Optimized Management of Ultra-wideband Photonics Switching Systems Assisted by Machine Learning

Ihtesham Khan,[1,*] **Lorenzo Tunesi,**[1] **Muhammad Umar Masood,**[1] **Enrico Ghillino,**[2] **Paolo Bardella,**[1] **Andrea Carena ,**[1] **and Vittorio Curri**[1]

[1] *Politecnico di Torino, Corso Duca degli Abruzzi, 24, 10129, Torino, Italy*
[2] *Synopsys, Inc., 400 Executive Blvd Ste 101, Ossining, NY 10562, United States*
[*] *ihtesham.khan@polito.it*

Abstract: Recent years have seen an unprecedented growth of data traffic driven by a continuous increase of connected devices and new applications. This trend will tend to saturate transparent optical networks that are the backbone of the whole telecommunication infrastructure. To improve the capacity of already deployed network infrastructures and maximize operators CAPEX returns, band-division multiplexing (BDM) has emerged as a promising solution to expand the fiber bandwidth beyond the existing C-band. Along with this, the demand for flexible and dynamically reconfigurable functionalities in each network layer is increasing. In this regard, optical networking is fast evolving towards the applications of the software-defined networking (SDN) paradigm down to the physical layer. The implementation of optical SDN requires the full abstraction and virtualization of each network element in order to enable complete control by a centralized network controller. To pursue this objective, photonics transmission components and their transmission functionalities must be abstracted to allow the definition of the control states and a real-time quality-of-transmission (QoT) evaluation of transparent lightpaths (LP). In this work, we propose an SDN based model of a photonic switching fabric that allows determining the control state and evaluating QoT degradation. Our investigations present a wideband optical switch design based on photonic integrated circuits (PICs), where QoT degradation is abstracted using a structure-agnostic approach based on machine learning (ML). The ML engine training and testing datasets are generated synthetically by software simulation of the photonic switch architecture. Results show the potential of the proposed technique to predict QoT impairments with high accuracy, and we envision its application in a real-time control plane.

1. Introduction

The continuous rise of global internet traffic and the latest evolving technologies such as 5G and internet of things (IoT) will require an increase of optical network capacity together with a demand for flexible and dynamic network functionalities at every layer. State-of-the-art optical transport networks are based on wavelength-division multiplexing (WDM) in the standard spectral window of ≈4.8 THz defined as C-band. An increase in network capacity can be obtained by adopting one of the two unique solutions; (i) exploiting the residual capacity of already installed infrastructure, (ii) deploying new network infrastructures. The initial solution of exploiting the residual capacity of already installed infrastructure is more valuable for network operators from a techno-economic viewpoint. In this scenario, a technique like BDM appears as a promising technology to exploit the residual capacity of existing WDM optical network infrastructure throughout the whole low-loss spectrum of optical fibers (e.g., ≈ 54 THz in ITU G.652.D fiber) [1].

The demand for flexible and dynamic network functionalities in each layer can be provided by implementing the SDN paradigm down to the physical layer [2, 3]. At this level, the SDN is

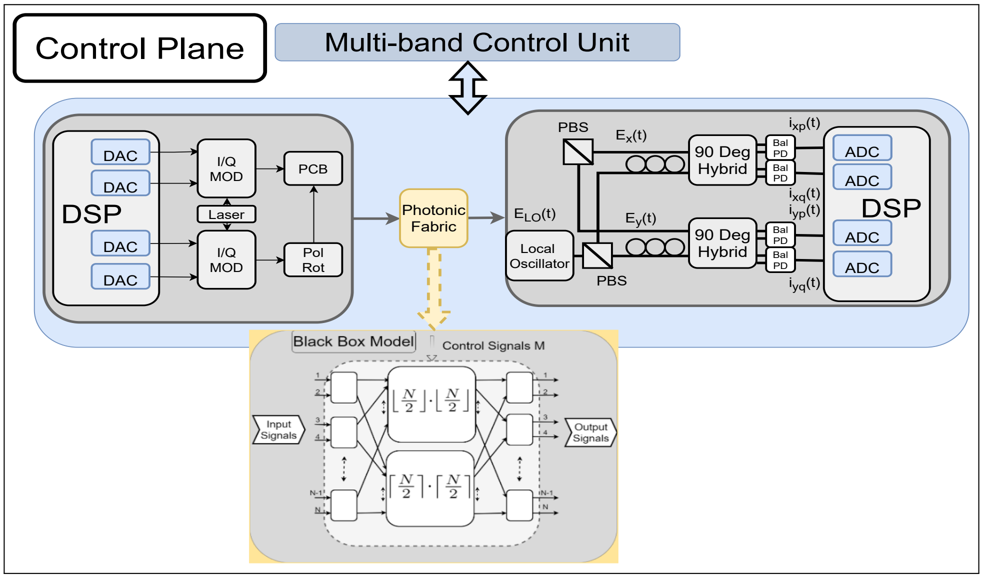


Fig. 1. Abstraction of the optical switching fabric for the control-plane of a SDN-controlled photonic transmission system.

based on a network controller that manages physical links and controls switching elements to optimize performance, i.e., to maximize transmission capacity. SDN needs an open interface for each transmission element with a model for its transmission impairment.

For physical channels, the SDN application requires the capability to summarize the QoT of links in a unique QoT meter, given by the availability of a QoT estimator (QoT-E) to compute it. The application of QoT-E to the WDM optical transport has been made easier by introducing transceivers (TRXs) based on dual-polarization multilevel modulation formats exploiting DSP technologies for spectral shaping and coherent detection. The advent of coherent TRXs has been a game-changer in link design: it allowed the introduction of the uncompensated approach, removing the need for dispersion compensating units that needed specific optimization. Exploiting this transmission technology in transparent LP within the system simplifies the QoT evaluation as the non-linear propagation impact can be easily modeled as an additive white Gaussian noise (AWGN) [4]. This AWGN-like approach defines the requested minimum optical-signal-to-noise ratio (OSNR) of TRXs in a back-to-back characterization and then uses it for determining LP deployment and feasibility. This property enables the full abstraction of the optical transport system through a QoT-E that computes the OSNR of each LP and compares it to the minimum OSNR requested by the coherent TRX [5]. Each transparent LP can be dynamically set from source to destination by setting the traversing switching elements. Consequently, models to evaluate the QoT degradation of each crossed switching element are also needed to compute the overall OSNR. Besides quantifying the QoT degradation due to the switching element, the optical network controller must define the control state for them. To this end, we need models to set operational modes minimizing the QoT degradation, defining the selected transparent LP by properly configuring the switching elements.

Optical network elements currently exploit PICs to carry out most of the complex functions at the photonic level; specifically, optical networks and data centers progressively utilize large-scale photonic switches and wavelength selective switches due to their wide-band abilities together with low latency and low power consumption. PIC-based photonic switches primarily work on the principle that the flow of light can be maneuvered by 2×2 electrically controlled elements, like mach-zehnder interferometers (MZI) [6] or optical micro-ring resonators (MRRs) [7]. Before the development of PIC technology, various switching machinery has been proposed, such as 3D micro-electro-mechanical systems (MEMS) [8] and beam-steering techniques [9]. These technologies offer stable optical switching and a satisfactory level of scalability, but the requirement to calibrate and install discrete components makes them considerably more costly and massive. Cost and size make it challenging to implement these technologies in the future UWB system. Consequently, it boosts the trend of using PICs-based modules. Moreover, this increased use of PICs-based switching systems creates a demand for a generic softwarized management model for photonic switches enabling complete control in the optical SDN context.

