Equalization performance and complexity analysis of dynamic deep neural networks in long haul transmission systems

OLEG SIDELNIKOV,[1,2,*] ALEXEY REDYUK,[1,2] AND STYLIANOS SYGLETOS[3]

[1]*Institute of Computational Technologies SB RAS, Novosibirsk, Russia*
[2]*Novosibirsk State University, Novosibirsk, Russia*
[3]*Aston Institute of Photonic Technologies, Aston University, Birmingham, UK*
[*]*o.s.sidelnikov@gmail.com*

Abstract: We investigate the application of dynamic deep neural networks for nonlinear equalization in long haul transmission systems. Through extensive numerical analysis we identify their optimum dimensions and calculate their computational complexity as a function of system length. Performing comparison with traditional back-propagation based nonlinear compensation of 2 steps-per-span and 2 samples-per-symbol, we demonstrate equivalent mitigation performance at significantly lower computational cost.

1. Introduction

Optical fibre communication systems face an enormous challenge of extending their capacity limits to deal with the exponential growth of internet data traffic [1]. Unlike linear channels where the capacity can be always improved by increasing the transmitted signal power [2], fiber-optic channels are non-linear, thus a power increase may create additional sources of distortion that degrade signal quality and cause loss of information [3]. These nonlinear distortions can become even more detrimental as we scale towards denser spectral efficiencies and higher number of launched optical channels. Therefore, signal transmission in the non-linear regime requires the development of new methods to mitigate non-linear impairments and to enable a substantial increase of system capacity.

With the availability of high speed digital signal processing (DSP) a number of compensation methods have been proposed in the electronic domain to deal with fibre non-linearities [4–13]. Most of them emulate an inverse fibre link propagation by means of a split-step Fourier (SSF) [6] method or Volterra Series Transfer Functions [7, 8] to counteract the non-linear interference accrued by the received signal. However, both methods require multiple computational steps along the link and as they need prior knowledge of the optical path's parameters, they can be applied only in static connections [9]. Although many research efforts have been devoted in improving their computational efficiency [10, 13], digital back propagation (DBP) methods are still far from real-time implementation.

The field of machine learning (ML) offers powerful statistical signal processing tools for the development of adaptive equalizers capable of dealing with nonlinear transmission effects. Contrary to back-propagation based reception, in machine learning the signal equalization and demodulation processes are treated jointly as a classification or regression problem by mapping the baseband signal onto a space determined by the direct interpretation of a known training sequence. This can bring an efficient adaptive performance and a significant reduction in the required number of computational steps potentially supporting real-time implementation. In addition, machine learning based equalizers can be periodically re-trained, which makes them suitable for operation in dynamically reconfigurable transmission environments.

Although machine learning based equalization techniques have been extensively studied in wireless systems [14, 15], only lately they have been considered for application in fibre transmission systems. A number of techniques, such as, k-nearest neighbors algorithm [16], affinity propagation clustering [17], statistical sequence equalizers [18], expectation maximization algorithms with Gaussian mixture models [19] and support vector machines [20–22], have been proposed combining the nonlinear equalization (NLE) functionality with optimum symbol classification. This means that they can adapt their decision boundaries to the residual nonlinear distortion of the received signal instead of performing hard decision. Therefore, such equalizers can achieve significant performance improvement, especially when they are used in memoryless systems, or dealing with nonlinear phase noise effects. A different category of machine learning algorithms treats nonlinear equalization as a regression problem by creating the inverse transfer function of the nonlinear link. In this category belong the neural network (NN) schemes [23–27], which have been mostly studied for application in orthogonal frequency division multiplexing (OFDM) systems. In [25] a NN based radial basis functions was used. In [26] authors made use an artificial neural network together with a genetic algorithm to compensate for non-linearity in dispersion-shifted, dispersion-managed and dispersion-unmanaged coherent optical communication links. Finally in [27], NN-based equalization was studied for application in short-reach Intensity-Modulation Direct-Detection (IM/DD) systems. All previous methods were based on static neural networks, and as such, they could demonstrate a substantial performance improvement only in short memory transmission systems. The application of NN for nonlinear equalization in typical wavelength-division multiplexing transmission systems characterized by long memory depth and high modulations rates has not been adequately explored.

In this paper we investigate the performance of deep neural networks (or multi-layer perceptron networks) in long haul transmission systems and we compare it with the linear compensation, as well as with digital back-propagation. Our results show that a static neural networks equalizer is unable to compensate the nonlinear channel response and outperform the linear equalizer. On the contrary, when using a dynamic neural network (dNN) architecture, we were able to calculate a Q^2-factor improvement of 1.5 dB for single channel transmission and of 1.4 dB for multi channel transmission, along a 1000 km fibre link. The required number of taps at the input of the NN has been identified as a function of the total system length. We also conduct extensive analysis of computational complexity of deep neural network training and prediction and show a significant superior performance of the NN scheme against conventional DBP methods.

2. Transmission system model

The simulated transmission link is depicted in Fig. 1. Single channel and five channel transmission scenarios were investigated. Each transmitter generated 16-QAM modulated root raised cosine pulses at 32 GBaud, with a Gray-coded constellation diagram, a roll-off factor of 0.001 and an oversampling factor of 16. In the multi-channel case the frequency spacing between the channels was equal to the reciprocal of the baudrate (i.e. 32 GHz). The central wavelength of the emitted signal band was at λ = 1550 nm.

