sources are compared against ‘N’ predicted PD sources; and averaged error for all coordinates are obtained. Important conclusions that have been drawn are as follows:

1) The obtained results depict that the proposed approach is robust and efficient, even for large values of measurement error $T_m$. While applying a measurement error of 24 ns, the average error values $\Delta r$, $\Delta \theta$, $\Delta \phi$, and $\Delta \delta$ for the proposed method, compared to the simple multi-DNN method, respectively see a significant approximate percentage decrease of 32 %, 24 %, 39 %, and 44 %.

2) While relying on single-valued system measurement error $T_s$ (reflected by a Gaussian distribution), the varying $T_m$ (reflected by a uniform distribution) fulfils the requirement of a large dataset for the multi-DNN model.

3) This helps realize the real PD signals’ measurement scenario with merely a single value; which is both time efficient and cost-effective. In addition, changing values of simulated error helps figure out those error values for which the proposed approach produces stable results.

4) After incorporating different sensor array designs and different test sizes, it was revealed that the overall best sensor array design for 3-D is Y-shaped array.

5) If standard deviation-based error is used instead of mean absolute error, comparatively better results are achieved.

6) Error vector plots both in polar and cartesian coordinate systems exhibit overall robustness of the proposed methodology. After a multidimensional study, MDNNM-VMM was found to outperform multi-DNN model for the PD detection and localization.

Future works focusing on PD localization, while aiming to simultaneously improve speed and accuracy, can improve this VMM-based approach by incorporating VMM during the training phase instead of testing phase.

REFERENCES

[1] A. Mas’ud, R. Albarracin, and J. Ardila-Rey, “Artificial neural network application for partial discharge recognition: Survey and future directions,” Energies, vol. 9, no. 8, p. 574, 2016.

[2] B. Barrios and G. Comech, “Partial discharge classification using deep learning methods—Survey of recent progress,” Energies, vol. 12, no. 13, p. 2485, 2485.

[3] V. M. Catterson and B. Sheng, “Deep neural networks for understanding and diagnosing partial discharge data,” in Proc. IEEE Elect. Insul. Conf. (EIC), Seattle, WA, USA, Jun. 2015, pp. 218–221.

[4] Z. Li, L. Luo, N. Zhou, G. Sheng, and X. Jiang, “A novel partial discharge localization method in substation based on a wireless UHF sensor array,” Sensors, vol. 17, no. 8, p. 1909, 1909.

[5] L. Ren, T. Jia, A. Sun, N. Zheng, F. Wu, and L. Luo, “Partial discharge angle of arrival estimation using UHF wireless sensor array,” IOP Conf. Ser.: Mater. Sci. Eng., vol. 486, no. 5, 2019, Art. no. 012031.

[6] H. Chai, B. T. Phung, and S. Mitchell, “Application of UHF sensors in power system equipment for partial discharge detection: A review,” Sensors, vol. 19, no. 5, p. 1029, 1029.

[7] J. Liu, Y. Hu, H. Peng, and A. U. M. Zaki, “Data-driven method using DNN for PD location in substations,” IET Sci., Meas. Technol., vol. 14, no. 3, pp. 314–321, May 2020.

[8] Z. Li, L. Luo, G. Sheng, Y. Luo, and X. Jiang, “UHF partial discharge localisation method in substation based on dimension-reduced RSSI fingerprint,” IET Gener., Transmiss. Distrib., vol. 12, no. 2, pp. 398–405, Jan. 2018.

[9] N. Zhou, L. Luo, G. Sheng, and X. Jiang, “Direction of arrival estimation method for multiple UHF partial discharge sources based on virtual array extension,” IEEE Trans. Dielectr. Electr. Insul., vol. 25, no. 4, pp. 1526–1534, Aug. 2018.

[10] M.-X. Zhu, Y.-B. Wang, D.-G. Chang, G.-J. Zhang, X. Tong, and L. Ruan, “Quantitative comparison of partial discharge localization algorithms using time difference of arrival measurement in substations,” Int. J. Electr. Power Energy Syst., vol. 104, pp. 10–20, Jan. 2019.

[11] M. Ataei and M. Osanloo, “Using a combination of genetic algorithm and the grid search method to determine optimum cutoff grades of multiple metal deposits,” Int. J. Surf. Mining, Reclamation Environ., vol. 18, no. 1, pp. 60–78, Jan. 2004.

[12] B. H. Cheng, R. E. Hudson, F. Lorenzelli, L. Vandenbergh, and K. Yao, “Distributed Gauss-Newton method for node localization in wireless sensor networks,” in Proc. IEEE 6th Workshop Signal Process. Adv. Wireless Commun., Jun. 2005, pp. 915–919.

[13] N. Zhou, L. Luo, J. Chen, G. Sheng, and X. Jiang, “Error correction method based on multiple neural networks for UHF partial discharge localization,” IEEE Trans. Dielectr. Electr. Insul., vol. 24, no. 6, pp. 3730–3738, Dec. 2017.

