Real-time Fault Localization in Power Grids With Convolutional Neural Networks

Wenting Li, *Student Member, IEEE* , Deepjyoti Deka, *Member, IEEE*,
Michael Chertkov, *Senior Member, IEEE*, Meng Wang, *Member, IEEE*,

***Abstract*—Diverse fault types, fast re-closures and complicated transient states after a fault event make real-time fault location in power grids challenging. Existing localization techniques in this area rely on simplistic assumptions, such as static loads, or require much higher sampling rates or total measurement availability. This paper proposes a data-driven localization method based on a Convolutional Neural Network (CNN) classifier using bus voltages. Unlike prior data-driven methods, the proposed classifier is based on features with physical interpretations that are described in details. The accuracy of our CNN based localization tool is demonstrably superior to other machine learning classifiers in the literature. To further improve the location performance, a novel phasor measurement units (PMU) placement strategy is proposed and validated against other methods. A significant aspect of our methodology is that under very low observability (7% of buses), the algorithm is still able to localize the faulted line to a small neighborhood with high probability. The performance of our scheme is validated through simulations of faults of various types in the IEEE 68-bus power system under varying load conditions, system observability and measurement quality.**

***Index Terms*—Fault Location, Deep Learning, Phasor Measurement Unit (PMU), Real-Time, PMU Placement, Feature Extraction**

I. Introduction

Efficient fault localization is an integral part of the system restoration, and it is necessary for improving power system stability and reliability. Although the status of circuit breakers (CBs) or relays are commonly utilized to locate the fault in the transmission system, many mis-operations of CBs and other devices have been reported to cause system-wide blackouts [1]. As increasing number of phasor measurement units (PMU) and smart meters are installed in power system, and large-scale datasets are generated, it becomes clear that data-driven methods can be used to automatically detect, locate and identify events in the power system.

Prior work on fault localization can be categorized into three groups, albeit with inherent limitations: (1) *impedance-based* methods that often assume the load to be static and are also sensitive to topology changes [2], [3]; (2) *traveling-wave-based* methods that typically require high sampling rates and accuracy of measurements [4]; (3) *existing Artificial Intelligence* methods that are data intense due to measurements with high sampling rates, like 2400 Hz [5], [6] and storage-wise expensive because of large dictionary [7]. Prior works on data-driven methods were also limited in scope due to DC flow model-based assumption with small power variations [8], [9], validity for the single type of faults [7], due to requirement of complete system observability [10] or three phase measurements [11], [12]. Several of such approaches also suffer from low physical interpretability.

Meanwhile, machine/deep learning algorithms have produced encouraging improvements in the fields of computer vision [13], natural language [14] and speech recognition [15], through the selection of correct data-features to use in classification and identification. Motivated by that, we discuss neural network based fault localization methods in power grids utilizing voltage data collected from PMUs. In particular, we show that Convolutional Neural Network (CNN) has much superior fault localization capability when compared with standard methods. The improvements are especially impressive at low system observability. This is important given that the presence of PMUs in current grids is not yet ubiquitous.

In the regime of low observability, the performance of any classifier used for localization greatly depends on the data features containing signatures of considered event's location. In the past, researchers have applied relative voltage angles variations as features to locate line outages through a classifier, but such methods are based on the DC power flow model with small power flow variations [9], clearly not appropriate for detection of the faults. In contrast, we base our newly proposed scheme on the recently reported observations [6], [10], [16] that significant fault currents are sparse and moreover located close to the faulted element of the system. The aforementioned "sparse fault current" phenomenon was explored in [16] under the assumption that PMU observations are available at all the terminal buses, and under partial observability via sparsity-enforcing l_1-regularized approach in [6], [10]. Even though sparsity of the fault current observations and strong correlations between location of the significant fault current and fault location was explored in [6], [10], the methods still suffer from complexity of tuning optimization parameters and non-uniqueness of optimal solutions when the PMU placement is sufficiently sparse.

We claim in this manuscript, by means of empirical experimentation, that the shortcomings of the previous approaches can be overcome through the use of the neural networks. We define the location feature by the estimation of the sparse fault current, which is explained in details in Section II, and train a CNN classifier to learn the correlations between the location

W. Li, M. Wang are with the Dept. of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY. Email: {liw14, wangm7}@rpi.edu.

D. Deka and M. Chertkov are with the Theory Division and the Center for Nonlinear Studies, Los Alamos National Laboratory, Los Alamos, NM. Email: {deepjyoti, chertkov}@lanl.gov

features of a large number of datasets and the fault locations.

Our CNN classifier outputs a fault probability score for all lines, among which the one with the highest probability score suggests location of the fault. We consider both symmetric and asymmetric faults with different impedance in IEEE test networks and show successful location by the classifier under varying load settings and availability of voltage measurements. We also show that the performance of CNN is significantly better than of the traditional classifiers, like Support Vector Machines (SVMs), especially when only a small number of buses are measured. At extremely low observability (7% buses monitored), our classifier is still able to assign the correct faulted line a score that is within the top 2-3 highest ranked lines. Furthermore, we show that lines with the high rank (high probability score) are consistently located within a small neighborhood of the correct fault. Therefore, despite much lower data requirements, our classifier is able to approximately localize the faulted line where others cannot. We relate this remarkably strong performance of the CNN classifier to the right selection of the feature vector based on fault current for the task at hand.

We also boost the fault location approach to solving another, even more challenging problem – designing a greedy algorithm suggesting a sparse PMU placement. We juxtapose the newly introduced CNN-enhancing placement-boosting algorithm to other topology-based placement strategies reported in the literature [17].

To summarize, we propose a data-driven CNN-based scheme which is capable to localize failures in power grids in the challenging case of an extremely low observability. Our work demonstrates that careful selection of proper system-based features and objective-aware placement of PMUs can enable advances in data analytics to significantly improve the performance of detection and estimation tools in power grids.

The organization of the rest of the paper is as follows: in the Section II the feature vector for the problem of fault localization is defined, based on the substitution theory, with proper physical interpretation provided. In Section III and IV, our newly-designed CNN classifier and the PMU placement booster are explained in details. Section V validates the effectiveness of the proposed methods through extensive simulations based on data synthetically generated for the case of the IEEE 68-bus power system. Finally, Section VI contains conclusions and discussions of the path forward.