The generated signals were subsequently launched into a transmission link that consisted of N-spans of 100 km single mode fibre each. An EDFA of 4.5 dB noise figure compensated the losses of each span. Signal propagation in a single polarization was considered, simulated by a typical symmetrized split-step Fourier method. The rest of the fibre link's parameters were: fibre loss α = 0.2 dB/km, dispersion D = 17 ps/(km-nm) and nonlinear factor γ=1.4 W^{-1}km^{-1}. After transmission the signals were coherently detected. Each channel was selected by a root raised cosine filter of the same roll-off factor as that of the transmitter and down-sampled to 2 samples per symbol. Then a linear equalization stage enabling ideal compensation of chromatic dispersion effects followed. After down-conversion to single sample per symbol the nonlinear equalization took place by means of a dynamic deep neural network architecture.
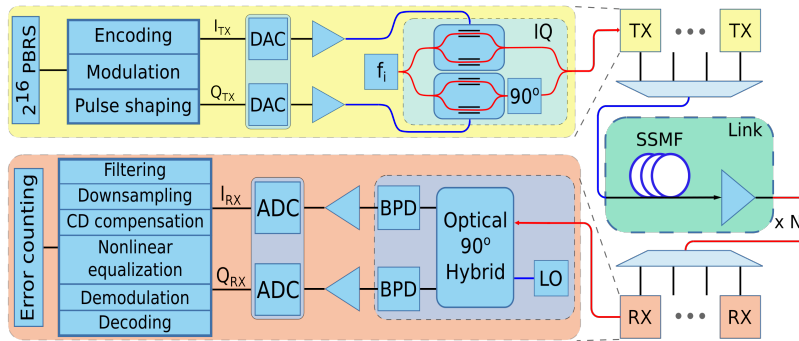
Fig. 1. Scheme of simulated transmission link.

Figure 2 shows the deep neural network architecture used in the work. Contrary to previous approaches [23] that employed separate neural networks for the real and imaginary part of the signal, here both signal features were fed into the same topology, reducing significantly the computational complexity. To take into account the channel memory effect we used delay blocks at the input of the NN architecture, so the overall neural network scheme was dynamic. Thus, for the equalization of each received symbol the preceding symbols in the stream were also used. Next, the received symbols were divided into their real and imaginary parts and forming the feature vector of the neural network. The size of the input layer was $2(N_{del} + 1)$, where N_{del} the number of delay blocks. The network had also two hidden layers of 16 neurons each and an output layer of two neurons, i.e. one for the real and one for the imaginary output. The hidden layer neurons had a hyperbolic tangent sigmoid transfer functions, whereas the neurons of the output layer had a linear transfer function. It should be noted that we didn't use bias nodes on any of the layers of neural network. Training was based on the Riedmiller's resilient-back propagation (Rprop) algorithm [28]. Although this method is more complex to implement, it is often faster than training with back propagation and it doesn't require to specify any free parameter values. The neural network equalizer had to be retrained for each launched power level to be able address the different nonlinear properties of the transmission channel.
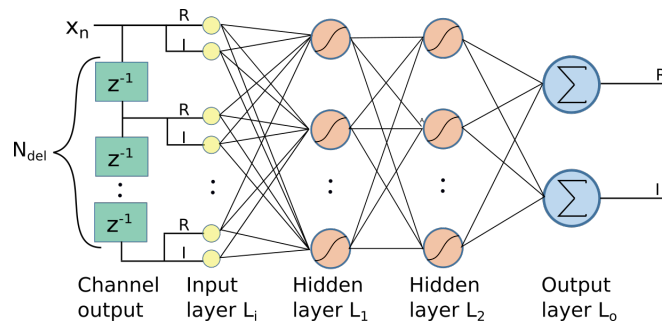

Fig. 2. Deep neural network architecture. The size of the input layer (L_i) is n_i, and the two hidden layers (L_1, L_2) and the output layer (L_o) have n_1, n_2 and n_o neurons, respectively

After the nonlinear equalization, the demodulation and signal decoding functionalities took place and finally the BER calculation. Every BER point was derived by averaging the error rate of 15 signal block transmissions, each containing 2^{16} symbols. From this block size, 2^{12} symbols were used for training (70%) and validation test (30%) and the remaining for the error rate calculation.

3. Numerical results

The role of the NN equalizer is to create an efficient inversion of the nonlinear transmission channel. Therefore, the first step in our study was to identify the optimum NN dimensions and how they are affected in different transmission scenarios. The number of delay block N_{del} at the NN input and the number of hidden layer neurons were the two critical dimensioning parameters of the equalizer. N_{del}, determined the ability of the equalizer to deal with degradations of finite time dependent response, whereas the number of hidden layer neurons the ability to approximate highly nonlinear channel inversions.

