A Real-Time OSNR Penalty Estimator Engine in the Presence of Cascaded WSS Filters

Jason Hoff, Tianliang Zhang, Andrea Fumagalli
The University of Texas at Dallas, Richardson, TX, USA
[Jason.Hoff, tianliang.zhang, andreaf]@utdallas.edu

Mustafa AL-QADI, Rongqing Hui
The University of Kansas, Lawrence, KS, USA
[mustafa.alqadi, hui]@ittc.ku.edu

GLOBECOM 2021 - 2021 IEEE Global Communications Conference | 978-1-7281-8104-2/21/$31.00 ©2021 IEEE | DOI: 10.1109/GLOBECOM46510.2021.9685715

***Abstract*—Virtual Optical Networks (VONs) offer scientists the opportunity to test and validate Software Defined Networking (SDN) controllers without requiring access to expensive optical equipment. A key component in VON is the real-time optical layer emulation (OLE) engine that must estimate the OSNR at the virtual receiver of every virtual lightpath that is established in the VON.**

This study investigates two algorithms for training a two-hidden layer Neural Network (NN) whose aim is to estimate the OSNR penalties caused by a series of virtual Wavelength Selective Switches (WSS) that are traversed by a virtual lightpath before reaching its virtual receiver. The NN solution yields good accuracy and only requires a few microseconds of computation time — a two key requirements for achieving a meaningful.

***Index Terms*—virtual optical network, WSS filter penalty, neural network, elastic optical networks, optical layer emulation**

I. Introduction

Transport networks combine multiple technologies spanning from the optical to IP layer. Experimental testing of these multi-layer solutions require significant investments, which become prohibitively high when testing with large size networks. Virtualization of switches and routers has been one of the most successful tools to enable scientists to experimentally test with layer 2 and layer 3 protocols, along with their Software Defined Networking (SDN) controllers [1]. Unfortunately, virtualization of optical networks is not a widely available tool at this time.

Optical networks have been extensively studied using analytical models, discrete-event simulators, and experimental testbeds. Simulation platforms like NS-3, OPNET, OMNET++ are quite popular as they offer the flexibility to rapidly implement new algorithms, protocols, and architectures. Julius, an emulation environment for SDN-enabled, multi-layer, and flexible IP/Optical networks was recently announced [2]. This environment combines Mininet [1] — for the SDN network emulation — and LINC-OE [3] — for the simulation of optical elements. While it offers web interfaces to configure the optical link properties, Julius does not support Quality of Transmission (QoT) models.

To achieve a realistic virtual optical network (VON) the emulation environment must provide a real-time optical physical layer emulation (OLE) engine that accounts for the desired QoT models in real-time. The OLE engine must be able to estimate relevant QoT performance indicators like pre-Forward Error Correction (FEC) Bit Error Rate (BER) and Optical Signal-to-Noise Ratio (OSNR) of any virtual lightpath that is established in the VON. Accounting for the fact that hundreds or even thousands of virtual lightpaths can be established in the VON, the OLE engine must make use of efficient techniques to compute these QoT performance indicators for each virtual lightpath in real-time.

The study in this paper aims to investigate the use of two-hidden layer Neural Networks (NNs) to accurately compute the OSNR penalty induced by a cascade of Wavelength Selective Switch (WSS) devices at the virtual receiver of a virtual lightpath routed through said virtual WSS devices. The NN solution is expected to account for a number of input parameters, including the modulation scheme, baud-rate, WSS bandwidth, and number of cascaded WSS devices. It is worth noting that in current and past deployments of Dense Wavelength Division Multiplexing (DWDM) networks the WSS induced penalties are for the most part negligible when compared to other factors, such as fiber non-linearity and optical amplifier noise. However, with the introduction of flexible grid allocation in Elastic Optical Network (EON) the fiber spectrum is expected to become more densely populated, making WSS induced OSNR penalties more relevant. Hence, a meaningful OLE engine must also account for the WSS induced OSNR penalties. Other OSNR penalties can be accounted for either independently [4] or inter dependently [5].

