

Deep Neural Network for Joint Nonlinearity Compensation and Polarization Tracking in the Presence of PDL

Reza Mosayebi, *Member, IEEE*, and Lutz Lampe, *Senior Member, IEEE*

Abstract—This paper presents a novel deep linear and nonlinear compensation network (DLNCN) that effectively addresses linear and nonlinear distortion in conjunction with polarization-dependent loss (PDL), while also accounting for changes caused by the joint impact of PDL and time-varying rotation of the state of polarization (RSOP). To accomplish this, we introduce neural network layers dedicated for PDL compensation, and we devise a transfer learning approach that selectively updates weights in layers affected by the variations while keeping the remaining weights unchanged. To monitor RSOP with PDL, we employ a pilot-based acquisition and a pilot-aided decision-directed tracking technique. Our numerical tests demonstrate successful RSOP tracking in the presence of PDL impairments, outperforming state-of-the-art schemes by an average of over 0.75 dB in Q-factor for a dual-polarized 960 km 32 Gbaud 64-QAM transmission with a polarization linewidth of 3 kHz. These results highlight the effectiveness of our proposed deep neural network structure, which includes a dedicated layer for PDL compensation, and its ability to work seamlessly with RSOP tracking.

Index Terms—Fiber optics, nonlinearity compensation, deep neural networks, polarization-dependent loss, transfer learning.

I. INTRODUCTION

Attenuation is a critical parameter that characterizes optical fibers and components [1]. As a single-mode fiber can support two polarization modes, the attenuation of the fiber may vary depending on the polarization of the propagating light. This phenomenon is typically insignificant for standard telecommunication fibers, but it can be significant for certain optical components such as erbium-doped fiber amplifiers (EDFAs), reconfigurable optical add-drop multiplexers (ROADMs), isolators, couplers, switches, and other similar elements that are known to be susceptible to polarization-dependent loss (PDL) [2], [3]. When multiple components with PDL are combined, they can degrade the overall performance of the communication link [4]–[6]. Furthermore, the interaction of the component-induced PDL with linear and nonlinear distortions introduced along the fiber compounds the challenge of nonlinearity compensation (NLC) and rotation of state of polarization (RSOP) tracking at the receiver [7]. Therefore, it is important to consider PDL compensation (PDL) techniques

when designing and optimizing fiber optical communication systems.

In recent years, machine learning (ML) techniques have gained significant attention for their potential in NLC within optical fiber communication. This is evident in notable studies such as for example [8], [9]. Among the ML methods for NLC available, model-based deep neural network (DNN) designs have emerged as effective solutions that strike a favourable balance between performance and complexity. A key advantage of these ML approaches, as opposed to “black-box” solutions, is their foundation on a thorough understanding of the underlying physical phenomena occurring in optical fibers. Specifically, learned digital back-propagation (LDBP) [8], [10] utilizes a DNN framework that emulates digital back-propagation (DBP) [11] and incorporates learned parameters when alternating linear and nonlinear operations. LDBP can also be seen as an instance of unfolding or unrolling of the DBP algorithm [12]. The effectiveness of these DNN architectures in addressing both linear and nonlinear impairments has been extensively discussed in works such as [8], [10], [13]–[16] and references therein. The deep convolutional recurrent neural network with distributed compensation of polarization mode dispersion (DCRNN-PMD) from [13] has been demonstrated to have a superior performance compared to other NLC methods, and it has been extended to the case of time-varying RSOP in [14]. However, we note that none of the existing ML-based methods have explicitly accounted for the effects of PDL.

In this paper, we aim to close this gap. We build on the DCRNN-PMD model and augment it by incorporating PDL. Our proposed solution entails the addition of a simple yet effective layer to the neural network architecture, while preserving the distributed nature of the overall impairment compensation mechanism. We refer to the new DNN model as deep linear and nonlinear compensation network (DLNCN). To train the proposed DLNCN model, like many other neural network structures, we face a practical challenge due to the non-convex nature of the optimization problem. This characteristic often results in the existence of multiple local optima. Consequently, uncertainty arises when seeking the best possible solution for a given architecture. To address this challenge, we have explored various approaches, including the utilization of different random and deterministic weight initializations, to discover an approximate optimal performance.

Comparisons of the DLNCN with the state-of-the-art DCRNN-PMD model reveal an enhanced performance for the former in the presence of PDL, assuming that the RSOP

The authors are with the Department of Electrical and Computer Engineering, University of British Columbia, Vancouver, BC V6T 1Z4, Canada, e-mail: (mosayebi@ece.ubc.ca, lampe@ece.ubc.ca).

This work was supported by Huawei Tech., Canada, and enabled in part through support provided by the Digital Research Alliance of Canada (www.alliancecan.ca).

remains static. When PDL occurs in conjunction with time-varying RSOP, we suggest the use of transfer learning via online training [17]–[20]. In particular, a DLNCN derived from offline training with a static RSOP is applied as the foundation model. Then, we selectively update the learned weights solely in layers affected by changes compared to the training scenario, keeping the remainder of the acquired model unaltered. Our proposed approach builds upon the training methodology employed in previous studies [8], [14], [21], [22]. We apply a pilot-based acquisition strategy followed by a pilot-aided decision-directed tracking algorithm to effectively monitor time-varying RSOP in the presence of PDL. During the acquisition stage, we adapt the layers responsible for accounting for disparities between the PDL and RSOP realizations in the dataset utilized for offline training and the one representing the operational scenario. Once a successful acquisition, characterized by a notable increase in Q-factor, is achieved, we resort to the use of intermittent pilots in the tracking stage. This reduction is facilitated by employing a modified decision-directed technique. Our findings demonstrate the efficacy of this approach in accurately tracking RSOP in the presence of PDL impairments within the fiber, thereby supporting the practicality of this ML detection method.

The remainder of this paper is organized as follows. Section II provides an overview of the theoretical foundation related to fiber impairments, which sets the stage for the subsequent discussions. In Section III, we introduce the DLNCN model and the online model adaptation scheme, which helps track RSOP in the presence of PDL. Section IV presents performance results for the proposed NLC techniques and comparisons with the state-of-the-art scheme in static and time-varying channels. We also discuss the details of the DNN training routine. Finally, Section V concludes the paper by summarizing the key insights from our study and suggesting future research directions.

II. SYSTEM MODEL

The propagation of a dual-polarized optical signal through a single-mode fiber is a complex process that can be subject to linear and nonlinear impairments. To effectively describe this process, the Manakov-PMD equation has been proposed in the literature [23]. The equation is shown at the bottom of the page as (1), where $\mathbf{E}(z, t) = [E_x(z, t), E_y(z, t)]^T$, and $E_x(z, t)$ and $E_y(z, t)$ represent the complex baseband signals of the X and Y polarizations, respectively. Note that the baseband signal is a function of both time $t \geq 0$ and location $0 \leq z \leq L$, where L represents the total length of the fiber. In (1), the coefficients α , β_2 , and γ correspond to the attenuation coefficient, the group velocity dispersion coefficient accounting for chromatic dispersion (CD), and the nonlinear coefficient, respectively. Additionally, the parameter $\Delta\beta_1$ describes the differential group delay (DGD) between the two polarizations and is related to the PMD coefficient

D_{PMD} through $D_{\text{PMD}} = 2\sqrt{2L_c}\Delta\beta_1$, where L_c represents the correlation length of the two polarizations. The matrix $\Sigma(z)$ describes the linear evolution of PMD along the fiber length [24].

It is important to note that (1) does not have a closed-form solution. Therefore, some approximation approaches have been proposed in the literature. We consider the widely used split-step Fourier method (SSFM) [25], which partitions the fiber into $G \times N$ small spatial steps where G is the total number of spans and N denotes the number of steps per span. Within each spatial step, we can assume that the linear and nonlinear effects can be treated independently as summarized in the following, allowing us to apply them successively to numerically solve (1).

A. CD

The impact of CD over a step of length L_s can be characterized in the frequency domain as

$$\tilde{\mathbf{E}}(z + L_s, f) = \exp(j2\beta_2\pi^2 f^2 L_s) \tilde{\mathbf{E}}(z, f), \quad (2)$$

where $\tilde{\mathbf{E}}(z, f)$ represents the Fourier transform of $\mathbf{E}(z, t)$.

B. Signal Attenuation

For each step of length L_s , the signal attenuation can be expressed in the time domain as

$$\mathbf{E}(z + L_s, t) = \exp\left(-\frac{\alpha L_s}{2}\right) \mathbf{E}(z, t). \quad (3)$$

At the end of each span of the fiber, represented by multiple SSFM steps, a lumped optical amplifier is applied to compensate for the attenuation. We note that this adds additive white Gaussian noise (AWGN) to the signal through the generation of amplified spontaneous emission (ASE).

