# Advanced Smart Grid Monitoring: Intelligent Cable Diagnostics using Neural Networks

Yinjia Huo, Gautham Prasad, Lutz Lampe, and Victor C. M. Leung
The University of British Columbia, Canada. Email: yortka@ece.ubc.ca

*Abstract*—Monitoring and control of network constituents are integral aspects of the smart grid. In this paper, we present a technique for monitoring one such network asset, the underground power cables, which are prone to degradation and damages, resulting in possible power outages. We propose an intelligent cable diagnostics solution using neural networks to determine the health of power cables to predict and prevent eventual faults. To this end, we reuse the communication channel state information inherently estimated by power line modems that are envisioned to enable smart grid communications. We advance the state-of-the-art machine learning based cable health monitoring techniques to present an automated diagnostics procedure using neural networks, which eliminates the need to manually extract features during operation. We demonstrate the architecture of our designed feed-forward neural network, the procedures involved in training, validating, and testing data, and the algorithms we use to train our machines. We evaluate our solution for medium voltage distribution network settings and show through simulation results that our method provides accurate diagnosis in detecting, locating, and assessing cable degradations.

## I. INTRODUCTION

Power line communication (PLC) has conventionally been conceived to enable reliable communication over existing infrastructure for indoor multimedia networks [1], outdoor smart grid communication [2, Ch. 9], and a host of other specific applications where power cables are reused for communication purposes, such as in vehicular networks [2, Ch. 10] and energy management systems [3], [4]. Recent advancements have further reused the ability to transmit high frequency communication signals over electrical power lines to extract information about the communication media, for network tomography [5], [6], smart grid fault diagnostics [7], anomaly detection [8], intrusion detection [9], and power cable health assessment [10]. In this paper, we focus on the latter, i.e., cable diagnostics using PLC modems, to present an intelligent health assessment solution.

Although cables are considered to be safe, relatively immune to accidental contact, and aesthetically palatable, one of the reasons hindering utilities from transitioning from overhead lines to underground cables is the challenge of detecting, locating, and rectifying faults that result from natural aging and/or external activities [11]–[13]. Power outages resulting from such faults lead to potentially dangerous situations and incur substantial economic losses [14]–[16]. Therefore, a rich body of work has focused on cable diagnostics targeted at

examining cable health [17, Ch. 6], including techniques that reuse PLC modems for this purpose [8], [10], [18]–[21].

PLC-based cable diagnostics methods present several benefits compared to legacy solutions by way of enabling online monitoring without requiring any portion of the network to be shut-down, providing a low-cost solution by reusing existing grid components, i.e., PLC modems, and allowing for continuous monitoring. State-of-the-art PLC-based solutions use machine learning (ML) techniques to further achieve automated diagnosis without requiring any manual intervention [10], [18], [19]. These methods reuse the intrinsically estimated channel state information (CSI), and examine channel behavior deviations to detect and locate cable abnormalities. They train dedicated machines to specifically distinguish such CSI changes-of-interest from those caused by other network activities, e.g., load variations. In this paper, we use such a method as our benchmark, and propose enhancements using neural networks to simplify the backend process involved in assessing the cable health. In particular, prior arts [10], [19] examined the use of support vector machines (SVMs) and boosting techniques for diagnostics. A crucial aspect of such a procedure is that the performance is highly dependent on the features extracted from the raw data. For instance, using peak locations and amplitudes from the estimated channel impulse response were found to be valuable in detecting and locating a possible localized cable degradation [10]. Two major drawbacks are evident from relying on such a procedure. First, the operation demands extracting the right set of features, which in turn requires prior domain knowledge, and therefore restricts layman accessibility of the solution. Additionally, the extracted features are tailored for one diagnostics task considered, a specific network topology, and a particular infrastructure setting, which therefore restricts its scalability across configurations. To alleviate these constraints, we investigate the use of neural network (NN) methods, which can match the performance of the state-of-the-art, without requiring any manual feature extraction process. We also draw inspiration from prior non-PLC works that have developed NN-based methods for fault diagnostics in transmission and distribution systems [22]–[24].

We design our NN framework to sequentially detect, locate, and assess localized and homogeneous types of cable degradation. We evaluate the performance of our solution under typical medium voltage (MV) distribution network settings and present simulation results of our performance accuracy. We show that, unlike prior arts, our method achieves successful cable diagnostics using only the CSI obtainable from legacy half-duplex PLC modems, i.e., using end-to-end channel fre-

quency response, in lieu of also requiring the reflection channel information available from PLC modems enabled with the in-band full-duplex (IBFD) operation [25], [26].