More precisely, the study investigates the use of two NN training algorithms, i.e., Levenberg-Marquardt (LM) and Bayesian Regularization (BR) [6]. LM is generally the fastest among all backpropagation algorithms. BR is based on LM and typically requires more training time, but can result in good generalization for difficult, small, or noisy input datasets. The considered NN solution is shown to yield accurate computation of the OSNR penalty caused by a cascade of multiple WSS filters, achieving less than 1 dB mean squared error for OSNR penalties that are most relevant to the OLE engine and while accounting for a wide range of virtual lightpath establishment scenarios. By making use of two hidden layers and 128+128 neurons the considered NN solution requires limited computation capabilities, thus making it possible for the OLE engine to estimate the OSNR penalties of 100,000 virtual lightpaths in less than one second.

This work was supported in part by NSF Grants No. ACI-1541461, CNS-1531039, CNS-1956137, and CNS-1956357.

II. Modeling a Cascade of WSS Devices

In transparent optical networks a lightpath occupying a fraction of the fiber optics spectrum (wavelength) is routed from the source to the destination node through multiple intermediate nodes. Lightpath routing and termination at these network nodes are achieved through Reconfigurable Optical Add/Drop Multiplexer (ROADM). ROADMs make use of Wavelength Selective Switch (WSS) devices, which are configured to selectively allow fractions of the optical spectrum to propagate through the device. WSS also enables advanced functionalities in the ROADM referred to as colorless, directionless, and contentionless (CDC) [7].

Ideal WSS devices with flat top and steep edges of frequency response are hard to built. Realistic WSS transfer functions have non-flat top and limited roll-off rates at the edges, which together result in QoT deterioration every time the signal goes through one WSS device. This WSS filter penalty is cumulative, in that every WSS device traversed by the lightpath further affects the integrity of the transmitted signal. This problem is exacerbated in flexible grid deployments whereby channel spectral spacing between lightpaths and their respective guard bands are both reduced to accommodate more transmission capacity in the same optical fiber cable.

The frequency response of a WSS device is typically modelled by (1) (2) [8], where BW_{OTF} is the roll-off factor and B is the bandwidth of the aperture of the WSS (usually set to match channel spacing). This model shows good agreement with WSS device experimental measurements.

$$S(f) = \frac{1}{2}\sigma\sqrt{2\pi}\left[erf\left(\frac{\frac{B}{2}-f}{\sqrt{2}\sigma}\right) - erf\left(\frac{-\frac{B}{2}-f}{\sqrt{2}\sigma}\right)\right] \quad (1)$$

$$\sigma = \frac{BW_{OTF}}{2\sqrt{2ln2}} \quad (2)$$

Cascading multiple WSS devices has the net effect of multiplying their bandpass filter frequency responses together. Fig. 1 shows the equivalent 6 dB bandwidth after a number of cascaded WSS modules, assuming that the WSS modules have identical frequency responses. The resulting effect of

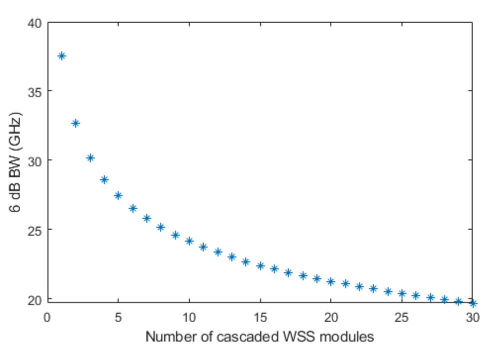


Fig. 1: Equivalent 6 dB bandwidth of cascaded WSS modules, where $BW_{OTF} = 10.5$ GHz and $B = 37.5$ GHz.

the applied 6 db bandwidth on the signal OSNR degradation (penalty) can only be investigated experimentally or through simulation, the latter being more economical.