C. Nonlinearity

The Kerr nonlinearity can be modeled by considering its impact over a step of length L_s according to

$$\mathbf{E}(z + L_s, t) = \exp\left(j\frac{8}{9}\gamma \|\mathbf{E}(z, t)\|^2 L_{\text{eff}}\right) \mathbf{E}(z, t), \quad (4)$$

where

$$L_{\text{eff}} = 1 - (1/\alpha) \exp(-\alpha L_s). \quad (5)$$

D. PMD and Time-Varying RSOP

The effect of PMD on each step n with a length of L_s within each span g can be represented as [24]

$$\tilde{\mathbf{E}}(z + L_s, f) = \mathbf{D}_{g,n} \mathbf{U}_{g,n} \tilde{\mathbf{E}}(z, f). \quad (6)$$

$$\frac{\partial \mathbf{E}(z, t)}{\partial z} = -\frac{\alpha}{2} \mathbf{E}(z, t) + \Delta\beta_1 \Sigma(z) \frac{\partial \mathbf{E}(z, t)}{\partial t} - j\frac{\beta_2}{2} \frac{\partial^2 \mathbf{E}(z, t)}{\partial t^2} + j\gamma \frac{8}{9} \mathbf{E}(z, t) \|\mathbf{E}(z, t)\|^2 \quad (1)$$

The matrix $D_{g,n}$ represents the DGD and can be expressed as

$$D_{g,n} = \begin{bmatrix} \exp(j\pi f\tau_{g,n}) & 0 \\ 0 & \exp(-j\pi f\tau_{g,n}) \end{bmatrix}, \quad (7)$$

where $\tau_{g,n}$ is the DGD for the n -th step of the g -th span. The matrix $U_{g,n}$ is a random unitary matrices that apply SOP rotations along the birefringent axes. This matrix, which can also be interpreted as a polarization scrambler, can be represented as a time-invariant rotation matrix of the form

$$U_{g,n} = \begin{bmatrix} \exp(j\phi_1) \cos \theta & \exp(j\phi_2) \sin \theta \\ -\exp(-j\phi_2) \sin \theta & \exp(-j\phi_1) \cos \theta \end{bmatrix}, \quad (8)$$

where ϕ_1 , ϕ_2 , and θ are the angles in Stokes space that are typically different for each pair $(g, n) \in \{1, \dots, G\} \times \{1, \dots, N\}$.

The above-mentioned phase scramblers $U_{g,n}$ are static. However, in practical optical communication systems, the RSOP of a signal may change over time due to various factors, such as temperature fluctuations, pressure changes, vibrations, and cable bends. To model this effect, the well-known hinge model [26] assumes that most of the rotation matrices are static, with only a few polarization scramblers at the ‘‘hot spots’’ (i.e., hinges) evolving over time. These hinges may serve as amplifier locations which are added at the end of each span or fiber segments that are susceptible to temperature or mechanical fluctuations in maintenance huts, railroad bridges, and similar structures. Assuming that the fiber has H hinges, according to the hinge model, the RSOP at each hinge h at time instance k can be represented by a 2×2 *time-varying* matrix $J_{k,h}$ in the Jones space. This matrix can be expressed as [27]

$$J_{k,h} = \exp(-j\zeta_{k,h} \cdot \vec{\sigma}) J_{k-1,h}, \quad (9)$$

where the random variable $\zeta_{k,h}$ in (9) is drawn from a Gaussian distribution with mean $\mathbf{0}$ and covariance $2\pi\Delta p T \mathbf{I}_3$. That is,

$$\zeta_{k,h} \sim \mathcal{N}(\mathbf{0}, 2\pi\Delta p T \mathbf{I}_3). \quad (10)$$

In (10), Δp represents the polarization linewidth, which determines the rate of change of time-varying RSOP, T is the time duration between updates, $\vec{\sigma} = (\sigma_1, \sigma_2, \sigma_3)$ is a tensor of the Pauli spin matrices [28], and \mathbf{I}_3 is a 3×3 identity matrix. The dot operation $\zeta_{k,h} \cdot \vec{\sigma}$ can be interpreted as linear combination of the three Pauli spin matrices as

$$\zeta_{k,h} \cdot \vec{\sigma} = \sum_{i=1}^3 \zeta_{k,h}[i] \sigma_i \quad (11)$$

where $\zeta_{k,h}[i]$ is the i -th element of $\zeta_{k,h}$.

E. PDL

Suppose that the fiber contains U inserted elements (amplifiers, switches, etc.) that each create a power imbalance between the two polarizations. To model PDL for each element u , a 2×2 matrix Γ_u is used in Jones space to describe the impact of that local PDL component on $\mathbf{E}(z, t)$. Therefore, the

effect of PDL in Jones format can be expressed as $\Gamma_u \mathbf{E}(z, t)$ with the following general model for PDL [28]

$$\Gamma_u = \exp\left(-\frac{\alpha_u}{2}\right) \exp\left(\frac{\alpha_u \cdot \vec{\sigma}}{2}\right), \quad (12)$$

where $\alpha_u = \|\alpha_u\|$ is the loss coefficient of the inserted element, and vector α_u is the PDL vector in Stokes space, and $\exp(\cdot)$ in (12) represents matrix exponential. When $\alpha_u = [\alpha_u, 0, 0]^T$, Γ_u reduces to a diagonal matrix, given by

$$\Gamma_u = \begin{bmatrix} 1 & 0 \\ 0 & e^{-\alpha_u} \end{bmatrix}, \quad (13)$$

which can be further normalized to

$$\Gamma_u \propto \begin{bmatrix} \sqrt{1+\gamma_u} & 0 \\ 0 & \sqrt{1-\gamma_u} \end{bmatrix} \quad (14)$$

with $\gamma_u = \tanh \alpha_u$. This normalized matrix is widely used in the literature to describe a diagonal PDL matrix [29]–[31], and we adopt it in this paper.

In our study, we introduce the concept of the *local PDL* measured in decibels (dB) for the u -th component. The local PDL, denoted as ρ_u , is defined mathematically as

$$\rho_u = 10 \log_{10} \left(\frac{1 + \gamma_u}{1 - \gamma_u} \right). \quad (15)$$

This measure quantifies the magnitude of the PDL induced by the specific element, indicating its severity.

A schematic model for the g -th span of a fiber, which includes a time-varying RSOP hinge [24], [27] and a component incurring PDL, is illustrated in Fig. 1. In this figure, it is assumed that there is a hinge and a PDL component at the end of the span. As can be seen, the signal in the g -th span undergoes N DGD and $N - 1$ static RSOP and finally experiences a time-varying RSOP and a PDL impairment.

F. Simulation Model

Based on the preceding discussions, we simulate fiber by applying various effects to each spatial step n within each span g . This involves applying CD, signal attenuation, nonlinearity, and PMD with static RSOP based on (2), (3), (4), and (6), respectively. To compensate for attenuation, a lumped amplifier is added at the end of each span, which introduces AWGN. Additionally, the impact of each local PDL element is emulated using (14). Lastly, when time-varying RSOP is present in the fiber, at each hinge h we calculate $J_{\lfloor k/F \rfloor, h}$ using (9) and apply it to $\mathbf{E}(z, kT_s/F)$ by computing the product $J_{\lfloor k/F \rfloor, h} \mathbf{E}(z, kT_s/F)$ for each time instance k/F , where $1/T_s$ denotes the baud rate, F is the sample-per-symbol rate of the SSFM, and $\lfloor \cdot \rfloor$ indicates the floor operation. In our simulations, we incorporate a PDL element at the end of each span. Moreover, in the presence of time-varying RSOP, we take into account the inclusion of a hinge at the end of each span. The initial matrices $J_{0,h}$ are selected uniformly at random from the set of all 2×2 unitary matrices ensuring uniformly distributed SOP across the Poincaré sphere [27].

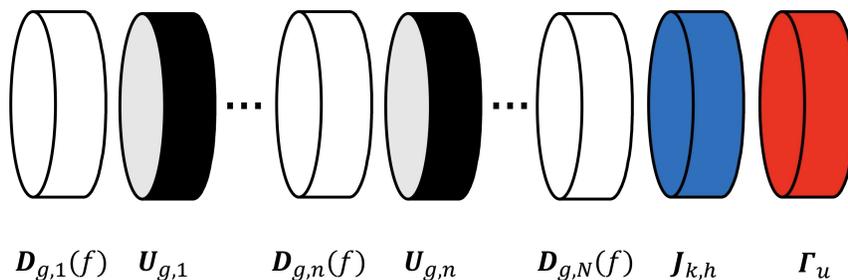


Fig. 1. A schematic model of PMD, PDL, and time-varying RSOP impairments in the g -th span of a fiber, assuming the presence of a hinge and a PDL component at the end of the span.

G. Transmitted and Received Signals

For the sake of notation convenience, in the following sections, we will use the notations $x(k)$ and $y(k)$ to represent the quadrature amplitude modulation (QAM) data symbols at time instance k for the X and Y polarizations, respectively. This can be expressed as

$$\mathbf{E}(0, t) = \sum_{k=-\infty}^{\infty} p(t - kT_s) \begin{bmatrix} x(k) \\ y(k) \end{bmatrix}, \quad (16)$$

where $p(t)$ is a root-raised cosine (RRC) pulse shape.

At the receiver, we assume that the received signal $\mathbf{E}(L, t)$ is coherently detected. It is then subjected to low-pass filtering using an optical filter of the form

$$\text{LPF}(f) = \exp\left(-\left(\frac{f}{f_{\text{cut}}}\right)^\xi\right), \quad (17)$$

where f_{cut} is the cutoff frequency, and the power exponent ξ determines the rate of decrease after the cutoff frequency. Following the low-pass filtering, the signal is sampled at a rate of 2 samples per symbol before being fed into the NN models for further processing. These inputs to the NN models are denoted by x_0 and y_0 for X and Y polarizations, respectively, and are defined as

$$\begin{bmatrix} x_0(k) \\ y_0(k) \end{bmatrix} = \mathbf{E}(L, kT_s/2). \quad (18)$$

III. LEARNING-BASED IMPAIRMENTS MITIGATION AND TRACKING IN THE PRESENCE OF PDL

In this section, we present the proposed DLNCN model with PDL layers and describe it in detail. We also discuss how to efficiently transfer knowledge from a trained model with static RSOP to the case where RSOP changes over time. Throughout the paper, we employ the mean squared error (MSE) as the chosen loss function for all training/adapting schemes.

