Performance Analysis of a Data-Driven Quality-of-Transmission Decision Approach on a Dynamic Multicast-Capable Metro Optical Network

T. Panayiotou, S. P. Chatzis, and G. Ellinas

***Abstract*—The performance of a data-driven quality-of-transmission (QoT) model is investigated on a dynamic metro optical network capable of supporting both unicast and multicast connections. The data-driven QoT technique analyzes data of previous connection requests and, through a training procedure that is performed on a neural network, returns a data-driven QoT model that near-accurately decides the QoT of the newly arriving requests. The advantages of the data-driven QoT approach over the existing *Q*-factor techniques are that it is self-adaptive, it is a function of data that are independent from the physical layer impairments (PLIs) eliminating the requirement of specific measurement equipment, and it does not assume the existence of a system with extensive processing and storage capabilities. Further, it is fast in processing new data and fast in finding a near-accurate QoT model provided that such a model exists. On the contrary, existing *Q*-factor models lack self-adaptiveness; they are a function of the PLIs, and their evaluation requires time-consuming simulations, lab experiments, specific measurement equipment, and considerable human effort. It is shown that the data-driven QoT model exhibits a high accuracy (close to 92%–95%) in determining, during the provisioning phase, whether a connection to be established has a sufficient (or insufficient) QoT, when compared with the QoT decisions performed by the *Q*-factor model. It is also shown that, when sufficient wavelength capacity is available in the network, the network performance is not significantly affected when the data-driven QoT model is used for the dynamic system instead of the *Q*-factor model, which is an indicator that the proposed approach can efficiently replace the existing *Q*-factor model.**

***Index Terms*—All optical networks; Multicast routing; Quality of transmission; Neural networks.**

I. Introduction

Recent advances in optical networks, which are expected to support traffic that will be heterogeneous in nature (i.e., capable of supporting unicast and multicast traffic), have made bandwidth-intensive point-to-multipoint applications widely popular. However, if such applications are offered in transparent optical networks, the physical layer impairments must be taken into consideration to ensure that the signal can be correctly detected at the receiver [1]. Several works exist in the literature that use different representations for modeling the most important physical layer effects that can accumulate in a transparent optical network, such as amplified spontaneous emission (ASE) noise, crosstalk, optical filter concatenation, and polarization mode dispersion (PMD), among others [1,2]. One approach for the modeling of the physical layer is based on the physical path *Q*-factor, which is subsequently used to calculate the bit error rate (BER) of the system, a parameter that is difficult to evaluate upfront. In these representations, worst-case budget values are usually included for accounting for the physical layer effects that are difficult to be accurately evaluated (e.g., crosstalk, PMD, polarization dependent gain/loss, etc.) and have been previously evaluated statistically [3]. The latter approach involves time-/frequency-domain Monte Carlo simulations on true measurements for each one of the PLIs that are present in a transparent optical network [3]. Although these representations are valuable at the engineering and performance evaluation stages of a network, upon network changes the aforementioned time-consuming procedure needs to be repeated for updating the *Q*-factor model.

In this work, a robust QoT decision approach is explored, which is based on a data-driven technique from the context of pattern recognition. In particular, a state-of-the-art feed-forward neural network is utilized for analyzing QoT data of previously established connections aiming at finding a QoT decision model of high accuracy. The approach inherits the advantages of neural networks; thus it is self-adaptive, fast in finding an accurate model (provided that such a model exists), fast in processing new data, and does not assume the existence of a system with extensive processing and storage capabilities [4]. Most importantly, the proposed approach is independent from the PLIs, eliminating the specific measurement equipment that *Q*-factor models require for evaluation/re-evaluation purposes.

A data-driven QoT approach for multicast-capable metro networks was first introduced in our previous work

Manuscript received July 25, 2016; revised November 9, 2016; accepted November 14, 2016; published December 22, 2016 (Doc. ID 272409).

T. Panayiotou (e-mail: panayiotou.tania@ucy.ac.cy) and G. Ellinas are with the Department of Electrical and Computer Engineering and the KIOS Research Center for Intelligent Systems and Networks, University of Cyprus, Cyprus.

S. P. Chatzis is with the Department of Electrical Engineering, Computer Engineering, and Informatics, Cyprus University of Technology, Cyprus.

https://doi.org/10.1364/JOCN.9.000098

presented in [5]. In [5], the connection features that effectively contribute to the data-driven QoT model were for the first time identified, and the problem was formulated according to the selected features and according to the specific feed-forward neural network (NN) chosen for training/validating the QoT model. Historical QoT data were generated from the *Q*-factor model described in detail in [2] assuming the multicast-capable architecture/engineering reported in [5] and references therein. The approach was validated for two metro area networks exhibiting a high accuracy for both networks. For data-generation purposes, a static system was assumed utilizing a single wavelength.

In this paper, we extend the previous work [5] by exploring a data-driven QoT approach on a dynamic impairment-aware unicast/multicast routing and wavelength assignment (IA-UMC-RWA) system in which multiple wavelengths are present, and the connections arrive and terminate in a dynamic (online) fashion (provisioned and terminated on the fly). Thus, the novelty of this work stems from the fact that the data-driven QoT problem is re-formulated by taking into account the existence of different wavelengths in the network and without *a priori* knowledge of all the connections in the network. Four cases are investigated. In each case, the network utilizes a different number of wavelengths, that is, 4, 8, 16, or 32 wavelengths for each link of the network. Upon the establishment of a connection, the wavelength assigned to that connection is reserved, and it is released when the connection terminates.

Historical data are generated from the dynamic IA-UMC-RWA algorithm and are used for training/validating a neural network [6], according to a known supervised learning algorithm [7]. The accuracy of the trained data-driven QoT model is evaluated by comparing the QoT decisions of the data-driven model to the QoT decisions of the *Q*-factor model, from which the historical data were extracted. Additionally, the network performance is evaluated when the data-driven QoT model is used in conjunction with the IA-UMC-RWA algorithm instead of the conventional *Q*-factor model. For every wavelength case examined, the data-driven QoT model exhibited a high accuracy (close to 92%–95%) in determining, during the provisioning phase, whether a connection to be established has a sufficient (or insufficient) QoT, when compared with the QoT decisions performed by the *Q*-factor model. Also, when sufficient wavelength capacity is available in the network, it was shown that the network performance was not significantly affected by the data-driven QoT model, an indication that the proposed approach can efficiently replace the existing *Q*-factor model.

Given the increased problem complexity, compared with the one investigated in [5], the data-driven QoT model is now trained on an NN utilizing a larger number of units. The dropout regularization technique [6] was used to allow for the network (with more parameters) to be trained effectively, without getting prone to overfitting. Dropout has been the first and the most popular regularization technique for NNs [6]. In essence, it consists of randomly dropping different units of the network on each iteration of the training algorithm. In doing so, only the parameters related to a subset of the network units are trained on each iteration. This ameliorates the associated network overfitting tendency, and it does so in a way that ensures that all network parameters are effectively trained. NNs with dropout are trained using stochastic optimization techniques, which allow for lower memory requirements. This includes algorithms such as Adam [7], Adagrad [8], and others. In this work, we opt for Adam, as it has been shown in [7] to converge faster than Adagrad in conjunction with the dropout regularization technique.