Signal penalties induced by WSS filtering are typically negligible in conventional 50-GHz grid DWDM networks when using standard symbol rates such as 28 Gbd and 32 Gbd. However, as already mentioned, these penalties become significant in EON as a result of their reduced channel spacing [9]. Fig. 2 shows the WSS filtering OSNR penalty in a 32 Gbd 256 Gbps optical signal modulated as PM-16QAM with 37.5 GHz channel spacing as it gets routed through WSS devices. Considering that a minimum of 4 cascaded WSS devices are needed to provision an optical circuits between two ROADMs (a single hop path), the resulting OSNR penalty (14.4 dB) quickly becomes prohibitively high. This example illustrates how the WSS filtering effect can induce significant transmission degradation in EON with narrow channel spacing. It must also be noted that besides the frequency response shape and number of traversed WSS devices, the resulting OSNR penalty depends on the relative position of the channel inside the WSS passband and the signal modulation format. Considering that numerical simulation techniques to estimate the OSNR penalty for each of the possible network configurations are time consuming, a faster and comprehensive alternative solution is highly desirable when designing a OLE engine.

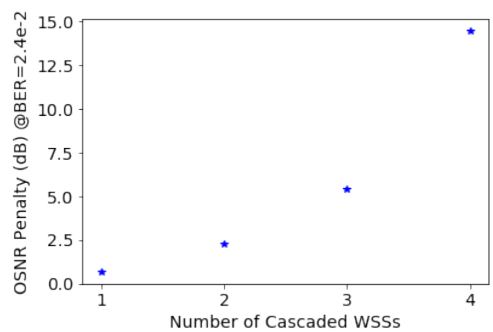


Fig. 2: OSNR penalty for 32 Gbd 256 Gbps optical signal modulated as PM-16QAM with 37.5 GHz channel spacing due to cascaded WSS filtering.

III. Neural Network Architecture and Training Algorithms

Neural network (NN) [10] is a modeling technique with varying structures that can be applied to solve most complex numerical problems through an effective input-output variable mapping. Structures may vary based on the specific problems under consideration and include artificial neural networks, convolutional neural networks, and recurrent neural networks. Classification and regression are two typical tasks neural networks are designed for. In this paper, artificial neural network is chosen to model the WSS filtering characteristics and estimate the resulting OSNR penalty through non-linear regression.

A generalized artificial neural network structure is shown in Fig. 3. The input layer contains a number of neurons matching the number of input variables (features). Between the input and output layer, one or more hidden layers are deployed, each hosting a number of neurons. Each neuron is fully connected with the neurons in the previous and successive layers and applies a non-linear activation function to the weighted sum of the incoming values to produce its outgoing value. This NN type is also known as feedforward neural network for supervised learning. In this study, the ReLU [11] activation function is used.

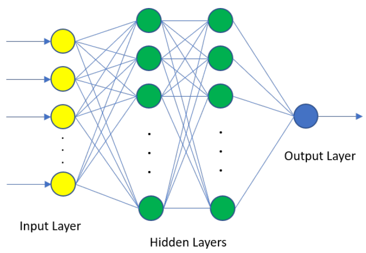


Fig. 3: A generalized artificial neural network structure.

The NN training process requires a backpropagation algorithm that calculates the gradients and Jacobians iterating backward from the last layer to the first layer [12]. Any network that has differentiable transfer functions, weight functions, and net input functions can be trained by a backpropagation algorithm. Variable optimization algorithms are used within the backpropagation procedure. In this paper, we mainly focus on Levenberg-Marquardt (LM) and Bayesian Regularization (BR) algorithms [6]. LM is generally the fastest among all backpropagation algorithms. BR is based on LM and typically requires more time to complete the NN training. However, it provides good generalization for difficult, small or noisy datasets. Batch training is used as it is significantly faster and produces smaller errors compared to incremental training. Mean squared error (MSE) — defined in (3) — is used as key indicator for the trained NN performance, where t_i is the target output and p_i is the predicted output.

$$MSE = \frac{1}{N}\sum_{i}^{N}(e_i)^2 = \frac{1}{N}\sum_{i}^{N}(t_i - p_i)^2 \tag{3}$$

IV. Results

A. Dataset Generation

Due to insufficient experimental data on WSS filtering penalty currently available to the authors, the training dataset is generated with the help of a simulator. Features (illustrated in Fig. 4) and restrictions are described next.