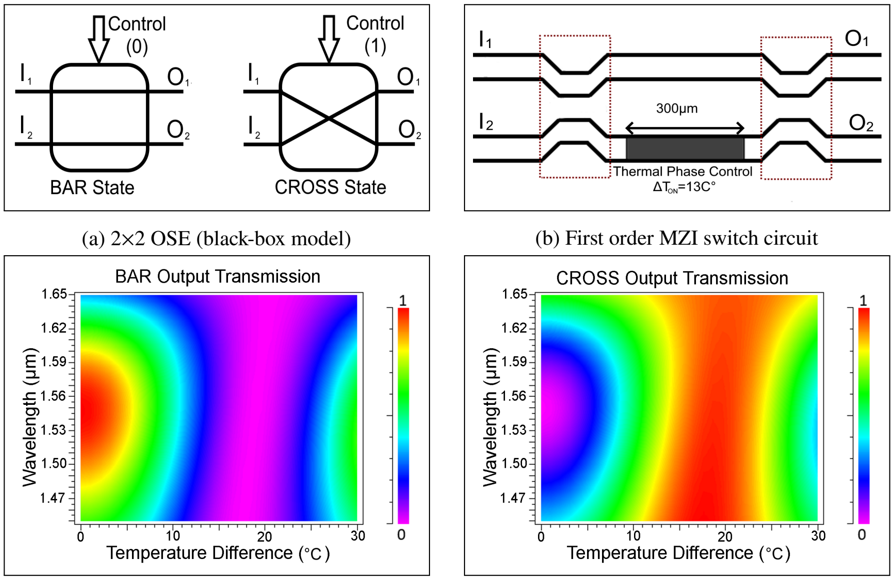
In this work, we focus on the definition of an SDN model of a photonic switching fabric for both control and QoT degradation as pictorially shown in Fig. 1. The principal aim of this investigation is to present the design of a wideband optical switch based on PICs and then model this $N\times N$ ultra-wideband (UWB) switching system at two different levels of abstraction: the routing behavior and the QoT relation to the applied control signals. The routing problem is solved by considering the black-box abstraction of the 2×2 cross-bar switching units on a simplified version of the circuit, taking only into account the ideal link between the elements and the binary control state of each fundamental switch. For the QoT, an ML-based framework is proposed to predict the QoT degradation due to the real switching element. This method is a topological and technological agnostic *blind* approach which exploits neural network methods to model the QoT impairments of the $N\times N$ photonic switch. The introduced data-driven structure is trained on a dataset obtained by considering a $N\times N$ photonic switch. The training dataset can either be obtained experimentally or *synthetically* by using a software simulator for components. The trained ML model provides the QoT impairments in real-time as ML models only require time during the initial training; once the model is substantially trained, it can provide results in real-time for the given application.

The remainder of the paper is organized as follows. In Section 2, we briefly describe the background and the previous related investigations. Section 3, presents switching structures, topologies and their related performances in a UWB system. In Section 4, the simulation model, together with synthetic data generation and analysis, are reported. In Section 5, the orchestration of the ML engine is presented. Then, in Section 6, we describe the results in detail. Finally, conclusions and future research directions are presented in Section 7.

2. Background of the study

Research related to the softwarized management of photonic switching systems has been sparsely performed and reported. The management model for optical switches is essential because of their path-dependent nature [10] as compared to electronic switches, where the performance of all routes is identical [11]. The variations in performance for optical switches are mainly due to the photonic circuit topology, but they can also depend on mask-level design flaws. Usually, the deterministic routing algorithms presented in literature can efficiently determine the control state of internal switches for any given output permutations. The effectiveness of these algorithms comes from their topology dependent nature, which enables a faster and efficient assessment of the multiple-stage networks. On the contrary, these traditional deterministic routing algorithms do not offer all the equivalent paths for a given channels permutation [12–14]. In contrast with traditional routing algorithms, we propose a routing algorithm that produces all the equivalent paths for a given output permutation. The provided routing scheme is then paired up with an ML-based agent capable of predicting the QoT degradation of each calculated path due to the switching and coupling units, thus allowing for the identification of the best control set.

The ML-based approach has already been explored in the area of PIC design and control for different functionalities. An algorithm-driven by the artificial neural network is proposed in [15] to regulate 2×2 dual-ring assisted-MZI switches. In [16] ML is used to assess QoT of PIC in



(a) 2×2 OSE (black-box model)

(b) First order MZI switch circuit

(c) BAR-state transmission $[I_1 \rightarrow O_1]$

(d) CROSS-state transmission $[I_1 \rightarrow O_2]$

Fig. 2. (a) Optical switching element (OSE) black-box model. (b) MZI switching circuit. (c) and (d) Dependence of transmission properties in BAR and CROSS state on wavelength and temperature.

order to reduce the system margin. In [17] ML module is used in SDN enabled optical network to provide the full abstraction of a PIC. In [18], the authors experimentally demonstrated a complete self-learning and reconfigurable photonic signal processor based on an optical neural network chip. The proposed chip executes a variety of functions by self-learning, such as multi-channel optical switching, optical multiple-input-multiple-output de-scrambling, and tunable optical filtering. We proposed an ML-based model for modeling the elementary control states of the PIC $N\times N$ switches in a structural agnostic way in [19, 20]. Similarly, in [21] ML-based model is used for the accurate prediction of QoT impairments of photonic switches in a SDN context. In [22], the deep reinforcement learning (DRL) technique is used to reconfigure the silicon photonic flexible low-latency interconnect optical network switch (Flex-LIONS) giving to the traffic attributes in high-performance computing systems. Additionally, a novel reinforcement ML-based framework called DeepConf is presented in [23], for automatically learning and implementing a range of data center networking techniques.

3. Ultra-wideband switching system

The device under analysis consists of an electronically controlled integrated transparent photonic switch, able to perform the routing operation without electro-optical conversion of the transmitted signals. The two main characteristics of the system are related to the frequency range of operation, allowing switching in the spectral range covering S+C+L bands, as well as the logical routing requirement, as every permutation of the input signal must be achievable at the egress stage of the device, referred as non-blocking switching.

Different solutions have been described in the literature, with $N\times N$ multistage switching networks being one of the most widespread implementations. In this class of devices, the routing operation is achieved by cascading various stages of elementary 2×2 switches, referred to as optical switching elements (OSE), arranged in different topologies depending on the required properties of the routing operation. Each OSE of the network can be controlled independently through an electrical signal. In this work, we apply the proposed approach to an 8×8 switching device, with MZI-based OSEs, analyzed in Section 3.1, interconnected through the Beneš network topology, described in Section 3.2. The switching network size $N = 8$ has been chosen as a trade-off between realistic implementation sizes for photonic integrated circuits, circuit complexity and data-set size. The chosen size acts as a reasonable simulation test-bed to verify the proposed control scheme and abstraction, while providing a large enough component cascade to highlight the physical devices behavior.