For the dimensioning, we considered single channel transmission along fibre links of 16, 20 and 25 spans (i.e. 1600 km, 2000 km and 2500 km total length). At the point of optimum launched power, identified assuming linear equalization at the receiver, we calculated the BER as a function of the number of delay taps, see Fig. 3(a). As we increased the number of delay taps the equalization performance improved until a starting point of a BER floor region where there was no need to further increase the complexity of the equalizer. This point defined an optimum number of delay-taps for the specific link length. Repeating the same optimization procedure we mapped the optimum N_{del} as a function of the transmission link's length, see Fig. 3(b). We see clearly a linear dependence of the required delays taps to the number of spans. So, for a 1600 km link we need at least 36 taps, while for 2600 km the required number of taps becomes 51.
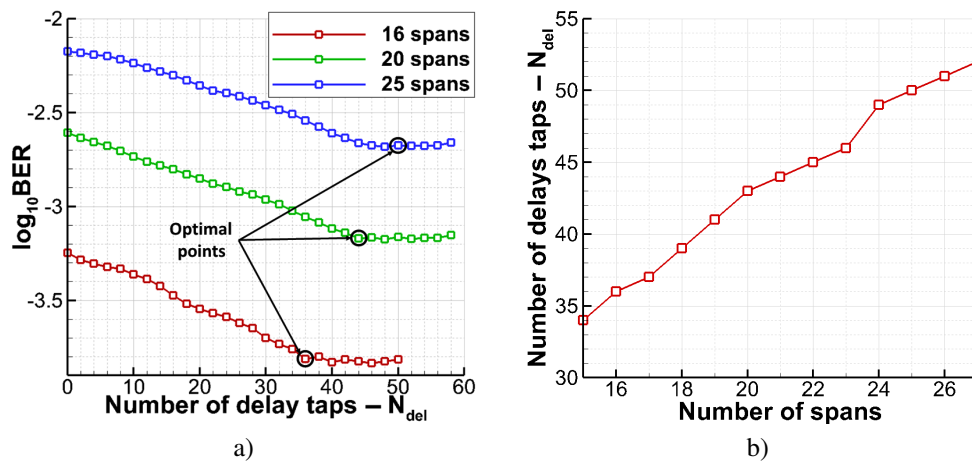

Fig. 3. (a) BER as a function of number of delay taps N_{del} for different number of spans; (b) number of required delay taps N_{del} a function of number of spans in transmission links. Span length equals 100 km.

In the aforementioned simulations we had considered each of the two hidden layer of the NN having 16 neurons. We tested the equalization performance also with a different number of neurons. Figure 4(a) shows the BER performance as a function of the launched signal power for the cases of 4, 8, 16 and 20 neurons at each of the two hidden layers. The results corresponded to a transmission link of 2000 km (i.e. 20 spans) and in all three cases we used 43 delay taps in accordance with Fig. 3(b). We notice a substantial decrease of the equalization performance when the number of neurons per hidden layer drops from 16, to 8 or 4, while the cases of 16 and 20 neurons provide practically the same results. For comparison, we investigated also the case of DBP based equalization of 2 steps-per-span. We see that the DBP based equalizer outperforms a neural network of 4 or 8 neurons at each hidden layer, whereas a dNN based equalizer of 16 or 20 neurons per layer provides better performance than the DBP. We also tested the equalization performance of the dNN for different number of hidden layers. Figure 4(b) shows the BER
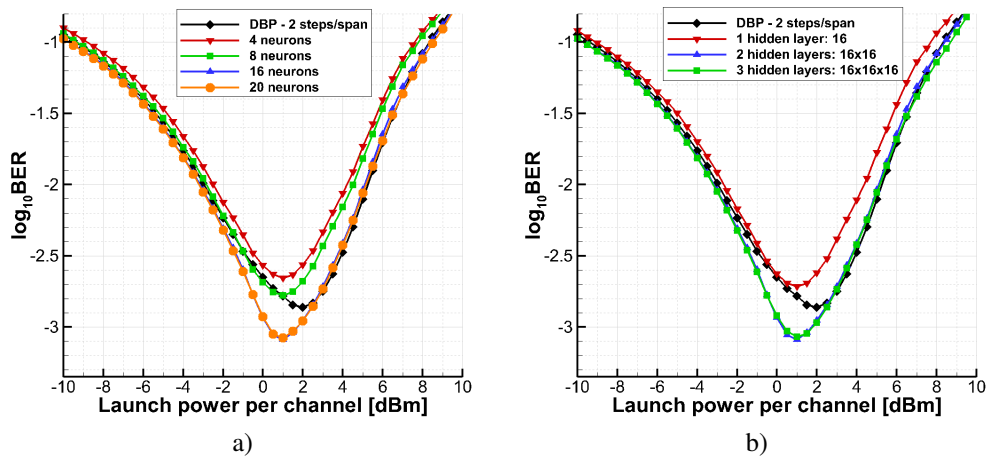
Fig. 4.   (a) BER as a function of launch power for different number of neurons per layer; (b) BER as a function of launch power for different number of hidden layers.

performance as a function of the launched signal power for the cases of one, two and three hidden layers with 16 neurons at each layer. Increasing the layer number from one to two improves significantly the BER performance, whereas a further increase in the number of layer does not give any BER improvement but only adds to the computational complexity. Therefore, for the rest of the simulation results two hidden layers of 16 neurons each were considered.
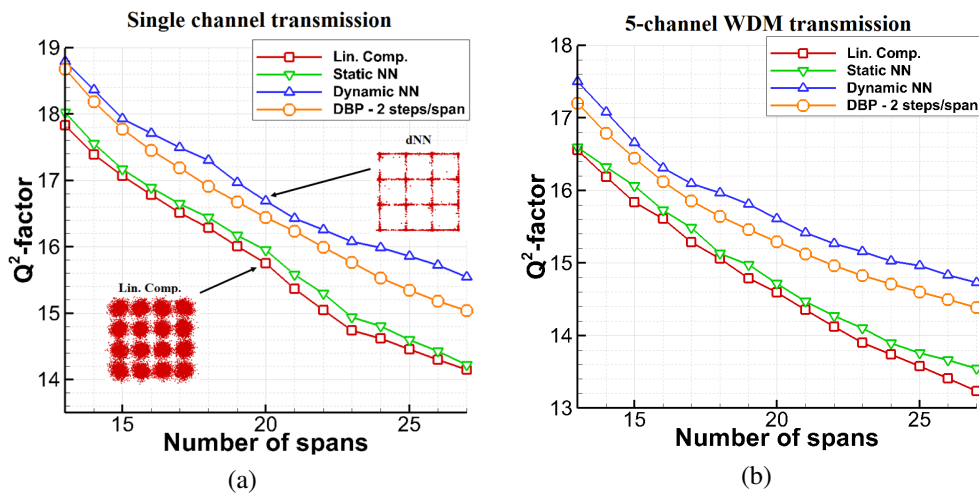