A. Deep Linear and Nonlinear Compensation Network (DLNCN)

The proposed DLNCN model employs a distributed compensation architecture comprising five distinct types of layers, each designed to compensate for a specific channel impairment. Fig. 2 shows the five layers in the i -th step, and the entire DLNCN is a concatenation of M such steps, which is

referred to as the number of DLNCN steps. The 2-samples-per-symbol sequences that are inputs to the i -th step are denoted as

$$\{x_i\} = \dots, x_i(k-1), x_i(k), x_i(k+1), \dots$$

for the X polarization and

$$\{y_i\} = \dots, y_i(k-1), y_i(k), y_i(k+1), \dots$$

for the Y polarization. The neural network receives the input sequences $\{x_0\}$ and $\{y_0\}$ from X and Y polarizations, respectively, and processes these sequences through different layers which will be explained in detail in the following. Finally, the output sequences of the model, denoted by $\{x_M\}$ and $\{y_M\}$ for the X and Y polarizations, respectively, are input to an RRC-matched filter (MF), whose output is sampled at symbol rate and compared with the true transmitted symbols $x(k)$ and $y(k)$ in order to train the model.

Remark 1: In several related works [9], [21], [32]–[34] including our previous work [14] the MF is applied before the NN model. This retains the position of the MF in the receiver processing chain when only using linear CDC and PMD compensation. However, since the role of the NN, as well as conventional DBP, is to invert the optical channel, it is more meaningful to place the MF after the NN.¹

The first part of the DLNCN model is a complex-valued convolutional layer that compensates for CD equally on both polarizations. The second part compensates for DGD using a short real-valued filter with the same coefficients for both polarizations but in a reversed order. To effectively mitigate the impact of PDL, we add a third layer dedicated to PDL. Its primary function is to rectify power disparities between the two distinct polarization modes. Accordingly, one polarization signal is subject to multiplication by a trainable parameter p_i , while the signal of the other polarization remains unaltered. The fourth layer applies a rotation matrix to the signal to compensate for RSOP, and finally, a bidirectional recurrent layer is used to account for nonlinearity compensation. As mentioned above, these five layers are repeated M times, and the output of the last step is used to estimate the transmitted symbols. Fully training the NN models from scratch is called *initial training* which will be further evaluated in Section IV-A.

Remark 2: It is worth noting that by removing the PDL layer, the network can be reduced to the DCRNN-PMD model

¹We acknowledge the reviewer of [14] who raised the question about the position of the MF during the review process.

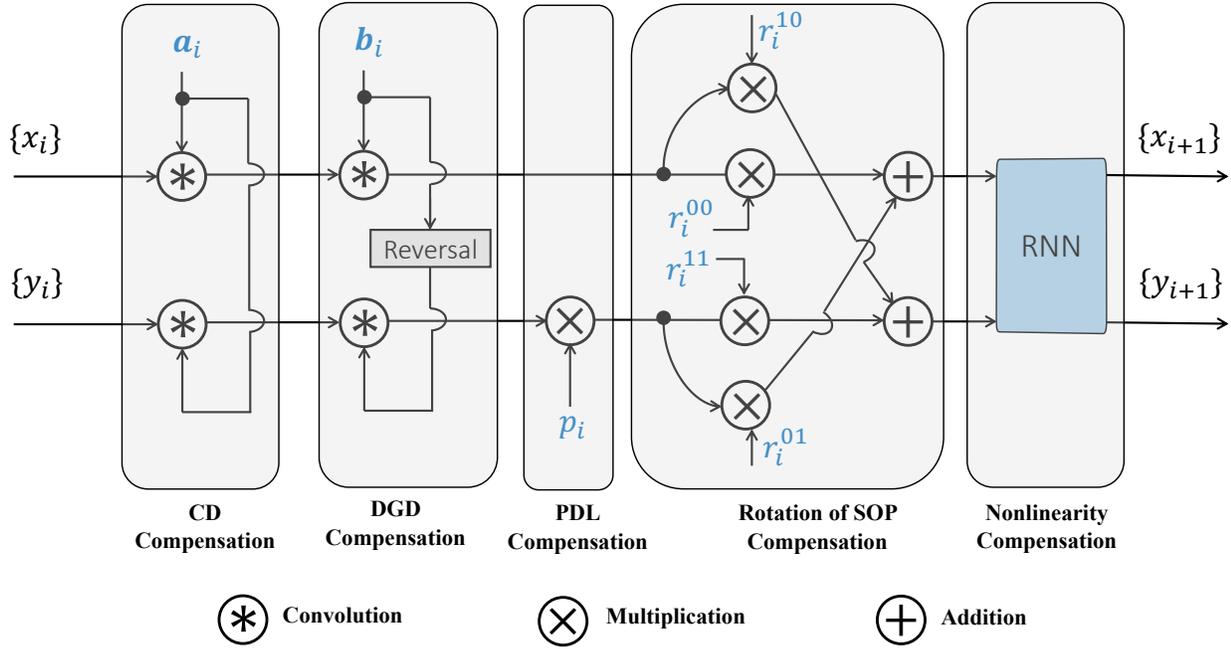


Fig. 2. Architecture of the proposed DLNCN model at the i -th step.

introduced in [14]. Although the addition of the PDL layer may seem like a minor modification, our numerical results demonstrate that it considerably improves the quality of the output symbols when static RSOP is present and even more so when time-varying RSOP is present, as the difference in the power of the X and Y polarizations can be substantial.

Remark 3: To gain preliminary insights into the effect of PDL on the transmission chain, let us momentarily assume that there is no RSOP in the fiber link. Under this assumption, as both DGD and PDL matrices are diagonal, we can consider the combined impact of DGD and PDL as given by

$$\Gamma_u \mathbf{D}_{g,n} = \begin{bmatrix} \exp(j\pi f \tau_{g,n}) \sqrt{1 + \gamma_u} & 0 \\ 0 & \exp(-j\pi f \tau_{g,n}) \sqrt{1 - \gamma_u} \end{bmatrix} \quad (19)$$

By examining (19) and comparing it to (7), we observe that to compensate for both DGD and PDL, a *scaled* version of the reversed filter for the other polarization must be employed. This will be accomplished by introducing the trainable coefficient p_i for each step i of the distributed neural network.

B. Adaptation of Learned DLNCN Model

Transfer learning [17]–[20] plays a vital role in simplifying the training process by leveraging similarities in distribution and features to modify a pre-trained neural network model from a source domain to a target domain, thus enabling effective online learning. In our specific case, dealing with time-varying RSOP in the presence of PDL, we build upon the concept of transfer learning in distributed NLC networks, which has been previously explored in works like [8], [14], [21]. Leveraging the interpretability of the layer roles in the DLNCN model, we selectively update only the weights that

correspond to changes when compared to the source domain, facilitating seamless adaptation to new datasets and changing environmental conditions. In this section, we discuss how to perform transfer learning, which includes retraining on different datasets and online training through acquisition and tracking.

1) *Retraining:* Retraining a pre-trained DLNCN network on a different dataset enables us to leverage knowledge learned from a source domain to improve performance on a target domain. For efficient retraining, we can exploit the fact that the DLNCN layers mirror impairments introduced by the optical fiber. Hence, we can freeze layers, i.e., keep weights fixed, for layers whose associated impairments we can assume to remain constant when transitioning from one dataset to another. For example, for a given optical link setup, we expect that CD and nonlinearity of the fiber are constant and thus we would keep the CDC and NLC layers fixed. On the other hand, as the PMD and PDL realizations of a new dataset are likely to be different from the ones used for initial training, we will adapt the layers corresponding to these impairments during the retraining process. In Section IV, we will demonstrate that DLNCN can be optimized with a relatively small number of epochs during retraining. This is possible because the network has already learned foundational features from the source dataset. By fine-tuning the adaptable layers with the target dataset, DLNCN can quickly adapt to the new domain, reducing training time and computational complexity.

2) *Acquisition and Tracking:* Retraining still uses a full dataset and adapts the NN weights through processing that dataset for several epochs. Adaptation of the NN in a deployed communication system should be performed via online learning. Nevertheless, efficient retraining via adapting weights of specific layers guides the implementation of online learning for

the DLNCN. We propose a two-stage online training process via acquisition and tracking. During acquisition, the layers of the DLNCN network that correspond to impairments that may have changed from the initial training, e.g., DGD, PDL, and RSOP compensation layers, are adapted using pilot data $x(k)$ and $y(k)$. That is, the MSE loss at the k -th time instance

$$\text{Loss}^{\text{Acq.}}(k) = \frac{\|\mathbf{x}_M(k) - \mathbf{x}(k)\|^2 + \|\mathbf{y}_M(k) - \mathbf{y}(k)\|^2}{2B}, \quad (20)$$

where B is the batch size and

$$\mathbf{x}_M(k) = [x_M(k-B+1), \dots, x_M(k-1), x_M(k)]^T, \quad (21)$$

$$\mathbf{y}_M(k) = [y_M(k-B+1), \dots, y_M(k-1), y_M(k)]^T, \quad (22)$$

$$\mathbf{x}(k) = [x(k-B+1), \dots, x(k-1), x(k)]^T, \quad (23)$$

$$\mathbf{y}(k) = [y(k-B+1), \dots, y(k-1), y(k)]^T, \quad (24)$$

is used to perform one update of the selected layers.

