Efficient Deep Learning of Nonlinear Fiber-Optic Communications Using a Convolutional Recurrent Neural Network

Abtin Shahkarami, Mansoor Yousefi, Yves Jaouën

Efficient Deep Learning of Nonlinear Fiber-Optic Communications Using a Convolutional Recurrent Neural Network

Abtin Shahkarami
dept. Communications and Electronics
Institute Polytechnique de Paris
Telecom Paris
Paris, France
abtin.shahkarami@telecom-paris.fr

Mansoor I. Yousefi
dept. Communications and Electronics
Institute Polytechnique de Paris
Telecom Paris
Paris, France
yousefi@telecom-paris.fr

Yves Jaouen
dept. Communications and Electronics
Institute Polytechnique de Paris
Telecom Paris
Paris, France
yves.jaouen@telecom-paris.fr

***Abstract*—Nonlinear channel impairments are a major obstacle in fiber-optic communication systems. To facilitate a higher data rate in these systems, the complexity of the underlying digital signal processing algorithms to compensate for these impairments must be reduced. Deep learning-based methods have proven successful in this area. However, the concept of computational complexity remains an open problem. In this paper, a low-complexity convolutional recurrent neural network (CNN+RNN) is considered for deep learning of the long-haul optical fiber communication systems where the channel is governed by the nonlinear Schrödinger equation. This approach reduces the computational complexity via balancing the computational load by capturing short-temporal distance features using strided convolution layers with ReLU activation, and the long-distance features using a many-to-one recurrent layer. We demonstrate that for a 16-QAM 100 G symbol/s system over 2000 km optical-link of 20 spans, the proposed approach achieves the bit-error-rate of the digital back-propagation (DBP) with substantially fewer floating-point operations (FLOPs) than the recently-proposed learned DBP, as well as the non-model-driven deep learning-based equalization methods using end-to-end MLP, CNN, RNN, and bi-RNN models.**

***Index Terms*—Fiber-optic communications, deep learning, nonlinear channel impairments, convolutional recurrent neural networks.**

I. Introduction

Signal propagation in long-haul optical fibers is subject to chromatic dispersion (CD), Kerr nonlinearity, and noise. This causes the signal to undergo distortions while propagating in the channel. As a result, digital signal processing is usually performed at the receiver (RX) to equalize the signal. A popular equalization method for optical fiber channels modelled by nonlinear Schrödinger (NLS) equation, is digital back-propagation (DBP) [1]. DBP reverses the deterministic effects of the channel by propagating the signal backward in distance using the split-step Fourier method (SSFM) [2].

In addition to limited performance, DBP suffers from high computational complexity associated with a large number of spatial segments and processing bandwidth [3]. This high complexity has hindered the real-time implementation of DBP in practice [4]. To address this limitation, several low-complexity equalizers have been proposed in lieu of DBP, including neural receivers that are most promising [5]–[10].

There are generally two classes of neural network-based receivers for fiber-optic communications: those driven by a model-based deep learning approach, also called deep unfolding [11], and those obtained on the basis of end-to-end deep learning of the transmitted symbols given the sampled waveform at RX (or typically the waveform after CD compensation) by using a neural network that does not incorporate the channel model [12].

In model-based deep learning methods, a neural network architecture is considered with a computation graph based on the channel model. The model parameters are then tuned using variants of the stochastic gradient descent and back-propagation algorithm. An example is learned DBP (LDBP) [5], [6], [13], which uses the computation graph generated by SSFM as a blueprint for the neural network design, resulting in a convolutional neural network (CNN) with a trainable activation function. It is shown that LDBP provides 50% complexity reduction over DBP for a comparable Q-factor performance in a 16-QAM transmission at 20 G symbol/s for a 32×100km optical fiber link [5]. A roughly similar approach is also adopted by [14] to simulate DBP in dual-pol wavelength-division-multiplexing (WDM) systems.

Although LDBP achieves good performance, it remains computationally complex (in addition to requiring training). The computation graph of LDBP with N_{sp} spans and N_{stps} steps per span (StPS) is a convolutional neural network (CNN) with $\ell = 3 \times N_{sp} \times N_{stps}$ successive linear and non-linear layers [13, Sec. 4]. For $N_{sp} = 32$ and $N_{stps} = 3$ considered in [13], this results in $\ell = 288$ layers, which is high.