- R_s: Symbol rate, is a continuous value from 2 Gbd to 42 Gbd. The 2 Gbd lower limit is set to account for Subcarrier Multiplexing (SCM) channels which may use only part of B. The 42 Gbd upper limit is set to account for most signal widths that would fit in the 50 GHz upper limit value for B.
- B: WSS filter bandwidth (channel spacing), is a continuous value from $\max(6.25\text{ GHz}, Rs)$ to 50 GHz. The minimum nominal central frequency granularity defined by the ITU-T is 6.25 GHz [13]. The upper limit is 50 GHz as it is the grid of traditional fixed grid DWDM network. Other values such as 67.5 GHz, 100 GHz, etc. are also acceptable.
- Δf: Signal center frequency relative to the center of the optical filter, is a continuous value from $(-B+R_s)/2$ to $(B-R_s)/2$ that accounts for all possible relative positions of a SCM channel.
- σ: Signal root-raised-cosine roll-off factor, is a continuous value from 0.01 to 1.
- M: Modulation format, is one of the following eight categories: PM-BPSK, PM-QPSK, PM-8QAM, PM-16QAM, PM-32QAM, PM-64QAM, PM-128QAM, and PM-256QAM.
- n: Number of cascaded WSS devices, is a discrete value from 1 to 20.
- $OSNR_{WSS}$: WSS filtering OSNR penalty, is the output value the neural network needs to make regression on.

By selecting these key features that may affect the WSS filtering penalty, the trained neural network is expected to be a valid estimation tool for a broad range of scenarios beyond current commercial configurations. Feature values are randomly selected within their respective ranges when generating the training dataset. The complete dataset contains 48,258 samples and is divided to form three disjoint sets: training (70%), validation (15%), and testing (15%).

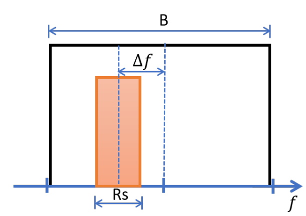


Fig. 4: Parameters used to generate the training dataset.

Pre-processing of the input features is necessary to increase the NN accuracy. For continuous-valued features, i.e., R_s, B, Δf, and σ, each feature is scaled by its maximum absolute value. For discrete-valued features, values are encoded using binary digits. Three digits are needed for feature M and five digits are needed for feature n. A total of 12 features are used.

B. Neural Network Training and Evaluation

The NN training is performed using a computer with the following specifications:

- CPU : Intel Core i9-9900k @ 4.7GHz.
- RAM : G.Skill Trident Z Neo F4-3600C18Q-128GTZN DDR4-3600MHz CL18-22-22-42 1.35V 128GB (4x32GB).
- Environment: MATLAB R2021b.

We use four groups of data to train the NN, labelled as “Whole Data,” “[0,5],” “[0,10],” and “[0,15],” respectively.

Whole Data includes the complete dataset of 48,258 samples. The highest value of OSNR penalty in the Whole Data group is 36.891 dB, while the lowest value of OSNR penalty is 0 dB. The distribution of OSNR penalty in the dataset is shown in Fig. 5. The other three groups are obtained by removing the

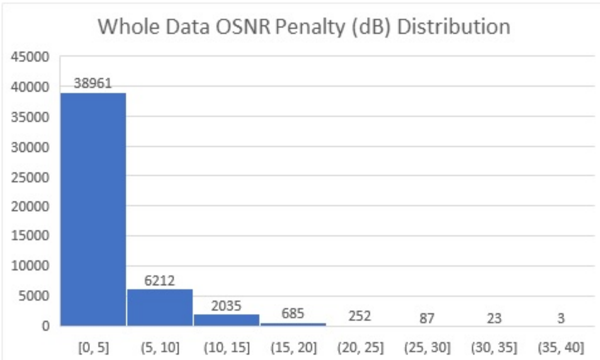


Fig. 5: Whole Data OSNR penalty (dB) distribution.

samples in the original dataset whose OSNR penalty does not fall in the [0,x] range (in dB). Table I shows the total number of samples and average OSNR penalty for each training group.