## II. CABLE DIAGNOSTICS USING NEURAL NETWORKS

In this section, we propose our neural networks based cable diagnostics procedure beginning with a brief summary of the required preliminaries.

---

**Algorithm 1** A generalized ML-based cable diagnostics.

---

1: **Task$-1$**: Identify degradation (classification)
2: **if** degradation exists **then**
3:    **Output**: *Cable degradation exists*
4:    **Task$-2$**: Identify type of degradation (classification)
5:    **if** degradation is localized **then**
6:       **Output**: *Localized degradation exists*
7:       **Task$-3$**: Detect location (regression)
8:       **Output**: degradation location
9:       **Task$-4$**: Predict severity (regression)
10:       **Output**: extent of damage
11:    **else**
12:       **Output**: *Homogeneous degradation exists*
13:       **Task$-5$**: Predict severity (regression)
14:       **Output**: extent of damage
15:    **end if**
16: **else**
17:    **Output**: *Cable is intact*
18: **end if**

---

### A. Preliminaries

The foundational ML diagnostics framework that we previously developed is summarized in Algorithm 1 [10]. We sequentially detect the presence of a degradation, assess the type of damage, and determine its severity and/or location depending on whether the degradation is homogeneous or localized along a section of the cable. While detecting and locating cable damage can assist in deploying targeted corrective measures, assessing the extent of degradation is critical in estimating the remaining life expectancy of the cable and in predicting a possible imminent in-service failure. To this end, we use supervised machine learning for both classification and regression tasks by training separate machines beforehand for each of the diagnostics tasks.

As part of the training process, we use synthetically generated PLC channel transfer functions modeled using the bottom-up approach [27], [28]. This allows us to emulate the signal propagation along any portion of the cable with the flexibility to introduce degradation of arbitrary severity. The overall transfer function can then be computed as a concatenation of these individual sections. We use the transmission line (TL) theory for computing the channel frequency response (CFR) by viewing every section of the cable with the same degradation severity as a uniform line with electrically small cross-sectional dimensions, in which case, the PLC
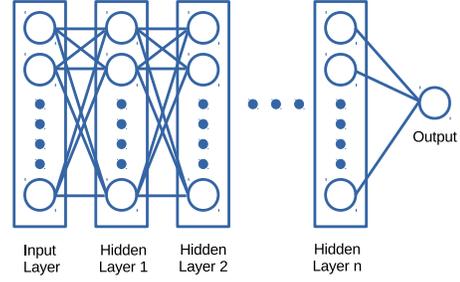


Fig. 1. Architecture of a feed-forward neural network.

signal travels in the quasi-transverse-electromagnetic (quasi-TEM) propagation mode. We then solve the TL equations for every section of the cable, where the per-unit-length (PUL) parameters vary depending on the extent of cable damage.

### B. Feed-Forward Neural Networks

For a first investigation of applying NNs for cable diagnostics, we use the architecture of a feed-forward neural network (FFNN), since its structure is ideally suited for the various supervised ML tasks described in Algorithm 1. The fully connected architecture of an FFNN is shown in Fig. 1, with every two neurons in adjacent layers connected with a weighted link. We train a dedicated NN for each ML task and feed the synthetically generated CFR at the input layer for training the machine. When the trained machine is deployed in the real-world, the PLC modems are fed with the inherently estimated CFRs for the prediction.

*1) Structure:* Our designed FFNN consists of $n$ hidden layers between the input and output layers. We denote the input vector of CFR as $\vec{H}$, and the output as $y$. Note that for all tasks we consider, which are detecting a degradation, determining the type of degradation, locating its position, and estimating its severity, $y$ is always a scalar. Further, by denoting the $i$th hidden layer ($1 \leq i \leq n$) as the vector $\vec{p}_i$, we have

$$\vec{p}_1 = \sigma_1(\mathbf{W}_1 \vec{H} + \vec{\theta}_1), \tag{1}$$

$$\vec{p}_i = \sigma_i(\mathbf{W}_i \vec{p}_{i-1} + \vec{\theta}_i), \qquad 1 < i \leq n, \tag{2}$$

where $\mathbf{W}_i$ is the weight matrix of the links, $\vec{\theta}_i$ is the bias vector, and $\sigma_i$ is the activation function, all between the layers $i$ and $i-1$. Consequently, the output is

$$y = \sigma(\vec{w} \cdot \vec{p}_n + \theta), \tag{3}$$

where $\vec{w}$ is the weight vector, $\theta$ is the bias, and $\sigma$ is the activation function, all between the hidden layer $n$ and the output, and $\cdot$ is the scalar product operator.