II. Feature Selection for Fault localization

We consider a power grid of n buses (see Fig. 1) with a single line fault that may either be one of the following: three phase short circuit (TP), line to ground (LG), double line to ground (DLG) and line to line (LL) faults. Assuming that fault detection through known techniques [18] is successful, we are interested in real-time fault localization using PMU measurements collected before and during the fault from a subset of the grid buses. To this end, we propose to use a neural network based fault localization method using power-system features derived from the collected data. As mentioned in the Introduction, selection of right features play a critical role in the success of data-driven classification methods. We now describe the selection of the physical model driven feature vector ψ, first under complete and then under partial system observability.

Note: Vectors are marked as bold font or $\vec{\cdot}$ and the real number and complex number sets are respectively represented by $\mathbb{R}$ and $\mathbb{C}$.

A. Substitution Theory and Features for Full Observability

In the case of a n-bus power system without un-transposed lines[1], we apply the substitution theory [16] to derive the equations related to pre and during-fault system variables. Given that three phase measurements may not be available from all the meters, we use only positive sequence data to represent the quantities.

In the steady state regime prior to the fault, bus voltages $U^0 \in \mathbb{C}^{n\times 1} = [U_1^0, \cdots, U_n^0]^T$, currents $I^0 \in \mathbb{C}^{n\times 1} = [I_1^0, \cdots, I_n^0]^T$ and bus admittance matrix $Y^0 \in \mathbb{C}^{n\times n}$ satisfy the Ohm's law in (1), where the jth entry in the ith row of Y^0 is $Y_{ij}^0, i, j = 1, \cdots, n$, denoting the admittance between the bus i and j,.

$$I^0 = Y^0 U^0 \tag{1}$$

When the line between the bus i and j is faulted at point F, the during-fault admittance matrix, $Y^F \in \mathbb{C}^{(n+1)\times(n+1)}$, with the fault point F as the $(n+1)$th node can be constructed as

$$Y^F = \left[\begin{array}{ccccc|c} Y_{11} & \cdots & \cdots & \cdots & Y_{1,n} & \\ \cdots & \cdots & \cdots & \cdots & \cdots & \\ \cdots & Y'_{ii} & \cdots & Y'_{ij} & \cdots & \\ \cdots & \cdots & \cdots & \cdots & \cdots & \mathbf{y}_{f_1} \\ \cdots & Y'_{ji} & \cdots & Y'_{jj} & \cdots & \\ \cdots & \cdots & \cdots & \cdots & \cdots & \\ Y_{n,1} & \cdots & \cdots & \cdots & Y_{n,n} & \\ \hline & & \mathbf{y}_{f_1}^T & & & \mathbf{y}_{f_2} \end{array}\right] = \left[\begin{array}{c|c} Y' & \mathbf{y}_{f_1} \\ \hline \mathbf{y}_{f_1}^T & \mathbf{y}_{f_2} \end{array}\right], \tag{2}$$

where $Y' \in \mathbb{C}^{n\times n}$ is the during-fault admittance matrix of n buses, $\mathbf{y}_{f_1} = [Y_{1,n+1}^F, \cdots, Y_{n,n+1}^F]^T \in \mathbb{C}^{n\times 1}$ is the admittance between the F and other buses, $\mathbf{y}_{f_2} = Y_{n+1,n+1}^F \in \mathbb{C}$ is the self-admittance of the faulted point F.

During-fault current and voltage $I' \in \mathbb{C}^{n\times 1}, U' \in \mathbb{C}^{n\times 1}$ of the buses, the fault point current and voltage I_f, U_f satisfy the relationship

$$\begin{bmatrix} I' \\ I'_f \end{bmatrix} = Y^F \begin{bmatrix} U' \\ U'_f \end{bmatrix} = \left[\begin{array}{c|c} Y' & \mathbf{y}_{f_1} \\ \hline \mathbf{y}_{f_1}^T & \mathbf{y}_{f_2} \end{array}\right] \begin{bmatrix} U' \\ U_f \end{bmatrix} \tag{3}$$

$$\Rightarrow I' = Y'U' + \mathbf{y}_{f_1} U_f \tag{4}$$

Replacing the Y' by $Y' = Y^0 - Y^u$, where Y^u is a 4-sparse[2] matrix that only has four nonzero entries $Y_{ii}^u = Y_{ii} - Y'_{ii}, Y_{ij}^u = Y_{ij} - Y'_{ij}, Y_{ji}^u = Y_{ji} - Y'_{ji}, Y_{jj}^u = Y_{jj} - Y'_{jj}$, we obtain

$$I' = (Y^0 - Y^u)U' + \mathbf{y}_{f_1} U_f = Y^0 U' - \Delta I^u \tag{5}$$

where the *unbalanced current* $\Delta I^u = Y^u U' - \mathbf{y}_{f_1} U_f$ is a 2-sparse vector with nonzero entries $\Delta I_i^u, \Delta I_j^u$ given in (6).

[1]The un-transposed lines have different mutual impedance between buses and are beyond our analysis.

[2]k-sparsity means there are only k nonzero entries.

Notice that these nonzero entries are just the terminal buses i, j of the faulted line.

$$\Delta I_i^u = (Y'_{ii} - Y^0_{ii})U'_i + (Y'_{ij} - Y^0_{ij})U'_j - Y'_{i(n+1)}U'_{n+1} \quad (6)$$

$$\Delta I_j^u = (Y'_{ji} - Y^0_{ji})U'_i + (Y'_{jj} - Y^0_{jj})U'_j - Y'_{j(n+1)}U'_{n+1} \quad (7)$$

If we define variations of voltage and current as $\Delta U = U' - U^0$, $\Delta I = I' - I^0$ and combine (1) and (5), then their relationships with the pre-fault admittance Y^0 become:

$$Y^0 \Delta U = \Delta I^u + \Delta I \quad (8)$$

The *feature vector* $\psi \in \mathbb{C}^{n\times 1}$ is defined according to (9) in terms of the bus voltages variations ΔU before and during the faults and the admittance matrix Y^0 before the faults

$$\psi = Y^0 \Delta U. \quad (9)$$

Because both imaginary and real parts of ψ can reflect the location, and the imaginary parts show a better performance in a large number of classification experiments, we choose the imaginary part ψ as the feature input to the classifier to avoid unnecessary complication.