Fig. 5.   (a) Q^2-factor as a function of number of spans for single channel transmission; (b) Q^2-factor as a function of number of spans for 5-channel Nyquist-WDM transmission. Results correspond to the middle channel.

Having identified the optimum number of delay taps for each transmission distance we subsequently characterized the equalization performance of the dynamic NN scheme. Figure 5(a) shows the calculated Q^2-factors for a single channel transmission system as a function of the number of spans and for different equalization methods applied at the receiver. The Q^2-factor values have been extrapolated from the calculated BER according to [29]: $Q^2 = 20 \log_{10}\left[\sqrt{10}\,\text{erfc}^{-1}(8BER/9)\right]$ at the point of optimum launched power and for optimal number

of taps, as defined in Fig. 4. Obviously, the system with the linear compensator was the worst performing. On the other hand, when using a static deep neural network (i.e. without delay blocks), the achieved improvement was extremely small. A dynamic deep neural network with an optimally dimensioned number of delay taps gave an improvement between 1 dB and 1.5 dB when the system length varied between 1500 km and 2700 km. This was slightly higher than the performance of a symmetric digital back-propagation algorithm with 2 calculation steps per span and 2 samples per symbol. The received constellation diagrams, for the cases of linear and dNN based equalization taken at the point of optimum launched power after 2000 km of signal transmission, are shown in the inset of Fig. 5(a). Similar conclusions were drawn for the 5-channel Nyquist WDM transmission scenario in Fig. 5(b), where the use of an optimum NN equalizer applied on the middle channel of the transmission band gave 1.4 dB Q^2-factor improvement when compared to linear compensation and slightly better results than conventional DBP of 2 steps per span [30].

4. Computational complexity analysis

Subsequently we compared the computational complexity of a receiver that was based on the proposed dynamic NN architecture with a receiver that used the DBP method. The comparison was achieved in terms of the total number of real multiplications per transmitted bit required by each of the two nonlinear compensation schemes.

We start our analysis with the DBP based receiver. As mentioned above, the simplest implementation of the DBP algorithm was considered, where each propagation step comprised a linear part for dispersion compensation followed by a nonlinear phase cancellation stage. The linear part was achieved with a zero-forcing equalizer by transforming the signal in the frequency domain and multiplying with the inverse dispersion transfer function of the propagation section. For a signal block size of N-points this stage required $N \log_2 N$ complex multiplications for the implementation of the two FFT- transforms and N complex multiplications for the static equalization of the dispersion [31]. The frequency domain filtering of the signal block was achieved with an overlap-and-save method, which introduced a processing overhead of $N_D - 1$ samples. As a result, the complexity of the linear stage, defined by the number of complex multiplication per transmitted bit was written as:

$$C_{LinDBP} = \frac{N \left(\log_2 N + 1 \right) n_s}{(N - N_D + 1) \log_2 M}, \tag{1}$$

where n_s is the oversampling factor, M is the constellation order and $N_D = n_s \tau_D / T$, where τ_D corresponds to the dispersive channel impulse response and T is the symbol duration. The nonlinear compensation stage was performed in the time domain and required one complex multiplication per sample. For calculating the overall complexity of the DBP algorithm the total number of propagation steps $N_{Span} N_{StpSp}$ along the link was considered, where N_{Span} is the total number of spans and N_{StpSp} is the number of propagation steps per span. Furthermore, we multiplied by 4 to express the result in terms of real multiplications per transmitted bit, which gave :

$$C_{DBP} = 4 N_{Span} N_{StpSp} \left(\frac{N \left(\log_2 N + 1 \right) n_s}{(N - N_D + 1) \log_2 M} + n_s \right). \tag{2}$$

Subsequently, we evaluated the computational complexity of the deep neural network based receiver. As a single equalizer, this architecture can compensate both chromatic dispersion and fibre nonlinearity effects at the same time. However, this would increase significantly the network size, by requiring more delay taps at the input, and slow down the convergence of the training algorithm. Decoupling the mitigation of the linear from the non-linear effects can lead, instead, to faster and computationally more efficient equalization structure. In our case, compensation of

the accumulated chromatic dispersion along the transmission link was achieved before the neural network, with the use of a typical, zero-forcing, frequency domain equalizer (FDE). Since this was equivalent to the linear step of the DBP algorithm, except that the impulse response of the chromatic dispersion was for the whole transmission link, the corresponding complexity was given by :

$$C_{FDE} = 4 \frac{N \left(\log_2 N + 1 \right) n_s}{\left(N - N'_D + 1 \right) \log_2 M}. \tag{3}$$