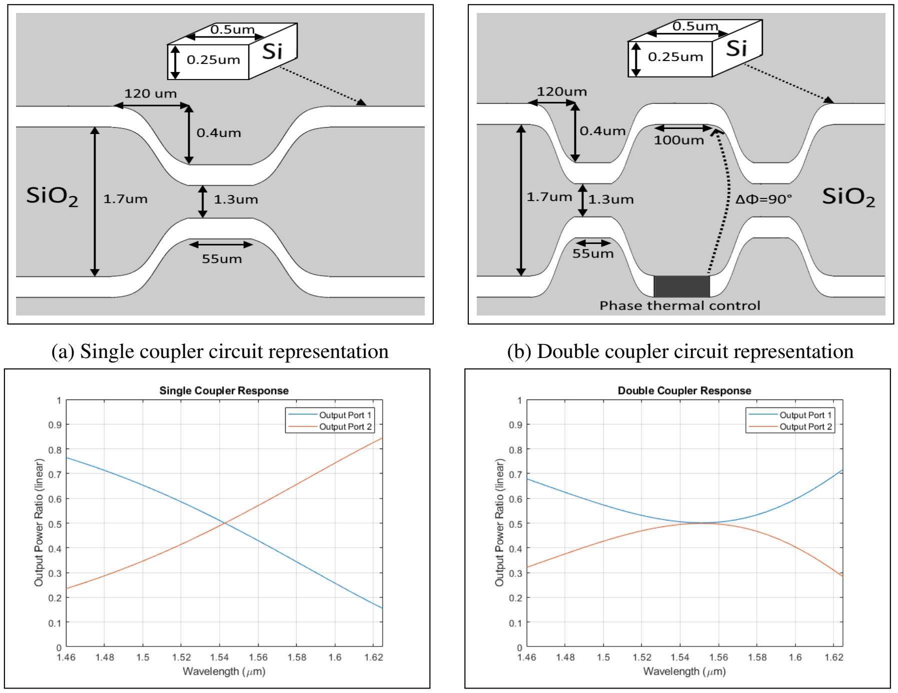
3.1. Optical switching element

The OSE is the fundamental block required for the switching action, introducing limitations on the operating frequencies and imposing some QoT degradation. At the logical level, the OSE 2×2 cross-bar switch can be modelled as a black-box (see Fig. 2a) with two available routing states: the BAR configuration ($[I_1, I_2] \rightarrow [O_1, O_2]$) and the CROSS configuration ($[I_1, I_2] \rightarrow [O_2, O_1]$) which can be appropriately toggled by a given binary control signal. The OSE can be implemented with two main solutions described in the literature: the MRR filter and the MZI. Due to the bandwidth limitations of the MRR solutions, we propose in this paper a device based on the MZI principle.

The most straightforward MZI device is structured as shown in Fig. 2b: the signal is divided into the two waveguides by the first 3 dB coupling section and recombined in the egress 3 dB coupler, with the thermally-controlled phase shift region acting as the control section for the routing state. The routing state is controlled electrically by increasing the temperature of the phase control waveguide in the MZI arms. This increase in temperature introduces a phase shift in the propagating signal, changing the output recombination waveguide in the egress coupler. The signal transmission is depicted in Fig. 2(c) and Fig. 2(d) as a function of the signal wavelength as well as the temperature shift between the MZI arms. In the OFF state ($\Delta T = 0\,°\mathrm{C}$) the bandwidth limitation of the device is clear, with the range of operation covering roughly half of the S+C+L band. The bandwidth limitation is due to the 3 dB coupling regions where phase velocity dispersion of the physical waveguides causes asymmetry in the signal propagation, with uneven power splitting and recombination, leading to significant crosstalk with the incorrect output port.

3.1.1. Higher order coupling regions

The critical component for achieving the UWB range of operation is the coupler region, required before and after the thermal phase control section. While the coupler has a 3 dB power ratio for the centre design frequency, the waveguide dispersion causes increasing asymmetry as the signal frequency moves away from the center point. One of the simplest solutions to compensate for this effect is cascading two identical couplers while introducing a constant phase shift between the two waveguides ($\Delta\phi = 90°$), as shown in Fig. 3. This solution reduces the dispersion effect on the power ratio, leading to a larger and flatter bandwidth near the design frequency while also reducing the overall asymmetry at the limits of the chosen bandwidth, depicted in Fig. 3d. More advanced solutions, like a complex waveguide, tapered structures or advanced 3D structure [24] can still enlarge the bandwidth of the 3 dB coupling region. For the intended applications of a multi-stage switching structure, the rapid increase in circuit complexity and production requirements may become prohibitive as the scale of the overall network increases, which leads to a trade-off between the cost and effectiveness of the solution. The analyzed device, implemented through the second-order coupling structure, is depicted in Fig. 4a: the bandwidth of operation covers the target transmission windows, with increases in crosstalk and penalty observed only at the edges of the operating region, as shown for both routing states in Fig. 4b-Fig. 4c.



(a) Single coupler circuit representation

(b) Double coupler circuit representation

(c) Single coupler frequency response

(d) Double coupler frequency response

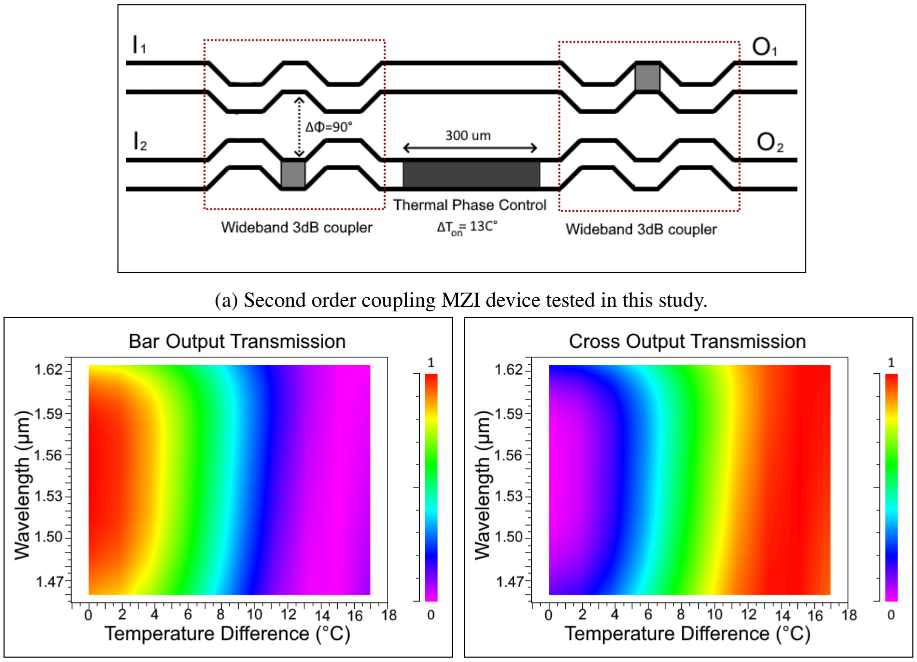
Fig. 3. (a)-(b) MZI switch structures: single coupler vs. double coupler. (c)-(d) Corresponding transmission bandwidth properties.

3.2. Beneš topology

After defining the fundamental 2×2 OSE, any generic $N \times N$ circuit can be modeled following the topology of choice, for example, the Beneš network. The Beneš network has been chosen here for various reasons, due to both the target application requirements and minimization of the circuit footprint. Different classes of switching networks exist: topologies based on the Clos network paradigm, like the Beneš structure, allow both a reduction of the number of switching elements, as well as guaranteeing non-blocking capabilities, avoiding routing conflict inside the device mesh [25]. The non-blocking property is fundamental for our target application, as all possible permutations of the N input signals must be achievable at the egress stage of the network, allowing full control of the routing component for any given output request. The $N \times N$ Beneš device is characterized by a recursive structure (see Fig. 5a), described in details in [26], with a number of OSE equal to $N_{\text{sw}} = N \times \log_2 N - \frac{N}{2}$ distributed in $2\log_2 N - 1$ stages, as depicted for an 8×8 Beneš in Fig. 5b. One important characteristic of this device is the solution multiplicity for any given target output state: given that the N_{sw} OSEs lead to several control configurations $N_{\text{conf}} = 2^{N_{\text{sw}}}$, larger than the number of unique output permutations $N_{\text{out}} = N!$, multiple solutions must exist for each unique output permutation request, whose multiplicity depends on the specific target output permutation. In Section 5 we propose a ML-based approach to determine the *best* configuration between the nominally identical ones returned by the proposed routing algorithm.

4. Simulation environment and dataset generation

The device is modeled under two different levels of abstraction to characterize the dependence on the control signals of both the routing behavior and the impact on QoT of the switching operation.

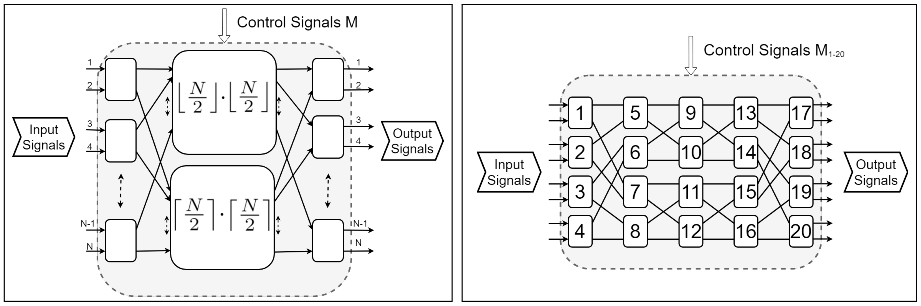


(a) Second order coupling MZI device tested in this study.