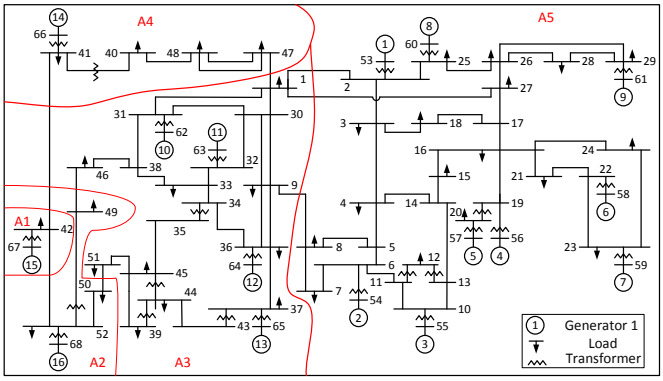

Fig. 1: IEEE 68-bus system with five coherence groups [19].

B. Physical Interpretation of the Features

Physical interpretation of ψ is revealed by the two components in (8). The dominant component is ΔI^u, which is a 2-sparse vector with nonzero values exactly corresponding to the terminal buses of the faulted line. Distribution of the ψ's entries is indicative of the faulted line location.

Consider the line between bus i and j as faulted. The kth ($k \neq i, j$) entry ψ_k is not related directly to the faulted line,

$$\psi_k = \Delta I_k + \Delta I_k^u = \Sigma_{j\in\mathcal{N}_k} Y^0_{kj} \Delta U_j = \Sigma_{j\in\mathcal{N}_k} \Delta I_{kj} \quad (10)$$

where $\mathcal{N}_k$ denotes the neighbor of the bus k, and I_{kj} is the line currents between the bus k and j. Therefore, ψ_k is nonzero if line currents variations in its neighborhood are nonzero. The minor components in ΔI are therefore useful indicators in the neighborhood of the faulted line. (This conjecture will be post-factum validated below.)

Numerical Example: We simulate in the power system toolbox (PST), based on nonlinear models [20], a three phase short circuit fault lasting 0.2 seconds at the line 5-6 in the IEEE 68-bus power system. The feature vector ψ is computed according to (9). The imaginary parts of ΔI^u and $\psi \in \mathbb{C}^{68\times 1}$ shown in Fig. 2 demonstrate that ΔI^u is a sparse vector with nonzero entries corresponding to the two terminal buses (5 and 6) of the faulted line, while ψ_5 and ψ_6 have relatively large values than others. Further, many other buses $(7, 8, 37, 53)$ and $(54-68)$ have nonzero values. These buses are either some PV buses [21] with large current variations or in the neighborhood of the faulted line.

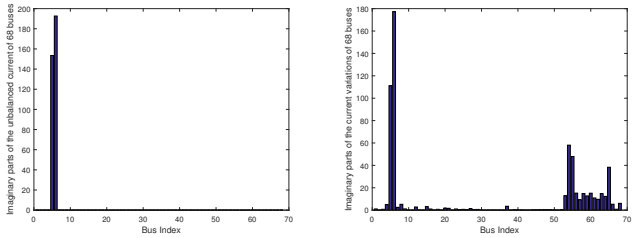

Fig. 2: The imaginary parts of the unbalanced currents ΔI^u(left) and of the feature vector $\psi_i, i = 1, \cdots, 68$ (right) after a three phase short circuit fault on the line 5-6 in Fig. 1

C. Feature Extraction under Partial Observability

Assume that only $s < n$ buses are measured and their pre-fault and during-fault voltages are provided, then we derive at the observed buses, $\Delta\bar{U} = \bar{U}^0 - \bar{U}'$. The feature vector $\bar{\psi} \in \mathbb{C}^n$ of s buses is defined as:

$$\bar{\psi} = \bar{Y}^0 \Delta\bar{U} \quad (11)$$

where $\bar{Y}^0 \in \mathbb{C}^{n\times s}$ denotes the submatrix of the pre-fault admittance matrix. The main reason to select $\bar{\psi}$, and not $\Delta\bar{I}^u$, as the feature vector is that otherwise measurements of all buses need to be known to ensure the nonzero entries of $\Delta\bar{I}^u$ are included, but in reality not all the buses are measured by PMUs. After representing all faults in the dataset by their feature vectors, we label them by their locations. For the system of m lines, we label the dataset into $(m + 1)$ classes with the $(m+1)$th class denoting the normal condition. In the next Section, we examine performance of the classifier.

III. Classification

With features ψ extracted, a number of machine learning classifiers, e.g. support vector machine (SVM) and fully-connected neural network (NN), were tested in [22]. We use a CNN [23] because, as will be shown below, it results in a better classification.

A. CNN classifier

Although there is no uniform way of designing the structure of CNN, and novel architectures are frequently proposed, several basic components are typically considered together for better classification accuracy in a wide range of applications. These components include convolutional, ReLU, Pooling, and fully connected operators. The size of the kernel matrices in these operators and the number of layers are hyper-parameters that are designed to fit the input. In this manuscript we follow the common practical suggestion - to adopt a scheme which has already shown a competitive advantage in other applications. We choose to work with the AlexNet model [13].

1) Architecture: We input the imaginary parts of the extracted feature vectors ψ^j and labels $y^j, j = 1, \cdots, N$, then the CNN optimizes all the parameters layer by layer.