*2) Parameters:* We determine the trainable parameters of the FFNN, $\mathbf{W}_i$, $\vec{w}$, $\vec{\theta}_i$, and $\theta$ during training to minimize a loss function $\eta$. For the regression tasks, we select the mean squared error as the loss function,

$$\eta_{\text{MSE}} = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2, \tag{4}$$

where $N$ is the number of samples used, and $y_i$ and $\hat{y}_i$ are the known and predicted output values for the $i$th sample, respectively. On the other hand, for the classification tasks, the cross entropy is usually adopted as the loss function [29, Eq. 2.112]. However, the conventional labeling scheme for classification results in the convergence of the training algorithm to a local minimum for our tasks. To this end, we divide the classification task into two stages, beginning with a regression that we perform with temporary labeling and followed by a thresholding-based classification, as we illustrate in Section III-B1. Next, before we initiate the training procedure, we specify the hyperparameters, i.e., $\sigma_i$, $\sigma$, $n$, and the size of each hidden layer $i$, which encode our prior beliefs of the structure of the machine to be trained [29, Ch. 3].

*Strategies for Setting Hyperparameters:* To determine $n$, we consider the multi-layer FFNN where the initial layers extract low-level features, and as the layers approach the output, the abstraction is higher and the number of neurons is fewer. The gradually reducing size of the hidden layers restricts over-fitting and empowers the FFNN with superior generalization ability, i.e., higher performance accuracy on unseen test data, since the information required to minimize the loss function is confined within a small number of neurons towards the output. Next, we choose an activation function to introduce non-linearity and improve the representation power of the FFNN. The common choice for $\sigma$ is the linear function for regression to allow for a wide range of output values, and the sigmoid function,

$$\sigma_{\text{sigmoid}}(x) = \frac{\exp(x)}{1 + \exp(x)}, \tag{5}$$

for classification to allow $y$ to represent the class probability ranging from $0$ to $1$. Within the hidden layers, we also use two other activation functions,

$$\sigma_{\text{tanh}}(x) = \frac{\exp(x) - \exp(-x)}{\exp(x) + \exp(-x)} \tag{6}$$

$$\sigma_{\text{relu}}(x) = \max(0, x). \tag{7}$$

In particular, we choose $\sigma_{\text{relu}}$ to simultaneously allow for certain degree of non-linearity and avoid the issue of vanishing or exploding gradient [30].

*3) Training:* We use batch normalization (BN) techniques to accelerate the training speed and to reduce the reliance of the NN performance on the initialization as well as the random dropout [31]. Specifically, we normalize the activations of the $i$th hidden layer and then apply the linear transformations with coefficients $\overrightarrow{\gamma}_i$ and $\overrightarrow{\beta}_i$ as

$$\overrightarrow{p}_i^* = \overrightarrow{\gamma}_i \cdot \left( \frac{\overrightarrow{p}_i - \text{mean}(\overrightarrow{p}_i)}{\text{std}(\overrightarrow{p}_i)} \right) + \overrightarrow{\beta}_i, \tag{8}$$

where $\text{mean}(\cdot)$ and $\text{std}(\cdot)$ are the functions to compute the mean and standard deviation, respectively, over the batch-wise number of samples empirically. We then apply the weights and the activation functions to obtain the activations of the next hidden layer,

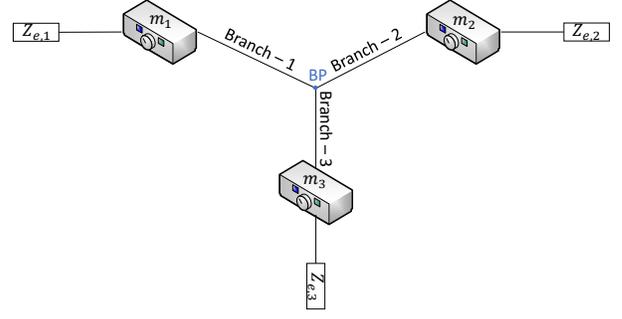$$\overrightarrow{p}_{i+1} = \sigma_{i+1}(\mathbf{W}_{i+1} \overrightarrow{p}_i^*), \tag{9}$$



Fig. 2. A generic sub-network of an MV distribution network. (BP: Branch point)

to replace (2) when BN is applied to the activations of the hidden layer $i$. This procedure introduces new learnable parameters, $\overrightarrow{\gamma}_i$ and $\overrightarrow{\beta}_i$, which are differentiable and can be learned together with the other learnable parameters during the training process. Various algorithms have been designed for such an optimization process based on iterative back-propagation [32]. For each iteration, we use a certain portion, i.e., a batch, of the total available training data to update the trainable parameters. After multiple iterations, the back-propagation algorithm completes one epoch after seeing all of the training data once. We run several such epochs until we obtain a satisfactory performance or until the validation performance ceases to improve. We use the RMSProp algorithm in our design, that is built on the back-propagation algorithm and uses an adaptive learning rate [33, Ch. 11].