TABLE I: Data Used in Neural Network Training.

Training Group	Total Number of Samples	Average OSNR Penalty (dB)
Whole Data	48258	2.899
[0,15]	47208	2.531
[0,10]	45173	2.101
[0,5]	38961	1.316

The training results use the following naming convention, with NN representing either LM or BR training algorithm:

- 256x1NN: 1 hidden layer with 256 neurons, NN trained with Whole Data.
- 128x2NN: 2 hidden layers with 128 neurons each, NN trained with Whole Data.
- 128x2NN5: 2 hidden layers with 128 neurons each, NN trained with OSNR penalties in [0,5] dB.
- 128x2NN10: 2 hidden layers with 128 neurons each, NN trained with OSNR penalties in [0,10] dB.
- 128x2NN15: 2 hidden layers with 128 neurons each, NN trained with OSNR penalties in [0,15] dB.
- 64x4NN: 4 hidden layers with 64 neurons each, NN trained with Whole Data.

Fig. 6 shows MSE (y-axis) vs. number of epoches (x-axis) for the two training algorithms (LM and BR) and three NN structures listed in Tables II and III. MSE values are computed using three datasets:

- Training Performance. This performance is based on the samples of data presented to the NN during training. The NN is adjusted based on this MSE.
- Validation Performance. This is the group of samples used to measure the NN's ability to generalize its estimates to samples that are not in the training dataset, and to stop training when its estimates stop improving.
- Testing Performance. This is the group of samples that is not used by the training procedure and provides an independent measure of the NN generalized performance.
- Overall Performance: The trained NN is used to predict all data samples to show the overall performance, indicating with MSE.

It is worth noting that:

- The graph of BR shows Training Performance in the titles, while the Test Performance is manually reported in the y-axis value of the graph at x = 100. For LM, the titles of the graph report the Validation Performance, while the Training Performance is reported in the tables.
- The results are rounded up to 3 decimals.
- The training stopping point for LM is as defined in MATLAB R2021b. For BR, the training stops once the pre-determined maximum number of epochs (100, in our case) is reached.

We first investigate the effect of the numbers of the NN hidden layers on the training performance and mean squared error of the trained NN. We used 1, 2, and 4 hidden layers, while keeping the total number of neurons at 256. Table II and Table III contain the training results for BR and LM, respectively. By increasing the number of NN hidden layers most

TABLE II: Training Results for BR Neural Network.

Training Mode	Training Performance	Test Performance	Overall Performance
256x1BR	2.614	3.564	2.757
128x2BR	0.370	1.781	0.582
64x4BR	0.357	1.045	0.461