Let the input of the kth convolutional layer ($k = 1, \cdots, l$) be $X_k \in R^{w_k \times h_k \times d_k}$, then the feature vector ψ^j of the jth dataset is the input of the first layer $X_1 = \psi^j$.

$$C_k^j = X_k \otimes W_k, \tag{12}$$

where the output of the kth convolutional layer is C_k^j, which is locally connected with the entries of X_k through kernels $W_k \in R^{c_k, r_k, m_k}$ by the convolution operator $\otimes$ in (12) [24]. These kernels element-wise multiply local parts of X_k and also move with the user-defined stride size over the entire input X_k. To maintain uniform operations in boundary elements, zeros may be padded to X_k.

$$R_k^j = \max(C_k^j, 0) \tag{13}$$

The convolutional layer is followed by the non-linear ReLU activation function in (13), which discards the negative items of C_k^j without changing the size.

$$P_k^j = Pooling(R_k^j) \tag{14}$$

In order to reduce the size of the input at the next layer, the max pooling operator is applied to R_k^j in (14). Kernels in the pooling operator pick the maximum within a small neighborhood of R_k^j and then move to the next neighborhood with a user-defined stride similar to the convolution operator. Likewise, the user also can pad the R_k^j with zeros to make sizes of the neighborhood and of the kernel equal.

The P_k^j is delivered to the next layer as input $X_{k+1} = P_k^j$. Applying these operators form (12) to (14) in all the l layers, the final output P_l^j is vectorized into a long vector $\vec{P}^j$,

$$\bar{y}^j = g(W_o^T \vec{P}^j + B_o) \tag{15}$$

where W_o, B_o are the output kernel and the bias respectively, and $g(\cdot)$ is the softmax function $g(x) = \frac{e^x}{1+e^x}$. $\vec{P}^j$ is fully connected with the output probability $\bar{y}_i^j, i = 1, \cdots, m$ of m lines by (15). The line with the highest probability determines the output class or the fault location.

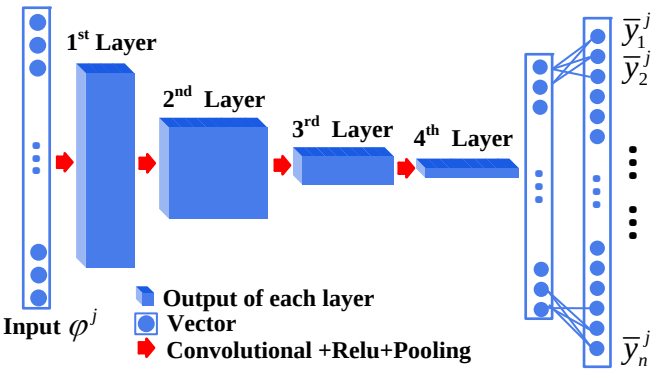

Fig. 3: The structure of our CNN

2) Training Process: We denote the set of all the CNN parameters Θ. The optimal Θ is found by minimizing a loss function. Interpreting the output of different classes related to different lines as probabilities of a fault, the cross-entropy loss function [23] together with a regularization term $\lambda\|\Theta\|_F^2$ to avoid overfitting is the common recipe (16):

$$l(\Theta) = \frac{1}{N} \Sigma_{j=1}^{N} \Sigma_{i=1}^{n} y_i^j \log f_{\Theta,\mathcal{S}}(\bar{\psi}^j) + \lambda\|\Theta\|_F^2 \tag{16}$$

where $\mathcal{S}$ is the set of measured buses, $\bar{\psi}^j$ is defined in (11) with $s \in \mathcal{S}$, $y_i^j \in R^m$ is unity if the label of the j-th dataset is i, and it is zero otherwise, and $\bar{y}_i^j = f_{\Theta,\mathcal{S}}(\bar{\psi}^j)$ is the output probability of CNN for the fault location of the j-th dataset to be at line i. $f_{\Theta,\mathcal{S}}(\cdot)$ denotes functions of (12) $\sim$ (15) parameterized by Θ given the set $\mathcal{S}$ to estimate the probability. λ is the regularization coefficient.

To solve this optimization problem, the stochastic gradient descent method or some of its extensions like Adam [25] and RMSprop [26], are shown to achieve high classification accuracy in a number of tests. Although rigorous convergence proofs of gradient-descent based methods are lacking, there are many techniques that are useful in reducing the effects of initial conditions and also improving the classification accuracy. Examples are "early stop" terminating iterations if the loss function does not decrease for l^* times [27]; "batch normalization" is effective to the issue of covariance shift [28].

In the next Section we describe how PMU placement helps to reduce fault localization error in the case of partial observability.

IV. PMU Placement for Fault Localization under Partial Observability

If the number of PMUs is limited, their correct placement can play a significant role in keeping the quality of the fault localization algorithm described in the preceding Section III. In this Section we propose a greedy algorithm to place K PMUs. PMU placement algorithms discussed in the literature, e.g. [6], [29], [30], are devised to guarantee complete system observability. However, locating faults may work well with some but not necessarily complete observability. Since the accuracy of the fault localization in our case is determined by the loss function of the classifier in (16), we suggest optimizing PMU placement to reduce the loss function (17).

$$\min_{\Theta,\mathcal{S}} \quad l(\Theta, \mathcal{S}) \tag{17}$$

$$\text{s.t. } |\mathcal{S}| = K \tag{18}$$

We propose a data-driven placement algorithm that is aware of both the fault localization and the learning mechanism (optimization of the loss function of CNN). To optimize the PMU placement for fault location, the optimal set $\mathcal{S}$ can be obtained by minimizing loss function (17) satisfying (18), but to find the optimal set $\mathcal{S}$ of size K is an NP-complete problem. Thus we propose an algorithm to greedily increase the number of measured buses until the total number K is reached in Algorithm 1.