On the other hand, non-model-driven neural network-based

This project has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No 766115.

equalizers try to learn the equalization using a relatively shallow neural network. Several models have been proposed in this category [10]. A number of end-to-end multi-layer perceptron (MLP) based approaches is reviewed in [7], [12]. A number of CNN+MLP based methods are also presented in [15]–[17]. These papers report a good BER performance for the proposed models in short-reach single-mode fiber (SMF) transmission. Particularly, [16] presents a relatively low-complexity CNN+MLP model with 5 hidden layers, outperforming the Volterra nonlinear equalizers in a 128 Gbps PAM-4 EML-based optical-link over 40 km SMF transmission. The performance of bidirectional recurrent neural networks (bi-RNNs) models is also investigated in [9], [18], [19]. [9] shows that a bi-LSTM can achieve the BER performance of DBP with lower computational complexity in 16-QAM transmission at distances over 1000 km. The same performance as bi-LSTMs, at near-optimal launch powers, is argued to be achievable by bi-GRUs with approximately 20% fewer floating-point operations (FLOPs) [19]. A comparative study of the performance and complexity of several neural network architectures for end-to-end deep learning-based equalization in fiber-optic communications is provided in [20].

In this paper, we note that different neural network models achieve nearly the same BER with optimized hyperparameters, however, with different computational costs (measured by the number of FLOPs). This is associated with the efficiency of the models to capture the features in data. Therefore, an architecture more aligned with the considered learning task could lead to a notable optimization in computational complexity. We note that although CNNs can efficiently capture short-temporal-distance features in the signal, they may not efficiently capture long-distance features in terms of the model size and complexity. On the other hand, even though RNNs have turned out to be efficient in capturing long-temporal-distance features, they are not as powerful as CNNs in capturing short-temporal-distance dependencies.

We thus propose a hybrid CNN+RNN model for efficient end-to-end deep learning-based equalization of the sampled waveform at RX (after CD compensation) in fiber-optic communication systems. We reduce the computational complexity of the model using two techniques. First, we apportion the learning task into capturing short temporal-distance features using a cascade of CNN layers and capturing long temporal-distance features using a many-to-one recurrent layer. Second, we reduce the dimensionality of data prior to the RNN layer via strided convolutions in the CNN block. We show that for a 16-QAM 100 G symbol/s SMF transmission over a 20×100km optical-link, the proposed CNN+RNN model results in, respectively, 98%, 77%, 32%, and 34% reduction in the number of FLOPs over LDBP and the non-model-driven end-to-end deep learning-based approach using MLP, CNN+MLP, and RNN models adopted in the literature.

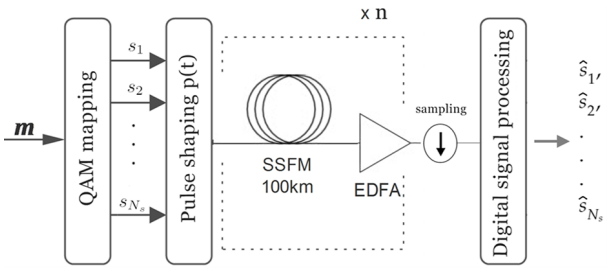

Fig. 1: The schematic of the fiber-optic communication system under consideration.

II. System model

The propagation of signal in one polarization in SMF is governed by the NLS equation

$$\frac{\partial q(t,z)}{\partial z} = -\frac{\alpha}{2}q(z,t) - \frac{j\beta_2}{2}\frac{\partial^2 q}{\partial t^2} + j\gamma|q(t,z)|^2 q(t,z) \quad , \quad 0 \le z \le \mathcal{L}, \quad (1)$$

where $q(t,z)$ is the complex envelope of the signal propagating in fiber as a function of time t and distance z. Here, $\mathcal{L}$ is the fiber length, α and β_2 are respectively attenuation and chromatic dispersion coefficients, and γ is the nonlinearity parameter.

The fiber-optic link is typically divided into a number of so-called spans. Amplification is performed using Erbium-doped fiber amplifiers (EDFA) after each span to compensate for the span loss.

The NLS equation is numerically solved using SSFM. The algorithm can be used to reverse the deterministic effects of the channel by calculating the evolution of the signal in backward direction using the negated channel parameters. The BER performance of this approach, called digital back-propagation [1], is extensively studied [21].

In SSFM, after conceptually splitting the fiber into M spans of length L_{sp}, the evolution of the signal in each span is calculated as follows: both space and time are uniformly discretized into the sets $\{z_0, z_1 \ldots, z_K\}$ and $\{t_0, t_1, \ldots, t_{\mathfrak{L}}\}$, respectively, i.e. $z_k = z_0 + \Delta_z k$ and $t_u = t_0 + \Delta_t u$, where $K = N_{stps}$. The evolution of the signal $\underline{q}(z)$ from the position z_0 to position z_K is computed as follows:

$$\underline{q}(z_{k+1}) = \mathbf{F}^\dagger D_L \mathbf{F} D_N \underline{q}(z_k), \quad (2)$$