## III. RESULTS

In this section, we present simulation results to demonstrate the performance of our proposed solution. Along the same lines as the state-of-the-art technique [10], we evaluate our method for diagnosing water-tree degradation that commonly impacts underground MV cross-linked polyethylene (XLPE) cables [11], [34].

### A. Simulation Settings

*1) Network Topology:* We evaluate our solution for a generic Y-network topology shown in Fig. 2, with the three branches of equal length of $500$ m. Given the transmission range of BB-PLC modems [35], a larger MV network consists of several building blocks of such Y-sub-networks. Each of the three PLC modems in our considered Y-network, $m_k$, $k = \{1, 2, 3\}$, are also connected to an extended equivalent impedance, $Z_{\text{e},k} \sim \mathcal{U}(0, 50) + j\mathcal{U}(-50, 50)$ $\Omega$ where $\mathcal{U}(a, b)$ denotes a uniform distribution between $a$ and $b$, and $j = \sqrt{-1}$, which emulate realistic network extensions beyond the considered sub-network.

*2) Channel Generation:* We use the open-source PLC channel synthesizer of [36] to generate PLC CFRs for training, validating, and testing our machines. We generate CFRs with inter-frequency spacing of $24.414$ kHz between $2 - 30$ MHz conforming with the IEEE 1901 BB-PLC standard [37]. We evaluate our solution under two different test conditions. The first, when only the end-to-end link CFR, $H_{\text{e2e}}$, is available

TABLE I
NUMBER OF TRAINING AND TESTING SAMPLES CHOSEN FOR EACH TASK

| Task | 1 & 2 | 3 | 4 | 5 |
|---|---|---|---|---|
| $n_{TR}$ | 32000 | 14000 | 14000 | 8000 |
| $n_{TE}$ | 13200 | 2000 | 2000 | 2000 |

for diagnostics, and the second when the reflection CFR is also available. The reflection CFR is also known as the self-interference CFR, $H_{SI}$, which is estimated by PLC modems that are enabled with the IBFD communication ability as part of the echo cancellation procedure [25], [26]. Further, since our goal is to evaluate the potential of extracting diagnostics information from CFRs, we ignore the impact of power line noise on the practical CFR estimates in our evaluations.

*3) Cable Degradation:* We model the water-tree degradation by replicating its impact on the cable insulation dielectricity [38], [39]. We then feed the values of intact and degraded insulation permittivity into our PLC channel emulator to generate healthy and damaged cable CFRs. We subject the cable to two types of degradation, namely, localized and homogeneous. A homogeneous degradation (HD) affects the cable insulation uniformly and is quantified by $\gamma_{homo}$, which represents the extent of degradation relative to the total insulation thickness. For instance, $\gamma_{homo} = 0.03$ indicates that $3\%$ of the cable insulation is corroded by water treeing. We limit $0 \leq \gamma_{homo} \leq 0.048$, where $\gamma_{homo} = 0$ represents a healthy cable and $\gamma_{homo} = 0.048$ corresponds to the maximum extent of HD caused due to water treeing over a service time of $t_{sr} = 30$ years [38], which is the typical lifespan of an MV XLPE cable. On top of a possible HD, we also introduce localized degradation (LD) that affects a concentrated section of the cable for a length of $\ell_{WT}$. The extent of LD is quantified by $\gamma_{local}$, similar to $\gamma_{homo}$, indicating the extent of cumulative LD relative of the insulation depth, i.e., $\gamma_{local} = 0.5$ represents that water tree degradation has impacted $50\%$ of the cable insulation in the LD section. For all our simulation evaluations, we set $\gamma_{homo} \sim \mathcal{U}(0, 0.05)$, $\gamma_{local} \sim \mathcal{U}(0.1, 1)$, $\ell_{WT} \sim \mathcal{U}(100, 300)$ m, and the center of the LD to be randomly located within 100 m from the center of the six branches (three in the considered sub-network and three extensions outward from every $m_k$).

*4) FFNN Operation:* We choose the number of training and testing samples, $n_{TR}$ and $n_{TE}$, respectively, individually for each diagnostics task (refer Algorithm 1 for the list of tasks). We specify $n_{TR}$ and $n_{TE}$ for each task in Table I. We choose a sufficient $n_{TR}$ to obtain a saturated performance of the trained machine and an adequate $n_{TE}$ to achieve a clear performance trend of the testing samples.