Given the total number of measured buses K, this algorithm greedily increases the size of the set $\mathcal{S}$ from the initial set $\mathcal{S}_0$

Algorithm 1 Greedy Algorithm for PMU Placement

Input: $K, y^j, \bar{\psi}^j, d_i, \beta, \mathcal{S}_0, j \in [1, N], i \in [1, n]$
Initialize : $\mathcal{S} = \mathcal{S}_0, l = \infty$
while $|\mathcal{S}|$ is less than K **do**
  for bus $i \notin \mathcal{S}$ **do**
    Let $\mathcal{S}_i = \{\mathcal{S} \cup i\}$, and compute the loss function $l_i = \min_\Theta l(\Theta, \mathcal{S}_i)$
  end for
  $i^* = \arg\min_i(\frac{\beta}{d_i} + l_i)$, where d_i is the degree of bus i, β is a weight parameter.
  if $l_{i^*} < l$ **then**
    $\mathcal{S} = \mathcal{S}_{i^*}$
    $l = l_{i^*}$
  end if
end while
Output: $\mathcal{S}$

one by one until K, where $\mathcal{S}_0$ includes a few buses having the largest degree d_i or being significantly crucial. For each step, the set $\mathcal{S}$ is updated by adding the i^*th bus that minimizes the loss function l_i plus the item of β/d_i. Note that the item β/d_i is added to the loss function to account for the effect of grid topology in determining the selected bus. The weight coefficient $\beta \in (0, 1)$ adjusts the significance of the bus degree and of the loss function to prioritize the buses with large degree. This item takes effect obviously when the set $\mathcal{S}$ is large and the difference of the loss function l_i becomes small. Meanwhile, a number of experimental results show that adding a bus with larger degree tends to have better performances. Based on all of the above, our algorithm tries to enforce the selected buses to achieve a larger degrees by minimizing the loss function augmented with the β/d_i item.

V. Numerical Results

Four types of line faults, including three phase short circuit (TP), line to ground(LG), double line to ground (DLG) and line to line (LL) faults, with different fault impedance are simulated in the IEEE 68-bus power system by PST [20]. In order to mimic the ambient data, active and reactive loads are introduced to generate fluctuations around the initial base condition with random values ϵ drawn from the normal distribution, $\epsilon \sim \mathcal{N}(0, 0.1I)$ where $I \in R^{n\times n}$ is the $n \times n$ identity matrix. These random load fluctuations are simulated by adding random number ϵ to the active and reactive modulation controls through the function *mlsig* and *rmlsig* respectively. The fault impedance is calculated by the negative sequence impedance, Z_2, and the zero sequence impedance, Z_0, [21]. The fault is cleared after 0.1 seconds.

Given voltage measurements and admittance matrix in the normal condition, the complete feature vectors ψ in (9) or partial $\bar{\psi}$ in (11) are computed. The fault location performance is evaluated by the location accuracy rate (**LAR**) η defined in (19).

$$\eta = \frac{\text{The number of faults correctly located}}{\text{total number of faults}} \tag{19}$$

A. Dataset Selection

There are a total of 86 different locations of faults in the system and one normal condition, thus total 87 classes are labeled. We take the data rate of PMU to be 30 samples per second. As mentioned, the initial conditions of each fault in the dataset is varying due to load fluctuations. We assume that active and reactive loads $z \in \mathbb{R}^{2n}$ are drawn from the Gaussian distribution $\mathcal{N}(\mu, \Lambda)$ with mean $\mu \in R^{2n}$ and covariance matrix $\Lambda \in R^{2n\times 2n}$, where the mean value of the load μ is given by the standard dataset and the covariance matrix is defined as $\Lambda = \text{diag}(0.1\boldsymbol{\mu})$. There is a total of 1428 training datasets and 884 (about 221 for each type) testing datasets that cover the four types of faults with zero sequence impedance changing from 0.05 to 0.0001.

B. Structural Parameters of CNN

Table I: The Size of Layers of the Designed CNN

Layer	Operator	Kernel	Stride	Padding	Output
The 1^{st}	Convolution	4 @ 5	1	VALID	4 @ 64
	Max Pooling	2×1	2	SAME	4 @ 32
The 2^{nd}	Convolution	8 @ 5	1	VALID	8 @ 28
	Max Pooling	2×1	2	SAME	8 @ 14
The 3^{rd}	Convolution	8 @ 3	1	VALID	8 @ 12
	Max Pooling	2×1	2	SAME	8 @ 6
The 4^{th}	Convolution	8 @ 3	1	VALID	8 @ 4
	Max Pooling	2×1	2	SAME	8 @ 2
Fully	Vectorize	-	-	-	16
Output	Regression	16 × 87	-	-	87

For this 68-bus power system, a CNN with four convolutional layers is designed to classify the feature vectors. The specific parameters are summarized in the Table I, where "4 @ 5 " denotes that there are four kernels of the size 5 by 1, "4 @ 64" denotes that the output volume is four vectors of the size 64 by 1, and in the column of " Padding", the notations "VALID" and "SAME" mean not padding zeros and padding zeros respectively. The size of the kernels is mainly determined by the size of each layer input.

C. Performance under complete PMU Observability

When the system is fully observable, we compare the LAR (19) of CNN with that of two other machine learning classifiers, including multi-class support vector machine (MSVM) [31], [32] and "fully-connected" neural network (NN). The MSVM classifier is based on the coupling pairwise or "one vs one" method with the radial basis function kernel to find the global solution. NN of two $\sim$ four layers are tested and the two-layer NN is selected as it achieves the optimal performance as discussed later in Fig. 5. The parameter and bias matrices for the first layer of NN are $W^1_{NN} \in R^{68\times 32}, b^1_{NN} \in R^{32}$ and for the second layer are $W^2_{NN} \in R^{32\times 16}, b^2_{NN} \in R^{16}$, and the activation function is ReLU function $f(x) = \max(x, 0)$. The RMSprop optimizer with decay coefficient $\alpha = 0.9$ is employed to train both NN and CNN after comparing with Adam and stochastic gradient descent methods. The "early stop" is applied if the loss function does not increase for 10 consecutive iterations.

Table II: The LAR η (%) of different classifiers on the different faults with various fault impedance

Z_0 (p.u.)	η of MSVM (%)				η of NN (or CNN) (%)			
	TP	LG	DLG	LL	TP	LG	DLG	LL
0.05	100	100	98.6	98.6	100	100	100	100
0.01	100	100	100	99.6	100	100	100	100
0.001	100	100	99.5	94.6	100	100	100	100
0.0001	100	100	94.5	93.6	100	100	100	100

The LAR of MSVM for the four types of faults with different fault impedances is shown in Table II. In general, the LAR is greater than 95%, while that of CNN or NN are all 100%. Although CNN performs slightly better than MSVM, the advantage of CNN so far does not look overwhelming. However, in the next Section, we will see that in the regime of a partial observability CNN outperforms other methods by a large amount.