where $\underline{q}(z_k)$ is the $\mathfrak{L} \times 1$ length-vector of sample values $q(t_u, z_k)$, $u = 0, 1, \ldots, \mathfrak{L}-1$, at position z_k, $\mathbf{F}$ and $\mathbf{F}^\dagger$ are respectively discrete Fourier transform (DFT) and inverse DFT (IDFT) matrices, D_N is a diagonal matrix operator with entries

$$e^{j\gamma|q(t_u,z_k)|^2\Delta z}, \quad u = \mathfrak{L}/2, \ldots, \mathfrak{L}-1, \quad (3)$$

and D_L is a diagonal matrix with entries:

$$\begin{aligned} &e^{-(\frac{\alpha}{2}+j\frac{\beta}{2}u^2/(\mathfrak{L}\Delta_t)^2)\Delta_z}, && u = 0, 1, \ldots, \mathfrak{L}/2-1, \\ &e^{-(\frac{\alpha}{2}+j\frac{\beta_2}{2}(\mathfrak{L}-u)^2/(\mathfrak{L}\Delta_t)^2)\Delta_z}, && u = \mathfrak{L}/2, \ldots, \mathfrak{L}-1\,. \end{aligned} \quad (4)$$

The EDFA after each span is imitated by applying the multiplicative factor $G^{-1} = e^{\frac{\alpha}{2}L_{sp}}$ to the signal, followed by a

$\mathfrak{L} \times 1$ length-vector of zero-mean white circularly symmetric complex Gaussian noise with the power spectral density σ^2 derived from the amplifier noise figure.

Fig. 1 depicts the schematic of the communication system under consideration. First, the input bit-stream block $\mathbf{m} = (m_1, m_2, \ldots, m_{N_b})$, $m_i \in \{0, 1\}$, is mapped to a sequence of symbols $\mathbf{S} = (s_1, s_2, .., s_{N_s})$, where s_i are drawn from a QAM-constellation. Next, the sequence of symbols is transformed to the waveform $q(t, 0) = \sum_{i=-\infty}^{\infty} s_i p(t - i/R_s)$, where $p(t)$ is the pulse shape and R_s is the baud rate. At RX, following the sampling of the received waveform $q(t, \mathcal{L})$, the corresponding vector $\mathbf{x}$ is passed to a DSP unit to retrieve the original sequence of symbols. The retrieved sequence of symbols and the corresponding bit-stream at RX are denoted by $\hat{\mathbf{S}}$ and $\hat{\mathbf{m}}$, respectively.

The goal is to implement an efficient neural network-based DSP unit in Fig. 1 in terms of complexity-performance trade-off. The computational complexity is measured as the number of FLOPs, and performance is evaluated using BER defined as

$$\text{BER} = \frac{1}{N_b} \sum_i \mathbb{1} \ \{m_i \neq \hat{m}_i\}, \tag{5}$$

where $\mathbb{1}$ is the indicator function, equal to 1 if the bracket condition is met and 0 otherwise. N_b is the number of bits (bit-stream size). Another performance metric is the effective signal-to-noise ratio (SNR), defined as

$$\text{effective SNR} = \frac{||\mathbf{S}||_2^2}{||\mathbf{S} - \hat{\mathbf{S}}||_2^2}, \tag{6}$$

where $||\mathbf{S}||_2 = (\Sigma |s_i|^2)^{1/2}$ is the L^2-norm of $\mathbf{S}$.

III. Proposed Hybrid CNN+RNN Architecture

The motivation underlying our approach is to use the properties of the channel memory and adapt the network architecture to obtain a less complex model. This is achieved by leveraging a many-to-one recurrent layer, thanks to its hidden state property. However, a purely recurrent architecture is not efficient either. This is due to the corresponding computational complexity in the case of passing raw data to the recurrent layer, and also the inefficiency of the RNN in capturing the short-temporal dependencies. We suggest that the many-to-one recurrent and strided convolution layers should be combined suitably to efficiently represent the memory and capture the dependencies within the sampled waveform.

Fig. 2 depicts the proposed neural network architecture. This architecture consists of an initial reshaper mapping the complex-valued input vector to two parallel channels containing the real and imaginary parts. The output of the reshaper is passed to a cascade of strided convolution layers (in our implementation, we set it to 3 layers), each followed by an activation function. The low-complexity ReLU activation is adopted in this respect. This is as against the relatively high-complexity model-based activation $\sigma_\ell(x) = xe^{-j\gamma_\ell |x|^2}$ (γ_ℓ is a trainable parameter for the layer ℓ) used in LDBP, which is adapted from Kerr induced nonlinearity effect of the fiber-optic channel. Each strided convolution layer captures the short-temporal dependencies within the data and reduces the dimensionality by taking the sampled waveform to a latent space. Considering that the dimensionality reduction can only be applied to an extent, reducing the length of the input matrix