After successful acquisition, the second stage utilizes a decision-directed scheme to update the network based on the output decisions made by the DLNCN. Typically, we can assume that the tracking scheme involves updating only the RSOP compensation layers while keeping the other layers frozen. The decision-directed approach employs these decisions as true target data and calculates a loss function to update the network's weights. Accordingly, the loss at the k -th time instance can be written as

$$\text{Loss}^{\text{Track}}(k) = \frac{\|\mathbf{x}_M(k) - [\mathbf{x}_M(k)]_D\|^2 + \|\mathbf{y}_M(k) - [\mathbf{y}_M(k)]_D\|^2}{2B} \quad (25)$$

where

$$\begin{aligned} [\mathbf{x}_M(k)]_D &= [x_M(k-B+1)_D, \dots, x_M(k-1)_D, x_M(k)_D]^T \\ [\mathbf{y}_M(k)]_D &= [y_M(k-B+1)_D, \dots, y_M(k-1)_D, y_M(k)_D]^T. \end{aligned} \quad (26)$$

are the hard-decision symbols obtained from $\mathbf{x}_M(k)$ and $\mathbf{y}_M(k)$, respectively.

This process enables the DLNCN to adapt to the changing conditions during data transmission. However, although the decision-directed scheme is generally effective in tracking changes over time, it is sensitive to decision errors. That is, there may be instances where the performance drops and fails to recover promptly, or it may never recover at all. To address this limitation, we propose augmenting the approach with a threshold-based signal processing technique and the use of intermittently transmitted small sets of pilots. This combination helps the DLNCN to minimize the occurrence of scenarios where performance drops and to recover quickly when it does.

The threshold-based signal processing technique is applied during decision-directed tracking and used to determine whether to update the learned model or not. A symbol decision is deemed reliable enough for updating filter coefficients of the learned model if the Euclidean distance between estimated

soft symbols and hard-decision symbols does not exceed a threshold value Th , i.e.,

$$\sqrt{\frac{\|\mathbf{x}_M(k) - [\mathbf{x}_M(k)]_D\|^2 + \|\mathbf{y}_M(k) - [\mathbf{y}_M(k)]_D\|^2}{2B}} \leq \text{Th}. \quad (27)$$

As we will illustrate in Section IV, this selective filter updating based on reliability improves tracking performance. The proposed tracking scheme is summarized in Algorithm 1.

Algorithm 1: Tracking scheme in online learning

- 1: Network has already been updated through acquisition.
 - 2: Keep all layers frozen but RSOP compensation layers.
 - 3: **while** true **do**
 - 4: Compute $\mathbf{x}_M(k)$ and $\mathbf{y}_M(k)$
 - 5: **if** $\mathbf{x}(k)$ and $\mathbf{y}(k)$ are available **then**
 - 6: Compute $\text{Loss}^{\text{Acq.}}(k)$.
 - 7: Backpropagate.
 - 8: **else**
 - 9: Compute
$$Q(k) = \sqrt{\frac{\|\mathbf{x}_M(k) - [\mathbf{x}_M(k)]_D\|^2 + \|\mathbf{y}_M(k) - [\mathbf{y}_M(k)]_D\|^2}{2B}}$$
 - 10: **if** $Q(k) \leq \text{Th}$ **then**
 - 11: Compute $\text{Loss}^{\text{Track}}(k)$.
 - 12: Backpropagate.
 - 13: **else**
 - 14: Do not update weights.
 - 15: **end if**
 - 16: **end if**
 - 17: **end while**
-

In summary, transfer learning is applied in the DLNCN for addressing the challenges posed by the time-varying RSOP in the presence of PDL. Specially, employing transfer learning in the two-stage online training process allows DLNCN to operate in a practical receiver and to track dynamic time-varying impairments via acquisition and tracking. The proposed threshold-based signal processing approach further enhances the DLNCN's capability to recover and improve performance, improving robustness in changing environments.

IV. NUMERICAL RESULTS

In this section, we conduct a numerical performance evaluation of the proposed DLNCN architecture and compare its performance with that of DCRNN-PMD, specifically in scenarios where PDL is present in the fiber.

For concreteness, we adopt the simulation setup from [14], with slight adjustments made to the parameters. The details are provided in Table I. Different from [14], we consider EDFAs with a 6 dB noise figure and a local PDL of 3 dB (for the initial training case with static RSOP) and 2 dB (for the retraining and time-varying RSOP cases) at the end of each span, respectively. In order to manage complexity, both NN models consist of $M = 12$ steps, which is the same as the number G of fiber spans, unless stated otherwise. While this 12-step configuration is assumed for most numerical tests, we also study the impact of varying the number of steps on

TABLE I
SIMULATION PARAMETERS FOR INITIAL TRAINING.

Parameter	Value
Modulation	64 QAM
Polarization	Dual
Center wavelength	1552.93 nm
Symbol rate	32 GBd
Cutoff frequency(f_{cut})	18.75 GHz
Filter power exponent (ξ)	40
SSFM bandwidth	128 GHz ($F = 4$ samples per symbol)
Receiver bandwidth	64 GHz (2 samples per symbol)
Pulse shape	Root-raised cosine
Pulse roll-off	0.06
Span length	80 km
# Spans (G)	12
Fiber loss(α)	0.21 dB/km
Dispersion parameter	16.8 ps/nm/km
Nonlinearity parameter (γ)	1.14 1/W/km
PMD	0.1 ps/ $\sqrt{\text{km}}$
EDFA noise figure	6 dB
Local PDL (ρ_u)	3 dB
SSFM steps per span (N)	500
# Training symbols	5×10^5
# Test symbols	5×10^5
Learning rate scheduling	Cosine annealing
Max. learning rate	10^{-3}
Min. learning rate	10^{-4}
# Epochs	2×10^3
Batch size	10^4
# DLNCN steps (M)	12

the performance of the proposed DLNCN model. Table I also includes the learning rate schedule used for training the NN models, with more details provided in the following.

We use the Q-factor, defined as

$$Q\text{-factor} = \sqrt{2} \operatorname{erfc}^{-1}(2 \text{BER}), \quad (28)$$

to assess the effectiveness of each model on the provided datasets. The Q-factor is obtained from the empirical bit error rate (BER) via the inverse complementary error function (erfc^{-1}) and can be interpreted as an equivalent signal-to-noise ratio (SNR).

A. Initial Training

Optimization problems performed on neural networks are generally non-convex, which means they can have multiple local optima, saddle points, and lack a globally optimal solution [35]–[39]. This poses a practical challenge in training neural networks, as there is no guarantee of finding the best possible solution for a given architecture. To address this challenge, researchers have explored different approaches, including the use of various random and deterministic weight initializations [40]–[43].

In optical communication systems, a number of studies have investigated the impact of different initialization methods on neural network performance [8], [10], [14]. Notably, in [10], the authors found that random initialization did not yield satisfactory results. However, contrasting these findings, [14] stated that random initializations provided the best outcomes. This suggests that the choice of initialization method may depend on the specific structure of the neural network model including hyper-parameters and the communication system settings.

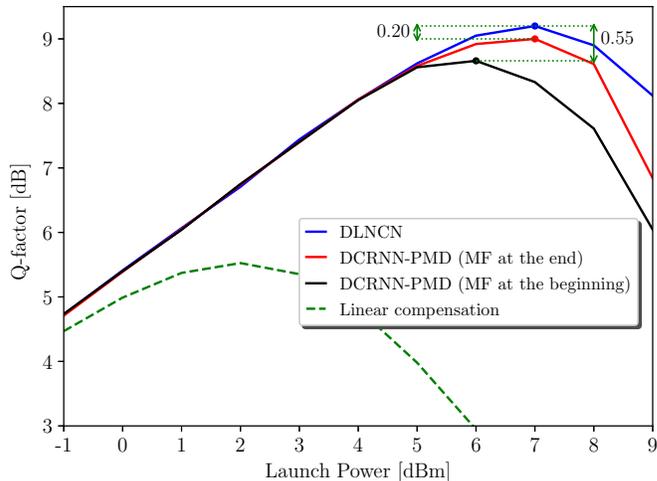


Fig. 3. Performance comparison of DCRNN-PMD and DLNCN models for $\rho_u = 3$ dB.

Given this insight, we performed experiments with multiple random and deterministic weight initializations, tailored to our network architecture. While deterministic weight initialization simplifies training and ensures consistent results across different runs, our investigations consistently revealed that the best outcomes were achieved through random initialization. More precisely, our training attempts show that the best results for both DCRNN-PMD and DLNCN models are obtained by using the Adam optimizer [44] along with employing Xavier weight initialization [45] for CD, DGD, rotation, and NL compensation layers along with initialization via a real-valued normal distribution with mean 1 and variance 10^{-4} for the PDLC layer. For the following figures, we have chosen the best result obtained with 20 random initializations. This approach allows us to showcase the performance of the neural network using the most favorable initialization instance identified during our experiments. For all training attempts, we have employed the well-known cosine annealing schedule for learning rate adaptation with the parameters given in Table I.

Fig. 3 depicts the Q-factor as a function of launch power for DCRNN-PMD and DLNCN in the presence of PDL with $\rho_u = 3$ dB. For a fair comparison, we also include the curve for DCRNN-PMD with the MF after the NN in the figure. As can be seen, in the linear regime (launch power less than 5 dBm), the DLNCN and DCRNN-PMD models perform almost the same. However, as the launch power increases and the system transitions into the nonlinear regime, the DLNCN model outperforms the original DCRNN-PMD model with the MF at the beginning of receiver digital signal processing by about 0.55 dB and the modified DCRNN-PMD model with the MF at the end of receiver by 0.20 dB. The gain can be attributed to the introduction of the PDLC layer. The difference is more pronounced in the presence of time-varying RSOP, which will be discussed later in Fig. 8. We note that since having the MF at the end of receiver chain provides better results, we also consider this setup for the benchmark DCRNN-PMD scheme in the sequel.