The rest of the paper is organized as follows: In Section II related work is discussed followed in Section III by the problem statement and the approach overview. In Section IV the data-driven QoT problem is formulated, and in Section V the dropout technique [6], used during training the data-driven QoT model, is described. Section VI describes the data-generation procedure for the dynamic network; Section VII discusses the experimental procedure used for validation purposes, and Section VIII describes some of the practical feasibility issues that arise in this work. Finally, Section IX offers concluding remarks.

II. Related Work

Related works exist in the literature that examine the inferential QoT framework by either focusing on designing a software defined network (SDN) platform capable of supporting data-driven QoT decisions or by proposing data-driven approaches for accurate QoT decisions [9–12]. Specifically, the authors of [9] proposed a transport SDN architecture and presented new data for devices, network elements, and SDN applications, in order to enable optical networks to support new services and virtualization with flexibility and scalability. Experimental results were shown, demonstrating optical network self-adaptation for sustaining QoT in the advent of optical impairments. In [9], only point-to-point connections were considered, and an optical signal-to-noise ratio (OSNR) monitoring scheme was utilized for keeping track of the physical impairments. However, no specific machine learning techniques were proposed.

Addressing the problem from a different point of view, the authors of [10] explored the benefits of utilizing data-driven models for QoT decisions. However, in [10], only point-to-point connections were assumed, and no specific machine learning approaches were proposed. This work was extended in [11], where a data-driven QoT estimator was proposed for classifying lightpaths into high- or low-quality categories in impairment-aware wavelength-routed optical networks. In particular, the technique presented was based on case-based reasoning (CBR), an artificial intelligence technique that solves new problems by exploiting previous experiences, which are stored in a knowledge database.

Finally, in [12] a Gaussian noise (GN) model was proposed that is able to estimate, quickly and accurately, the OSNR of the optical channels in uncompensated

coherent transmission systems. The GN model seems to be a promising candidate for being a useful tool for system and network analysis design and control, as it is more easily exploitable, both analytically and numerically, compared with other models that were previously published (i.e., time-consuming Monte Carlo simulations are not required). However, in that work, no specific network scenarios were addressed. In summary, for all aforementioned works, only point-to-point connections were considered. To the best of our knowledge, the data-driven QoT decisions problem addressing multicast connection was for the first time examined in our previous work, as presented in [5].

Specifically, in [5], a data-driven technique for analyzing quality-of-transmission (QoT) data of previous connection requests was proposed for accurately deciding the QoT of the newly arriving multicast requests in metro optical networks. In [5] the approach was examined on a static impairment-aware multicast routing system assuming the use of a single wavelength. In this paper, the work in [5] is significantly extended by examining the data-driven QoT decision problem on a dynamic multicast-capable network utilizing multiple wavelengths.

III. Problem Statement and Approach Overview

The existing QoT model for multicast connections [2,13] is merely a function of the physical layer impairments and is based on the Q-factor modeling approach presented in [3]. The Q-factor model was evaluated with time-consuming Monte Carlo simulations, which are based on stochastic numerical sampling from distributions, and each PLI (e.g., distortion-induced penalty due to filter concatenation, crosstalk, ASE noise, polarization mode dispersion, etc.) was evaluated by conducting experiments in the lab with the appropriate measurement equipment. A detailed analysis for the evaluation of each impairment and the procedure followed can be found in [3] and references therein. Although the existing analytical Q-factor model is valuable for evaluating the QoT of a connection prior to its establishment, upon network changes (i.e., network upgrades, equipment repairs, equipment replacement, aging, etc.) the time-consuming procedure described in [3] must be conducted all over again. This, however, entails the usage of specific equipment, lab measurements, considerable human effort, and a process for deciding whether model re-evaluation is really necessary or not. In this work we explore a data-driven QoT decisions approach that is capable of finding a model that is self-adaptive; it does not require extensive lab measurements or specific equipment, it is time efficient, and it does not require the existence of a system with extensive processing and storage capabilities.

For finding such a model, a feed-forward neural network is utilized, as feed-forward neural networks have been reported to be fast in model evaluation and processing new data, result in compact models (as they require only a few training parameters), be adaptive during training, and also exhibit a high generalization performance. A detailed discussion on feed-forward neural networks and their advantages over other optimization methods in the context of pattern recognition can be found in Chapter 3 of [4].

The general framework of the proposed approach, which entails the utilization of the neural network, is briefly described as follows:

(1) Data from the analytical Q-factor model are generated (or real QoT data are utilized if these are available).
(2) These data are represented in a vector form that is independent from the PLIs but capable of describing the QoT of the connections requesting to be established in the network of interest.
(3) The feed-forward neural network is trained on a training data set.
(4) The accuracy of the neural network model is validated on a data set other than the training data set.

For training purposes, an on-line procedure can be used instead of a batched (off-line) procedure, in which patterns are sequentially fed into the neural network allowing each time for model updates [4]. According to the on-line training procedure, the model can be updated sequentially in the evolving network with insignificant processing overhead. Thus, the technique is self-adaptive, while re-evaluation decisions are not really necessary. Note that the above procedure can be used for any network scenario, provided that the QoT data are represented in a way that is independent from the PLIs but reflects the QoT of the connections requesting to be established in the network of interest.

IV. Problem Formulation

The data-driven QoT problem is treated as a binary classification problem in which the goal is to take an input vector $\boldsymbol{x}$ and to assign it to a discrete value y, where $y \in \{0, 1\}$ [4]. In this work the set $\mathcal{D} = \{(\boldsymbol{x(j)}, y(j)) | j = 1, \ldots, n\}$ is defined to be the historical data set in which pattern j represents lightpath j, $y(j) \in 0, 1$ is the target value, and $\boldsymbol{x(j)}^T = [x_1(j), x_2(j), x_3(j), x_4(j), x_5(j), x_6(j)]$. Specifically,

- $x_1(j)$ is the nominal path length of j;
- $x_2(j)$ is the number of erbium doped fiber amplifiers (EDFAs) in j;
- $x_3(j)$ is the nominal maximum link length of j;
- $x_4(j)$ is the degree of the destination node in j;
- $x_5(j)$ is the nominal wavelength on which j is established; and
- $x_6(j)$ is the bias b of the first layer of the neural network.

Regarding the target values, $y(j) = 0$ when the QoT of j is insufficient, and $y(j) = 1$ otherwise.

A detailed analysis on how the features $x_1(j)$, $x_2(j)$, $x_3(j)$, and $x_4(j)$ are extracted and selected for successfully describing the multicast connections and their associated QoT can be found in our previous work in [5]. In this paper, another feature is included, denoted as $x_5(j)$, representing

the specific wavelength on which lightpath j is established. Briefly, in [5], it was experimentally shown that none of the x_1, x_2, x_3, and x_4 features can be considered separately for accurate QoT decisions, due to the unacceptable high uncertainty regions that were created between the sufficient and insufficient QoT input features, resulting from a single type of feature (i.e., resulting only for type x_i feature). Rather it was shown that, if a number of different types of features are considered jointly, two near-separable sets are possible.