TABLE I: Fiber and Noise Parameters

a_{dB}	0.2 dB/km	fiber loss
D	17 ps/(nm-km)	chromatic dispersion
γ	1.4 $\text{W}^{-1}\text{km}^{-1}$	nonlinearity parameter
h	$6.626 \times 10^{-34}\,\text{J} \cdot \text{s}$	Planck's constant
c	3×10^8	speed of light
λ_0	1.55 μm	carrier wavelength
L_{sp}	100 km	span length
NF	5 dB	EDFA noise figure

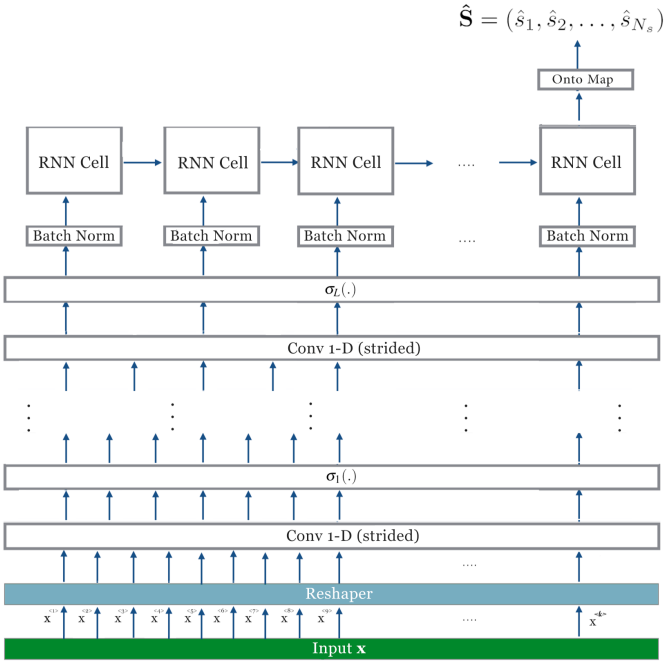

Fig. 2: The architecture of the proposed CNN+RNN model.

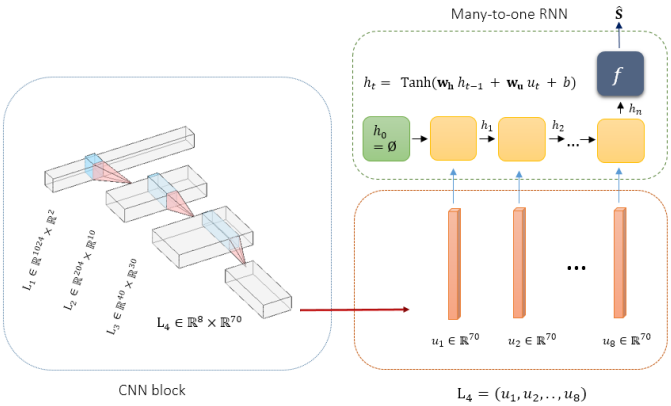

Fig. 3: The process flow in the CNN+RNN model. Each array in the depth axis of the resulting feature map output by the CNN block is passed to one RNN cell.

TABLE II: The complexity measures for the implemented LDBP, CNN+MLP, RNN, and the proposed CNN+RNN

Model	#Params /symbol	#FLOPs /symbol	Memory requirment (training mode)	#Layers
LDBP-2StPS	0.17K	$\sim$ 750K	$>$ 1GB	120
MLP	37K	$\sim$ 37K	80MB	3
CNN+MLP	4.6K	$\sim$ 12.6K	148MB	4
RNN	0.8K	$\sim$ 13K	92MB	1
CNN+RNN	0.75K	$\sim$ 8.5K	107MB	4

at each convolution layer should be accompanied by increasing the depth of the matrix but in a smaller ratio. This is carried out by applying the same number of filters as the desired depth on the input matrix.

Following the CNN block, the batch-normalization layer normalizes the resulting feature map based on a learned optimal mean and variance. This has a positive contribution in speeding up the convergence rate of the model. The normalized feature map is then passed to a many-to-one recurrent layer, such that each array in the depth axis is passed to one RNN cell.

Each RNN cell in the recurrent layer processes the input vector while it also considers the information received from the previous cells. The hidden state of the final RNN cell is subsequently passed to a linear mapping $f(\mathbf{x}) = \alpha\mathbf{x}+\mu$, where $\alpha, \mu \in \mathbb{R}$ are trainable parameters, to output the sequence of symbols $\hat{\mathbf{S}} = (\hat{s}_1, \hat{s}_2, \ldots, \hat{s}_{N_s})$. It is important to note that the model outputs two values for each $\hat{s}_i$, one for the real part and one for the imaginary part. Fig. 3 depicts an illustrative description of the process flow in the proposed CNN+RNN model.