### B. Simulation Results

*1) Tasks−1 and 2: Identifying a Degradation and Classifying its Type:* We perform the two tasks of identifying a possible degradation and classifying its type simultaneously. To this end, we train each $m_k$ to detect an LD between itself and the branch point (BP). A positive result not only indicates the presence of an LD, but also detects the branch on which it resides. A negative outcome from all $m_k$ conveys that no
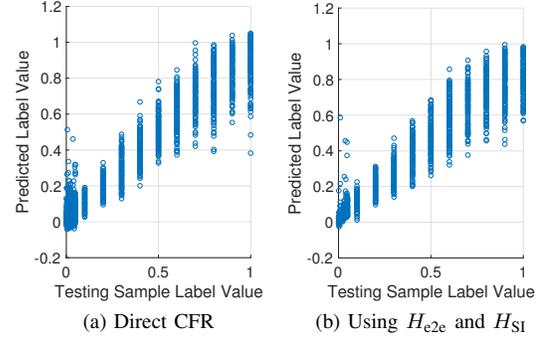


Fig. 3. The scatter plot of prediction results of the label value.

(a) Direct CFR    (b) Using $H_{e2e}$ and $H_{SI}$

LD is present in the sub-network and that the cable is only subject to possible HD.

Binary classification requires only a bi-level labeling during training. However, we discovered that the NN converges only to one of its local minima with such a setting. We therefore label the positive samples, i.e., CFRs with an LD present between $m_k$ and BP, with the associated $\gamma_{local} \in [0.1, 1]$, and the negative samples with only the associated $\gamma_{homo}$, even when an LD is present either on an extended network branch or a branch between $m_j$ and BP, $j \in \{1, 2, 3\}$, $j \neq k$.

Following our design architecture outlined in Section II, we construct an FFNN with $n = 7$. The first five hidden layers are followed by a BN layer. The sizes of the hidden layers are 1024, 512, 256, 128, 64, 32, and 16 from the input to the output layer direction. We use $\sigma_{tanh}$ and $\sigma_{sigmoid}$ as the activation functions for the first two hidden layers, respectively, and $\sigma_{relu}$ for the remaining ones. We apply the RMSProp algorithm to train for 100 epochs, with the training samples split evenly between positive and negative ones. Among the 16000 negative samples, 6000 contain HD and 2000 each contain an LD on the other five branches, i.e., excluding the branch between $m_k$ and BP.

The 13200 testing samples are split evenly into 11 groups of 1200 samples each containing an LD with $\gamma_{local} = \{0.1, 0.2, ..., 1\}$ and an HD. For each condition with $\gamma_{local} > 0$, we generate 200 samples for LD on each of the six branches, i.e., 200 positive and 1000 negative testing samples. The resultant scatter plots for these testing samples are shown in Fig. 3, which demonstrates the performance of our method for varying degrees of $\gamma_{local}$.

While the aforementioned results display the regression ability in predicting the labels, our eventual goal is to translate these predicted labels into classification decisions. To this end, we apply a classification threshold, $\gamma_{th}$, for decision making. We then use the predicted label, $\hat{\gamma}$, to obtain the final classification output, $y_{classify}$, as

$$y_{classify} = \begin{cases} 0, & \text{if } \hat{\gamma} < \gamma_{th} \\ 1, & \text{otherwise,} \end{cases} \tag{10}$$

where $y_{classify} = 0$ and $y_{classify} = 1$ indicate negative and positive outputs of the classification task, respectively. Since the positive samples are labeled with $\gamma_{local} \geq 0.1$ while

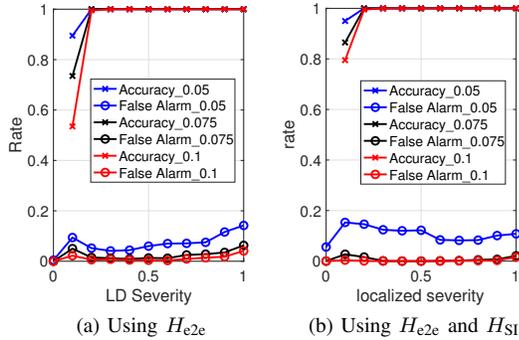(a) Using $H_{e2e}$      (b) Using $H_{e2e}$ and $H_{SI}$

Fig. 4. Results for Tasks 1 and 2, where the legend specifies the chosen $\gamma_{th}$.