(b) BAR-state transmission [$I_1 \rightarrow O_1$]

(c) CROSS-state transmission [$I_1 \rightarrow O_2$]

Fig. 4. Transmission dependence on wavelength and thermal tunability for the second order coupling MZI device tested in the paper.



(a) Generic Beneš network recursive structure

(b) 8×8 Beneš network

Fig. 5. Beneš network topologies.

4.1. Routing model

Given the black-box abstraction of the 2×2 cross-bar OSEs, the routing problem can be solved on a simplified version of the circuit, taking only into account the logical link between input-output ports as a function of the binary control state of each fundamental switch. To this end, a virtual topological structure was generated in MATLAB®, in order to analyze the routing and then to evaluate the logical output for the QoT transmission-level simulation. Given the simple recursive structure of the network, coupled with the non-polynomial increase in the solution space ($N_{\text{conf}} = O(2^{N \log N})$, $N_{\text{out}} = O(N!)$), brute-force solution together with look-up tables are not a scalable method to obtain the states configurations for the target output request. This introduces the need for a scalable deterministic algorithm to tackle the problem complexity and provide the equivalent paths routing the same output permutation. While it is fundamental to be

Algorithm 1 Beneš routing algorithm

```
Require: Number of input and output channels $\mathcal{N}$, Input signals labels $\mathcal{I}_{[1:\mathcal{N}]}$, Output signals labels $\mathcal{O}_{[1:\mathcal{N}]}$
Ensure: A control state, chosen randomly among all the control states giving the requested target output permutation
Number of switch per stage $\mathcal{N}_{\text{sw/st}} = \frac{N}{2}$
Number of stages $\mathcal{N}_{st} = \log_2 N$
for layer index $l = 1$ to $\frac{\mathcal{N}_{st}}{2} - 1$ do
    Initialize to -1 the routing matrix $M_l \in \mathcal{R}^{\frac{N}{2} \times \frac{N}{2}}$
    Initialize to 0 the label matrix $T_l \in \mathcal{R}^{\frac{N}{2} \times \frac{N}{2}}$
    for label index $j = 1$ to $\mathcal{N}$ do
        $SW_I \leftarrow \left\lceil \frac{j}{2} \right\rceil$
        $SW_O \leftarrow \left\lceil \frac{k}{2} \right\rceil$ with $k$ such that $O_k = \mathcal{I}_j$
        if $M_l(SW_I, SW_O)$ is -1 then
            if column $SW_O$ of $M_l$ contains a 0 then
                $M_l(SW_I, SW_O) \leftarrow 1$   %%% 1 i.e. bottom network routing
            else if column $SW_O$ of $M_l$ contains a 1 then
                $M_l(SW_I, SW_O) \leftarrow 0$   %%% 0 i.e. top network routing
            else
                set randomly $M_l(SW_I, SW_O)$ to 0 or 1
            end if
            $T_l(SW_I, SW_O) \leftarrow \mathcal{I}_j$
        else if $M_l(SW_I, SW_O)$ is not -1 then
            $M_l(SW_I, SW_O) \leftarrow 2$   %%% 2 i.e. same input-output couple request
            $T_l(SW_I, 1) \leftarrow T_l(SW_I, SW_O)$
            $T_l(SW_I, 2) \leftarrow \mathcal{I}_j$
        end if
    end for
    check for conflict → Algorithm 2
    update $\mathcal{I}_j$, $O_j$ for the evaluated routing
end for
```

able to generate a single routing solution for a target output permutation, in order to minimize the penalties in a device-agnostic scenario a more general algorithm is needed to evaluate all equivalent routing solutions for each required signal output permutation. The device-agnostic scenario is introduced to generalize the analysis without the need of assuming the QoT behavior to the physical and device-level structure: a simpler approach to the optimization of the QoT could be to minimize the number of interconnecting crossings encountered by each signal, as these elements are typically the leading cause of signal attenuation. However, this relies on a device assumption which could not always be accurate, so to avoid the issue the problem is split into two main sections, under a "divide and rule" paradigm: the routing model is tasked with generating all equivalent routings for the target signal output, without introducing assumptions on the underlying transmission penalty, while the ML agent proposed in the later sections handles the QoT optimizations, selecting between the solution space the best-predicted solution.

The proposed solution represents a generalization of the matrix-based algorithm described in [27]. Having defined an $N \times N$ Beneš, with number of switches per stage $N_{\text{sw/st}} = \frac{N}{2}$ and number of stages $N_{\text{st}} = \log_2 N$, the proposed algorithm is divided in the following steps (Algorithm 1-Algorithm 2):

- For each layer of the network up to the half-point stage, generate two empty matrices $\mathcal{M}, \mathcal{T} \in \mathcal{R}^{\frac{N}{2} \times \frac{N}{2}}$, representing respectively the control states of the OSEs in the layer and the rearranged signal order after the layer.

- By comparing the input signals order of the ingress layer with respect to the output signals order of the egress layer, for every signal map, the relation between input switch and target output switch. The ingress and egress layers are symmetrical with respect to the middle stage $N_{\text{middle}} = \frac{N_{\text{st}}}{2}$ (ingress: layer (i), egress: layer $(N_{\text{st}} - i)$, for $i \in [1 : N_{\text{middle}}]$)

Algorithm 2 Routing conflict algorithm

```
**Require:** *Routing matrix M*
**Ensure:** *Balanced conflict-avoiding routing*
 1: conflict=**false**
 2: Conflicts=empty list
 3: **for** row index $r = 1$ **to** $N$ **do**
 4:     $R$=row $r$ of $M$
 5:     **if** $R$ contains two 0s or two 1s **then**
 6:         conflict=**true**
 7:         append $r$ to Conflicts
 8:     **end if**
 9: **end for**
10: **if** conflict **then**
11:     $r^*$= randomly selected element **in** Conflicts
12:     $R^*$=row $r^*$ of $M$
13:     $c^*$ = randomly selected position of any element of $R^*$ equal to 0 or 1
14:     $C^*$=column $c^*$ of $M$
15:     exchange 0 and 1 in $C^*$
16:     **loop** to 1
17: **else**
18:     **return** → **Algorithm 1** (update matrix M)
19: **end if**
```

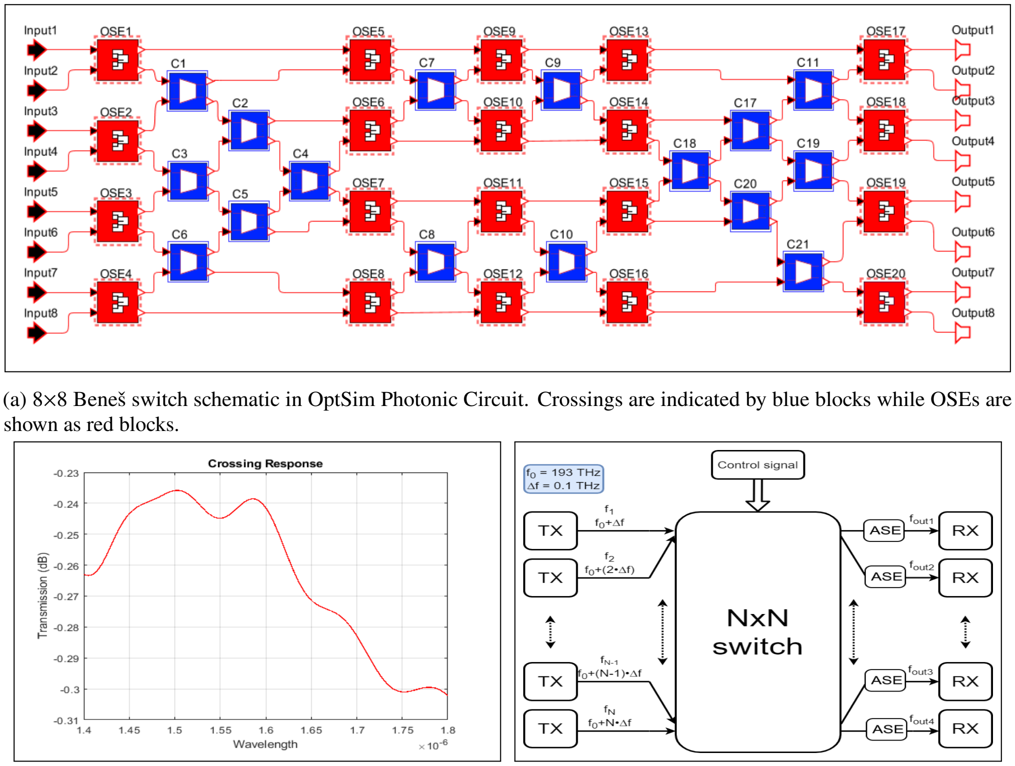
- Fill the matrix $\mathcal{M}$ with $[0, 1]$ using the input-output switch relationship to select the row-column pair respectively. The matrix $\mathcal{T}$ contains the label of the signal corresponding to that input-output switch pair.