We highlight that since no information should be forgotten or reset, in the recurrent layer pipeline, from the first RNN cell to the last, there is no need to use LSTM or GRU cells. Furthermore, as the short-temporal dependencies have already been captured and the neighboring symbols are efficiently grouped via the CNN block, the number of time-steps passed to the recurrent layer is quite limited and can be managed well by simple RNNs.

IV. Numerical Results

We consider a single-polarization 16-QAM 100 G symbol/s point-to-point fiber-optic communication system over 20x100km SMF optical-link, according to the system model illustrated in Fig. 1. The fiber and noise parameters are mentioned in Table. I. In this system, root-raised cosine (RRC) filters, with roll-off of 0.1, were used for pulse shaping. Forward propagation was simulated using 8 SpS and 50 StPS in SSFM (increasing either value did not affect the results). The sampling rate at RX was also set to 8 SpS.

A. Comparison of the neural equalizers

Equalization was considered using CD compensation (CDC), DBP, LDBP, MLP, CNN (followed by fully connected layers), RNN, and the proposed hybrid CNN+RNN models. These models were trained over 120 epochs on a dataset containing 2^{16} signals of size 2^7 symbols, i.e., $N_b = 2^9$, divided into mini-batches of size 2^9. A validation set of size 2^{13} signals was used during training, and a test set of size 2^{20} was leveraged to evaluate the models. All the models were created and trained using Tensorflow 2.0 in Python. For all the models, the loss function was set to mean absolute error (MSE). For all the models, the optimization algorithm was set to Adam with the learning rate of 0.001, having the reduce on plateau property with the patience of 7 and the factor of 0.7, $\beta_1 = 0.85$, and $\beta_2 = 0.999$. The source code of the simulated system is accessible at [22].

The models that are used in the comparison are described here. LDBP-2StPS is a fully CNN approach, as it is discussed in [13], with 3 layers (2 linear and 1 nonlinear) per step, resulting in 120 layers in total for our scenario. In LDBP, the input to the neural network is the sampled waveform at

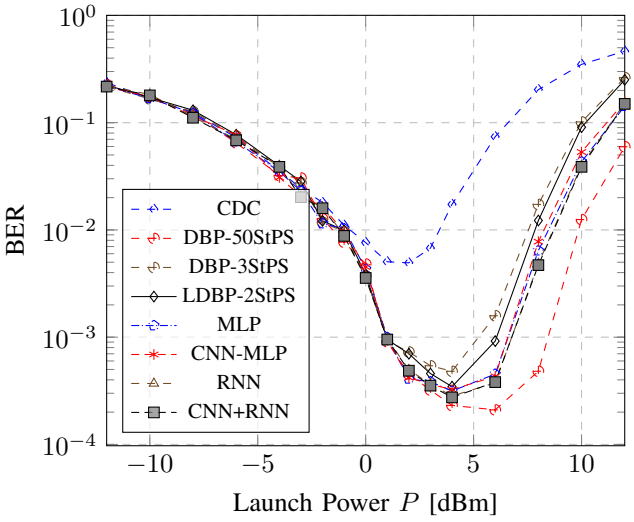

Fig. 4: The BER performance of the models as a function of the launch power.

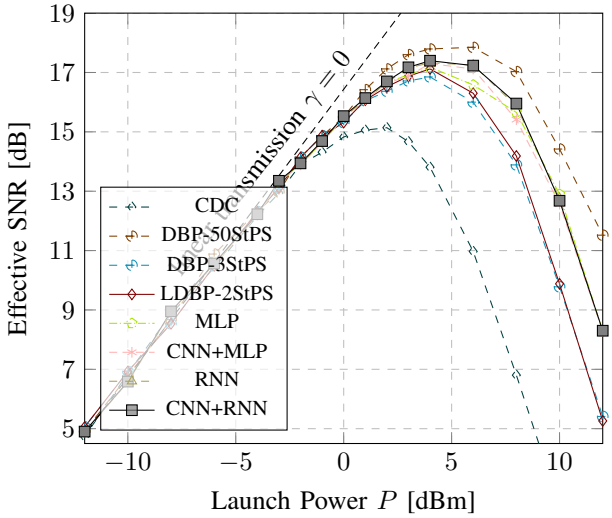

Fig. 5: The effective SNR of the models as a function of the launch power.