Apart from the data features x_1, x_2, x_3, and x_4, other features were also extracted (for the work in [5]) as candidate input data features to the learning algorithm. Such features were the degree of the source node and the minimum/average link lengths of each lightpath. However, they were not considered in the learning algorithm, as they were either not found to contribute additional information or were found to be uncorrelated to the QoT, at least for the network architecture/engineering under consideration. As an example, the splitting losses of source/intermediate node/s in the lightpaths are uncorrelated to the QoT because, in the engineering case considered, variable optical attenuators are used to equalize the signals at the EDFAs just after the optical splitters. Thus, the equalization procedure "hides" the amount of attenuation at these nodes, as the attenuators equalize the total power of incoming wavelengths to a specified worst-case value determined by the maximum splitting losses (maximum degree node) in the network (eventually, the power of all signals into the EDFAs is equal). At the destination nodes, the signals are dropped before facing equalization. Thus, only the destination's node degree is included as a feature correlated to the QoT (valuable feature for the learning process).

V. Neural Networks With Dropout

As pointed out, during training, the dropout technique [6] is adopted. Dropout is a regularization method for preventing units from co-adapting too much by randomly dropping units (along with their connections) from the neural network during training.

For describing the dropout neural network model, consider the neural network of Fig. 1 with one hidden layer. Let $l \in 0, 1$ index the input layer and the hidden layer of the network, respectively. Let $z^{(l)}$ denote the vector of inputs into layer l and $o^{(l)}$ denote the vector of outputs from layer l, with $o^{(0)} = x$ being the input. $W^{(l)}$ and $b^{(l)}$ are the weights and biases at layer l. The feed-forward operation of the standard neural network of Fig. 1 is described by Eqs. (1) and (2), (for any hidden unit k and for $l = 0$):

$$z_k^{(l+1)} = w_k^{(l+1)} o^l + b_k^{(l+1)}, \tag{1}$$

$$o_k^{(l+1)} = f(z_k^{(l+1)}), \tag{2}$$

where f can be any activation function. In this work the activation function is given by Eq. (3):

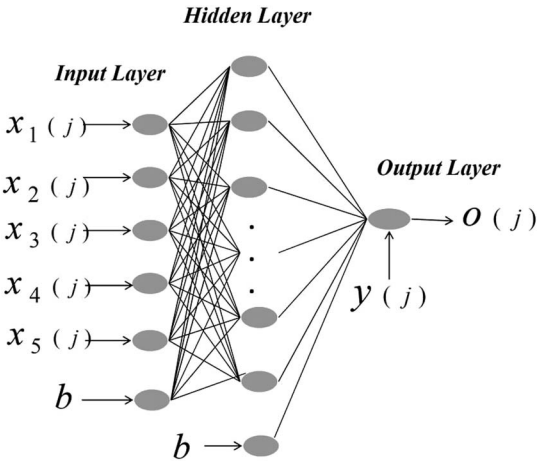

Fig. 1. Neural network with six input units, one hidden layer with several units, and one output.

$$f(x) = \tanh(x) = \frac{2}{1 + \exp(-2x)} - 1. \tag{3}$$

With dropout [6], the feed-forward operation becomes

$$r_v^{(l)} \sim \text{Bernoulli}(p), \tag{4}$$

$$\tilde{o}^{(l)} = r^{(l)} * o^{(l)}, \tag{5}$$

$$z_k^{(l+1)} = w_k^{(l+1)} \tilde{o}^{(l)} + b_k^{(l+1)}, \tag{6}$$

$$o_k^{(l+1)} = f(z_k^{(l+1)}), \tag{7}$$

where $*$ denotes an element-wise product, and $r^{(l)}$ is a vector of independent Bernoulli random variables, each of which has probability p of being 1. This vector is sampled and multiplied element-wise with the outputs of that layer, $o^{(l)}$, to create the thinned outputs $\tilde{o}^{(l)}$. An example of a thinned network produced by applying dropout to the network is illustrated in Fig. 2. The thinned outputs are then used as input to the next layer, and this process is applied at each layer. For learning, the derivatives of the loss function are backpropagated through the subnetwork. In this work, the mean-square loss function is utilized:

$$E = \sum_{j=1}^{n} (y(j) - o(j))^2, \tag{8}$$

where n is the number of the input patterns, $y(j)$ is the expected output value for pattern j, and $o(j)$ is the output evaluated by the training algorithm after the presentation of pattern j to the network.

Dropout neural networks are trained using stochastic gradient descent in a manner similar to standard neural networks, with the only difference being that, for each training case in a mini-batch, a thinned network is sampled by dropping out units. Forward and backpropagation for that training case are performed only on this thinned network.

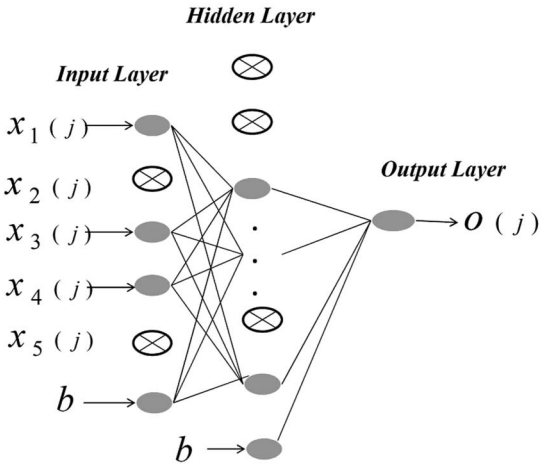

Fig. 2. Example of a thinned network produced by applying dropout to the network of Fig. 1. Crossed units have been dropped.

The gradients for each parameter are averaged over the training cases in each mini-batch. Any training case that does not use a parameter contributes a gradient of zero for that parameter [6]. In this work, for training the dropout neural network, the Adam algorithm is used, as described in [7]. In general, the Adam algorithm is an efficient stochastic optimization method that only requires first-order gradients with little memory requirements; it is well suited for problems that are large in terms of data and/or parameters, and it has been shown to outperform other stochastic optimization methods [7]. At test time, the trained weights are scaled as $\boldsymbol{W}_{\text{test}}^{(l)} = p\boldsymbol{W}^{(l)}$, and the resulting neural network is used without dropout [6].

VI. Data Set Generation

In this section we describe how the data set $\mathcal{D}$ is generated from a dynamic IA-UMC-RWA system. For data generation, the network of Table I is used. Note that the network of Table I was generated in such a way so as to meet the characteristics of a metro area network for which the Q-factor model is valid (in terms of average distance between the network nodes, number of network nodes, average node degree). The network statistics are given in Table II.