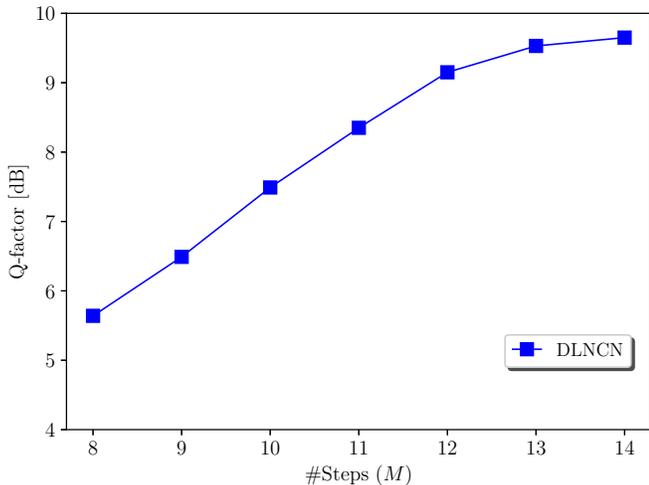


Fig. 4. Offline training performance versus different number of steps M for a transmit power of 7 dBm.

In Fig. 4, we examine the impact of varying the number of steps on the DLNCN model’s performance, analyzing the initial training performance in terms of Q-factor against different step counts, M , at a transmit power of 7 dBm. The figure illustrates that the DLNCN model’s performance increases with a larger number of steps, first more sharply and then more gradually at around the one-step-per-span point. The performance improvement can be attributed to an increase in the number of learnable parameters, which in this test is linearly proportional to M . One may speculate that $M = G$ is a favourable setting as the NN model and the link structure are matched in some way. However, we note that one could also increase the number of parameters per layer for the cases of $M < G$ to achieve an optimized performance. Furthermore, choosing $M > G$ results in additional performance gains. Overall, finding an optimal balance between the model’s compactness and its learning capabilities is essential for achieving a desired trade-off.

Before concluding the initial training results section, we provide the scatter plot of the output of the DLNCN model in Fig. 5 for a launch power of 7 dBm. The plot reveals that the output samples exhibit a complex Gaussian-like distribution centered around the transmitted data symbols. This finding supports the selection of MSE as the loss function, as the NN training indeed minimizes the uncertainty about the transmitted data in the NN outputs.

B. Retraining

Applying a neural network model trained on a dataset representing a specific scenario of an optical fiber link to a different dataset even for a nominally identical link is likely to yield an unacceptably low Q-factor. This is because of the different instances of random link parameters between the datasets, in particular the different realizations of RSOPs, DGDs, and local PDLs. To deal with this distribution shift from one scenario to another, we employ transfer learning, as introduced in Section III-B1. In this approach, we set the

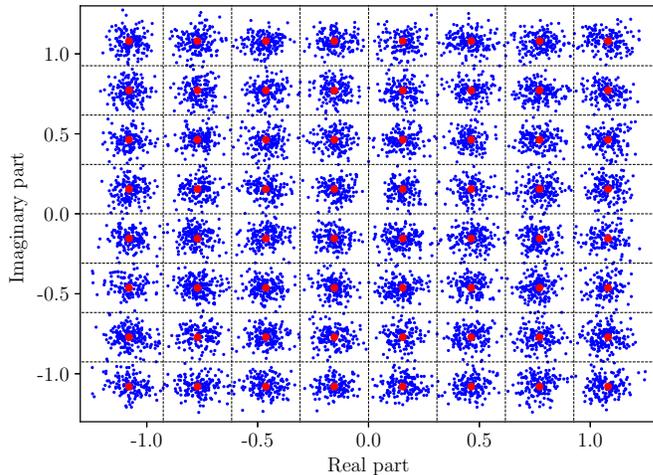


Fig. 5. Scatterplot of the output symbols from the DLNCN at launch power 7 dBm.

initial weights as the final weights of the pre-trained network and retrain it by only updating the DGD, rotation matrix, and PDL filters, while keeping other weights and layers unaltered.

The learning curve for training from scratch (initial training) with 2000 epochs is depicted alongside the learning curve for retraining over 100 epochs, both in terms of the MSE loss function, in Fig. 6, where both schemes are trained on the same new dataset. To implement the retraining scheme, we borrowed the initial network weights from a fully trained network with training parameters given in Table I, and performed retraining over a different dataset with different DGD and RSOP realizations and as well as a different local PDL of $\rho_u = 2$ dB, using maximum and minimum learning rates of 10^{-2} and 10^{-3} , respectively. This learning rate schedule is one order of magnitude larger than the learning rate used for training from scratch. As Fig. 6 shows, with only 100 epochs and by updating only the DGD, RSOP, and PDL compensation layers of the network, we were able to successfully retrain the model on a new dataset, resulting in an obtained Q-factor of 9.78 dB. This example demonstrates a significant overhead reduction of 95% compared to the case without transfer learning, while only incurring a loss of approximately 0.1 dB.

Next, we compare the performances of the proposed DLNCN model and the DCRNN-PMD model from [14]. Both models are first trained on a dataset according to the parameters in Table I and then we examine four datasets with $\rho_u = 2$ dB, each featuring different realizations of DGD and RSOP. Table II shows the figures for the Q-factor obtained after retraining. We also include the aggregated PDL, which represents the ratio between the average powers for the X and the Y polarizations at the output of the optical fiber link. The results from this retraining procedure reveal that the DLNCN model exhibits superior performance over the DCRNN-PMD model, surpassing it by a margin of more than 0.3 dB when PDL is present. Notably, when the aggregated PDL is higher, the disparity between the proposed model and the benchmark model becomes more pronounced. This difference can be as large as 3.5 dB, as evident in the table for Dataset #4. These

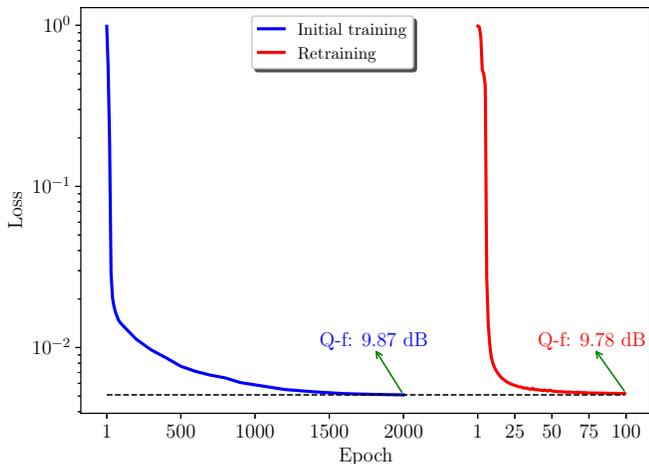


Fig. 6. Validation loss versus epoch for DLNCN model during initial training and retraining, both on the same dataset with $\rho_u = 2$ dB at launch power 7 dBm.

TABLE II
PERFORMANCE COMPARISON WITH RETRAINING.

NN model	Dataset for retraining	Aggregated PDL	Q-factor obtained after retraining
DCRNN-PMD	Dataset #1	2.20 dB	9.28 dB
			DLNCN
DCRNN-PMD	Dataset #2	2.46 dB	9.58 dB
			DLNCN
DCRNN-PMD	Dataset #3	0.60 dB	10.00 dB
			DLNCN
DCRNN-PMD	Dataset #4	2.34 dB	6.07 dB
			DLNCN

results demonstrate the effectiveness of the DLNCN model in addressing PDL in conjunction with other linear and nonlinear channel impairments, and its suitability for efficient retraining.

In considering the results in Table II, it is also important to recognize that while the aggregated PDL values are indicators for the severity of impairments, the specific PMD realizations, which vary between datasets, also play an important role. For example, Dataset #2, despite having a slightly larger aggregated PDL than Dataset #1, shows a slightly better Q-factor.

C. Acquisition and Tracking

We now turn to the case of online training using acquisition and tracking with only a single epoch. Acquisition is similar to retraining in that known data in the form of pilot symbols are available. Therefore, the insights gained from retraining above can directly be applied here. In addition, online training via acquisition and tracking is also meant to render the DLNCN operating successfully when dynamic impairment are present, in particular time-varying rotations of SOP along the fiber as discussed in Section II-D. In such scenarios, due to the continuous change in the data distribution, the concept of offline training with multiple epochs is not feasible. Instead, agile tracking schemes with a single epoch should be employed to effectively track these changes.

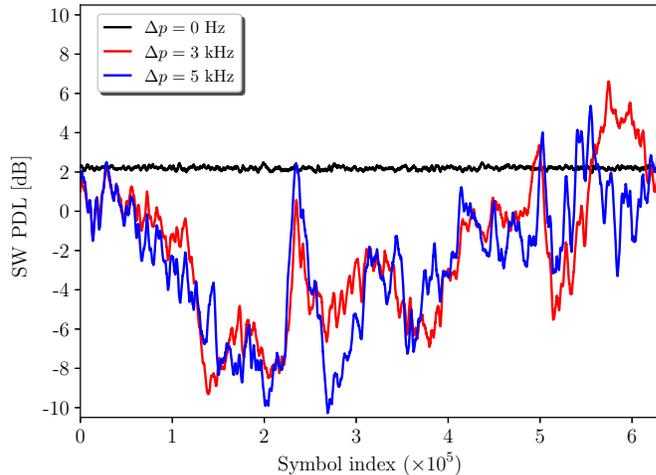


Fig. 7. PDL over time for $\Delta p = \{0, 3, 5\}$ kHz with $\rho_u = 2$ dB.