[14] M. Karimi, M. Majidi, H. MirSaeedi, M. M. Arefi, and M. Oskuee, “A novel application of deep belief networks in learning partial discharge patterns for classifying corona, surface, and internal discharges,” IEEE Trans. Ind. Electron., vol. 67, no. 4, pp. 3277–3287, Apr. 2020.
[15] Y. Bengio, A. Courville, and P. Vincent, “Representation learning: A review and new perspectives,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 35, no. 8, pp. 1798–1828, Aug. 2013.

[16] A. Zaki and Y. Hu, “2D and 3D partial discharge localization with a deep neural network based on a data enhancement method,” CSEE J. Power Energy Syst., 2020.

[17] Y. Hu, Z. Zeng, J. Liu, J. Wang, and W. Zhang, “Design of a distributed UHF sensor array system for PD detection and location in substation,” IEEE Trans. Instrum. Meas., vol. 68, no. 6, pp. 1844–1851, Jun. 2019.

[18] H. Yue, W. Jianwen, and P. Xiangyang, “An improved partial discharge location method in substations: Error reduction based on Monte-Carlo simulation,” in Proc. Conf. Precis. Electron. Meas. (CPEM), Ottawa, ON, USA, Jul. 2016, pp. 1–2.

[19] P. Li, K. Dai, T. Zhang, Y. Jin, Y. Liu, and Y. Liao, “An accurate and efficient time delay estimation method of ultra-high frequency signals for partial discharge localization in substations,” Sensors, vol. 18, no. 10, p. 3439, 3439.

[20] Q. Liu, M.-X. Zhu, Y.-B. Wang, J.-B. Deng, Y. Li, G. J. Zhang, X.-J. Shao, and X.-F. Zhao, “UHF antenna array arrangement optimization for partial discharge direction finding in air-insulted substation based on phased array theory,” IEEE Trans. Dielectr. Electr. Insul., vol. 24, no. 6, pp. 3657–3668, Dec. 2017.

[21] S. Lee and T. Jeong, “Forecasting purpose data analysis and methodology comparison of neural model perspective,” Symmetry, vol. 9, no. 7, p. 108, 2017.

[22] Y. Zhao, J. Chen, and H. V. Poor, “A Learning-to-Infer method for real-time power grid multi-line outage identification,” IEEE Trans. Smart Grid, vol. 11, no. 1, pp. 555–564, Jan. 2020.

[23] B. Jin, D. Li, S. Srinivasan, S.-K. Ng, K. Poolla, and A. Sangiovanni-Vincentelli, “Detecting and diagnosing incipient building faults using uncertainty information from deep neural networks,” in Proc. IEEE Int. Conf. Prognostics Health Manage. (ICPHM), San Francisco, CA, USA, Jun. 2019, pp. 1–8.

[24] J. Tong, H. Wu, Y. Lin, Y. He, and J. Liu, “Fog-computing-based short-circuit diagnosis scheme.” IEEE Trans. Smart Grid, early access, Jan. 8, 2020, doi: 10.1109/TSG.2020.2964805.

**ATTU UL MUNIM ZAKI** was born in Rawalakot, Azad Kashmir, Pakistan, in 1989. He received the B.S. degree in electrical engineering from the University of Engineering and Technology, Lahore, Pakistan, in 2011, and the M.S. degree in electrical engineering from the University of Engineering and Technology, Taxila, Pakistan, in 2015. He is currently pursuing the Ph.D. degree in electrical engineering with Shanghai Jiao Tong University, Shanghai, China. He is on study leave nowadays from the Department of Electrical Engineering, University of Azad Jammu and Kashmir, Muzaffarabad, Pakistan; where he had served as Laboratory Engineer, from 2014 to 2016. His current research interests include the partial discharge detection and location using virtual measurement methods alongside data-driven approaches.

**YUE HU** (Member, IEEE) was born in Yueyang, Hunan, China, in 1978. He received the B.S. and M.S. degrees from Shanghai Jiao Tong University, Shanghai, China, in 2001 and 2007, respectively, and the Ph.D. degree from the Politecnico di Turin, Turin, Italy, in 2010. In 2010, he joined the Department of Applied Electromagnetism, Istituto Nazionale di Ricerca Metrologica, Turin, related to the characteristic of partial discharge (PD) instruments. Since 2011, he has been with Shanghai Jiao Tong University. His current research interests include the PD detection and location, and nanogrid energy systems.

**XIUCHEN JIANG** was born in Shandong, China, in January 1965. He received the B.E. degree in high voltage and insulation technology from Shanghai Jiaotong University, Shanghai, China, in 1987, the M.S. degree in high voltage and insulation technology from Tsinghua University, Beijing, China, in 1992, and the Ph.D. degree in electric power system and automation from Shanghai Jiao Tong University, in 2001. He is currently a Professor with the Department of Electrical Engineering, Shanghai Jiaotong University. His research interests include electrical equipment online monitoring, condition-based maintenance, and automation.