The next step was to evaluate the computational complexity of the dynamic deep neural network, which dealt with the nonlinear degradations. Our calculations took into account the computational cost not only of the prediction phase in the neural network operation, where the signal equalization was performed, but also of the training phase. Generally, the neural network training is carried out in three steps. The first step invloves the random initialization of all the connection weights. In the second step, known as forward propagation, neuron activation takes place, starting from the input layer and moving towards the output. Finally, in the back-propagation step, the computation of the error as a sum-of-squares difference between the outputs and the targets is performed. The error is fed backwards through the network for updating the weights of the hidden layers and of the input layer, defining a cycle (i.e. single epoch), which is repeated until the error from the validation set reaches a point of minimum indicating over-fitting on the training set [32].

The random initialization of the neural network's weights was based on the Nguyen-Widrow algorithm [33] which provided conditions for fast training by selecting values that distributed the active region of each neuron approximately evenly across the layer's input space. The number of multiplications that were required for the activation of each neuron was equal to the number of input connections. Thus, to activate all the three layers (L_1, L_2 and L_o) of the neural network during a single epoch of the forward propagation step, we needed $n_i n_1 + n_1 n_2 + n_2 n_o$ real multiplications per sample. Since we had to consider the output samples of the whole training set, the total number of real multiplication in the forward propagation step was calculated as $(N_{ts} + N_{vs}) (n_i n_1 + n_1 n_2 + n_2 n_o)$, where N_{ts} and N_{vs} the number of samples in the training and validation sets, respectively.

Subsequently, we evaluated the number of multiplications required by the the Riedmiller's resilient back-propagation algorithm [28]. A main feature of this method was that the direction of the weight change was determined only by the sign of the partial derivative of the error with respect to the corresponding weight. The size of the weight update $\Delta \omega_{i,j}^{(t)}$ at each epoch was defined by the weight parameter $\Delta_{i,j}^{(t)}$ as follows:

$$\Delta \omega_{i,j}^{(t)} = \begin{cases} -\Delta_{i,j}^{(t)}, & \text{if } \frac{\partial E^{(t)}}{\partial \omega_{i,j}} > 0, \\ +\Delta_{i,j}^{(t)}, & \text{if } \frac{\partial E^{(t)}}{\partial \omega_{i,j}} < 0, \\ 0, & \text{else,} \end{cases} \tag{4}$$

where $E^{(t)}$ is the error function calculated for the entire training set. The update values $\Delta_{i,j}^{(t)}$ of each step were defined by a sign-dependent adaptation process as follows:

$$\Delta_{i,j}^{(t)} = \begin{cases} \eta^+ \Delta_{i,j}^{(t-1)}, & \text{if } \frac{\partial E^{(t-1)}}{\partial \omega_{i,j}} \frac{\partial E^{(t)}}{\partial \omega_{i,j}} > 0, \\ \eta^- \Delta_{i,j}^{(t-1)}, & \text{if } \frac{\partial E^{(t-1)}}{\partial \omega_{i,j}} \frac{\partial E^{(t)}}{\partial \omega_{i,j}} < 0, \\ \Delta_{i,j}^{(t-1)}, & \text{else,} \end{cases} \tag{5}$$

where $0 < \eta^- < 1 < \eta^+$, and η^- and η^+ are the decrease and increase factors, respectively. When the partial derivative changes sign, which means that we have jumped over the optimum point,

the algorithm reduces the update value $\Delta_{i,j}^{(t)}$ by the decrease factor η^-. When the sign of the partial derivative does not change, the algorithm increases the weight parameter $\Delta_{i,j}^{(t)}$ by η^+.

For the back-propagation step we calculated the partial derivatives of the error function $\frac{\partial E}{\partial \omega_{i,j}}$ for all the weights $\omega_{i,j}$ of the neural network connections. For the output layer weights (i.e. $\omega_{i,j}$ with $i \in L_2$ and $j \in L_o$) this calculation was given by :

$$\frac{\partial E}{\partial \omega_{i,j}} = x_i \delta_j = x_i(x_j - t_j), \tag{6}$$

where x_i and x_j are the values of ith and j^{th} neurons, respectively, and t_j is the target value of the training set at the output of the jth neuron. For a single weight each calculation took 1 real multiplication and since each output neuron had n_2 connections, the entire output layer required $n_2 n_o$ real multiplications.

Subsequently, we moved backwards and calculated the partial derivative of the error $\frac{\partial E}{\partial \omega_{i,j}}$ for the weights of the second hidden layer (i.e. $\omega_{i,j}$ with $i \in L_1$ and $j \in L_2$) :

$$\frac{\partial E}{\partial \omega_{i,j}} = x_i \delta_j = x_i \left(\sum_{l \in L_o} \delta_l \omega_{j,l} \right)(1 + x_j)(1 - x_j). \tag{7}$$

Here, the calculation of a single weight took $n_o + 3$ multiplications. Thus, the total number of real multiplications that corresponded to the second hidden layer was $(n_o + 3) n_1 n_2$. Repeating a similar procedure for first hidden layer we calculated $(n_2 + 3) n_i n_1$ real multiplications.