(a) Using $H_{e2e}$      (b) Using $H_{e2e}$ and $H_{SI}$

Fig. 6. Results for Task - 3b: Estimating $\ell_{WT}$.



(a) Using $H_{e2e}$      (b) Using $H_{e2e}$ and $H_{SI}$

Fig. 5. Results for Task - 3a: Estimating $b_{k,near}$.



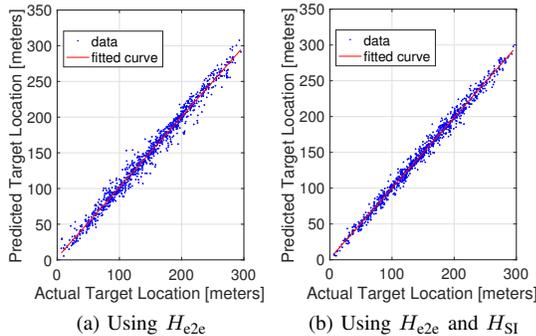(a) Using $H_{e2e}$      (b) Using $H_{e2e}$ and $H_{SI}$

Fig. 7. Results for Task - 4: Estimating $\gamma_{local}$.

the negative ones with $\gamma_{homo} \leq 0.05$, we may choose any $0.05 \leq \gamma_{th} \leq 0.1$ by considering the trade-off between detection accuracy and false alarm. In Fig. 4, we present results for our classification task for different values of $\gamma_{th}$ under both conditions when only $H_{e2e}$ is available, and when $H_{SI}$ can also be extracted. Fig. 4 demonstrates that we nearly match the detection and false alarm rates achieved by the state-of-the-art [10], without requiring any manual feature extraction.

*2) Task−3: LD Location:* Following the diagnostics framework outlined in Algorithm 1, we proceed to locating the LD when the result of Task−2 indicates the presence of one. As part of this task, we locate the position of the two ends of the LD, i.e., the near end towards $m_k$, $b_{k,near}$, and the far end towards the BP, $b_{k,far}$. For this purpose, we first use regression to predict $b_{k,near}$, and then estimate $\ell_{WT}$ to determine $b_{k,far} = b_{k,near} + \ell_{WT}$. Since the presence of an LD causes additional signal reflections resulting from impedance mismatches, we use the time domain versions of $H_{e2e}$ and $H_{SI}$ for this task. We show the results in Fig. 5 and Fig. 6, which clearly illustrate that the fit line of the prediction is nearly of unit slope and passes through the origin, indicating a high degree of prediction accuracy. We also note that while the state-of-the-art solution [10] requires a dual-stage prediction of $\ell_{WT}$ to resolve location ambiguities, our proposed FFNN architecture presents superior performance and can predict the results in a single step. This indicates that our solution is better suited to provide scalability across different network topologies.
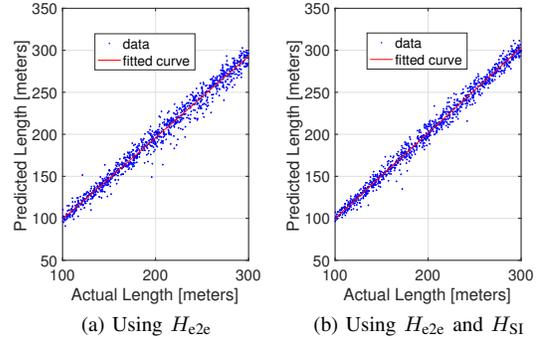
*3) Task−4: LD Severity Prediction:* As the final diagnosis of an LD, we estimate the extent of its damage to assist us in predicting a future in-service fault. As with Task−3, we use the time-domain versions of the CFRs for prediction, whose results are shown in Fig. 7. We notice that while the fit lines under both evaluation conditions, i.e., using only $H_{e2e}$, and together with $H_{SI}$, are similar, the individual prediction variance is improved by using the added insight provided by $H_{SI}$. With sufficient averaging of predictions over time, however, the performance of both conditions are nearly identical.

*4) Task−5: HD Severity Prediction:* When the classification result from Task−2 indicates the case of an HD, we perform HD severity prediction similar to Task−4 for an LD. Note that this condition also includes the case of an intact cable with $\gamma_{homo} = 0$. For this task, we tweak the FFNN architecture by applying BN for the first six layers, instead of five as used previously. Once we predict $\gamma_{homo}$ from our machine, we translate this into an artificial equivalent age, $t_{eq}$, of the cable to provide an intuitive indication into the homogeneous aging severity [10, Eq. 2]. Our estimation results are shown in Fig. 8. We notice that while the machine under-fits a part of the age range, the trend line of the predictions is nearly ideal.