In the presence of time-varying RSOP, the powers of X and Y polarizations change over time. To highlight the interplay of PDL and RSOP dynamics, Fig. 7 shows the time-varying PDL experienced by the signal at the fiber output measured for a sliding window (SW) of 2^{12} symbols and different polarization linewidths Δp , when $\rho_u = 2$ dB and the launch power is set to 7 dBm. The figure illustrates that in the absence of time-varying RSOP ($\Delta p = 0$), the SW PDL remains relatively constant (it will become constant and identical to the aggregated PDL as the SW size increases). However, increasing the polarization linewidth results in a rapidly changing SW PDL, and it reaches levels of up to 10 dB in our example. The rate of change of the SW PDL is proportional to the level of RSOP dynamics as represented by Δp .

1) *Acquisition*: To test the effectiveness of the proposed online learning scheme, we conducted acquisition experiments on a modified version of Dataset #1 (see Table II) that also features time-varying RSOP with a polarization linewidth of 3 kHz. To investigate the impact of various factors on the performance of the DCRNN-PMD and DLNCN models, we tested different batch sizes, learning rates, and adaptation methods for each model. The best results obtained from these experiments for each model are presented in Fig. 8 (top two curves), which shows the SW Q-factor over symbol index (time) for acquisition on each model scheme. Similar to the SW PDL results, also the SW Q-factor is averaged over a SW of 2^{12} symbols.

Through numerical tests we observed that a batch size of only one symbol yields the best results for both models. Furthermore, stochastic gradient descent (SGD) is found to be the optimal adaptation method for DCRNN-PMD, while the Adam optimizer performs best for the proposed DLNCN model. This can be attributed to the fact that the adaptive learning rates of Adam may be better suited for the architecture of DLNCN model that has additional layers for PDLC. The optimal learning rate for the DLNCN model is 10^{-4} , one order of magnitude smaller than the DCRNN-PMD model (10^{-3}), highlighting their differences and the need for a specialized adaptation strategy to address RSOP changes over time in

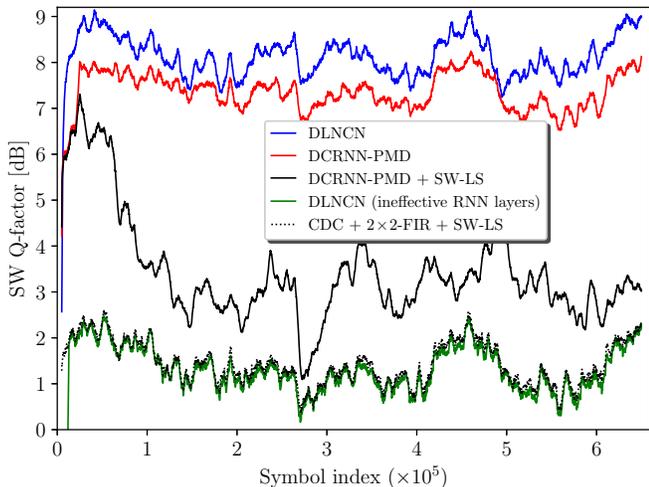


Fig. 8. Performance of acquisition for $\Delta p = 3$ kHz, $\rho_u = 2$ dB, and batch size equal to 1.

the presence of PDL. Most importantly, the DLNCN notably outperforms the state-of-the-art DCRNN-PMD model when PDL and time-varying RSOP are present. The average Q-factor gain is over 0.75 dB. This showcases the effectiveness of the proposed DNN structure with a dedicated layer for PDL and its ability to work in tandem with RSOP adaptation.

We have included three additional schemes for benchmarking in Fig. 8 (bottom three curves). Two of them include the SW least-squares (SW-LS) algorithm from [30], which has been developed for RSOP tracking and PDL compensation in the absence of other linear and nonlinear distortions. We thus concatenate it with the DCRNN-PMD model as well as a linear equalizer consisting of CDC and a 2×2 -FIR filter. We adapt the DCRNN-PMD model and the 2×2 -FIR filter up to 20,000 symbols for effective compensation of static PMD, then halt their adaptation, and feed their output into the SW-LS algorithm for further RSOP tracking and PDL compensation. The SW-LS parameters have been set to $L = 50$ and $\nu = 1$. Lastly, we consider another NN-aided model for comparative analysis. This addition is the DLNCN with ineffective RNN layers, i.e., it is similar to linear equalization but with integrated RSOP tracking and PDL compensation.

The first notable observation is the comparable performance between the conventional linear equalizer with the SW-LS algorithm (dotted black curve) and the DLNCN with ineffective RNN layers (green curve). It is noteworthy that the DLNCN scheme with ineffective RNN layers, despite not having access to the actual system parameters and not compensating for nonlinearity, performs almost on par with the conventional linear receiver that is informed of the CD parameter. As both benchmarks do not address nonlinearity, their performance falls short when compared to the other, more comprehensive benchmarks. In addition, the results clearly demonstrate that when the adaptation in the DCRNN-PMD model is ceased and its output is transitioned to the SW-LS algorithm (solid black curve), there is a significant decrease in Q-factor. This observation supports the notion that the continuous adaptation of the

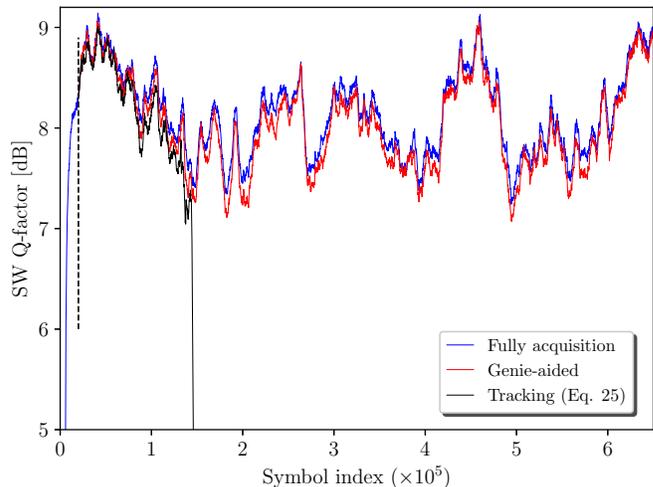


Fig. 9. Performance of tracking for $\Delta p = 3$ kHz, $\rho_u = 2$ dB, and batch size equal to 1. Blue and red curves serve as upper bounds.

neural network layers, as proposed in Section III-B2, is critical for handling dynamic data streams. Consequently, integrating conventional RSOP tracking as a standalone component at the end of the neural network, in lieu of the transfer-learning based adaptation scheme, is not recommended.

2) *Tracking*: In the acquisition phase discussed in previous section, pilot data are used to permit the learned receiver to adjust to the channel realization. We next proceed to the tracking phase, where hard decisions from outputs generated by the current model are used in the loss function (25) to adapt the network. The decision-directed scheme is evaluated against two upper bounds. The first upper bound represents the acquisition case, assuming the availability of all transmitted data ($x(k)$ and $y(k)$) for adaptation. The second upper bound is a *genie-aided* scheme, updating weights at time instance k only if $[x_M(k)]_D = x_M(k)$ and $[y_M(k)]_D = y_M(k)$. We expect that this provides a tighter upper bound on the performance of any practical tracking scheme.

Fig. 9 shows the SW Q-factor results for decision-directed tracking and the two upper bounds and the same setting as in Fig. 8. For all curves, an acquisition phase using 2×10^4 pilot symbols is performed. The comparison between the proposed tracking and the fully acquisition scheme reveals the ability of the former to track the time-varying RSOP well, up until the symbol index 1.5×10^5 . Although there are instances where performance momentarily drops, it gradually recovers and effectively aligns with the Q-factor curve of the proposed upper bounds. However, after the symbol index 1.5×10^5 , where the Q-factor drops, the tracking mechanism fails to recover the signal, resulting in a complete loss of track. This can be attributed to the error propagation present in the decision-directed process.

In order to enhance the performance of the tracking scheme, we consider the application of thresholds on decision-directed updates, as detailed in Section III-B2. The performances achieved with this tracking scheme and different values for Θ in (27) are shown in Fig. 10. When a small threshold value,

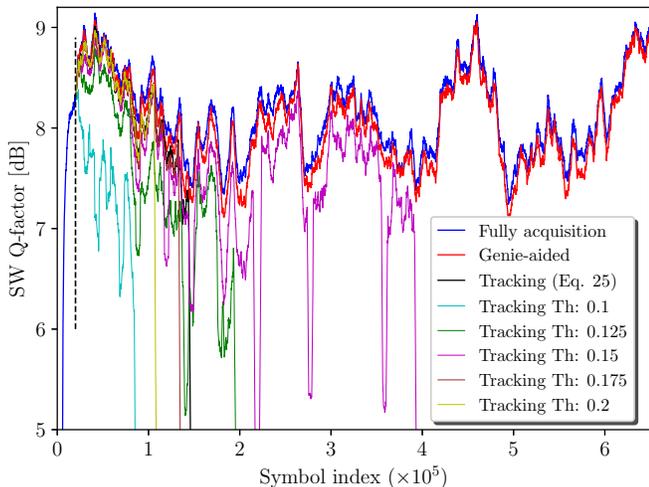


Fig. 10. Performance of tracking for $\Delta p = 3\text{kHz}$ and $\rho_u = 2\text{ dB}$ with reliable decisions and batch size equal to 1.