For generating the data set $\mathcal{D} = \{(\boldsymbol{x}(j), y(j)) | j = 1, \ldots, n\}$, the Q-factor model described in detail in [2,13] in combination with the multicast-capable network architecture/engineering with fixed TXs/RXs reported in [13] were used. In order to generate the data set $\mathcal{D}$, 10,000 requests were generated with the multicast group sizes varying between 1 and 7, thus accounting for both multicast and unicast connections. Requests were generated according to a Poisson process with exponentially distributed holding times with a unit mean, for a network load of 100 Erlangs. Data were generated assuming that 4, 8, 16, and 32 C-band wavelengths were available. The Q-threshold was set at 9 dB, corresponding to a BER of 10^{-15}. Note that such a small BER was set, as FEC coding was not assumed in the network receivers. Moreover, doing so, we have managed to generate a data set with a large number of patterns with insufficient QoT for training/validating the data-driven QoT model.

TABLE I
Network Connectivity

Link	Distance (km)	Link	Distance (km)
(1, 2)	100	(1, 3)	100
(2, 3)	75	(2, 4)	100
(1, 9)	80	(3, 6)	100
(4, 11)	70	(4, 5)	60
(5, 6)	75	(5, 7)	60
(6, 8)	100	(6, 13)	90
(7, 9)	60	(9, 10)	60
(8, 10)	100	(10, 12)	75
(11, 14)	100	(10, 14)	100
(12, 13)	100	(13, 14)	60
(1, 4)	100	(4, 10)	60
(2, 9)	70	(5, 11)	90
(3, 14)	30	(13, 8)	40
(3, 5)	50	(9, 14)	25
(6, 14)	50	(9, 12)	40
(10, 6)	25	(7, 11)	40
(3, 12)	30	(6, 9)	35
(13, 4)	40	(7, 1)	60
(12, 5)	100	(3, 10)	90
(11, 12)	80	(8, 9)	100
(11, 1)	100	(10, 11)	60
(4, 14)	20	(5, 14)	100
(2, 12)	30	(3, 13)	20
(5, 10)	40	(5, 9)	30
(10, 13)	30	(9, 3)	100

TABLE II
Network Statistics

Number of nodes	14
Number of bidirectional links	50
Average distance (km)	67
Maximum distance (km)	100
Minimum distance (km)	20
Average node degree	7.15
Minimum node degree	4
Maximum node degree	10
Diameter (km)	160
Diameter (hops)	3

Figure 3 shows a flowchart of the dynamic IA-UMC-RWA algorithm used for data generation. For routing the multicast connections the Steiner tree (ST) heuristic [14] was used, for routing the unicast connections the Dijkstra's algorithm [15] was used, and for the wavelength assignment the first-fit algorithm [16] was utilized. Briefly, the dynamic IA-UMC-RWA algorithm, for each call, calculates on wavelength u a unicast/multicast lightpath/lightree (P/T), and the call is accepted if

- there are available TXs/RXs to support the connection;

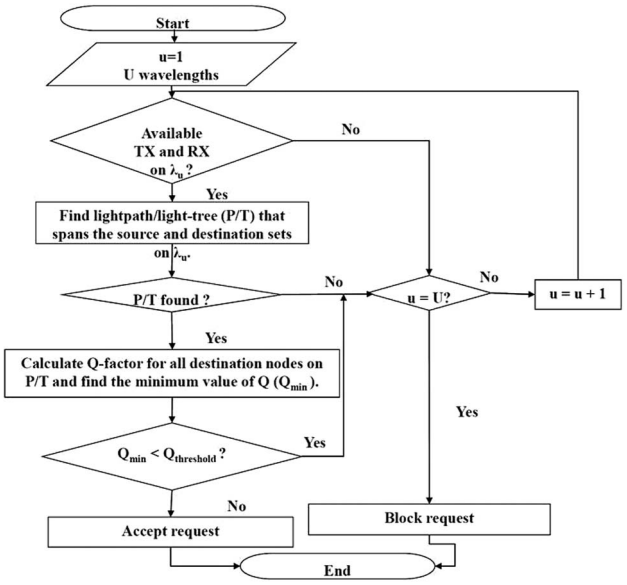

Fig. 3. Flowchart of the dynamic IA-UMC-RWA algorithm utilizing the Q-factor model. Note that U denotes the total number of wavelengths in the network, P denotes a point-to-point lightpath, and T denotes a multicast light-tree.

- there exists a lightpath/light-tree from the source to every destination node in the call; and
- the Q-factor is above the Q-threshold for every destination node in the call.

If at least one of the constraints is not satisfied, a new wavelength assignment is implemented, and the heuristic is repeated until no new wavelength assignment is possible. In that case, the call is blocked.

The data set $\mathcal{D}$ was created during the execution of the dynamic IA-UMC-RWA algorithm. In particular, features $\boldsymbol{x}(j)$ and $y(j)$ were extracted from each lightpath/light-tree j attempted to be established into the network. For each light-tree a number of patterns was generated by decomposing each light-tree to its constituent lightpaths. The Q-factor of each destination was compared with the predetermined Q-threshold and converted to the binary target value $y(j)$; $y(j) = 0$ for a Q-factor below the predetermined Q-threshold, and $y(j) = 1$ otherwise. Details regarding the number of patterns generated for each wavelength case, $U = 4$, $U = 8$, $U = 16$, and $U = 32$ can be found in Table III.

TABLE III
PATTERNS GENERATED FOR EACH WAVELENGTH CASE

	$U = 4$	$U = 8$	$U = 16$	$U = 32$
# patterns in $\mathcal{D}$	50,322	82,010	91,221	91,338
# patterns with $y(j) = 1$	36,152	60,966	69,594	69,300
# patterns with $y(j) = 0$	14,170	21,044	21,627	22,038

VII. EXPERIMENTAL RESULTS

For the experiments, the neural network (NN) toolbox developed by the author of [17] was used on a MATLAB machine with a CPU at 2.60 GHz and 8 GB RAM. The data set $\mathcal{D}$ was divided into the training data set $\mathcal{D}^r$ and the test data set $\mathcal{D}^t$ in such a way that an equal number of patterns with $y(j) = 1$ and $y(j) = 0$ was included in each one of the sets. In general, the training set is used to teach the network, while the test set provides a completely independent measure of network accuracy. For every wavelength case examined, the neural network was trained with 100 hidden units and a batch size of 1000 patterns.

A. Model Accuracy

Tables IV–VII summarize the accuracy achieved within the test data set $\mathcal{D}^t$, for the cases of $U = 4$, $U = 8$, $U = 16$, and $U = 32$, respectively. Information regarding the number of patterns in the training data set ($\mathcal{D}^r$) and the test data set ($\mathcal{D}^t$) is also shown in Tables IV–VII. Note that, for each wavelength case examined, the data-driven QoT model was evaluated for three $\mathcal{D}^r$ data sets with each one containing a different number of patterns. This was done in order to investigate the impact of the number of training patterns to the model accuracy and time convergence. For each $\mathcal{D}^r$ data set, information regarding the dropout fraction (p), the number of epochs required for model convergence, and the training time is also reported in Tables IV–VII.