RX, and the output is the equalized waveform. On the other hand, the MLP, CNN, RNN, and CNN+RNN models were trained in an end-to-end learning fashion, based on the pairs of the sampled waveform at RX after CD compensation and the corresponding transmitted symbols. The MLP has 2 hidden layers, similar to [7], [23], with the sizes $2 \times N_s \times f_{sps}$ and $2 \times N_s$, respectively, where f_{sps} is the sampling rate per symbol. The CNN+MLP is a CNN model with 4 hidden layers consists of 3 non-strided convolution layers, similar to [16], followed by a fully-connected layer. The kernel size for the CNN layers was set to 192 to cover 12 adjacent symbols. The output layer of the MLP and CNN+MLP models is a fully-connected layer, without activation, which has two units per output symbol, one for the real part and one for the imaginary. The RNN is a many-to-one RNN architecture consists of simple RNN cells. The input size to each RNN cell is $16 \times f_{sps}$. The activation function in MLP, CNN+MLP, and RNN models is Tanh (it was observed in the simulations to have a better performance than sigmoid, ELU, and ReLU activations). The CNN+RNN consists of a cascade of 3 strided convolution layers (followed by ReLU activations) with both kernel size and stride of 5 for all the layers and the depths of 10,30,70, respectively. A many-to-one RNN layer follows the CNN block with Tanh activation.

The resulting BER and effective SNR plots of the models are illustrated in Fig. 4 and Fig. 5, respectively. Expectedly, all the models (excluding CDC, low complexity DBPs, and the corresponding LDBPs) reached roughly the same level of error performance. However, their complexities are quite different, as shown in Table. II.

It follows from Fig. 5 and Table. II that the proposed CNN+RNN model achieves the error performance of the other models with respectively, 98%, 77%, 32%, and 34% fewer FLOPs than LDBP, MLP, CNN, and RNN models.

The learning curve of the models is also depicted in Fig. 6. According to this figure, the CNN+RNN model takes more epochs to converge in the training process compared to the other models. However, training is an offline process, and therefore training convergence speed is a quite low-impact criterion for model adoption.

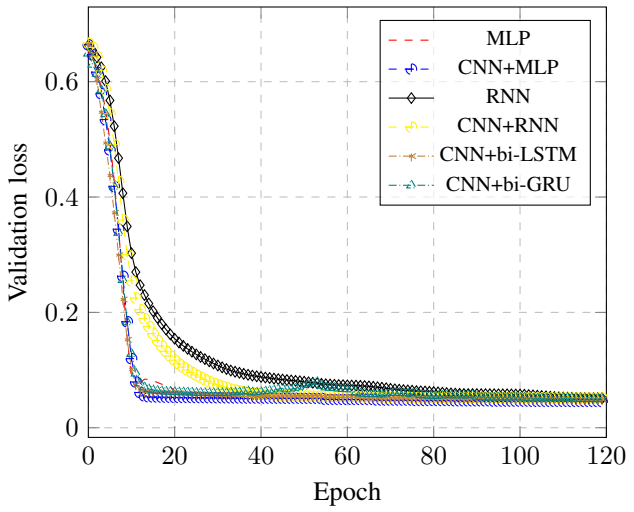

Fig. 6: Learning curve of the models. Resulting loss (MSE) at optimal launch power (4 dBm) on the validation dataset as a function of the number of epochs.

B. The benefits of RNN

As discussed, the lower complexity of the CNN+RNN model is owing to the efficient distribution of the computational load between the CNN and RNN blocks such that each block is primarily responsible for capturing the dependencies it is more efficient at capturing. For example, in the CNN+MLP approach, a kernel size of 192 had to be used to capture the long-range temporal dependencies. This results in a high memory requirement and leads to inefficiency in the number of FLOPs. On the other hand, in the pure RNN approach, a high computational load is tolerated to capture the short temporal dependencies, which is inefficient.

The memory requirement mentioned in Table. II is obtained by analyzing the training process of the models on an Intel(R) Core(TM) i5-2410 CPU @ 2.30 GHz 2.30 GHz, 16.00 GB RAM, Win10 PC 64-bit system. As it is typically envisioned, MLP has a lower memory requirement for training than the other models owing to its relatively lightweight computations for the back-propagation process; whereas a CNN model, due to the computational complexity of the back-propagation for the convolutional layers, and moreover, the size of the kernel, consumes a noticeable amount of memory. The RNN model, on another side requires a lower memory requirement for training than the CNN due to the simple dynamics of the simple RNN layer (similar to the MLP) for the back-propagation. However, the lower memory requirement of the MLP and the RNN models accompanies by a higher number of FLOPs in a disproportionate manner. Furthermore, they do not have the same parallel processing capability as CNNs.

The CNN+RNN model captures the so-called equilibrium point in the memory-FLOPs dilemma. In the problem under our consideration, it reduces the number of FLOPs by 34% compared to the pure RNN approach using 16% higher memory. Plus that it facilitates the possibility of parallel processing in the CNN layers. Furthermore, note that memory is a highly more available resource than the processing speed.