Overall, the calculation of the partial derivatives $\frac{\partial E}{\partial \omega_{i,j}}$ across all neural network weights required $n_o n_2 + n_o n_2 n_1 + 3 n_2 n_1 + n_2 n_1 n_i + 3 n_1 n_i$ real multiplications. To this, we added two extra multiplications for the update process of each weight value, resulting in $2(n_i n_1 + n_1 n_2 + n_2 n_o)$ multiplications for the whole neural network. Therefore, the total number of real multiplications required in the back-propagation step became equal to $3 n_o n_2 + n_o n_2 n_1 + 5 n_2 n_1 + n_2 n_1 n_i + 5 n_1 n_i$.
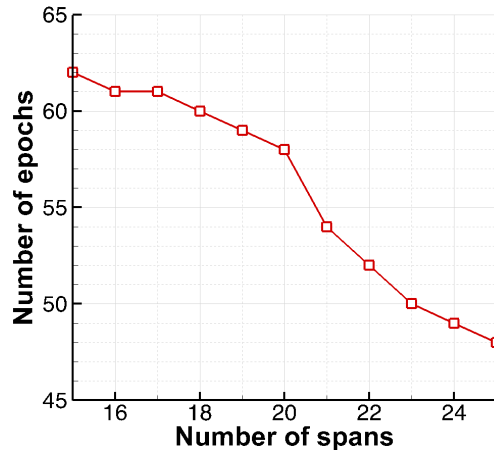

Fig. 6. The dependency of the number of performed epochs on number of spans.

The aforementioned calculations corresponded to a single epoch. If the training process takes N_{ep} epochs to be completed, the corresponding complexity can be written as:

$$C_{dNN}^{train} = \frac{N_{ep} \left[(N_{ts} + N_{vs}) \left(n_i n_1 + n_1 n_2 + n_2 n_o \right) + 3 n_o n_2 + n_o n_2 n_1 + 5 n_2 n_1 + n_2 n_1 n_i + 5 n_1 n_i \right]}{\left(N_{ps} + N_{ts} + N_{vs} \right) \log_2 M}, \tag{8}$$

where N_{ps} is the number of transmitted symbols that are processed during the prediction phase. Figure 6 shows the number of performed epochs for different propagation distances. We can see that as we increase in the number of spans, the number of required epochs decreases since performance convergence occurs at higher BER levels.

After the training, the deep neural network was used to process signals and to make the symbol prediction. The number of multiplications required by the this stage coincided with the forward propagation step in the training process. Therefore, the corresponding complexity of the prediction phase was given by:

$$C_{dNN}^{predict} = \frac{N_{ps}\,(n_i n_1 + n_1 n_2 + n_2 n_o)}{(N_{ps} + N_{ts} + N_{vs})\log_2 M}. \tag{9}$$

Finally, the complexity of the overall equalization scheme (FDE+dNN) was calculated according to :

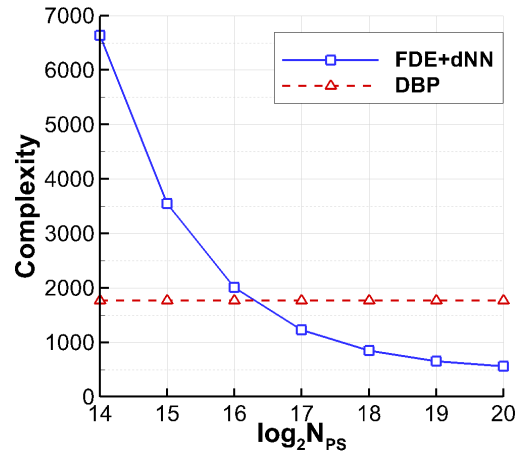$$C_{FDE+dNN} = C_{dNN}^{train} + C_{dNN}^{predict} + C_{FDE}. \tag{10}$$

Fig. 7. Dependency of complexity for DBP and (FDE+dNN) based nonlinear equalization schemes as a function of the number of transmitted symbols N_{ps}.

We should note that the training phase of the neural network is computationally more intensive than the prediction phase. However, its impact on the overall complexity depends on how frequent is the repetition of this process during the neural network operation, which is also reflected in the number N_{ps} of processed transmitted symbols. Performing frequent re-training of the neural network, to address highly dynamic channel conditions, allows a limited number of transmitted symbols in each operation cycle and may lead to a computationally inefficient performance. On the other hand, in semi-static channels the training may be repeated at much longer time periods, allowing for a higher number of transmitted symbols N_{ps}, and a significant reduction of the training impact on the overall computational complexity. This is also shown in Fig. 7 which presents the complexity calculations for the two equalization scenarios, as a function of the number of transmitted symbols N_{ps}, for a 20 span transmission system and with the following neural network parameters: $n_i = 2\,(N_{del} + 1)$, $n_1 = n_2 = 16$, $n_o = 2$. The complexity performance of the DBP-based receiver is straight line parallel to the horizontal axis because there is no dependence on the number of transmitted symbols. On the other hand, the complexity of the dynamic deep neural network based compensation scheme (FDE+dNN) decreases with
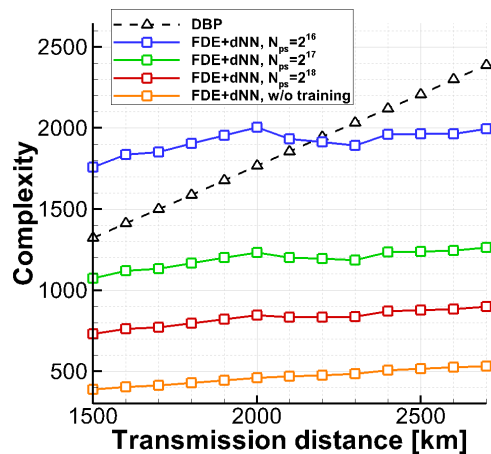
Fig. 8. Complexity of DBP and (FDE+dNN) based nonlinear equalizers as a function of transmission length

increasing N_{ps} and eventually drops much below the complexity level of the 2 steps-per-span DBP method (2 samples per symbol).