According to Tables IV–VII, the training time for every wavelength case and for every $\mathcal{D}^r$ data set examined indicates a fast model convergence (the model converged within 1.5–5.3 min). Regarding the impact of the number of patterns to the training time, the results indicate that

TABLE IV
ACCURACY RESULTS FOR $U = 4$

# of patterns in $\mathcal{D}^r$	6000	12,000	24,000
# of patterns in $\mathcal{D}^t$	2000	2000	2000
Dropout fraction (p)	0.8	0.8	0.8
# of epochs	700	700	700
Training time in min	0.7	1.5	3
Class 1 Acc. (%)	90.8	87.6	89
Class 2 Acc. (%)	87.3	100	99.9
Total Acc. (%)	89	93.8	94.45

TABLE V
ACCURACY RESULTS FOR $U = 8$

# of patterns in $\mathcal{D}^r$	6000	12,000	24,000
# of patterns in $\mathcal{D}^t$	2000	2000	2000
Dropout fraction (p)	0.8	0.8	0.8
# of epochs	700	700	700
Training time in min	0.6	1.5	3
Class 1 Acc. (%)	88.5	85	90
Class 2 Acc. (%)	96	100	99
Total Acc. (%)	92.25	92.5	94.5

TABLE VI
ACCURACY RESULTS FOR $U = 16$

# of patterns in $\mathcal{D}^r$	8000	18,000	36,000
# of patterns in $\mathcal{D}^t$	2000	2000	2000
Dropout fraction (p)	0.75	0.75	0.75
# of epochs	800	800	800
Training time in min	1	2.6	5.3
Class 1 Acc. (%)	74.4	87.5	93.3
Class 2 Acc. (%)	100	99.9	96.4
Total Acc. (%)	87.2	93.7	94.8

TABLE VII
ACCURACY RESULTS FOR $U = 32$

# of patterns in $\mathcal{D}^r$	8000	18,000	36,000
# of patterns in $\mathcal{D}^t$	2000	2000	2000
Dropout fraction (p)	0.75	0.75	0.75
# of epochs	800	500	800
Training time in min	1	2	5.3
Class 1 Acc. (%)	91.7	87.7	93.7
Class 2 Acc. (%)	86.6	99.1	97.4
Total Acc. (%)	89.15	93.4	95.5

the training time increases as the number of patterns increases, but this increase is not significant (up to 5.3 min for the largest in size dataset). Regarding the accuracy results, each table shows the overall model accuracy achieved (Total Acc.), and the model accuracy achieved in each one of the two Classes, 1 (Class 1 Acc.) and 2 (Class 2 Acc.). Note that, in Tables IV–VII, Class 1 refers to the patterns with sufficient QoT, and Class 2 refers to the patterns with insufficient QoT. The number of correctly classified patterns in the entire data set $\mathcal{D}^t$ can be evaluated by multiplying the total accuracy percentage by the number of patterns in the $\mathcal{D}^t$ data set (Total Acc. × 2000). The number of correctly classified patterns in each class can be evaluated by multiplying the class accuracy by half of the number of patterns in the $\mathcal{D}^t$ data set (i.e., Class 1 Acc. × 1000).

According to the results shown in these tables, the approach achieved an overall high accuracy for every wavelength case examined, particularly for the larger training data sets (i.e., the $\mathcal{D}^r$ data sets that include more than or equal to 12,000 patterns). For the $\mathcal{D}^r$ data sets that include fewer patterns (i.e., 6000 or 8000 patterns), the results in most wavelength cases ($U = 4$, $U = 16$, $U = 32$) cannot be considered to be of high accuracy. This indicates the importance of considering a large enough $\mathcal{D}^r$ data set during the training procedure. Note that the $\mathcal{D}^r$ data set size, which is capable of returning an accurate enough model, can only be evaluated empirically for each network scenario. In general, as the number of patterns in the $\mathcal{D}^r$ data set increases, so does the total accuracy. According to the results of the largest in size $\mathcal{D}^r$ data sets, the approach achieved an overall accuracy of 94.45% for the case of $U = 4$, 94.5% for the case of $U = 8$, 94.8% for the case of $U = 16$, and 95.5% for the case of $U = 32$. These results correspond to the rightmost column of each one of the Tables IV–VII.

Regarding the accuracy results in each class, the approach achieved a high accuracy in both classes, for every wavelength case examined. Note that the results show that Class 2 achieved an accuracy nearing 100%, meaning that a pattern with insufficient QoT will almost never be perceived as a pattern with sufficient QoT (the end-users will almost never experience a connection with unacceptable QoT). The above can be observed by inspecting the last two columns of Tables IV–VII for which the approach achieved an overall high accuracy (92%–95%) for every wavelength case. In comparison, it is interesting to note that, when the BP algorithm [18] was also used for training the neural network as described in [5], the resulting data-driven QoT model achieved, for every wavelength case examined, an accuracy close to 89%. This accuracy was the best result achieved after a number of trials that examined different learning rates, numbers of hidden units, and weight initializations.

B. Performance Evaluation

The network performance, in terms of blocking probability, is also evaluated when the QoT decisions are taken utilizing the data-driven QoT model and is compared with the network performance achieved when the Q-factor model is used. In order to evaluate the impact of the QoT constraint on the network performance, results were also obtained for the conventional UMC-RWA algorithm, which does not account for the QoT constraint. Thus the network performance is evaluated for three cases:

- Case 0: The QoT constraint is not considered during the dynamic UMC-RWA algorithm.
- Case 1: The Q-factor model is utilized in the dynamic IA-UMC-RWA algorithm for the QoT decisions.
- Case 2: The data-driven QoT model is utilized in the dynamic IA-UMC-RWA algorithm for the QoT decisions.

Figure 4 shows a flowchart for the dynamic IA-UMC-RWA algorithm in which the data-driven QoT model, denoted as $o_d(j) = M(\boldsymbol{x}_d(j), \boldsymbol{W})$, is used for the QoT decisions. Note that $\boldsymbol{x}_d(j)$ is the features vector extracted from the lightpath j that terminates at destination node d, $\boldsymbol{W}$ are the trained parameters of the neural network, and $o_d(j)$ is the output (zero or one) evaluated by the model M. In Fig. 4 each new lightpath/light-tree (P/T) j is decomposed to its constituent lightpath/s, one for each destination node $d \in P/T$ with D being the total number of destination nodes in P/T. According to the dynamic IA-UMC-RWA algorithm of Fig. 4, a connection is accepted into the network if

- there are available TXs/RXs to support the connection;
- there exists a lightpath/light-tree from the source to every destination node in the connection; and
- $\Sigma_{d=1}^{D} o_d(j) = D$, where $o_d(j) = M(\boldsymbol{x}_d(j), \boldsymbol{W}) \forall\ d \in P/T$, indicating that the QoT is sufficient for every destination node in the connection.