- Once the matrix for the layer is compiled, verify that no repetitions occur both row-wise and column-wise. Only one instance of "0" and "1" can occur in any given row or column. If repetitions occur, flip the element column-wise until the conditions are solved.

- Iterate for all layers $i \in [1 : N_{\text{middle}}]$).

In the described algorithm the "0" and "1" flags of the $\mathcal{M}$ matrices correspond to the propagation direction of the signal in each switching element, relative to the following stages: considering the recursive structure of the Beneš topology, as well as its symmetry, at every stage, two equivalent paths can be found in the respective top and bottom following sub-network. Two additional flags values are used in the proposed algorithm: every matrix cell is initially set to "-1" to indicate non-allocated requests or empty cells. An additional flag is required in the routing matrix in order to account for equivalent routings in some specific cases: while typically the input signals of an ingress switch must be routed to different egress switches when both input signals are targeting the same output switch, only one single cell of the routing matrix can be targeted: to this end, the flag "2" represents the path equivalence between the top and bottom network, with the implied value of both ("1","0") and ("0","1").

Once the procedure is completed, the state of the switches can be obtained by comparing the order of the signals of each layer, taking into account the interconnects and the top/down direction provided by the compiled $\mathcal{M}_i$ matrices. With a slight modification to the presented algorithm, the evaluation of all equivalent paths in terms of permutation of the output signals becomes trivial: once the output permutation is set, each valid matrix $\mathcal{M}_i$ represents a different equivalent routing possibility. For every routing of the previous layer, the process is iterated, generating a recursive exploration of all switching states for the required output. Using the proposed algorithm, the control unit can generate different solutions depending on the required task: if all equivalent routing solutions are evaluated, the proposed ML agent can optimize the QoT, finally choosing the path with minimum transmission penalty. Suppose a simpler control unit is required; the algorithm can provide a single control configuration for the device, generating



(a) 8×8 Beneš switch schematic in OptSim Photonic Circuit. Crossings are indicated by blue blocks while OSEs are shown as red blocks.

(b) Response of the waveguide crossing

(c) Transmission environment under simulation

Fig. 6. Simulation model schematic and characteristics

one random routing compatible with the required signal output permutation without exploring all equivalent paths.

4.2. Transmission model

To evaluate the impact of the switching fabric on the QoT, numerical simulations have been first carried out in the Synopsys OptSim$^{\mathrm{TM}}$ Photonic Circuit simulation environment [28], testing an 8×8 Beneš switch base on an OSE implemented with the second order coupling MZI previously described. Due to the relative low-loss flat-band behavior of the OSE, the critical components in the device, especially concerning routing optimization, are the waveguide crossings, which introduce path-dependent losses and attenuation in the propagating signals. It must also be remarked that for strict-sense Beneš structures ($N = 2^x$, $x \in \mathcal{N}$) the number of switches encountered by each signal is equal, independently from the OSE control signals, as shown in Fig. 6a, highlighting the critical task in characterizing the control states dependent QoT impairments due to the stages interconnects.

The designed waveguide crossing introduces an average 0.2 dB–0.3 dB loss for each instance, with a small spectral variance, as depicted in Fig. 6b. While the crossings have been accounted for the penalty evaluation, the interconnect waveguides and bent sections have not, due to their generally negligible effect in a properly designed layout. The general schematic of the simulated setup is depicted in Fig. 6c. We assumed eight input signals spaced $\Delta f = 100$ GHz with a central frequency of $f_c = 193$ THz. The simulated signals consisted of PM-16-QAM modulated streams at $B_r = 60$ GBaud, which are then demodulated at the receiver side, extracting the Bit-Error Rate (BER) as a function of the OSNR. These measurements are then expressed as QoT Penalty

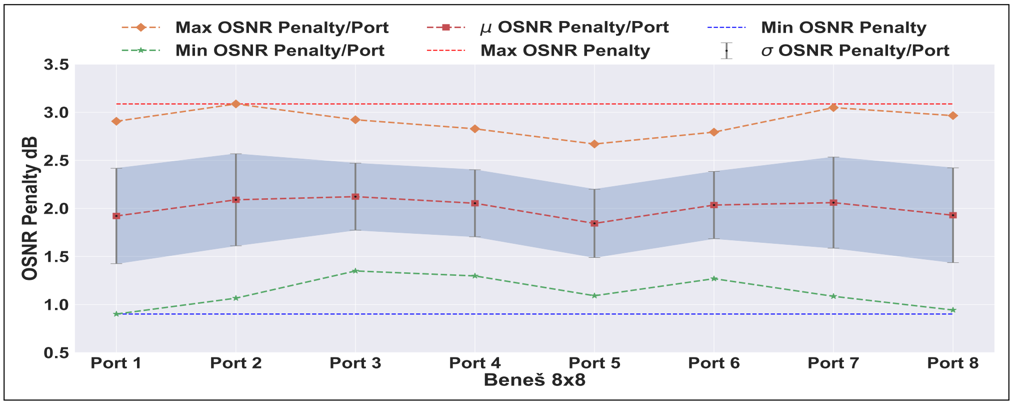


Fig. 7. Statistical analysis of OSNR Penalties for each output port.

(in decibel), comparing to the trend of the back-to-back TX/RX system evaluated without the switching fabric. Due to the previously discussed non-polynomial increase of the solution space, the characterization of the full system through a look-up table solution is not feasible, especially at the transmission level, due to the high computational costs of such simulations. In order to train the proposed ML algorithm, it is necessary to build a dataset of simulated configurations, measuring the QoT Penalty for a random sub-set of control signals.

The simulation dataset has been generated for $N_{\mathrm{sim}} = 5000$ random control configurations, allowing equivalent paths (output permutation) but enforcing individual control states to avoid erroneous training by repeating the same OSE states. The general distribution of the OSNR Penalties for the simulated dataset is shown in Fig. 7: as expected, the distribution has a relatively uniform average value of $\mu = 2\,\mathrm{dB}$ for every output port, with a comparable standard deviation. To characterize the device in SDN controlled environment, it is important to highlight the maximum value of the penalty: $\Delta\mathrm{OSNR}_{\mathrm{max}} \approx 3.1\,\mathrm{dB}$. Without a control unit capable of a reliable prediction of the expected penalty in real-time, the impact of switching on QoT must always be over-estimated to this maximum value, which represents an infrequent worst-case assumption. To this end, the ML agent allows more flexible control of the device, highlighting the cases where a higher transmission rate can be applied due to a lower penalty. Furthermore, in Fig. 7 every data-point corresponds to one of the different equivalent solutions, highlighting the average penalty for all the output ports, as well as the minimum and maximum values. It is clear how a real-time control strategy can be employed to optimize the performance of such a device. At the same time, the average port penalties are identical between configurations; the lower variance solutions offer a better alternative, as the QoT is more uniform between all the output ports of the configuration.