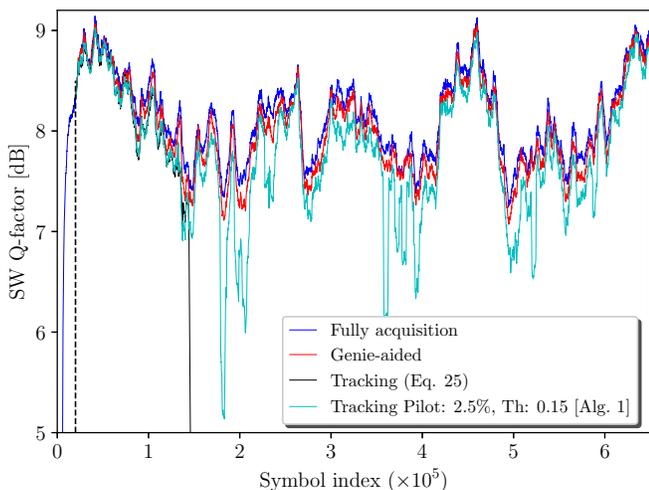


Fig. 11. Performance of tracking for $\Delta p = 3\text{ kHz}$ and $\rho_u = 2\text{ dB}$ with 2.5% pilots.

such as $\text{Th} = 0.1$, is used, the scheme’s performance begins to deteriorate earlier. This phenomenon can be attributed to the continuously changing RSOP present in the data. Failing to update the filter taps sufficiently frequently leads to significant performance degradation. On the other hand, while increasing the threshold value results in more frequent updates, the quality of these updates deteriorates due to decision errors. For the example considered in Fig. 10, setting the threshold to 0.15 seems to strike a favorable trade-off between frequency and quality of the updates. As can be seen in Fig. 10, using $\text{Th} = 0.15$ extends the tracking scheme’s capability to track the signal up to symbol index 3.9×10^5 . However, beyond this point, this approach proves to be ineffective in following the channel variations.

Finally, we integrate the proposed threshold-based tracking strategy with an intermittent use of pilots, introduced in Algorithm 1. The results obtained from this scheme with a pilot density of 2.5% are depicted in Fig. 11. As evident

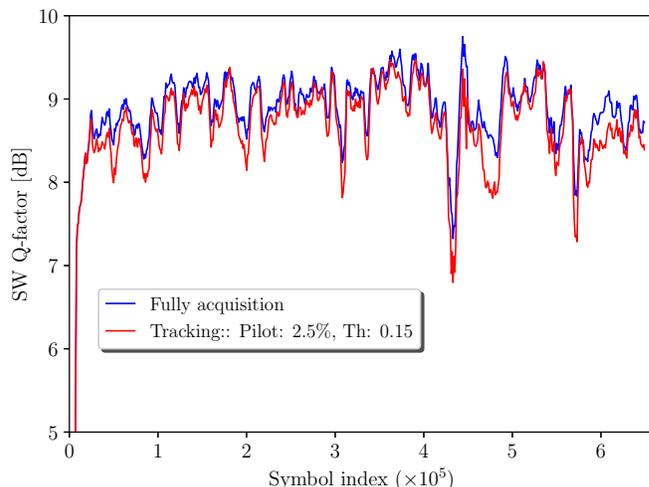


Fig. 12. Performance comparison of acquisition and tracking schemes with $\Delta p = 10\text{ kHz}$ and $\rho_u = 0.5\text{ dB}$, featuring a time-varying RSOP segment at a random location within each odd-numbered span.

from the results, the proposed scheme is able to seamlessly track the signal. Although there are occasional instances where performance dips, the algorithm promptly recovers and accurately follows the signal trend. It is important to note that our experimental findings reveal that these instances, characterized by a decrease in the Q-factor, typically involve a maximum of 1000 symbols. To mitigate the impact of these burst instances, we propose to utilize a suitable interleaver integrated with forward-error-correction (FEC) codes, so that each code block experiences an average of pre-FEC Q-factors.

3) *Robustness*: In the previous tests, each span consistently featured one time-varying RSOP segment, located at its end (see Fig. 1). During these tests, the RSOP linewidth was maintained at $\Delta p = 3\text{ kHz}$, and the local PDL was set to $\rho = 2\text{ dB}$. In the subsequent discussion, we explore variations in these settings.

First, we consider the scenario that a time-varying RSOP segment is placed at a *randomly-chosen location* within each odd-numbered span, i.e., spans 1, 3, 5, 7, 9, and 11, and the even-numbered spans do not include a time-varying RSOP. Additionally, the local PDL, applied at the end of each span, is set to a reduced value of $\rho_\ell = 0.5\text{ dB}$, and the polarization linewidth is increased to 10 kHz. The performance results are shown in Fig. 12. We observe that the proposed acquisition and tracking schemes operate effectively also when the time-varying components of the link are distributed irregularly across the fibre link.

Next, we compare the performances for the proposed DL-NCN model across various linewidths (and $\rho_u = 2\text{ dB}$) using pilot-based acquisition and pilot-aided decision-directed tracking modes with a *fixed learning rate* of 10^{-4} for different linewidths. Fig. 13 shows the symbol-error rate (SER) results as a function of the polarization linewidth normalized by the transmission baud rate. The SER is derived as a single value, averaged over the entire data sequence of 650,000 symbols, incorporating the impact of any fluctuations, such as dips in the performance observed in previous figures. We observe

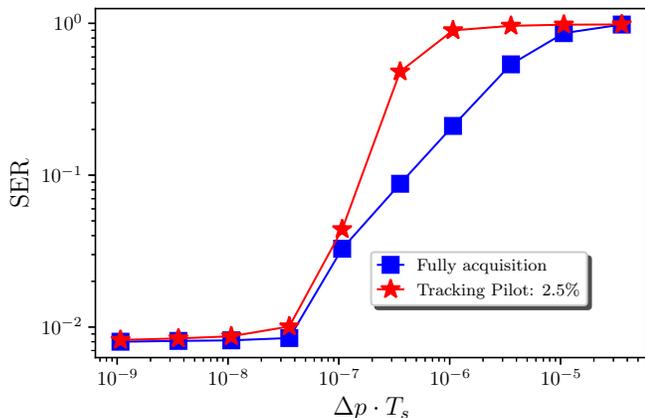


Fig. 13. Performance comparison between the proposed acquisition and tracking schemes versus different normalized polarization linewidths.

that our tracking scheme closely aligns with the upper-bound performance of the pilot-based acquisition scheme until a certain polarization linewidth and thus SER are exceeded. Notably, at a pre-FEC BER around 10^{-2} and normalized linewidth of 10^{-7} , the SER is approximately 6×10^{-2} , with minimal performance difference from the upper bound using only 2.5% pilots.

V. CONCLUSION AND FUTURE WORK

In this paper, we present an enhanced neural network model that effectively compensates linear and nonlinear impairments in optical fiber transmission. Our model incorporates a physics-based network architecture and a PDL compensation layer, resulting in improved symbol output quality compared to state-of-the-art approaches, particularly in the challenging nonlinear transmission regime. Our scheme outperforms existing methods in offline learning and retraining experiments. Moreover, the performance benefits of our model compared to the state-of-the-art schemes become more pronounced in time-varying environments where PDL is combined with time-varying RSOP. While the SW PDL may change drastically, our acquisition-tracking scheme offers a solution for online learning and effectively handles time-varying RSOP in PDL-based optical fiber systems.

However, our simulation results have also revealed occasional performance drops when the SW PDL changes drastically. While increasing the pilot portion and employing an interleaver in conjunction with FEC can alleviate these dips, a potential avenue for future research may lie in investigating more sophisticated tracking schemes. One possible direction could involve studying the scaling of learning rates based on the magnitude of very short SW PDLs or the actual error amount. Regarding the latter, weighting the learning rate inversely proportional to a power of loss via, for example,

$$lr_{\text{new}} = lr_{\text{old}} \times \left(|x_M(k) - [x_M(k)]_D|^2 + |y_M(k) - [y_M(k)]_D|^2 \right)^\nu$$

with $\nu < 0$, might be beneficial in further enhancing the tracking scheme's performance.

Finally, we note that we have not addressed the potential need for re-acquisition after losing track. We did not observe such instances when simulating acquisition/tracking for up to 650,000 symbols, but a study and/or mechanism for re-acquisition is required towards practical use.