D. Performance under Partial Observability

Real-world PMU deployment is not ubiquitous. We consider scenarios where only 15% $\sim$ 30% of the buses are covered by PMUs. Under such partial observability, the LAR of the MSVM, two-layer NN and CNN are compared for the four types of faults in Fig. 4. The observed buses for each classifier are selected according to the principles of algorithm 1 using their corresponding loss functions to demonstrate optimal performance. To elucidate the selection of two-layer NN in Fig. 4, performances of NN with different layer depths are compared in Fig. 5, which demonstrates that the two-layer NN has better performance than other schemes.

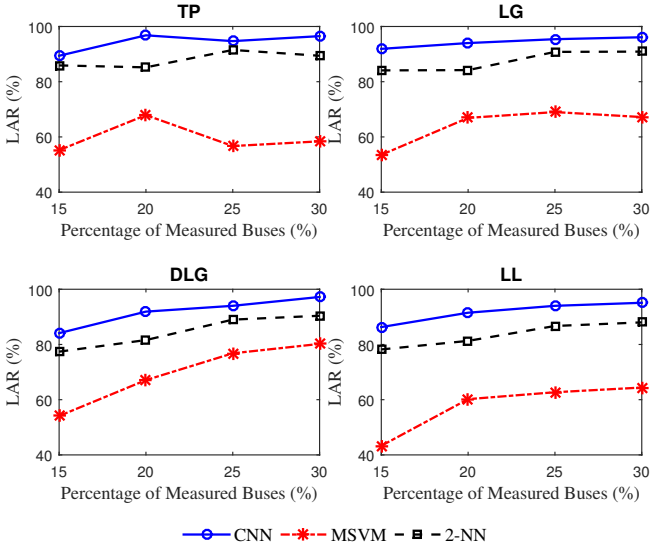

Fig. 4: The LAR of the CNN, MSVM, NN on the four types of faults in terms of different percentage of measured buses

The results in Fig. 4 demonstrate that when only 15% $\sim$ 30% buses are observed, fault localization by CNN is much better for the four types of faults than that shown by the other two classifiers. Observe that when 30% of buses are measured, CNN can reach an impressive fault localization accuracy of more than 95% for faults of the four types. It

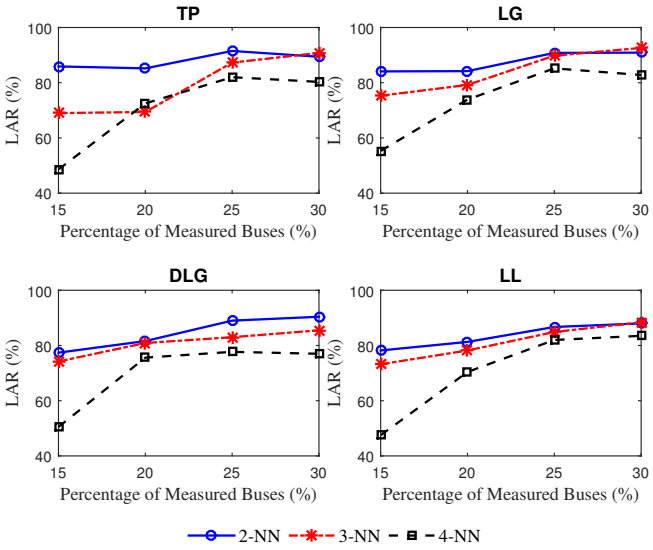

Fig. 5: The LAR of NN classifier with different layer depths in terms of different percentage of measured buses

is worth investigating the performance of the CNN classifier when less than 15% of all buses are measured. In this case one would guess that LAR of CNN cannot be better than 90%. However we observe that even if the CNN does not predict the fault location exactly, it is still able to associate a relatively large probability of failure (though not the largest) to the correct faulted line. To analyze this, we sort the lines according to the output probability $\bar{y}^j$ of CNN in descending order and then record the rank r_j of the correct line of the jth fault. We define a new performance metric "average rank of the correct line" (**ARC**) for the N testing faults as $\bar{r} = \frac{1}{N}\Sigma_{j=1}^{N} r_j$. The ARC indicates how many high-probability lines need to be considered on average to show the correct faulted line. Note that a lower ARC reflects better average performance with the ARC of exact localization being 1.

E. The ARC of CNN under $\leq$ 15% of nodal observability

Table III: The ARC of CNN for different type of faults when the ratio of measured buses is less than 15%

Measured Ratio	TP	LG	DLG	LL
7%	1.32	1.48	1.92	1.56
10%	1.38	1.28	1.66	1.54
15%	1.38	1.23	1.57	1.54

The ARC of the four types of faults is shown in the Table III when no more than 15% of buses are measured. It is significant that the ARC for all types of faults is less than 3 when only 7% to 15% of buses are measured. This observation suggests that despite the low PMU coverage, the operator needs to check only a few lines to identify the fault. Crucially, as discussed next, under low PMU coverage, CNN is also able to localize the fault to a small graphical neighborhood of its true location.

F. Neighborhood property of high probability lines

The lines with high output probability $\bar{y}_i^j$ demonstrate neighborhood property in Fig. 6, where the line between bus

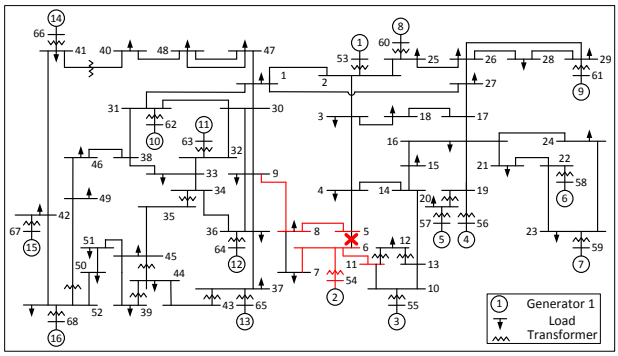

Fig. 6: The lines of top-5 high probability (above) from CNN are marked in red IEEE 68-bus power system