5. Machine learning modeling for QoT impairment

This section illustrates the details of the proposed ML framework and explains the complete workflow of training and testing phases. It also describes the architecture of the main cognitive engine of the proposed ML module, along with the definition of features, labels, and additional tuning and control parameters. The final ML module will be integrated as an application program interface (API) inside the controller.

The proposed supervised ML-based framework works in a complete black-box manner, requiring only substantial training data to develop a cognitive model without considering the photonic circuit internal topology. Like all other supervised ML techniques, to complete the training and prediction procedures, the proposed model requires defining the features and labels

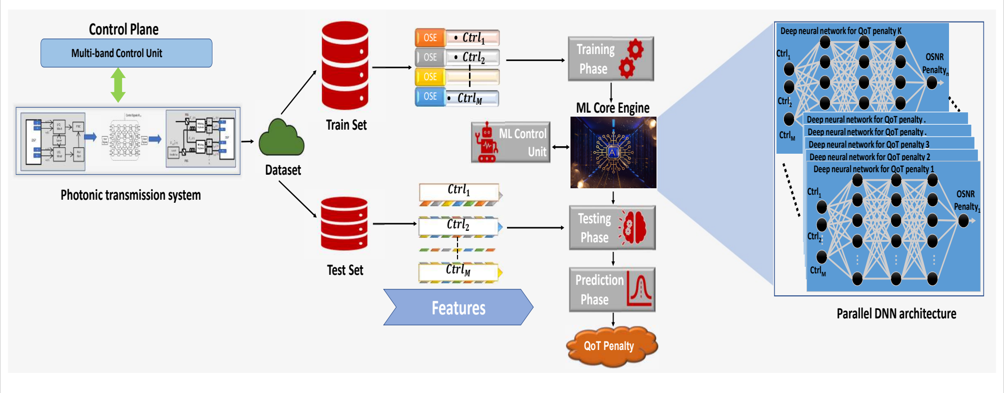

Control Plane
Multi-band Control Unit
Photonic transmission system
Dataset
Train Set
Test Set
OSE
• $Ctrl_1$
• $Ctrl_2$
• $Ctrl_M$
$Ctrl_1$
$Ctrl_2$
$Ctrl_M$
Features
Training Phase
ML Core Engine
ML Control Unit
Testing Phase
Prediction Phase
QoT Penalty
Deep neural network for QoT penalty K
Deep neural network for QoT penalty .
Deep neural network for QoT penalty .
Deep neural network for QoT penalty 3
Deep neural network for QoT penalty 2
Deep neural network for QoT penalty 1
$Ctrl_1$
$Ctrl_2$
$Ctrl_M$
OSNR $Penalty_n$
OSNR $Penalty_1$
Parallel DNN architecture

Fig. 8. Schematic of the ML module with Parallel DNN architecture

that represent the system inputs and outputs, respectively. The manipulated features include the different permutations of the OSE control signals (Ctrl_1, Ctrl_2, Ctrl_3, $\cdots$, Ctrl_M) at the control ports of the photonic switch and utilize QoT Penalty of the k-th output port of the considered photonic switch as labels shown in Fig. 8.

A deep neural network (DNN) [29] is considered to develop the cognition in the ML engine as it is the potential tool that is frequently used in different applications in various fields. The proposed DNN is built by using a higher-level API of the open-source TensorFlow© library [30], which offers a variety of learning algorithms along with data processing functions to improve the quality of the generated dataset. The core engine of DNN is configured by various optimized hyper-parameters such as the *training steps*, set to 1000; the optimizer is loaded with the *adaptive gradient algorithm (ADAGRAD)* Keras optimizer, with a default *learning rate* of 10^{-2} and L_1 regularization is set to 10^{-3} to acquire the computational advantage by avoiding the features with zero coefficients [31]. Additionally, numerous non-linear activation functions such as *Relu, tanh, sigmoid.* have been tested during the model build-up. Later, *Relu* has been selected to feed in DNN as it outperforms the others in terms of prediction and computational load [32].

Furthermore, another essential DNN hyper-parameter is the number of hidden-layers. The proposed model core engine has been tuned on considerable numbers of hidden-layers and neurons to reach the best trade-off between accuracy and computational time. Even though an increase in the number of layers and neurons enhances the accuracy of the DNN up to a certain level, a further increase in the values of these parameters introduces diminishing returns that cause over-fitting of the model and, at the same time, increases the computational time. After this complex trade-off assessment, we decided upon a DNN with three hidden-layers with ten artificial neurons for each hidden layer optimized for the considered dimension N. To enhance prediction performance, we propose to use a parallel architecture for the DNN as shown in Fig. 8. In reality, we have an autonomous DNN to predict the QoT Penalty against each k-th output port of the considered $N{\times}N$ photonic switch. The parallel architecture of DNN better exploits the augmented information in the provided dataset for each output port, which gives better cognition to the core DNN engine and consequently achieves high efficiency in terms of prediction. Initially, the core DNN engine training is performed; after that, the trained model is tested on a separate subset of the dataset: the conventional rule of thumb 70 % and 30 % have been opted to split the generated dataset. The train set is 70 % while the test set is 30 % of the total generated dataset reported subsection 4.2. Each of the individual DNN modules in the parallel architecture is provided with the same set of features (Ctrl_1, Ctrl_2, Ctrl_3, $\cdots$, Ctrl_M, i.e., OSE control signals) as an input during training and retrieves OSNR $\mathrm{Penalty}_{i,k}$ for each port k

of the proposed UWB Beneš switch as an output label. In order to prevent over-fitting of the DNN, the *training step* is considered as the stopping factor while the *mean square error* (MSE) is applied as a loss function, given by

$$\text{QoT MSE} = \frac{1}{n}\sum_{i=0}^{n}\left(\frac{1}{N}\sum_{k=1}^{N}\left(\text{OSNR Penalty}_{i,k}^{p} - \text{OSNR Penalty}_{i,k}^{a}\right)^{2}\right) \tag{1}$$

where n is the number of test realizations, N is the total number of input/output ports of the specific $N{\times}N$ switching system and OSNR $\text{Penalty}_{i,k}^{p}$ – OSNR $\text{Penalty}_{i,k}^{a}$ are the predicted and actual OSNR Penalties of the k-th output port of the considered topology.

6. Results and discussion

This section demonstrates the accuracy of proposed ML modules in delivering QoT impairments predictions for UWB photonics switching architectures. The ML module exploits the deterministic switch control states to obtain the QoT impairments in terms of OSNR $\text{Penalty}_{i,k}$ for each port k of the proposed UWB Beneš switch. In addition to this, a complete case study is also analyzed to reveal the effectiveness of the proposed ML-based QoT Penalty estimation model for the photonic switching system.

The proposed ML cognitive engine manipulates the deterministic control states as input and exploits the QoT Penalty as an output. The metric utilized to assess the accuracy of the ML model is defined as:

$$\Delta\text{OSNR}_{i,k} = \text{OSNR Penalty}_{i,k}^{a} - \text{OSNR Penalty}_{i,k}^{p} \tag{2}$$

where the parameters reported in Eq. 2 have the same meaning as in Eq. 1. The reliability of the proposed ML-based QoT model is verified by analyzing its performance at each port of the proposed 8×8 Beneš switch. The distribution of ΔOSNRs of all the ports of the 8×8 Beneš are shown in Fig. 9, along with their mean (μ) and standard deviation (σ) statistics.

In Fig. 9, all the distributions of ΔOSNRs are split up into two slices by the red dotted line ($\Delta\text{OSNR} = 0$). The portion area where $\Delta\text{OSNRs} \leq 0$ is not critical as $\text{OSNR Penalty}_{i,k}^{a} \leq \text{OSNR Penalty}_{i,k}^{p}$ so, in this case we only waste some capacity but the system will never turn into out-of-service. In contrast the section where $\Delta\text{OSNRs} > 0$ is the critical one as $\text{OSNR Penalty}_{i,k}^{a} > \text{OSNR Penalty}_{i,k}^{p}$. In this case, it is required to deploy some margin on top of the ML prediction to keep the system working all the time. The maximum required margins (δ_k) for this case where $\Delta\text{OSNRs} > 0$ are shown as a green line for each port k of the 8×8 Beneš.