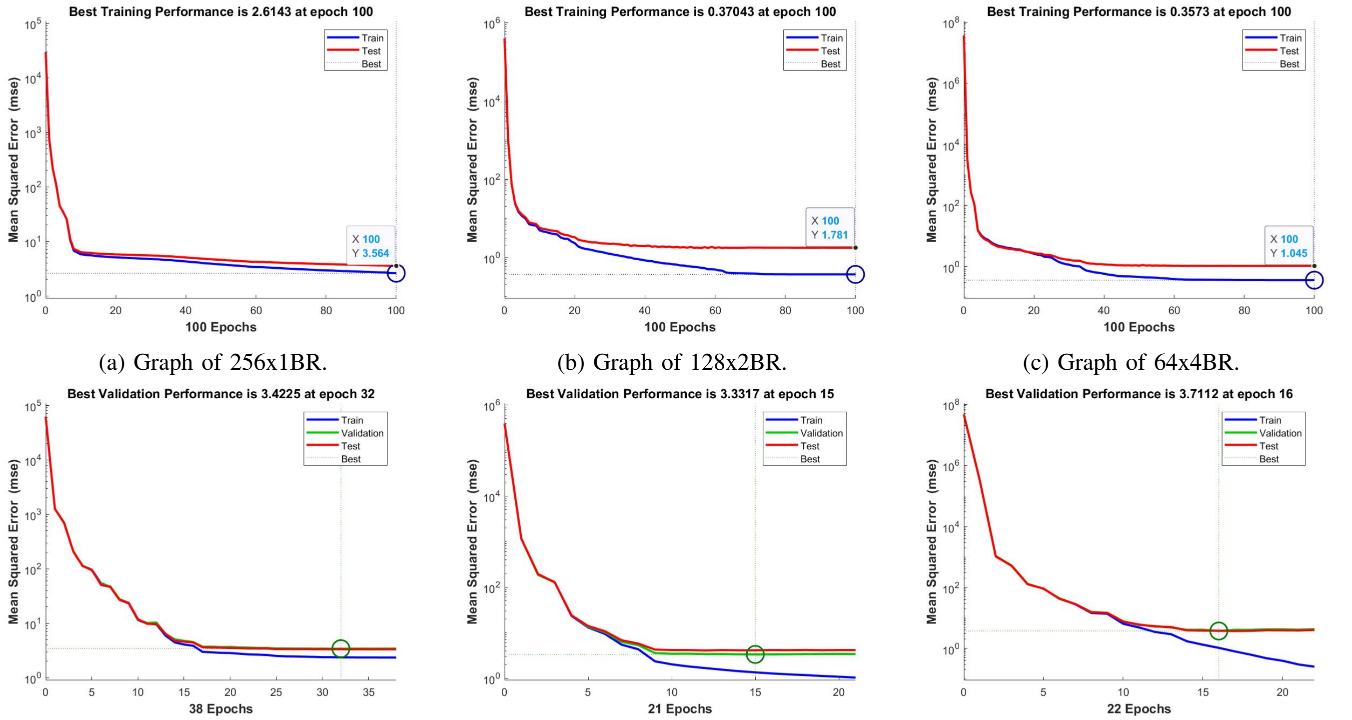
TABLE III: Training Results for LM Neural Network.

Training Mode	Training Performance	Validation Performance	Overall Performance
256x1LM	2.350	3.423	2.677
128x2LM	1.030	3.332	2.057
64x4LM	0.251	3.711	1.855

performance indicators improve. Most performance gains are already achieved by the two-hidden layer NN compared to the single hidden layer NN, with more modest performance gains being achievable by the four-hidden layer NN. In addition, while 64x4BR has the best training results, for LM, 128x2LM has the best Validation Performance of the three LM NNs. For these reasons the remainder of the paper focuses on the NN solutions that are based on 2 hidden layers, i.e., 128x2BR and 128x2LM.

To see the effect of dataset used on each training algorithm, the 4 different sets of samples in Table I are applied. Fig. 7 shows the training procedures for different NN setups with three training sets ([0,5], [0,10], [0,15]) and two algorithms (LM, BR) listed in Table IV and V.

With Fig. 5 in mind, the training results show us that the training performances of each algorithm depends on the dataset used to train the NN. It appears that training NN using data with higher OSNR penalties tends to yield higher mean squared error on Training Performance, Validation or



(a) Graph of 256x1BR. (b) Graph of 128x2BR. (c) Graph of 64x4BR.

(d) Graph of 256x1LM. (e) Graph of 128x2LM. (f) Graph of 64x4LM.

Fig. 6: Training evolution for different NN structures and algorithms.

TABLE IV: Four types of BR NN training mode.

Training Mode	Training Performance	Test Performance	Overall Performance
128x2BR	0.370	1.781	0.582
128x2BR15	0.065	0.291	0.099
128x2BR10	0.024	0.112	0.037
128x2BR5	0.005	0.023	0.008

TABLE V: Four types of LM NN training mode.

Training Mode	Training Performance	Validation Performance	Overall Performance
128x2LM	1.030	3.332	2.057
128x2LM15	0.141	1.013	0.475
128x2LM10	0.048	0.393	0.166
128x2LM5	0.019	0.135	0.064

Test Performance, and Overall Performance. Two main factors contribute to this outcome: 1) at higher OSNR values the absolute error is larger compared to lower OSNR values and 2) reaching a well trained NN is simpler over a narrowly defined range of OSNR values. Further investigation of this trend is discussed next.