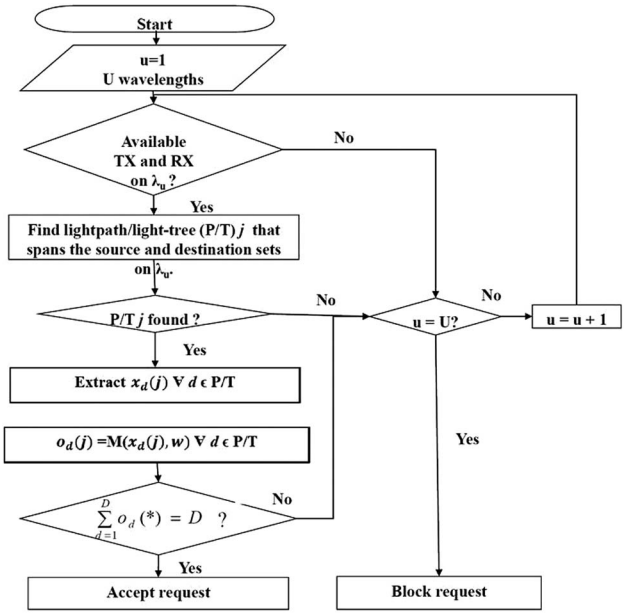

Fig. 4. Flowchart of the dynamic IA-UMC-RWA algorithm utilizing the data-driven QoT model (Case 2). Note that U denotes the total number of wavelengths in the network, P denotes a point-to-point lightpath, and T denotes a multicast light-tree.

If at least one of the constraints is not met, a new wavelength assignment is implemented, and the heuristic is repeated until no new wavelength assignment is possible. In the latter case, the call is blocked.

For the simulations, the network parameters described in Section VI were utilized in all three cases: Case 0 (the QoT constraint is removed), Case 1 (Algorithm of Fig. 3), and Case 2 (Algorithm of Fig. 4). Five simulation runs were performed for each case, and the results were averaged over all simulation runs. Tables VIII–XI summarize the performance results for the cases of $U = 4$, $U = 8$, $U = 16$, and $U = 32$, respectively. Specifically, all tables report the overall blocking probability (overall Pr\{blocking\}), which is followed by the breakdown of the results to the blocking probability due to insufficient QoT (Pr\{blocking\} due to QoT) and to the blocking probability due to the lack of wavelengths (Pr\{blocking\} due to wav.). For Case 0, where the QoT constraint is not considered, all blocking is caused due to the lack of wavelengths. For Case 2, two different data-driven QoT models were considered. These models are the trained models of Subsection VII.A that correspond to the last two columns of each one of the Tables IV–VII. For denoting the exact model considered each time in Case 2, the total accuracy of each model is included in parenthesis within the Case 2 column in Tables VIII–XI.

TABLE VIII
BLOCKING PROBABILITY RESULTS FOR $U = 4$

	Case 0	Case 1	Case 2 (93.8)	Case 2 (94.45)
Overall Pr\{blocking\}	0.42	0.44	0.44	0.44
Pr\{blocking\} due to QoT	–	0.15	0.25	0.21
Pr\{blocking\} due to wav.	0.42	0.29	0.19	0.23

TABLE IX
BLOCKING PROBABILITY RESULTS FOR $U = 8$

	Case 0	Case 1	Case 2 (92.5)	Case 2 (94.5)
Overall Pr\{blocking\}	0.09	0.11	0.15	0.12
Pr\{blocking\} due to QoT	–	0.04	0.1	0.06
Pr\{blocking\} due to wav.	0.09	0.07	0.05	0.06

For all three cases, the results show that, as the number of wavelengths increases, the blocking probability decreases. For Cases 1 and 2, when $U = 4$ (Table VIII) and $U = 8$ (Table IX) a significant percentage of the blocking is caused due to the QoT constraint. When $U = 16$ (Table X) and $U = 32$ (Table XI) the blocking probability caused due to the QoT constraint becomes insignificant. The impact of the QoT constraint to the system performance can be viewed by comparing the Case 0 results with the Case 1 and 2 results of Tables VIII–XI. It is clear that the impact of the QoT constraint reduces as the number of wavelengths increases.

As pointed out, for the wavelength cases $U = 16$ and $U = 32$, there is insignificant QoT blocking. However, this does not mean that connection requests of insufficient QoT did not appear during the IA-UMC-RWA procedure. In fact, Table III indicates that approximately 23% of the overall lightpaths that attempted establishment were of insufficient QoT. However, those requests were not blocked. The IA-UMC-RWA algorithm just attempted to provision them utilizing the next wavelength on the list. Because in the two aforementioned cases enough wavelength capacity exists in the network, a wavelength was eventually found for which the QoT was found to be sufficient, and these requests were ultimately accommodated. On the contrary, for the $U = 4$ and $U = 8$ wavelength cases, the capability of the IA-UMC-RWA algorithm to find alternate wavelengths with sufficient QoT is limited due to the small number of available wavelengths, thus resulting in significant blocking due to insufficient QoT.

It is also important to note that, when the QoT is taken into account (IA-UMC-RWA algorithm for Cases 1 and 2), a drop is observed in the blocking probability due to the lack of wavelengths, compared with the conventional UMC-RWA algorithm (Case 0). This effect is obvious only for the $U = 4$ and $U = 8$ wavelength cases where blocking is present in Case 0. This is true because blocking due to insufficient QoT means that more capacity (wavelengths) is

TABLE X
BLOCKING PROBABILITY RESULTS FOR $U = 16$

	Case 0	Case 1	Case 2 (93.7)	Case 2 (94.8)
Overall Pr\{blocking\}	0	0.0002	0.0014	0.0004
Pr\{blocking\} due to QoT	–	0.0002	0.0014	0.0004
Pr\{blocking\} due to wav.	0	0	0	0

now available for future connections to be established. Note, however, that even if additional lightpaths exist, it does not mean that these lightpaths will satisfy the QoT constraint. Thus, a rise in the total blocking probability may appear (especially when not enough wavelengths are present). This is indeed the case, as demonstrated by the results in Tables VIII and IX that clearly show that the overall blocking probability for Cases 1 and 2 is eventually higher compared to the blocking probability of Case 0.

Regarding the performance results of the data-driven QoT model (Case 2), Tables VIII–XI clearly show that the overall blocking probability is only minimally affected when the data-driven QoT model is used instead of the Q-factor model (Case 1), especially when the number of available wavelengths in the network is large enough. Specifically, when $U = 32$ (Table XI), the overall blocking probability is near zero for both Cases 1 and 2. Further, the breakdown of the results shows that the blocking probability due to insufficient QoT is only slightly increased, while the blocking probability due to the lack of wavelengths is not affected. This is due to the fact that the data-driven model fails by a small percentage to correctly classify the Class 1 patterns [a percentage of the Class 1 patterns (6.3% for the total accuracy of 95.5% and 12.3% for the total accuracy of 93.4%) are misclassified to Class 2]. This means that now the IA-UMC-RWA algorithm has to face more patterns with insufficient QoT, compared to the IA-UMC-RWA algorithm that utilizes the Q-factor model. Thus the QoT blocking probability increases. However, this slight misclassification of Class 1 patterns does not eventually affect the network performance, as the connection requests misclassified to Class 2 are eventually accommodated via an alternate wavelength for which the QoT is correctly classified to Class 1.

As the number of available wavelengths decreases, the fact that the data-driven model fails by a small percentage to correctly classify the Class 1 patterns negatively affects the network performance. Specifically, for $U = 16$ (Table X) and $U = 8$ (Table IX), this slight misclassification gives an increase in the blocking probability due to insufficient QoT that eventually affects the overall blocking probability. This is due to the fact that not enough wavelengths are now available to mitigate the impact of this misclassification (Table VI for $U = 16$ and Table V for $U = 8$). Note, however, that as the number of training patterns increases, and the data-driven model becomes more accurate (94.8% accuracy for $U = 16$ and 94.5% accuracy for $U = 8$), the overall network performance is insignificantly affected, compared to the network performance of Case 1.