Inspecting the required margin, we observe the high level of accuracy achieved operating ML model for QoT impairments estimation. The proposed 8×8 Beneš, the worst-case prediction performance is observed on port 5; the δ_5 is less than 0.12 dB. With the availability of such accurate prediction, we can envision that in practical applications, the OSNR Penalty margin on top of the ML prediction can be reduced to 0.12 dB for Beneš 8x8. Furthermore, the prediction asymmetry between the different port of the device, is due to the intrinsic randomness and limited size of the provided data-set, leading to better training for the prediction of certain paths. Under the envisioned case-study, a drastically smaller data-set has been provided by choice to the ML agent with respect to the complete device configuration set. Even under this limited training scenario, the asymmetry between the port predictions is still marginal with respect to the QoT optimization available through this method deployment.

The effectiveness of the proposed ML-based QoT impairments estimation model is further demonstrated by considering the optimality routing issue: the ML agent can be used to optimize the routing solution in conjunction with the previously described routing algorithm. Taking as an example a target output request such as [1, 2, 3, 4, 5, 6, 7, 8] → [7, 6, 3, 8, 5, 4, 1, 2], we observe that 32 different combinations of the control states exist leading to the desired output

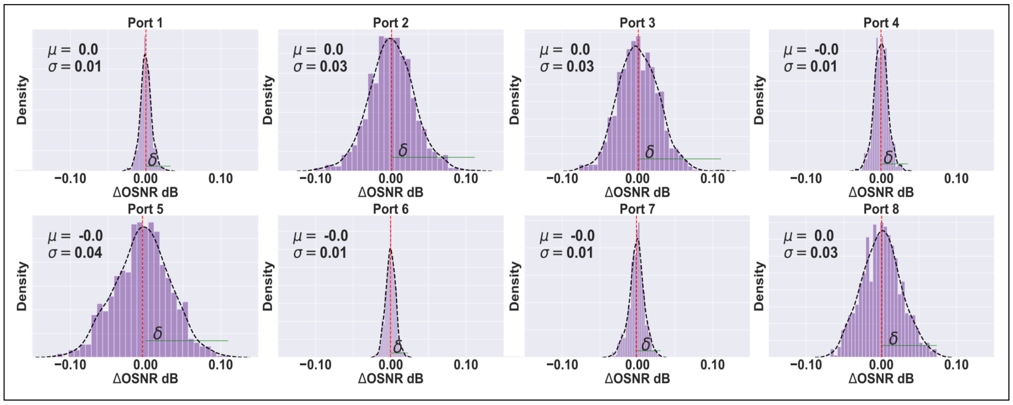


Fig. 9. Probability density functions of ΔOSNR for each port of the 8x8 Beneš switch. Average values μ and variances σ indicated for the individual cases are expressed in decibel.

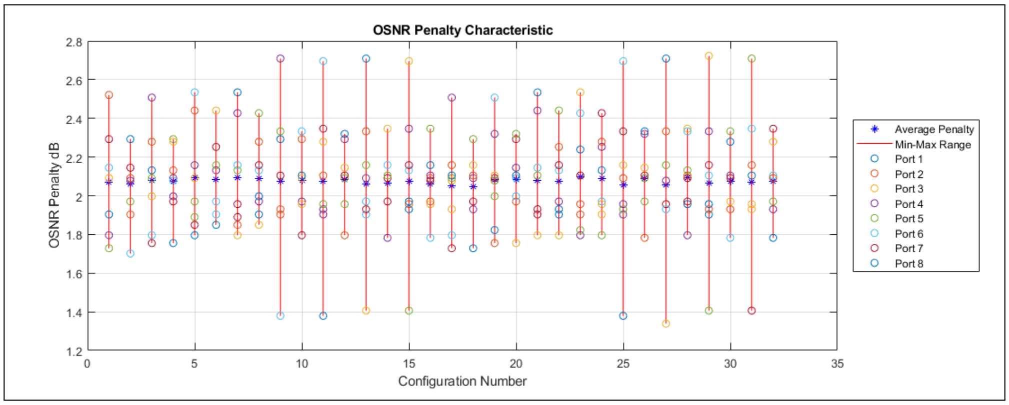


Fig. 10. OSNR Penalty distribution for 32 nominally equivalent control states generating the output pattern [7, 6, 3, 8, 5, 4, 1, 2]. A label from 1 to 32 has been assigned to each control state according to the order it is generated by the proposed algotithm.

pattern. The designed routing algorithm is able to evaluate all these nominally equivalent routing solutions, which have been tested in order to characterize their penalty and statistical distribution, as shown in Fig. 10. The average penalty for every equivalent configuration is reasonably similar, while the main difference is found between the standard deviation between the penalty of each port. The ML agent could provide real-time control optimization for this application, minimizing the overall penalty and avoiding high deviation solutions. This target goal allows for a similar penalty factor between all the output signals, minimizing the overall deviation, although different criteria could provide alternative solutions depending on the overall control goal. The choice of the *best* control state depends on the selected metric: considering the results introduced in Fig. 10, configuration number 18 provides the minimal deviation between the alternative routings, while solution number 27 could be selected if only the minimum penalty is considered as the optical metric.

7. Conclusions

Optical network elements currently exploit PICs to carry out most of the complex functions at the photonic level; specifically, optical networks and data centers progressively utilize large-scale photonic switches and wavelength selective switches due to their wide-band abilities together with low latency and low power consumption. This increased use of photonic switching systems creates a massive demand for a generic management model that works in an entirely topological and technological agnostic way.

This work introduced the concept of a softwarized and autonomous management of PIC-based UWB optical switches for software-defined open optical networks. The proposed method can model any $N\times N$ UWB switching system at two different levels of abstraction: the routing and the QoT levels related to the applied control signals. The routing level problem is solved by considering the black-box abstraction of the 2×2 cross-bar switching units. At the same time, for the QoT, an ML-based framework is proposed to predict the QoT degradation due to the switching element. The proposed model works in a topological and technological agnostic *blind* way, exploiting neural network to model the QoT impairments of any $N\times N$ UWB photonic switch.

The operated data-driven technique is easily scalable to larger input dimensions N as a high level of accuracy can be achieved with limited-size datasets. Besides this, the proposed two-level abstraction scheme can be further expanded to evaluate the performance of any $N\times N$ optical switch on the network layer metrics. Furthermore, the model achieved promising results in predicting QoT degradation; the error in predicting QoT degradation is less than 0.12 dB. With the availability of such accurate prediction, we can envision that in practical applications, the required QoT margin on top of the ML prediction can be reduced to 0.12 dB for the considered Beneš 8x8 architecture.

Funding. This work was supported by the Synopsys within the activities of a research MSA with Politecnico di Torino and by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement 814276.

Disclosures. The authors declare no conflicts of interest.

Data Availability Statement. Data underlying the results presented in this paper are not publicly available at this time but may be obtained from the authors upon reasonable request.