Table VI reports the NN training results when specific samples in the dataset are used during training, i.e., samples in the [0,5], [5,10], and [10,15] dB OSNR penalty range, respectively. These results confirm and quantify an intuitive conjecture according to which the NN trained using only samples in the [0,5] dB range [128x128BR5] is best suited to estimate OSNR penalties that fall in this range of values. However, they should not be used to estimate OSNR penalties that exceed 5 dB. Similarity, for scenarios in which the OSNR penalty is in the [5,10] dB range solution 128x2BR10 yields the best performance, and so on. The results in this table can be effectively leveraged to compute the most accurate OSNR penalty over a wide range of OSNR values, understanding that the NN structure that provides the most accurate value is determined by the estimated OSNR value itself.

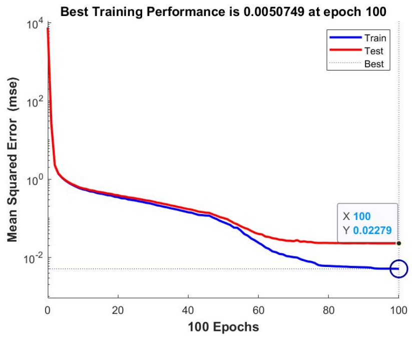
TABLE VI: Mean squared error breakdown for various OSNR penalty value (dB) ranges

Training Mode	[0,5] dB	[5,10] dB	[10,15] dB
128x2BR5	0.0078	-	-
128x2BR10	0.019	0.146	-
128x2BR15	0.036	0.208	0.966
128x2BR	0.141	1.010	2.903
128x2LM5	0.064	-	-
128x2LM10	0.117	0.471	-
128x2LM15	0.276	0.763	3.406
128x2LM	0.929	2.548	5.856

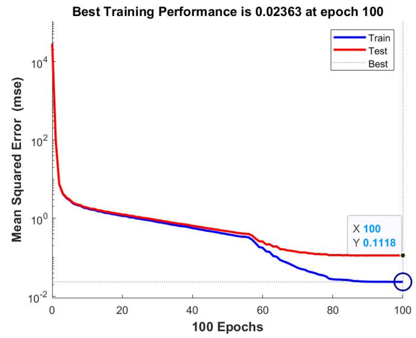
The typical NN execution time — defined as the time that is required by the NN to compute the OSNR penalty — is another important performance indicator in real-time applications. For completeness, the execution time is measured for every sample in the Whole Data set. The mean execution time using our computer is 3.380 μs for 128x2 structure.

V. Conclusion

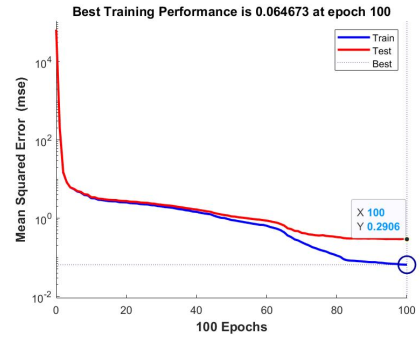
This study investigates the use of LM and BR trained NN for estimating the OSNR penalty induced by a cascade of



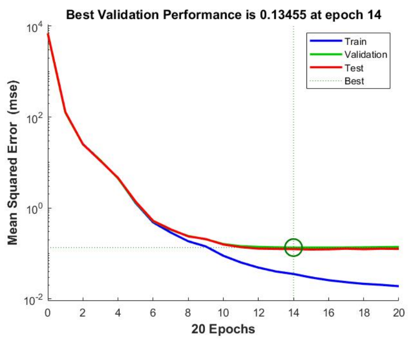
(a) Graph of 128x2BR5.