REFERENCES

- [1] N. Gisin, "Statistics of polarization dependent losses," *Optics Communications*, vol. 114, no. 5, pp. 399–405, 1995.
- [2] B. L. Heffner, "Automated measurement of polarization mode dispersion using Jones matrix eigenanalysis," *IEEE Photonics Technology Letters*, vol. 4, no. 9, pp. 1066–1069, 1992.
- [3] J. B. et al., "COST 2 I7 interlaboratory comparison of optical measurement on single-mode fibre couplers," *Symposium on Optical Fiber Measurements, Technical Digest*, pp. 7–10, 1990.
- [4] E. Lichtman, "Limitations imposed by polarization-dependent gain and loss on all-optical ultralong communication systems," *Journal of Lightwave Technology*, vol. 13, no. 5, pp. 906–913, 1995.
- [5] C. Poole and T. Darcie, "Distortion related to polarization-mode dispersion in analog lightwave systems," *Journal of Lightwave Technology*, vol. 11, no. 11, pp. 1749–1759, 1993.
- [6] B. T. Kirby, D. E. Jones, and M. Brodsky, "Effect of polarization dependent loss on the quality of transmitted polarization entanglement," *Journal of Lightwave Technology*, vol. 37, no. 1, pp. 95–102, 2019.
- [7] N. Rossi, S. Musetti, P. Ramantanis, and S. Almonacil, "The impact of kerr nonlinearity on the snr variability induced by polarization-dependent loss," *Journal of Lightwave Technology*, vol. 37, no. 19, pp. 5048–5055, 2019.
- [8] R. M. Büttler, C. Häger, H. D. Pfister, G. Liga, and A. Alvarado, "Model-based machine learning for joint digital backpropagation and PMD compensation," *Journal of Lightwave Technology*, vol. 39, no. 4, pp. 949–959, 2021.
- [9] P. J. Freire, V. Neskornuik, A. Napoli, B. Spinnler, N. Costa, G. Khanna, E. Riccardi, J. E. Prilepsky, and S. K. Turitsyn, "Complex-valued neural network design for mitigation of signal distortions in optical links," *Journal of Lightwave Technology*, vol. 39, no. 6, pp. 1696–1705, 2021.
- [10] C. Häger and H. D. Pfister, "Physics-based deep learning for fiber-optic communication systems," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 1, pp. 280–294, 2021.
- [11] E. Ip and J. M. Kahn, "Compensation of dispersion and nonlinear impairments using digital backpropagation," *Journal of Lightwave Technology*, vol. 26, no. 20, pp. 3416–3425, 2008.
- [12] V. Monga, Y. Li, and Y. C. Eldar, "Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing," *IEEE Signal Processing Magazine*, vol. 38, no. 2, pp. 18–44, 2021.
- [13] P. Jain, L. Lampe, and J. Mitra, "Deep convolutional recurrent neural network for fiber nonlinearity compensation," in *2022 European Conference on Optical Communication (ECOC)*, 2022, pp. 1–4.
- [14] —, "Joint PMD tracking and nonlinearity compensation with deep neural networks," *Journal of Lightwave Technology*, vol. 41, no. 12, pp. 3957–3966, 2023.
- [15] H. He, S. Jin, C.-K. Wen, F. Gao, G. Y. Li, and Z. Xu, "Model-driven deep learning for physical layer communications," *IEEE Wireless Communications*, vol. 26, no. 5, pp. 77–83, 2019.
- [16] N. Shlezinger, N. Farsad, Y. C. Eldar, and A. J. Goldsmith, "Model-based machine learning for communications," 2021. [Online]. Available: <https://arxiv.org/abs/2101.04726>
- [17] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.
- [18] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *International Conference on Neural Information Processing Systems (NIPS)*, 2014, p. 3320–3328.
- [19] A. Bobu, E. Tzeng, J. Hoffman, and T. Darrell, "Adapting to continuously shifting domains," in *International Conference on Learning Representations (ICLR)*, 2018.
- [20] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Learning transferable features with deep adaptation networks," in *International Conference on Machine Learning (ICML)*, 2015.
- [21] P. J. Freire, D. Abode, J. E. Prilepsky, N. Costa, B. Spinnler, A. Napoli, and S. K. Turitsyn, "Transfer learning for neural networks-based equalizers in coherent optical systems," *Journal of Lightwave Technology*, vol. 39, no. 21, pp. 6733–6745, Nov 2021.

- [22] O. Simeone, S. Park, and J. Kang, "From learning to meta-learning: Reduced training overhead and complexity for communication systems," in *2020 2nd 6G Wireless Summit (6G SUMMIT)*, 2020, pp. 1–5.
- [23] P. Wai and C. Menyak, "Polarization mode dispersion, decorrelation, and diffusion in optical fibers with randomly varying birefringence," *Journal of Lightwave Technology*, vol. 14, no. 2, pp. 148–157, 1996.
- [24] C. B. Czegledi, "Modeling and compensation of polarization effects in fiber-optic communication systems," Ph.D. dissertation, Chalmers University, 2018.
- [25] R. H. Hardin, "Application of the split-step Fourier method to the numerical solution of nonlinear and variable coefficient wave equations," *Siam Review*, vol. 15, p. 423, 1973.
- [26] M. Brodsky, N. J. Frigo, M. Boroditsky, and M. Tur, "Polarization mode dispersion of installed fibers," *Journal of Lightwave Technology*, vol. 24, no. 12, pp. 4584–4599, 2006.
- [27] C. B. Czegledi, M. Karlsson, P. Johannisson, and E. Agrell, "Temporal stochastic channel model for absolute polarization state and polarization-mode dispersion," in *Optical Fiber Communications Conference and Exhibition (OFC)*, 2017, pp. 1–3.
- [28] J. N. Damask, *Polarization Optics in Telecommunications*. New York, NY, USA: Springer, 2005.
- [29] A. Mecozzi and M. Shtaif, "The statistics of polarization-dependent loss in optical communication systems," *IEEE Photonics Technology Letters*, vol. 14, no. 3, pp. 313–315, 2002.
- [30] M. Farsi, C. Häger, M. Karlsson, and E. Agrell, "Polarization tracking in the presence of PDL and fast temporal drift," *Journal of Lightwave Technology*, vol. 140, no. 19, pp. 1–9, 2022.
- [31] H.-M. Chin, D. Charlton, A. Borowiec, M. Reimer, C. Laperle, M. O'Sullivan, and S. J. Savory, "Probabilistic design of optical transmission systems," *Journal of Lightwave Technology*, vol. 35, no. 4, pp. 931–940, 2017.
- [32] O. Sidelnikov, A. Redyuk, S. Sygletos, M. Fedoruk, and S. Turitsyn, "Advanced convolutional neural networks for nonlinearity mitigation in long-haul WDM transmission systems," *Journal of Lightwave Technology*, vol. 39, no. 8, pp. 2397–2406, 2021.
- [33] D. Orsuti, C. Antonelli, A. Chiuso, M. Santagiustina, A. Mecozzi, A. Galtarossa, and L. Palmieri, "Deep learning-based phase retrieval scheme for minimum-phase signal recovery," *Journal of Lightwave Technology*, vol. 41, no. 2, pp. 578–592, 2023.
- [34] J. Ding, T. Liu, T. Xu, W. Hu, S. Popov, M. S. Leeson, J. Zhao, and T. Xu, "Intra-channel nonlinearity mitigation in optical fiber transmission systems using perturbation-based neural network," *Journal of Lightwave Technology*, vol. 40, no. 21, pp. 7106–7116, 2022.
- [35] A. Choromańska, M. Henaff, M. Mathieu, G. B. Arous, and Y. LeCun, "The loss surfaces of multilayer networks," in *International Conference on Artificial Intelligence and Statistics*, 2014.
- [36] Y. N. Dauphin, R. Pascanu, C. Gulcehre, K. Cho, S. Ganguli, and Y. Bengio, "Identifying and attacking the saddle point problem in high-dimensional non-convex optimization," in *International Conference on Neural Information Processing Systems (NIPS)*, 2014, pp. 2933–2941.
- [37] Y. Li and Y. Yuan, "Convergence analysis of two-layer neural networks with ReLU activation," in *International Conference on Neural Information Processing Systems (NIPS)*, 2017, p. 597–607.
- [38] Y. N. Dauphin, R. Pascanu, C. Gulcehre, K. Cho, S. Ganguli, and Y. Bengio, "Identifying and attacking the saddle point problem in high-dimensional non-convex optimization," in *International Conference on Neural Information Processing Systems (NIPS)*, 2014, p. 2933–2941.
- [39] A. Daniely, R. Frostig, and Y. Singer, "Toward deeper understanding of neural networks: The power of initialization and a dual view on expressivity," in *International Conference on Neural Information Processing Systems (NIPS)*, 2016, p. 2261–2269.
- [40] P. Izmailov, D. Podoprikin, T. Garipov, D. P. Vetrov, and A. G. Wilson, "Averaging weights leads to wider optima and better generalization," in *Conference on Uncertainty in Artificial Intelligence*, 2018.
- [41] J. Zhao, F. T. Schaefer, and A. Anandkumar, "Zero initialization: Initializing neural networks with only zeros and ones," *Transactions on Machine Learning Research*, 2022.
- [42] M. Hardt and T. Ma, "Identity matters in deep learning," in *International Conference on Learning Representations*, 2017.
- [43] H. Zhang, Y. N. Dauphin, and T. Ma, "Fixup initialization: Residual learning without normalization via better initialization," in *International Conference on Learning Representations (ICLR)*, 2019.
- [44] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [45] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," *International Conference on Machine Learning (ICML)*, vol. 28, no. 3, pp. 249–256, 2010.

Reza Mosayebi (Member, IEEE) received his B.Sc., M.Sc., and Ph.D. degrees with honors in electrical engineering from Sharif University of Technology, Tehran, Iran, in 2012, 2014, and 2018, respectively. In 2017, he was a Visiting Research Scholar at Friedrich–Alexander University of Erlangen–Nürnberg, Erlangen, Germany. From 2019 to 2021, he served as a Postdoctoral Researcher at Universitat Pompeu Fabra (UPF), Barcelona, Spain. Currently, he is a Postdoctoral Fellow at the University of British Columbia (UBC), Vancouver, BC, Canada. He has been recognized as an "Exemplary Reviewer" for IEEE COMMUNICATIONS LETTERS in 2020 and for IEEE TRANSACTIONS ON COMMUNICATIONS in 2022, respectively. His research interests include optical fiber and wireless communications, signal processing, and machine learning.

Lutz Lampe (Senior Member, IEEE) received the Dipl.-Ing. and Dr.-Ing. degrees in electrical engineering from the University of Erlangen, Erlangen, Germany, in 1998 and 2002, respectively. Since 2003, he has been with the Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver, BC, Canada, where he is a Full Professor. His research interests include theory and application of wireless, optical wireless, optical fiber, power line, and underwater acoustic communications. He was an associate editor and a guest editor for several IEEE journals, and as the general co-chair and the technical program committee co-chair for IEEE conferences. He has been a Distinguished Lecturer of IEEE Communications Society and a co-recipient of a number of best paper awards. He is a Co-Editor of the book *Power Line Communications: Principles, Standards and Applications from Multimedia to Smart Grid* (2nd edition, John Wiley & Sons).