Finally, we compared the computational complexity of the two nonlinear equalization schemes as a function of system's transmission length and for different number of transmitted symbols N_{ps}, see Fig. 8. As expected, the computational complexity of the DBP based receiver increases linearly with the transmission distance, whereas there is minor impact on the dNN based receiver. As already mentioned the complexity of the latter scheme is mostly defined by the number of transmitted symbols. When N_{ps} equals 2^{16}, the complexity of both methods turns out to be comparable. However, already at $N_{ps} = 2^{17}$ the complexity of the dNN based receiver drops below the level of the DBP based scheme. When transmitting even higher number of symbols, to the extend where the complexity of the training process can be neglected, the deep neural network based compensation scheme shows significant superiority over the 2 steps-per-span DBP method.

5. Conclusion

We investigated the equalization performance of dynamic deep neural networks for long haul transmission. Our results showed that the use of dynamic neural networks along a 1000 km fibre link allows to improve the Q^2-factor by 1.5 dB in single channel transmission and by 1.4 dB in multi channel transmission, in comparison to linear equalization. Extensive analysis of the computational complexity has been also performed showing a reduction in the number of required real multiplication by transmitted bit by more than three times when compared to the use of traditional digital back propagation of 2 steps per span and 2 samples per symbol.

Funding

References

1. R. Tkach, "Scaling optical communications for next decade and beyond," Bell Labs Tech. J. **14**(4), 3–9 (2010).
2. C. E. Shannon, "A mathematical theory of communication," Bell Labs Tech. J. **27**(3), 379–423 (1948).
3. E. Temprana, E. Myslivets, B. P.-P. Kuo, V. Ataie, N. Alic, and S. Radic, "Overcoming Kerr-induced capacity limit in optical fiber transmission," Science **348**(6242), 1445–1448 (2015).
4. D. Rafique, "Fiber Nonlinearity Compensation: Commercial Applications and Complexity Analysis," J. Lightw. Technol. **34**(2), 544–553 (2016).