TABLE XI
BLOCKING PROBABILITY RESULTS FOR $U = 32$

	Case 0	Case 1	Case 2 (93.4)	Case 2 (95.5)
Overall *Pr*{blocking}	0	0	0.0004	0.0004
Pr{blocking} due to QoT	–	0	0.0004	0.0004
Pr{blocking} due to wav.	0	0	0	0

It is also important to note that, for $U = 8$ in Case 2, a drop occurs in the blocking probability due to the unavailability of wavelengths, compared with Case 1. As previously discussed, this drop is a consequence of the increased blocking due to insufficient QoT caused due to the fact that the data-driven model fails by a small percentage to correctly classify the Class 1 patterns. Thus, increased blocking due to insufficient QoT means that more capacity (wavelengths) is now available for future connections to be established. Again, even though additional lightpaths exist, it does not mean that these lightpaths will satisfy the QoT constraint. As shown in Table IX, the impact of this effect reduces as the model becomes more accurate (the inaccurate percentage of Class 1 patterns reduces from 15% to 10% when 24,000 training patterns are used instead of 12,000).

Finally, for the case of $U = 4$, the blocking probability is unacceptably high for all three Cases: 0, 1, and 2. The inaccurate percentage of Class 1 patterns causes an increase in the blocking probability due to insufficient QoT when the data-driven model is used, which, in turn, as discussed above, causes a drop in the blocking probability due to the lack of wavelengths. Both Cases, 1 and 2, achieve the same overall blocking probability (Table VIII), as the network performance is limited by the number of available wavelengths; that is, if the connections would not have been blocked due to insufficient QoT, they would have been blocked due to the lack of wavelengths. This conclusion is evident when the results of Cases 1 and 2 are compared with the results of Case 0 where the QoT constraint is not considered.

VIII. PRACTICAL FEASIBILITY CONSIDERATIONS

Regarding the practical feasibility of the proposed approach, we have assumed a centralized approach in which a centralized controller exists that receives the lightpath/light-tree requests and is responsible for path computation, wavelength assignment, validation of the QoT, and lightpath/light-tree setup. Thus, we have assumed a controller that maintains a centralized traffic engineering (TE) database with detailed wavelength availability information and is able to specify the full details of the lightpath/light-tree (i.e., all switches and ports along the lightpath/light-tree). The control plane can verify the feasibility of the lightpath/light-tree and performs the lightpath/light-tree provisioning. Then, the controller updates its traffic engineering database, and all the traffic engineering parameters are stored and updated within the controller.

From the traffic engineering database, lightpath/light-tree information can be extracted and stored in a knowledge database for training/validating the data-driven QoT model proposed. According to the proposed data-driven QoT approach, the input patterns x can either be readily found or easily computed by the information included in the centralized traffic engineering database (lightpath length, number of amplifiers, degree of destination node, maximum link length, assigned wavelength). Information

regarding the QoT of the lightpaths can be obtained after the establishment of the connections, at the destination nodes and/or by the appropriately placed monitoring equipment.

Centralized control plane solutions for supporting optical network technologies have been extensively considered [9,19–21]. In this work, the proposition of a specific control plane solution for supporting the proposed data-driven QoT approach is out of the scope and is planned for future work. However, regarding the existing efforts, the authors of [9,19–21] proposed control plane solutions for optical network technologies that are based on OpenFlow extensions. In [19–21], the QoT considerations are not specifically addressed. Only work in [9] considers the QoT constraint during the path computation, by performing OSNR monitoring. Further, in [22] a centralized approach has been proposed for encompassing physical impairments in transparent optical networks. In that work, a path computation element (PCE) was considered, capable of computing a network path or route based on a network graph and applying computation constraints during the path computation. The PCE is aware of the physical parameters that are stored by the PCE in a locally managed physical parameter database. The physical parameters (i.e., polarization mode dispersion, chromatic dispersion, self-phase modulation, worst-case penalties) are obtained by the management system or through a performance monitoring system. Note that, while similar solutions can be developed for supporting the proposed data-driven QoT approach, the physical parameters will not be needed, as now only a label is required indicating an acceptable or an unacceptable QoT for each lightpath attempted to be established into the network.

Another important issue that naturally arises regarding the practical feasibility of the proposed approach can be stated as follows: How much data are enough for training an accurate data-driven QoT model, and how much time would it take for collecting such a number of data? Clearly, there are no straightforward answers to these questions, as the amount of collected data and the time for this data collection depend on several factors such as the dynamicity of the network (how quickly does the network state change), the network topology, the number of available wavelengths, the diversity of data collected, the network equipment utilized, etc. Nevertheless, for ensuring the diversity and quick collection of data, appropriate schemes can be developed. For example, strategically placed monitoring equipment can be used for collecting QoT data. In doing so, QoT data at intermediate points of the established lightpaths (not only at the destination nodes) can be collected, enriching the historical data set. Further, lightpaths can be established, according to a strategic plan, only for data collection purposes. The investigation of such topics is also planned as future work.

As pointed out, in the Q-factor model, worst-case budget values need to be initially evaluated for each PLI in the network, either through measurements in the lab for certain PLIs, or theoretical models for other PLIs, and these values must be tested in the lab by performing experiments on a metro network. On the contrary, in the data-driven model, only OSNR measurements are needed. These can be obtained through lab experiments on a number of lightpaths that can be established. The rest of the features associated with the data-driven QoT model are more easily extractable from the lightpaths and do not depend on the PLIs (e.g., the length of the lightpath, the maximum link length of the lightpath, the specific wavelength on which the lightpath is established, and the degree of the destination node in the lightpath). These features do not need any specific measurement equipment or the utilization of theoretical models. They are readily available after the routing and lightpath establishment procedure takes place. Once the network is up and running, the data-driven model can subsequently be periodically re-trained on data extracted from the evolving network. In doing so, the data-driven model can become more and more accurate. Thus the savings of the proposed technique over the Q-factor model include, apart from the human effort that is needed for modeling the Q-factor, the equipment savings that are required for evaluating the PLIs.

Note that, for training an initial data-driven QoT model, the acceptable OSNR threshold can be set to a lower value than the one the network of interest can tolerate. In doing so, a safety margin can be included in the trained model, resembling the safety margin included in the Q-factor model. Once a data-driven QoT model of acceptable accuracy is trained, then the data-driven model can only become more accurate once it is exposed to true data collected from the evolving network. This can be achieved in an automatic manner by periodically retraining the model, as the training procedure is not time prohibitive (required approximately 5 min for a data set of 36,000 patterns). The latter, however, is not possible with the Q-factor model that includes worst-case budget values and can only be re-evaluated through a more complex procedure. Further, in that case, decisions would also be needed on whether the re-evaluation procedure of the Q-factor model is necessary or not.

