Machine Learning Assisted Management of Photonic Switching Systems

Ihtesham Khan[1], **M Umar Masood**[1], **Lorenzo Tunesi**[1], **Paolo Bardella**[1], **Enrico Ghillino**[2], **Andrea Carena**[1], **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: We propose a machine learning based approach to predict control signals for a photonic switching system. This solution is topology and technology agnostic and it can be employed in real-time switch control planes.

1. Introduction

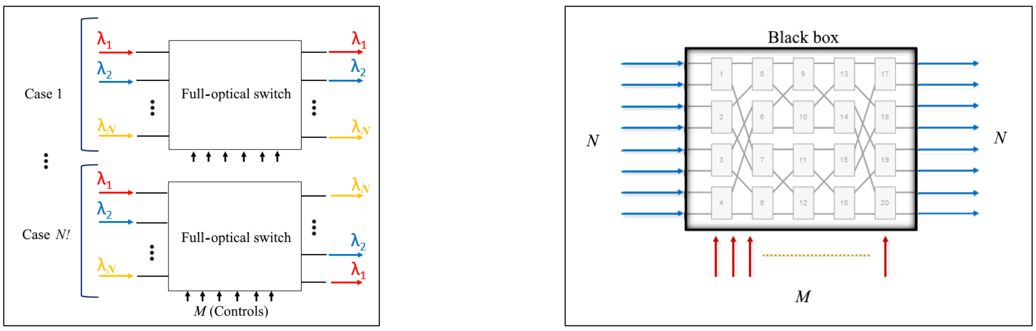
Photonic switching is a fundamental functionality both for core optical networks and data-centers, to allow dynamic provisioning to satisfying connection requests. The key components to enable such operations are the photonic integrated circuits (PICs) enabling spatial and wavelength optical routing. In recent years, silicon photonics emerged as a promising technology for low-cost and energy efficient solutions.

PIC based components internally rely on switches (optical micro-ring resonators or Mach-Zehnder modulators) to route optical inputs to output ports, and this is achieved by varying electrical control signals [1,2]. Each control signal is a binary set driving each individual switch in cross or bar condition. The considered generic structure and application is depicted in Fig. 1a: N input signals at different wavelengths are sent to input ports and by varying M control signals we can route any wavelength to any output port.

Properly connecting the M internal switches the goal of the $N \times N$ switching can be achieved. Many topological solutions have been proposed [3], each characterized by special properties: collision avoidance among the input channels, minimization of optical losses, reduction of the circuit footprint and of the operation power consumption. Among the mostly considered configurations are the butterfly networks (which requires $M = N^2/2$ switches), the Beneš network (a Clos network with 2×2 switches, $M = N\log_2 N - N/2$) and arbitrary-sized Beneš , where N is not constrained to be a power of 2. Beneš solutions are particularly suitable for the PICs integration due to the reduced number of switches required [4]. For all the non-blocking network topologies, the variation of the M control signals generates a total combination of 2^M, whereas $N!$ is the number of distinct permutations of the N input signals as shown in Fig. 1a. So, the control plane is required to define the control signals to set the $N \times N$ switches in the required condition.

Many deterministic algorithms can efficiently ($\mathcal{O}(N\log N)$ for the Beneš network case [5]) calculate the control state of the internal switches to obtain the requested wavelength permutation. Unfortunately each algorithm is specifically designed for the internal topology of the network , and no general algorithm exists [3].

In this work, we propose to train a topology agnostic machine learning (ML) agent to predicting control signals. Given a $N \times N$ photonic switch with arbitrary, and potentially unknown, internal structure, the proposed technique assumes it as a black-box component, as illustrated in Fig. 1b, and it efficiently allows to determine control states with accuracy exceeding 98%.



(a) (b)

Fig. 1: (a) Example of full optical routing with micro-ring based switches, (b) 8×8 Beneš network as a particular case of a the generic black box model with N inputs, N outputs and M control signals.