References

1. A. Ferrari, A. Napoli, J. K. Fischer, N. Costa, A. D'Amico, J. Pedro, W. Forysiak, E. Pincemin, A. Lord, A. Stavdas, J. P. F.-P. Gimenez, G. Roelkens, N. Calabretta, S. Abrate, B. Sommerkorn-Krombholz, and V. Curri, "Assessment on the achievable throughput of multi-band ITU-T G.652.D fiber transmission systems," J. Light. Technol. **38**, 4279–4291 (2020).
2. C.-S. Li and W. Liao, "Software defined networks," IEEE Commun. Mag **51**, 113–113 (2013).
3. M. Jinno, T. Ohara, Y. Sone, A. Hirano, O. Ishida, and M. Tomizawa, "Elastic and adaptive optical networks: possible adoption scenarios and future standardization aspects," IEEE Commun. Mag. **49**, 164–172 (2011).
4. V. Curri, A. Carena, A. Arduino, G. Bosco, P. Poggiolini, A. Nespola, and F. Forghieri, "Design strategies and merit of system parameters for uniform uncompensated links supporting Nyquist-WDM transmission," J. Light. Technol. **33**, 3921–3932 (2015).
5. V. Curri, "Software-defined wdm optical transport in disaggregated open optical networks," in *2020 22nd International Conference on Transparent Optical Networks (ICTON),* (2020), pp. 1–4.
6. K. Suzuki, R. Konoike, J. Hasegawa, S. Suda, H. Matsuura, K. Ikeda, S. Namiki, and H. Kawashima, "Low-insertion-loss and power-efficient 32 × 32 silicon photonics switch with extremely high-δ silica plc connector," J. Light. Technol. **37**, 116–122 (2019).
7. Q. Cheng, L. Y. Dai, N. C. Abrams, Y.-H. Hung, P. E. Morrissey, M. Glick, P. O'Brien, and K. Bergman, "Ultralow-crosstalk, strictly non-blocking microring-based optical switch," Photon. Res. **7**, 155–161 (2019).
8. J. Kim, C. Nuzman, B. Kumar, D. Lieuwen, J. Kraus, A. Weiss, C. Lichtenwalner, A. Papazian, R. Frahm, N. Basavanhally, D. Ramsey, V. Aksyuk, F. Pardo, M. Simon, V. Lifton, H. Chan, M. Haueis, A. Gasparyan, H. Shea, S. Arney, C. Bolle, P. Kolodner, R. Ryf, D. Neilson, and J. Gates, "1100 x 1100 port MEMS-based optical crossconnect with 4-dB maximum loss," IEEE Photonics Technol. Lett. **15**, 1537–1539 (2003).
9. A. N. Dames, "Beam steering optical switch," (2008). US Patent 7,389,016.

10. Y. Huang, Q. Cheng, Y.-H. Hung, H. Guan, X. Meng, A. Novack, M. Streshinsky, M. Hochberg, and K. Bergman, “Multi-stage 8 × 8 silicon photonic switch based on dual-microring switching elements,” J. Light. Technol. **38**, 194–201 (2020).
11. D. Opferman and N. Tsao-Wu, “On a class of rearrangeable switching networks part I: Control algorithm,” The Bell Syst. Tech. J. **50**, 1579–1600 (1971).
12. M. Ding, Q. Cheng, A. Wonfor, R. V. Penty, and I. H. White, “Routing algorithm to optimize loss and IPDR for rearrangeably non-blocking integrated optical switches,” in *2015 Conference on Lasers and Electro-Optics (CLEO),* (2015), pp. 1–2.
13. Y. Qian, H. Mehrvar, H. Ma, X. Yang, K. Zhu, H. Fu, D. Geng, D. Goodwill, P. Dumais, and E. Bernier, “Crosstalk optimization in low extinction-ratio switch fabrics,” in *2014 Optical Fiber Communication (OFC),* (2014), pp. 1–3.
14. Q. Cheng, Y. Huang, H. Yang, M. Bahadori, N. Abrams, X. Meng, M. Glick, Y. Liu, M. Hochberg, and K. Bergman, “Silicon photonic switch topologies and routing strategies for disaggregated data centers,” IEEE J. Sel. Top. Quantum Electron. **26**, 1–10 (2020).
15. W. Gao, L. Lu, L. Zhou, and J. Chen, “Automatic calibration of silicon ring-based optical switch powered by machine learning,” Opt. Express **28**, 10438–10455 (2020).
16. I. Khan, M. Chalony, E. Ghillino, M. U. Masood, J. Patel, D. Richards, P. Mena, P. Bardella, A. Carena, and V. Curri, “Effectiveness of machine learning in assessing QoT impairments of photonics integrated circuits to reduce system margin,” in *2020 IEEE Photonics Conference (IPC),* (2020), pp. 1–2.
17. I. Khan, M. Chalony, E. Ghillino, M. U. Masood, J. Patel, D. Richards, P. Mena, P. Bardella, A. Carena, and V. Curri, “Machine learning assisted abstraction of photonic integrated circuits in fully disaggregated transparent optical networks,” in *2020 22nd International Conference on Transparent Optical Networks (ICTON),* (2020), pp. 1–4.
18. H. Zhou, Y. Zhao, X. Wang, D. Gao, J. Dong, and X. Zhang, “Self-configuring and reconfigurable silicon photonic signal processor,” ACS Photonics **7**, 792–799 (2020).
19. I. Khan, L. Tunesi, M. Chalony, E. Ghillino, M. U. Masood, J. Patel, P. Bardella, A. Carena, and V. Curri, “Machine-learning-aided abstraction of photonic integrated circuits in software-defined optical transport,” in *Next-Generation Optical Communication: Components, Sub-Systems, and Systems X,* vol. 11713 (SPIE, 2021), p. 117130Q.
20. I. Khan, L. Tunesi, M. U. Masood, E. Ghillino, P. Bardella, A. Carena, and V. Curri, “Automatic management of N×N photonic switch powered by machine learning in software-defined optical transport,” IEEE Open J. Commun. Soc. **2**, 1358–1365 (2021).
21. I. Khan, L. Tunesi, M. U. Masood, E. Ghillino, P. Bardella, A. Carena, and V. Curri, “Machine learning assisted model of qot penalties for photonics switching systems,” in *Photonics in Switching and Computing 2021,* (Optical Society of America, 2021), p. M2A.3.
22. R. Proietti, X. Chen, Y. Shang, and S. J. B. Yoo, “Self-driving reconfiguration of data center networks by deep reinforcement learning and silicon photonic Flex-LION switches,” in *2020 IEEE Photonics Conference (IPC),* (2020), pp. 1–2.
23. S. Salman, C. Streiffer, H. Chen, T. Benson, and A. Kadav, “DeepConf: Automating data center network topologies management with machine learning,” in *Proceedings of the 2018 Workshop on Network Meets AI & ML,* (Association for Computing Machinery, New York, NY, USA, 2018), NetAI’18, pp. 8—-14.
24. R. Orta, G. Perrone, R. Tascone, A. Fincato, M. Lenzi, S. Lorenzotti, and P. Nugent, “Design technique for wideband optical couplers,” in *Fiber Optic Network Components,* vol. 2449 (SPIE, 1995), pp. 375 – 383.
25. C. Clos, “A study of non-blocking switching networks,” Bell Syst. Tech. J. **32**, 406–424 (1953).
26. C. Chang and R. Melhem, “Arbitrary size benes networks,” Parallel Process. Lett. **7**, 279–284 (1997).
27. A. Chakrabarty, M. Collier, and S. Mukhopadhyay, “Matrix-based nonblocking routing algorithm for Beneš networks,” in *Future computing 2009,* (IEEE, 2009), pp. 551–556.
28. E. Ghillino, E. Virgillito, P. V. Mena, R. Scarmozzino, R. Stoffer, D. Richards, A. Ghiasi, A. Ferrari, M. Cantono, A. Carena, and V. Curri, “The Synopsys software environment to design and simulate photonic integrated circuits: A case study for 400G transmission,” in *2018 20th International Conference on Transparent Optical Networks (ICTON),* (2018), pp. 1–4.
29. C. Bishop, *Pattern Recognition and Machine Learning: All "just the Facts 101" Material*, Information science and statistics (Springer (India) Private Limited, 2013).
30. https://www.tensorflow.org/.
31. J. Duchi, E. Hazan, and Y. Singer, “Adaptive subgradient methods for online learning and stochastic optimization,” J. Mach. Learn. Res. **12**, 2121–2159 (2011).
32. C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, “Activation functions: Comparison of trends in practice and research for deep learning,” in *2nd International Conference on Computational Sciences and Technology,* (2021), pp. 12–133.