## IV. CONCLUSION

In this paper, we presented first investigation results of using neural networks for cable diagnostics using power line modems. Our online monitoring solution reuses power line modems to intelligently diagnose the health of underground power cables through the inherently estimated power line communication channel state information. We illustrated the

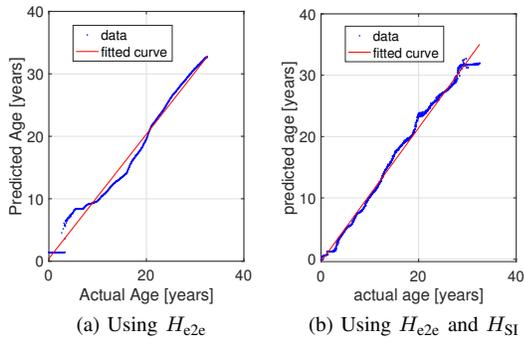(a) Using $H_{e2e}$      (b) Using $H_{e2e}$ and $H_{SI}$

Fig. 8. Results for Task 5: HD severity prediction.

architectures of our designed feed-forward neural network and presented simulation results to show that our method matches the performance of the state-of-the-art to achieve a high degree of accuracy in sequentially detecting, locating, and assessing different variations of water tree degradation, without requiring any manual intervention either for data analysis or feature extraction.

## REFERENCES

[1] S. Galli, H. Latchman, V. Oksman, G. Prasad, and L. Yonge, "Multimedia PLC systems," in *Power Line Communications*, pp. 473–508, Wiley Online Library, 2016.

[2] L. Lampe, A. M. Tonello, and T. G. Swart, *Power Line Communications: Principles, Standards and Applications from multimedia to smart grid.* John Wiley & Sons, 2016.

[3] J. Jousse, N. Ginot, C. Batard, and E. Lemaire, "Power line communication management of battery energy storage in a small-scale autonomous photovoltaic system," *IEEE Tran. Smart Grid*, vol. 8, no. 5, pp. 2129–2137, 2016.

[4] G. Prasad, Y. Huo, L. Lampe, and V. C. Leung, "Electromagnetic compatibility of power line communications in energy storage units," in *IEEE Int. Symp. Power Line Commun. Applicat. (ISPLC)*, pp. 1–6, 2018.

[5] M. O. Ahmed and L. Lampe, "Power line communications for low-voltage power grid tomography," *IEEE Trans. Commun.*, vol. 61, pp. 5163–5175, December 2013.

[6] F. Passerini and A. M. Tonello, "On the exploitation of admittance measurements for wired network topology derivation," *IEEE Transactions on Instrumentation and Measurement*, vol. 66, no. 3, pp. 374–382, 2017.

[7] G. Prasad, Y. Huo, L. Lampe, A. Mengi, and V. C. Leung, "Fault diagnostics with legacy power line modems," in *IEEE Int. Symp. Power Line Commun. Applicat. (ISPLC)*, pp. 1–6, 2019.

[8] F. Passerini and A. M. Tonello, "Smart grid monitoring using power line modems: Anomaly detection and localization," *IEEE Trans. Smart Grid*, vol. 10, no. 6, pp. 6178–6186, 2019.

[9] G. Prasad, Y. Huo, L. Lampe, and V. C. Leung, "Machine learning based physical-layer intrusion detection and location for the smart grid," in *IEEE Int. Conf. Commun. Control Comput. Tech. Smart Grid (SmartGridComm)*, 2019.

[10] Y. Huo, G. Prasad, L. Atanackovic, L. Lampe, and V. C. Leung, "Cable diagnostics with power line modems for smart grid monitoring," *IEEE Access*, vol. 7, pp. 60206–60220, 2019.

[11] H. Orton, "History of underground power cables," *IEEE Electr. Insul. Mag.*, vol. 29, no. 4, pp. 52–57, 2013.

[12] Western Power, "Head to head – overhead vs underground," 2017. [Online] Available: https://westernpower.com.au/community/news-opinion/head-to-head-overhead-vs-underground/.

[13] J. Densley, "Ageing mechanisms and diagnostics for power cables - an overview," *IEEE Electr. Insul. Mag.*, vol. 17, no. 1, pp. 14–22, 2001.

[14] Executive Office of the President. Council of Economic Advisers, *Economic Benefits of Increasing Electric Grid Resilience to Weather Outages.* 2013.

[15] J. Reichl, M. Schmidthaler, and F. Schneider, "The value of supply security: The costs of power outages to Austrian households, firms and the public sector," *Energy Economics*, vol. 36, pp. 256–261, 2013.