Table 1: Dataset Statistics

Benes size $N \times N$	**Permutations** $N!$	**Number of switches** M	**Combinations** 2^M	**Dataset**	**Train Set**	**Test Set**
8×8	40,320	20	1,048,576	100,000	70,000	30,000
10×10	3,628,800	26	67,108,864	300,000	210,000	90,000
15×15	1,307,674,368,000	49	562,949,953,421,312	1,000,000	700,000	300,000

2. Results & Conclusion

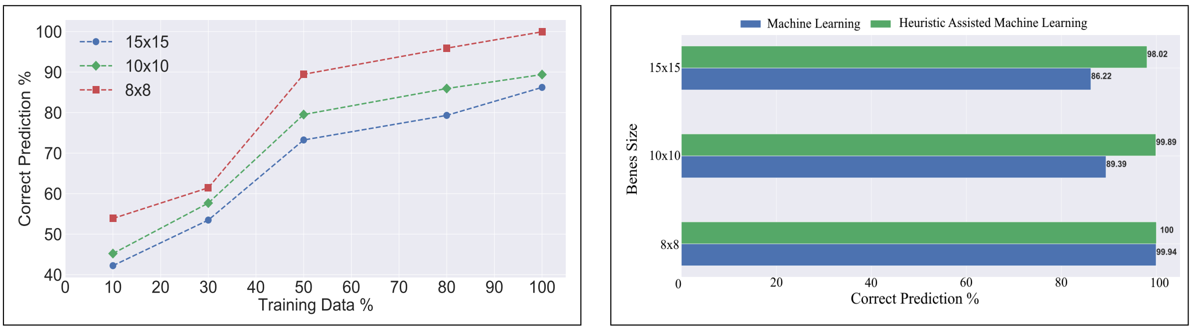
To generate the ML training and testing dataset, we considered a $N \times N$ Beneš network. To demonstrate the scalability of the proposed, approach we addressed three cases: $N = 8, 10, 15$; corresponding to the configurations with $M = 20, 26, 49$ internal switches, respectively. A subset of the total 2^M control combinations is used for the generation of the dataset, as reported in Tab. 1.

The datasets are used to train a supervised learning neural network. The proposed ML model, explicitly a deep neural network (DNN), is developed by using TensorFlow© platform: it incorporates 2 hidden-layers along with 20 neurons for each hidden-layer, having *ReLU* as activation function. The proposed DNN model is evaluated by mean square error (MSE) as loss function. The DNN model is configured for training-steps of 1,000 and learning-rate of 0.01. The train set for each $N \times N$ mode of Beneš topology consists of 70% of the dataset, while the test set consists of 30% of dataset as reported in Tab. 1. In the proposed ML agent we use as features the different combination of wavelengths at the output ports and as labels the M control signals.

Promising results are achieved for each considered Beneš size, as shown in Fig. 2a and Fig. 2b. In Fig. 2a the effect of training data size is shown. The trend shows that the prediction ability of ML model improves when increasing the training data size, but already with a 50% or training size good results are obtained. The rate of correct prediction is summarized for all three considered Beneš sizes by the blue bars in Fig. 2b. We observe an excellent level of accuracy ($> 86\%$) but with a trend to reduce the effectiveness of the prediction when increasing N: correct predictions reach 99.84%, 89.39% and 86.22% for N equal to 8, 10 and 15, respectively.

To improve the prediction capabilities of the ML approach, we added an auxiliary step based on a simple heuristic that we derived from observing wrong configurations. In most of the cases failing to determine the correct control state there was a single switch element in an incorrect state. The heuristic we propose is to simply try the correction of single ring error by flipping one switch at a time and comparing the output sequence against the desired output. For Beneš 8×8, 10×10 and 15×15, using ML assisted by heuristic the accuracy improves to 100%, 99.89%, 98.02% respectively.

In conclusion, we have demonstrated that a ML approach can effectively determine control states for a generic $N \times N$ photonic switch without any knowledge required on the topology. The ML is scalable to large N as we show high level of accuracy with limited size data-set.



(a) (b)

Fig. 2: Correct prediction vs. normalized training size (a), and with and without heuristic correction (b).

References

1. E. Ghillino et.al, “Assessing the impact of design options for an optical switch in network routing impairments,” in *2019 21st ICTON,* (IEEE, 2019), pp. 1–4.
2. I. Khan et.al, “Machine learning assisted abstraction of photonic integrated circuits in fully disaggregated transparent optical networks,” in *2020 22nd ICTON,* (2020), pp. 1–4.
3. S. Arora, T. Leighton, and B. Maggs, “On-line algorithms for path selection in a nonblocking network,” in *Proceedings of the twenty-second annual ACM symposium on Theory of computing,* (1990), pp. 149–158.
4. Q. Cheng et.al, “Silicon photonic switch topologies and routing strategies for disaggregated data centers,” IEEE JSTQE **26**, 1–10 (2020).
5. A. Chakrabarty, M. Collier, and S. Mukhopadhyay, “Matrix-based nonblocking routing algorithm for Beneš networks,” in *2009 Computation World: Future Computing, Service Computation, Cognitive, Adaptive, Content, Patterns,* (IEEE, 2009), pp. 551–556.