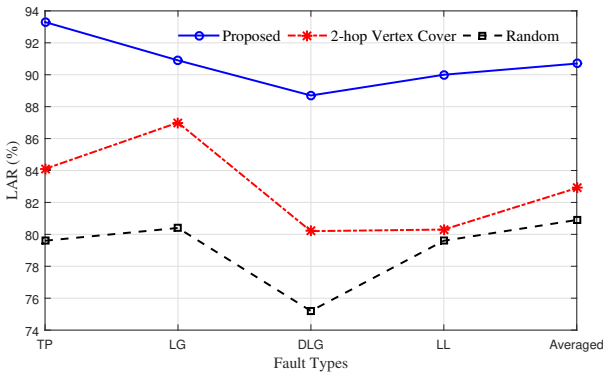

Fig. 7: The LAR of CNN when 12 measured buses are selected by three algorithms

5 and 6 has a three phase short circuit fault. All lines are sorted according to $\bar{y}_i^j$ from high to low, then those with the top-5 probabilities, marked as red, are in the neighborhood of the faulted line. Furthermore, we have verified that this neighborhood property is not a special case for this fault but extends to the majority of the tested faults. Moreover, this neighborhood property is determined by the feature vector in (10) and as such also applies to other tested classifiers, e.g. NN. Since, $\psi_k(k \neq i, j)$, defined in Section II-B as the total line currents in the neighborhood of bus k, lines in the neighborhood of the fault are identified with high probability.

Low ARC and neighborhood localization properties appear very useful to guide initial dispatch of a recovery/maintenance crew. Moreover, it should also be advantageous to use these features to determine the order of triggering relays or circuit breakers automatically for protection in the post-fault grid. We plan to study these directions in the future.

G. Comparison with other PMU placement algorithms

In this Subsection, we discuss the performance of the algorithm 1 for PMU placement. The proposed algorithm is compared with the "2-hop Vertex Cover (VC)" and the Random placement algorithms. The "2-hop VC" is a topology-based algorithm for PMU placement [17]. It places PMUs on a set of buses such that each edge in the graph is at-most two hops away from a PMU. The baseline of Random algorithm selects arbitrarily s buses. The LAR for faults of the four tested types is compared in Fig. 7 where the measured buses are suggested by the three placement algorithms.

As there are at least 12 buses that can satisfy the objective of "2-hop vertex cover" for this 68-bus power system, these three algorithms are compared when $s = 12$. The 12 buses selected by the Random algorithm include $[31, 3, 65, 46, 43, 28, 15, 44, 23, 58, 9, 57]$, one solution of the 2-hop VC algorithm is obtained by solving a linear programming approximating the 2-hop VC formulation, and the selected buses are $[3, 6, 13, 19, 23, 26, 30, 31, 36, 40, 44, 52]$, and the 12 buses selected by the method proposed in the manuscript are $[1, 9, 16, 30, 36, 23, 42, 61, 51, 57, 6, 37]$. Compared with the Random algorithm, the improvements of the proposed algorithm for different types of faults varies, however it always shows about 10% improvement in average over the other methods. The 2-hop VC method also has higher LAR than that of the Random algorithm, however it is still lagging behind the proposed algorithm showing the average improvement of 8%.

H. Sensitivity to noise

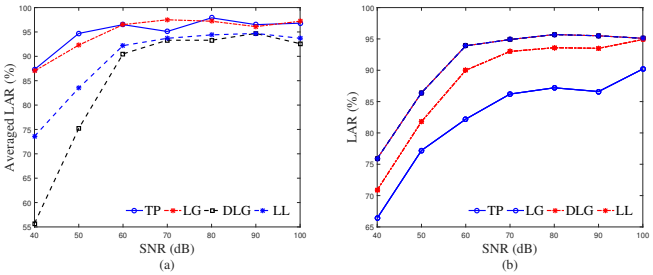

Fig. 8: (a) The LAR for different types of faults when 30% of buses measured with different SNR; (b)The Averaged LAR of all types of faults when 20% $\sim$ 30% of buses measured with different SNR

The IEEE Standard C37.118 only defines the measurement accuracy but does not specify the signal-noise-ratio(SNR) of PMU measurements [33], and the SNR of PMUs in different regions can vary. We select the experimental range of SNR from 40 dB to 100 dB [18], [34], [35] to test our method. Gaussian noise of the same SNR is added both to the training and testing datasets. The structure of the CNN is the same as before but the hyper-parameter decay coefficient, α, is changed from 0.9 to 0.7 in the noisy regime. Other parameters are kept the same.

The (a) of Fig. 8 demonstrates the LAR with different SNR when 30% of buses are observed, and the (b) indicates the average LAR of all types of faults when 20% $\sim$ 30% of buses are observed. Results in (a) indicate that the sensitivity of different types of faults to noise is different, and the three phase short circuit faults are relatively more robust to the noise. When SNR is higher than 60 dB, LAR for faults of all the types can achieve 90% or higher. The (b) reveals that, as expected, when more buses are measured the robustness to noise can be strengthened. Furthermore, when SNR is higher than 60 dB, the influence of the noise is contained and the performance does not improve or degrade noticeably.

VI. Conclusions

This manuscript builds a data-driven CNN classifier applied to the problem of fault localization under complete and partial PMU measurement availability. The performance of CNN is validated on IEEE test system, and it is shown to be better than of other data-driven approaches. The improvement is especially significant when PMUs are limited to a small number of (less than 30%) buses. At low observability, the CNN is able to localize the fault to a small region around the actual faulted line. The success is related to a proper choice of the input features for the learning algorithm. We also present a location and learning aware PMU placement scheme which maximizes performance of the CNN classifier compared to other placement options such as random and vertex cover based ones. The CNN is verified on faults of various types, load settings, measurement noise levels and system observability to benchmark its performance.

In the future, we will extend this work not just to locate the faulted line but also identify exact location of the fault along the line. Furthermore, we are interested in designing mitigation and protection strategies that take into account the data-driven approach proposed here. Testing the methodology on real-data (as opposed to synthetically generated data) is another direction for our future work.

Acknowledgement

The authors acknowledge the support from the Department of Energy through the Grid Modernization Lab Consortium, and the Center for Non Linear Studies (CNLS) at Los Alamos.

References

[1] D. Novosel, G. Bartok, G. Henneberg, P. Mysore, D. Tziouvaras, and S. Ward, "Ieee psrc report on performance of relaying during wide-area stressed conditions," *IEEE Trans. Power Del.*, vol. 25, no. 1, pp. 3–16, 2010.