Because no information is supposed to be forgotten or rest in the recurrent layer pipeline, we discussed that there is no need to use LSTM or GRU units in the recurrent layer. However, we tested the leveraging of LSTM and GRU cells to check this. In addition, we also investigated the performance of bidirectional recurrent layers. These models were designed based on the many-to-many architecture followed by a flattening and fully-connected (without activation) layers, similar to [9], [19].

Fig. 7 shows the resulting effective SNR plot of the CNN+LSTM, CNN+GRU, CNN+biLSTM, CNN+biGRU, and CNN+biSimpleRNN based equalizers. As anticipated, using LSTM and GRU layers led to no gain over using simple RNN. Moreover, no considerable performance improvement was

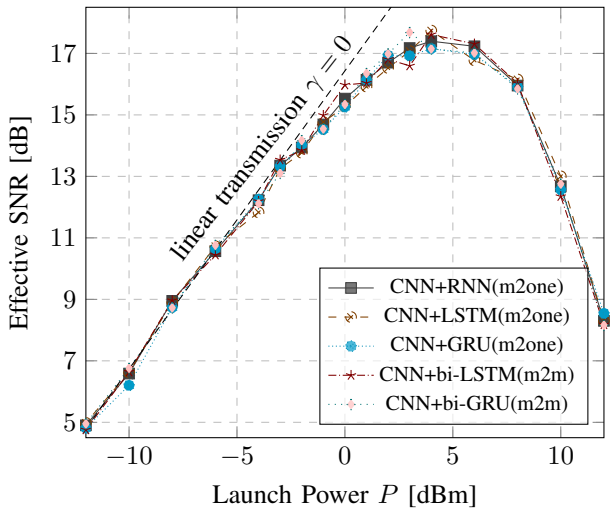

Fig. 7: The effective SNR of the CNN+(bi)RNN models as a function of the launch power. m2m and m2one respectively signifies many-to-many and many to one architecture.

observed for using many-to-many bi-RNNs in lieu of many-to-one uni-RNNs, despite the fact that the former architecture brings about higher computational complexity. We analyze the reason for this is that in the many-to-one architecture, all the information is gathered (by leveraging a memory emulated by the hidden state property of the RNNs) and processed collectively, without the need to propagate information in the reverse direction; whereas, in the many-to-many architecture, each RNN cell in the layer requires to know the state of the previous and next RNN cells in the layer, in order to output an accurate result.

V. Conclusion

We proposed a lightweight end-to-end hybrid CNN+RNN based equalizer trained on the pairs of the sampled waveform at RX after CD compensation and the corresponding transmitted symbols. In the proposed approach, the learning task was apportioned into capturing long-range dependencies using RNNs and short-term features using CNNs with ReLU activation. A many-to-one architecture was implemented for the recurrent layer to emulate a memory enabling efficient parameter sharing. Striding was also exploited in the CNN block to reduce the dimensionality of the data prior to the recurrent layer. It was demonstrated that for a 16-QAM transmission at 100 G symbol/s over a 20 x 100km SMF optical-link, the CNN+RNN model achieves the performance of DBP with around 98% fewer FLOPs than the recently-proposed model-driven LDBP approach. We furthermore showed that the proposed CNN+RNN model is computationally more efficient than using non-model-driven end-to-end MLP, CNN+MLP, RNN, and biRNN models for the same deep learning-based equalization task.

References

[1] E. Ip and J. M. Kahn, "Compensation of dispersion and nonlinear impairments using digital backpropagation," *J. Lightw. Technol.*, vol. 26, no. 20, pp. 3416–3425, 2008.

[2] G. Kramer, M. I. Yousefi, and F. R. Kschischang, "Upper bound on the capacity of a cascade of nonlinear and noisy channels," in *Proc. IEEE Inf. Theory Workshop*, 2015, pp. 1–4.

[3] M. Secondini, D. Marsella, and E. Forestieri, "Enhanced split-step Fourier method for digital backpropagation," in *Proc. Eur. Conf. Opt. Commun. (ECOC)*. IEEE, 2014, pp. 1–3.

[4] A. Napoli, Z. Maalej, V. A. Sleiffer, M. Kuschnerov, D. Rafique, E. Timmers, B. Spinnler, T. Rahman, L. D. Coelho, and N. Hanik, "Reduced complexity digital back-propagation methods for optical communication systems," *J. Lightw. Technol.*, vol. 32, no. 7, pp. 1351–1362, 2014.

[5] R. M. Butler, C. Hager, H. D. Pfister, G. Liga, and A. Alvarado, "Model-based machine learning for joint digital backpropagation and PMD compensation," *J. Lightw. Technol.*, pp. 1–1, 2020.