5. L. B. Du, D. Rafique, A. Napoli, B. Spinnler, A. D. Ellis, M. Kuschnerov, and A. J. Lowery, "Digital Fiber Nonlinearity Compensation: Toward 1-Tb/s transport," IEEE Commun. Mag. **31**(2), 46–56 (2014).
6. E. Ip, "Nonlinear Compensation Using Backpropagation for Polarization-Multiplexed Transmission," J. Lightw. Technol **28**(6), 939–951 (2010).
7. L. Liu, L. Li, Y. Huang, K. Cui, Q. Xiong, F. N. Hauske, C. Xie, and Y. Cai, "Intrachannel Nonlinearity Compensation by Inverse Volterra Series Transfer Function," J. Lightw. Technol. **30**(3), 310–316 (2012).
8. M. Gagni, F. Guiomar, S. Wabnitz, and A. Pinto, "Simplified high-order Volterra series transfer function for optical transmission links," Opt. Express **25**(3), 2446–2459 (2017).
9. E. Yamazaki, A. Sano, T. Kobayashi, E. Yoshidaa, and Y. Miyamoto, "Mitigation of Nonlinearities in Optical Transmission Systems," in *Optical Fiber Communication Conference/National Fiber Optic Engineers Conference 2011*, OSA Technical Digest (CD) (Optical Society of America, 2011), paper OThF1.
10. A. Napoli, Z. Maalej, V. Sleiffer, M. Kuschnerov, D. Rafique, E. Timmers, B. Spinnler, T. Rahman, L. Coelho, and N. Hanik, "Reduced Complexity Digital Back-Propagation Methods for Optical Communication Systems," J. Lightw. Technol. **32**(7), 1351–1362 (2014).
11. E. Ip and J. M. Kahn, "Compensation of Dispersion and Nonlinear Impairments Using Digital Backpropagation," J. Lightw. Technol. **26**(20), 3416–3425 (2008).
12. D. S. Millar, S. Makovejs, C. Behrens, S. Hellerbrand, R. I. Killey, P. Payvel, and S. J. Savory, "Mitigation of Fiber Nonlinearity Using a Digital Coherent Receiver," IEEE J. Sel. Top. Quantum Electron. **16**(5), 1217–1226 (2010).
13. D. Rafique, M. Mussolin, M. Forzati, J. Martensson, M. N. Chugtai, and A. D. Ellis, "Compensation of intra-channel nonlinear fibre impairments using simplified digital back-propagation algorithm," Opt. Express **19**(10), 9453–9460 (2011).
14. S. Li, B. Liu, B. Chen, and Y. Lou, "Neural network based mobile phone localization using Bluetooth connectivity," Neural Comput. Appl. **23**(3-4), 667–675 (2013).
15. S. Rajbhandari, Z. Ghassemlooy, and M. Angelova, "Effective Denoising and Adaptive Equalization of Indoor Optical Wireless Channel With Artificial Light Using the Discrete Wavelet Transform and Artificial Neural Network," J. Lightw. Technol. **27**(20), 4493–4500 (2009).
16. D. Wang, M. Zhang, M. Fu, Z. Cai, Z. Li, H. Han, Y. Cui, and B. Luo, "Nonlinearity Mitigation Using a Machine Learning Detector Based on k-Nearest Neighbors," IEEE Photon. Technol. Lett. **28**(19), 2102–2105 (2016).
17. E. Giacoumidis, I. Aldaya, J. L. Wei, C. Sanchez, H. Mrabet, and L. P. Barry, "Affinity propagation clustering for blind nonlinearity compensation in coherent optical OFDM," in *Conference on Lasers and Electro-Optics*, OSA Technical Digest (online) (Optical Society of America, 2018), paper STh1C.5.
18. T. Koike-Akino, C. Duan, K. Parsons, K. Kojima, T. Yoshida, T. Sugihara, and T. Mizuochi, "High-order statistical equalizer for nonlinearity compensation in dispersion-managed coherent optical communications," Opt. Express **20**(14), 15769–15780 (2012).
19. D. Zibar, O. Winther, N. Franceschi, R. Borkowski, A. Caballero, V. Arlunno, M. Schmidt, N. Gonzales, B. Mao, Y. Ye, K. Larsen, and I. Monroy, "Nonlinear impairment compensation using expectation maximization for dispersion managed and unmanaged PDM 16-QAM transmission," Opt. Express **20**(26), B181-B196 (2012).
20. M. Li, S. Yu, J. Yang, Z. Chen, Y. Han, and W. Gu, "Nonparameter Nonlinear Phase Noise Mitigation by Using M-ary Support Vector Machine for Coherent Optical Systems," IEEE Photon. J. **5**(6), 7800312 (2013).
21. D. Wang, M. Zhang, Z. Cai, Y. Cui, Z. Li, H. Han, M. Fu, and B. Luo, "Combatting nonlinear phase noise in coherent optical systems with an optimized decision processor based on machine learning," Opt. Commun. **369**, 199–208 (2016).
22. T. Nguyen, S. Mhatli, E. Giacoumidis, L.V. Compernolle, M. Wuilpart, and P. Megret, "Fiber Nonlinearity Equalizer Based on Support Vector Classification for Coherent Optical OFDM," IEEE Photon. J. **8**(2), 7802009 (2016).
23. M. A. Jarajreh, E. Giacoumidis, I. Aldaya, S. T. Thai, A. Tsokanos, Z. Ghassenmlooy, and N. J. Doran, "Artificial Neural Network Nonlinear Equalizer for Coherent Optical OFDM," IEEE Photon. Technol. Lett. **27**(4), 387–390 (2015).
24. E. Giacoumidis, S. Le, M. Ghanbarisabagh, M. McCarthy, I. Aldaya, S. Mhatli, M. Jarajreh, P. Haigh, N. Doran, A. Ellis, and B. Eggleton, "Fiber nonlinearity-induced penalty reduction in CO-OFDM by ANN-based nonlinear equalization," Opt. Lett. **40**(21), 5113–5116 (2015).
25. S. T. Ahmad and K. P. Kumar, "Radial Basis Function Neural Network Nonlinear Equalizer for 16-QAM Coherent Optical OFDM," IEEE Photon. Technol. Lett. **28**(22), 2507–2510 (2016).
26. D. Wang, M. Zhang, Z. Li, C. Song, M. Fu, J. Li, and X. Chen, "System impairment compensation in coherent optical communications by using a bio-inspired detector based on artificial neural network and genetic algorithm," Opt. Commun. **399**, 1–12 (2017).
27. J. Estaran, R. Rios-Muller, M. A. Mestre, F. Jorge, H. Mardoyan, A. Konczykowska, J.-Y. Dupuy, and S. Bigo, "Artificial Neural Networks for Linear and Non-Linear Impairment Mitigation in High-Baudrate IM/DD Systems," in *Proceedings of European Conference on Optical Communication (ECOC) (2016)*, paper M.2.B.2.
28. M. Riedmiller and H. Braun, "A direct adaptive method for faster backpropagation learning: the RPROP algorithm," in *Proceedings of the International Conference on Neural Networks* (IEEE, 1993), pp. 586–591.
29. R. A. Shafik, M. S. Rhahman, and R. Islam, "On the Extended Relationships Among EVM, BER and SNR as Performance Metrics," in *International Conference on Electrical and Computer Engineering* (ICECE, 2006), pp. 408–411.

30. X. Li, X. Chen, G. Goldfarb, E. Mateo, I. Kim, F. Yaman, and G. Li, "Electronic post-compensation of WDM transmission impairments using coherent detection and digital signal processing," Opt. Express **16**(2), 881–888 (2008).
31. B. Spinnler, "Equalizer Design and Complexity for Digital Coherent Receivers," IEEE J. Sel. Top. Quantum Electron. **16**(5), 1180–1192 (2010).
32. L. Prechelt, "Early Stopping – But When?" in *Neural Networks: Tricks of the Trade. Lecture Notes in Computer Science,* vol 7700, G. Montavon, G. B. Orr, and K. R. Muller, eds. (Springer, 2012).
33. D. Nguyen and B. Widrow, "Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights," in *Proceedings of the International Joint Conference on Neural Networks*, (IEEE, 1990), vol. 3, pp. 21–26.