IX. Conclusions

In this work, a data-driven technique is utilized for analyzing QoT data of previous unicast and multicast connection requests aiming at accurately deciding the QoT of newly arriving connections. A Q-factor model, which is a function of each impairment present, is first utilized for generating a QoT data set on a dynamic IA-UMC-RWA system, which is used for training and testing an NN model.

The resulting data-driven QoT model was tested for a metro area network utilizing 4, 8, 16, and 32 wavelengths. In every wavelength case, the training algorithm converged very fast (within minutes) in a model of high accuracy [close to 92%–95% accuracy in determining, during the provisioning phase, whether a connection to be established has a sufficient (or insufficient) QoT, when compared with the QoT decisions performed by the Q-factor model]. The network performance was also evaluated when the data-driven QoT model was used for the QoT decisions,

instead of the *Q*-factor model. The results showed that the network performance is not significantly affected when the data-driven QoT model is used instead of the *Q*-factor model, provided that enough wavelengths are present for ensuring a practical network performance.

The advantages of the proposed approach over the existing *Q*-factor model are several: it is self-adaptive, it is fast, and it does not require special measurement equipment or the existence of a system with extensive processing and storage capabilities. It can therefore replace the *Q*-factor model that lacks self-adaptiveness, requires lab experiments, specific measurement equipment, and considerable human effort. It is important to note that the proposed technique can be applied for any network scenario (e.g., flex-grid optical networks, different network architecture/engineering, different modulation formats, etc.), provided that the data selected for the data-driven QoT model are independent from the PLIs but reflect the QoT of the connections requesting to be established in the network of interest. Clearly, several practical feasibility issues are still open for investigation (control plane solution, number of data required, time required for data collection), and these are planned for future work.

References

[1] C. Politi, V. Anagnostopoulos, C. Matrakidis, and A. Stavdas, "Physical layer impairment aware routing algorithms based on analytically calculated *Q*-factor," in *Optical Fiber Communication Conf. and the Nat. Fiber Optic Engineers Conf. (OFC/NFOEC)*, Anaheim, CA, Mar. 2006, paper OFG1.

[2] G. Ellinas, N. Antoniades, T. Panayiotou, A. Hadjiantonis, and A. M. Levine, "Multicast routing algorithms based on *Q*-factor physical layer constraints in metro networks," *IEEE Photonics Technol. Lett.*, vol. 21, pp. 365–367, 2009.

[3] N. Antoniades, A. Boskovic, I. Tomkos, N. Madamopoulos, M. Lee, I. Roudas, D. Pastel, M. Sharma, and M. J. Yadlowsky, "Performance engineering and topological design of metro WDM optical networks using computer simulation," *IEEE J. Sel. Areas Commun.*, vol. 20, pp. 149–165, 2002.

[4] C. M. Bishop, *Pattern Recognition and Machine Learning*. New Jersey: Springer, 2006.

[5] T. Panayiotou, G. Ellinas, and S. P. Chatzis, "A data-driven QoT decision approach for multicast connections in metro optical networks," in *Proc. Optical Network Design and Modeling (ONDM)*, Cartagena, Spain, May 2016.

[6] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, 2014.

[7] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. for Learning Representations*, San Diego, CA, 2015.

[8] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *J. Mach. Learn. Res.*, vol. 12, pp. 2121–2159, 2010.

[9] J. Oliveira, J. Oliveira, E. Magalhães, J. Januário, M. Siqueira, R. Scaraficci, M. Salvador, L. Mariote, N. Guerrero, L. Carvalho, F. Hooft, G. Santos, and M. Garrich, "Toward terabit autonomic optical networks based on a software defined adaptive/cognitive approach [Invited]," *J. Opt. Commun. Netw.*, vol. 7, pp. A421–A431, 2015.

[10] T. Jiménez, J. Aguado, I. de Miguel, R. Durán, N. Fernandez, M. Angelou, D. Sánchez, N. Merayo, P. Fernández, N. Atallah, R. Lorenzo, I. Tomkos, and E. Abril, "A cognitive system for fast quality of transmission estimation in core optical networks," in *Optical Fiber Communication Conf. and Expo. and the Nat. Fiber Optic Engineers Conf. (OFC/NFOEC)*, Los Angeles, CA, Mar. 2012, paper OW3A.5.

[11] T. Jiménez, J. C. Aguado, I. Miguel, R. J. Durán, M. Angelou, N. Merayo, P. Fernández, R. M. Lorenzo, I. Tomkos, and E. J. Abril, "A cognitive quality of transmission estimator for core optical networks," *J. Lightwave Technol.*, vol. 31, pp. 942–951, 2013.

[12] P. Poggiolini, "The GN model of non-linear propagation in uncompensated coherent optical systems," *J. Lightwave Technol.*, vol. 30, pp. 3857–3879, 2012.

[13] T. Panayiotou, G. Ellinas, N. Antoniades, and A. Hadjiantonis, "Impairment-aware multicast session provisioning in metro optical networks," *Comput. Netw.*, vol. 91, pp. 675–688, 2015.

[14] L. H. Sahasrabuddhe and B. Mukherjee, "Multicast routing algorithms and protocols: A tutorial," *Network*, vol. 14, pp. 90–102, 2000.

[15] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numer. Math.*, vol. 1, pp. 269–271, 1959.

[16] S. Yung, J. Gu, and D. H. K. Tsang, "Multicast routing in all-optical wavelength-routed networks," *SPIE Opt. Netw. Mag.*, vol. 2, pp. 101–109, 2001.

[17] R. B. Palm, "Prediction as a candidate for learning deep hierarchical models of data," Master's thesis, Technical University of Denmark, Denmark, 2012.

[18] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, pp. 533–536, 1986.

[19] A. Giorgetti, F. Cugini, F. Paolucci, and P. Castoldi, "OpenFlow and PCE architectures in wavelength switched optical networks," in *Proc. Int. Conf. on Optical Network Design and Modeling (ONDM)*, Colchester, UK, 2012.

[20] M. Channegowda, R. Nejabati, M. Rashidi Fard, S. Peng, N. Amaya, G. Zervas, D. Simeonidou, R. Vilalta, R. Casellas, R. Martnez, R. Muoz, L. Liu, T. Tsuritani, I. Morita, A. Autenrieth, J. Elbers, P. Kostecki, and P. Kaczmarek, "Experimental demonstration of an OpenFlow based software-defined optical network employing packet, fixed and flexible DWDM grid technologies on an international multi-domain testbed," *Opt. Express*, vol. 21, pp. 5487–5498, 2013.

[21] M. Channegowda, R. Nejabati, and D. Simeonidou, "Software-defined optical networks technology and infrastructure: Enabling software-defined optical network operations [Invited]," *J. Opt. Commun. Netw.*, vol. 5, pp. A274–A282, 2013.

[22] P. Castoldi, F. Cugini, L. Valcarenghi, N. Sambo, E. Le Rouzic, M. J. Poirrier, N. Andriolli, F. Paolucci, and A. Giorgetti, "Centralized versus distributed approaches for encompassing physical impairments in transparent optical networks," in *Proc. 11th Int. Conf. on Optical Network Design and Modeling (ONDM)*, Athens, Greece, 2007.