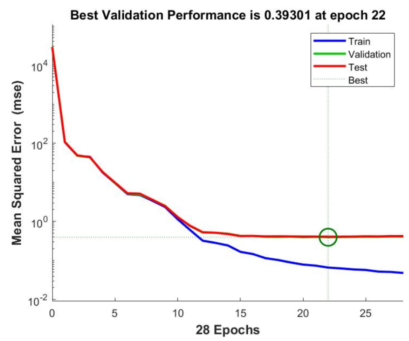
(b) Graph of 128x2BR10.



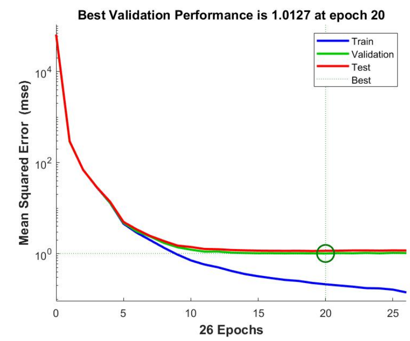
(c) Graph of 128x2BR15.



(d) Graph of 128x2LM5.



(e) Graph of 128x2LM10.



(f) Graph of 128x2LM15.

Fig. 7: Training evolution for different NN structures and algorithms validated against different subsets of whole dataset.

WSS devices that are used to route a lightpath to its intended receiver. The NN is trained using a dataset that is obtained from an optical transmission simulator. The considered NN solutions, and in particular 128x2BR15, may yield estimates with less than 1 dB of mean squared error for OSNR penalties up to 15 dB. The mean computation time required by the NN to produce one OSNR penalty estimate is less than 5 μs.

With its combined good accuracy and short computation time NN is a suitable solution for Optical Layer Emulation (OLE) engines, which must timely compute QoT performance indicators in Virtual Optical Network (VON) platforms. Large virtual optical networks with thousands of virtual lightpaths can then be emulated for testing the scalability of single- and multi-layer SDN controllers and orchestrators. These NN solutions may also find good applications in network planning tools and network controllers. Future work will investigate other promising solutions like the 64x4BR NN.

References

[1] Mininet. [Online]. Available: https://www.mininet.org

[2] A. Kretsis *et al.*, "An emulation environment for sdn enabled flexible ip/optical networks," in *Proc. 18th International Conference Transparent Optical Network*, 2016, pp. 1–4.

[3] Linc-oe. [Online]. Available: https://github.com/FlowForwarding/LINC-Switch/blob/master/docs/optical_extension.md

[4] T. Zhang, *et al.*, "Exploration of optical signal qot margin with intelligent wss filtering penalty estimator using neural network," in *2021 International Conference on Optical Network Design and Modeling (ONDM)*, 2021.

[5] A. Minakhmetov, *et al.*, "Accurate prediction via artificial neural network of osnr penalty induced by non-uniform wss filtering," in *2021 Optoelectronics and Communications Conference (OECC)*, 2021.

[6] M. T. Hagan, H. B. Demuth, and M. Beale, *Neural network design*. PWS Publishing Co., 1997.

[7] S. Perrin. (2015, Mar.) Next-generation roadm architectures & benefits. [Online]. Available: https://www.fujitsu.com/us/Images/Fujitsu-NG-ROADM.pdf

[8] C. Pulikkaseril *et al.*, "Spectral modeling of channel band shapes in wavelength selective switches," *Optics Express*, vol. 19, no. 9, pp. 8458–8470, Apr 2011.

[9] A. Morea *et al.*, "Impact of reducing channel spacing from 50ghz to 37.5ghz in fully transparent meshed networks," in *Proc. Optical Fiber Communications Conference and Exposition (OFC' 14)*, 2014, pp. 1–3.

[10] J. Mata *et al.*, "Artificial intelligence (ai) methods in optical networks: A comprehensive survey," *Optical Switching and Networking*, vol. 28, pp. 43 – 57, 2018.

[11] (2021) Relu. [Online]. Available: https://en.wikipedia.org/wiki/Rectifier_(neural_networks)

[12] H. Simon, *Neural networks and learning machines*. New York: Prentice Hall, 2009.

[13] (2012, Feb.) Spectral grids for wdm applications: Dwdm. [Online]. Available: https://www.itu.int