[2] A. A. Girgis, C. M. Fallon, and D. L. Lubkeman, "A fault location technique for rural distribution feeders," *IEEE Trans. Ind. Appl.*, vol. 29, no. 6, pp. 1170–1175, 1993.

[3] M. Farajollahi, A. Shahsavari, and H. Mohsenian-Rad, "Location identification of distribution network events using synchrophasor data," in *Proceedings of North American Power Symposium (NAPS)*, 2017.

[4] F. Han, X. Yu, M. Al-Dabbagh, and Y. Wang, "Locating phase-to-ground short-circuit faults on radial distribution lines," *IEEE Trans. Ind. Electron.*, vol. 54, no. 3, pp. 1581–1590, 2007.

[5] S. Azizi and M. Sanaye-Pasand, "A straightforward method for wide-area fault location on transmission networks," *IEEE Trans. Power Del.*, vol. 30, no. 1, pp. 264–272, 2015.

[6] M. Majidi, M. Etezadi-Amoli, and M. S. Fadali, "A sparse-data-driven approach for fault location in transmission networks," *IEEE Trans. Smart Grid*, vol. 8, no. 2, pp. 548–556, 2017.

[7] H. Jiang, J. J. Zhang, W. Gao, and Z. Wu, "Fault detection, identification, and location in smart grid based on data-driven computational methods," *IEEE Trans. Smart Grid*, vol. 5, no. 6, pp. 2947–2956, 2014.

[8] H. Zhu and G. B. Giannakis, "Sparse overcomplete representations for efficient identification of power line outages," *IEEE Trans. Power Syst.*, vol. 27, no. 4, pp. 2215–2224, 2012.

[9] M. Garcia, T. Catanach, S. Vander Wiel, R. Bent, and E. Lawrence, "Line outage localization using phasor measurement data in transient state," *IEEE Trans. Power Syst.*, vol. 31, no. 4, pp. 3019–3027, 2016.

[10] G. Feng and A. Abur, "Fault location using wide-area measurements and sparse estimation," *IEEE Trans. Power Syst.*, vol. 31, no. 4, pp. 2938–2945, 2016.

[11] M. Majidi, A. Arabali, and M. Etezadi-Amoli, "Fault location in distribution networks by compressive sensing," *IEEE Trans. Power Del.*, vol. 30, no. 4, pp. 1761–1769, 2015.

[12] M. Majidi, M. Etezadi-Amoli, and M. S. Fadali, "A novel method for single and simultaneous fault location in distribution networks," *IEEE Trans. Power Syst.*, vol. 30, no. 6, pp. 3368–3376, 2015.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[14] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[15] A.-r. Mohamed, G. E. Dahl, G. Hinton *et al.*, "Acoustic modeling using deep belief networks," *IEEE Trans. Audio, Speech & Language Processing*, vol. 20, no. 1, pp. 14–22, 2012.

[16] Q. Jiang, B. Wang, and X. Li, "An efficient pmu-based fault-location technique for multiterminal transmission lines," *IEEE Trans. Power Del.*, vol. 29, no. 4, pp. 1675–1682, 2014.

[17] D. Deka and S. Vishwanath, "Pmu placement and error control using belief propagation," in *Smart Grid Communications (SmartGridComm), 2011 IEEE International Conference on*. IEEE, 2011, pp. 552–557.

[18] L. Xie, Y. Chen, and P. R. Kumar, "Dimensionality reduction of synchrophasor data for early event detection: Linearized analysis," *IEEE Trans. Power Syst.*, vol. 29, no. 6, pp. 2784–2794, 2014.

[19] G. Rogers, *Power system oscillations*. Springer Science & Business Media, 2012.

[20] J. H. Chow and K. W. Cheung, "A toolbox for power system dynamics and control engineering education and research," *IEEE Trans. Power Syst.*, vol. 7, no. 4, pp. 1559–1564, 1992.

[21] P. Kundur, N. J. Balu, and M. G. Lauby, *Power system stability and control*. McGraw-hill New York, 1994, vol. 7.

[22] C. Robert, "Machine learning, a probabilistic perspective," 2014.

[23] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[24] S. Y. Fei-Fei Li, Justin Johnson. (2018) Convolutional neural networks. [Online]. Available: http://cs231n.github.io/convolutional-networks/

[25] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[26] G. Hinton, N. Srivastava, and K. Swersky, "Neural networks for machine learning-lecture 6a-overview of mini-batch gradient descent," 2012.

[27] Y. Bengio, "Practical recommendations for gradient-based training of deep architectures," in *Neural networks: Tricks of the trade*. Springer, 2012, pp. 437–478.

[28] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[29] Y. Zhao, A. Goldsmith, and H. V. Poor, "On pmu location selection for line outage detection in wide-area transmission networks," *arXiv preprint arXiv:1207.6617*, 2012.

[30] E. Abiri, F. Rashidi, T. Niknam, and M. R. Salehi, "Optimal pmu placement method for complete topological observability of power system under various contingencies," *International Journal of Electrical Power & Energy Systems*, vol. 61, pp. 585–593, 2014.

[31] S. Pöyhönen, A. Arkkio, P. Jover, and H. Hyötyniemi, "Coupling pairwise support vector machines for fault classification," *Control Engineering Practice*, vol. 13, no. 6, pp. 759–769, 2005.

[32] C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 415–425, 2002.

[33] K. E. Martin, "Synchrophasor measurements under the ieee standard c37. 118.1-2011 with amendment c37. 118.1 a," *IEEE Trans. Power Del.*, vol. 30, no. 3, pp. 1514–1522, 2015.

[34] M. Brown, M. Biswal, S. Brahma, S. J. Ranade, and H. Cao, "Characterizing and quantifying noise in PMU data," in *Power and Energy Society General Meeting (PESGM), 2016*. IEEE, 2016, pp. 1–5.

[35] W. Li, M. Wang, and J. H. Chow, "Real-time event identification through low-dimensional subspace characterization of high-dimensional synchrophasor data," *IEEE Trans. Power Syst.*, vol. 33, no. 5, pp. 4937–4947, 2018.