[16] A. Castillo, "Risk analysis and management in power outage and restoration: A literature survey," *Electric Power Systems Research*, vol. 107, pp. 9–15, 2014.

[17] P. Gill, *Electrical power equipment maintenance and testing.* CRC press, 2008.

[18] Y. Huo, G. Prasad, L. Lampe, and V. C. M. Leung, "Cable health monitoring in distribution networks using power line communications," in *IEEE Int. Conf. Commun. Control Computing Tech. Smart Grids (SmartGridComm)*, pp. 1–6, 2018.

[19] Y. Huo, G. Prasad, L. Atanackovic, L. Lampe, and V. C. M. Leung, "Grid surveillance and diagnostics using power line communications," in *IEEE Int. Symp. Power Line Commun. Appl. (ISPLC)*, pp. 1–6, 2018.

[20] Y. Huo, G. Prasad, L. Lampe, and C. V. Leung, "Smart-grid monitoring: Enhanced machine learning for cable diagnostics," in *IEEE Int. Symp. Power Line Commun. Appl. (ISPLC)*, pp. 1–6, 2019.

[21] A. M. Lehmann, K. Raab, F. Gruber, E. Fischer, R. Müller, and J. B. Huber, "A diagnostic method for power line networks by channel estimation of PLC devices," in *IEEE Int. Conf. on Smart Grid Commun. (SmartGridComm)*, pp. 320–325, 2016.

[22] G. P. Zhang, "Neural networks for classification: a survey," *IEEE Trans. Syst. Man Cybern.*, vol. 30, no. 4, pp. 451–462, 2000.

[23] A. O. Fernandez and N. K. I. Ghonaim, "A novel approach using a FIRANN for fault detection and direction estimation for high-voltage transmission lines," *IEEE Trans. Power Del.*, vol. 17, no. 4, pp. 894–900, 2002.

[24] S. Samantaray, P. Dash, and S. Upadhyay, "Adaptive kalman filter and neural network based high impedance fault detection in power distribution networks," *Intl. J. of Electr. Power Energy Syst.*, vol. 31, no. 4, pp. 167–172, 2009.

[25] G. Prasad, L. Lampe, and S. Shekhar, "In-band full duplex broadband power line communications," *IEEE Trans. Commun.*, vol. 64, no. 9, pp. 3915–3931, 2016.

[26] G. Prasad, L. Lampe, and S. Shekhar, "Digitally controlled analog cancellation for full duplex broadband power line communications," *IEEE Trans. Commun.*, vol. 65, no. 10, pp. 4419–4432, 2017.

[27] A. M. Tonello and F. Versolatto, "Bottom-up statistical PLC channel modeling—part I: Random topology model and efficient transfer function computation," *IEEE Trans. Power Del.*, vol. 26, no. 2, pp. 891–898, 2011.

[28] F. Versolatto and A. M. Tonello, "An MTL theory approach for the simulation of MIMO power-line communication channels," *IEEE Trans. Power Del.*, vol. 26, no. 3, pp. 1710–1717, 2011.

[29] K. Murphy, *Machine Learning: A Probabilistic Perspective.* MIT Press, 2012.

[30] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *Intl. Conf. Machine Learning*, pp. 1310–1318, 2013.

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

[32] D. E. Rumelhart, G. E. Hinton, R. J. Williams, *et al.*, "Learning representations by back-propagating errors," *Cognitive modeling*, vol. 5, no. 3, p. 1, 1988.

[33] A. Géron, *Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems.* O'Reilly Media Inc., 2017.

[34] E. Steennis and F. Kreuger, "Water treeing in polyethylene cables," *IEEE Trans. Electrical Insul.*, vol. 25, no. 5, pp. 989–1028, 1990.

[35] devolo A.G., "devolo BPL modem MV," https://www.devolo.com/en/SmartGrid/Products/devolo-BPL-Modem-MV.

[36] F. Gruber and L. Lampe, "On PLC channel emulation via transmission line theory," in *IEEE Int. Symp. Power Line Commun. Appl. (ISPLC)*, pp. 178–183, 2015.

[37] "IEEE standard for broadband over power line networks: Medium access control and physical layer specifications," *IEEE Std 1901-2010*, pp. 1–1586, 2010.

[38] J. P. Crine and J. Jow, "A water treeing model," *IEEE Trans. Dielectr. Electr. Insul.*, vol. 12, no. 4, pp. 801–808, 2005.

[39] G. Mugala, *High frequency characteristics of medium voltage XLPE power cables.* PhD thesis, KTH, 2005.