[6] C. Häger and H. D. Pfister, "Deep learning of the nonlinear Schrödinger equation in fiber-optic communications," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*. IEEE, 2018, pp. 1590–1594.

[7] C. Catanese, R. Ayassi, E. Pincemin, and Y. Jaouën, "A fully connected neural network approach to mitigate fiber nonlinear effects in 200G DP-16-QAM transmission system," in *Proc. Int. Conf. TRN. Opt. Netw. (ICTON)*, 2020, pp. 1–4.

[8] T. Koike-Akino, Y. Wang, D. S. Millar, K. Kojima, and K. Parsons, "Neural turbo equalization: Deep learning for fiber-optic nonlinearity compensation," *J. Lightw. Technol.*, vol. 38, no. 11, pp. 3059–3066, 2020.

[9] S. Deligiannidis, A. Bogris, C. Mesaritakis, and Y. Kopsinis, "Compensation of fiber nonlinearities in digital coherent systems leveraging long short-term memory neural networks," *J. Lightw. Technol.*, vol. 38, no. 21, pp. 5991–5999, 2020.

[10] F. Musumeci, C. Rottondi, A. Nag, I. Macaluso, D. Zibar, M. Ruffini, and M. Tornatore, "An overview on application of machine learning techniques in optical networks," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 2, pp. 1383–1408, 2019.

[11] A. Balatsoukas-Stimming and C. Studer, "Deep unfolding for communications systems: A survey and some new directions," in *Proc. IEEE Int. Workshop Signal Process. Syst. (SiPS)*. IEEE, 2019, pp. 266–271.

[12] C. Catanese, A. Triki, E. Pincemin, and Y. Jaouën, "A survey of neural network applications in fiber nonlinearity mitigation," in *Proc. Int. Conf. Trn. Opt. Netw. (ICTON)*. IEEE, 2019, pp. 1–4.

[13] C. Häger and H. D. Pfister, "Nonlinear interference mitigation via deep neural networks," in *Proc. Opt. Fiber Commun. Conf. Expo. (OFC)*, 2018, pp. 1–3.

[14] O. Sidelnikov, A. Redyuk, S. Sygletos, M. Fedoruk, and S. Turitsyn, "Advanced convolutional neural networks for nonlinearity mitigation in long-haul WDM transmission systems," *J. Lightw. Technol.*, vol. 39, no. 8, pp. 2397–2406, 2021.

[15] P. Li, L. Yi, L. Xue, and W. Hu, "56 Gbps IM/DD PON based on 10G-class optical devices with 29 dB loss budget enabled by machine learning," in *Proc. Opt. Fiber Commun. Conf. Expo. (OFC)*, 2018, pp. 1–3.

[16] C. Chuang, L. Liu, C. Wei, J. Liu, L. Henrickson, W. Huang, C. Wang, Y. Chen, and J. Chen, "Convolutional neural network based nonlinear classifier for 112-Gbps high speed optical link," in *Proc. Opt. Fiber Commun. Conf. Expo. (OFC)*, 2018, pp. 1–3.

[17] P. Li, L. Yi, L. Xue, and W. Hu, "100Gbps IM/DD transmission over 25km SSMF using 20G-class DML and PIN enabled by machine learning," in *Proc. Opt. Fiber Commun. Conf. Expo. (OFC)*, 2018, pp. 1–3.

[18] B. Karanov, M. Chagnon, V. Aref, D. Lavery, P. Bayvel, and L. Schmalen, "Optical fiber communication systems based on end-to-end deep learning : (invited paper)," in *Proc. IEEE Photon. Conf. (IPC)*, 2020, pp. 1–2.

[19] X. Liu, Y. Wang, X. Wang, H. Xu, C. Li, and X. Xin, "Bi-directional gated recurrent unit neural network based nonlinear equalizer for coherent optical communication system," *Optics Express*, vol. 29, no. 4, pp. 5923–5933, 2021.

[20] P. J. Freire, Y. Osadchuk, B. Spinnler, A. Napoli, W. Schairer, N. Costa, J. E. Prilepsky, and S. K. Turitsyn, "Performance versus complexity study of neural network equalizers in coherent optical systems," *arXiv:2103.08212*, 2021.

[21] R. Dar and P. J. Winzer, "Nonlinear interference mitigation: Methods and potential gain," *J. Lightw. Technol.*, vol. 35, no. 4, pp. 903–930, 2017.
[22] (2021) Optical fiber channel simulation using split-step Fourier method. Telecom Paris. [Online]. Available: https://github.com/FONTE-EID/fiber-optic-channel-simulation
[23] O. Sidelnikov, A. Redyuk, and S. Sygletos, "Equalization performance and complexity analysis of dynamic deep neural networks in long haul transmission systems," *Optics express*, vol. 26, no. 25, pp. 32 765–32 776, 2018.