

# First Scalable Machine Learning Based Architecture for Cloud-native Transport SDN Controller

Carlos Manso<sup>1</sup>, Noboru Yoshikane<sup>2</sup>, Ricard Vilalta<sup>1</sup>, Raul Muñoz<sup>1</sup>,  
Ramon Casellas<sup>1</sup>, Ricardo Martínez<sup>1</sup>, Cen Wang<sup>2</sup>, Filippos Balasis<sup>2</sup>, Takehiro  
Tsuritani<sup>2</sup>, Itsuro Morita<sup>2</sup>

<sup>1</sup> Centre Tecnològic de Telecomunicacions de Catalunya (CTTC/CERCA). Castelldefels (Spain)

<sup>2</sup> KDDI Research, Inc. Saitama (Japan)

carlos.manso@cttc.es

**Abstract:** We present a cloud-native architecture with a machine learning QoT predictor that enables cognitive functions in transport SDN controllers. We evaluate the QoT predictor training and auto-scaling capabilities in a real WDM/SDM testbed. © 2021 The Author(s)

## 1. Introduction

Software-Defined Networking (SDN) has enabled the centralized use of multi-layer, multi-domain and multi-vendor network equipment using open standard Application Programming Interfaces (APIs) instead of distributed managing through proprietary interfaces. This new centralized paradigm needs a new kind of SDN controller architecture able to deal with the aggregation of several transport layers and tens/hundreds of domains, while at the same time being resilient and capable of scaling to a cloud-level number of connections for provisioning and telemetry [1]. Cloud-native architectures are based on micro-services, a set of loosely coupled services that are connected by means of lightweight protocols, such as gRPC Remote Procedure Calls (RPC). SDN controllers are prone to this kind of architecture as their functional modules can easily be divided into micro-services. This software architectural style makes applications able to scale without limit, among other benefits, to deal with cloud-level load.

At the same time, the centralized SDN architecture has brought a lot more complexity that has to be dealt with. The control of the optical equipment for novel technologies, such Elastic Optical Network (EON) or Spatial Division Multiplexing (SDM) has added plenty degrees of freedom to the provisioning of connections that would be handed directly to the SDN controller. This kind of freedom in top of optical impairments, requests an accurate Quality of Transmission (QoT) estimation for the lightpath that do not introduce high margins, in order to maximize utilization.

Machine learning can aid these challenges by making QoT predictions with better performance and lower computational cost. Nonetheless, the ever-growing degrees of freedom in optical networks, add a time delay for real-time predictions that is unacceptable for real-time provisioning, making scalability very valuable. However, no architecture has been effectively demonstrated for SDN controllers that can provide a common framework for machine learning techniques, since most machine learning approaches are considered as an external algorithm not integrated into the SDN controller.

To address this issues, we extend a cloud-native SDN controller called uABNO [1] with a scalable machine learning framework that would enable multiple autonomic functions, such as the QoT predictor presented in this paper. The auto-scaling properties will also be demonstrated for the machine learning framework in a real WDM/SDM testbed using Docker containers orchestrated and scaled using Kubernetes.

## 2. Architecture and workflow

To be able to successfully deliver a cloud-native SDN controller, it is essential to segregate its functionalities into micro-services that are decoupled between each other. In Fig. 1(a), the micro-services of the uABNO are depicted interacting with the WDM/SDM testbed used in the experimental validation. In this architecture, it is possible to identify four different types of micro-services: a) HTTP micro-services, the NorthBound Interface (NBI), that acts as a gateway for incoming requests from the Operation Support Services and Business Support Services (OSS/BSS) or other clients such an upper SDN or NFV orchestrators. b) gRPC micro-services performing core functionalities such as path computation or communicating with forwarding elements through the SouthBound Interface (SBI), that interact between each other by means of the gRPC protocol. c) Database micro-services used to store information about the topology or the state of the connections served and requested among other data. d) Scalable micro-services, which are able to scale autonomously through the deployment of multiple replicas, and is one of the main contributions in this paper.

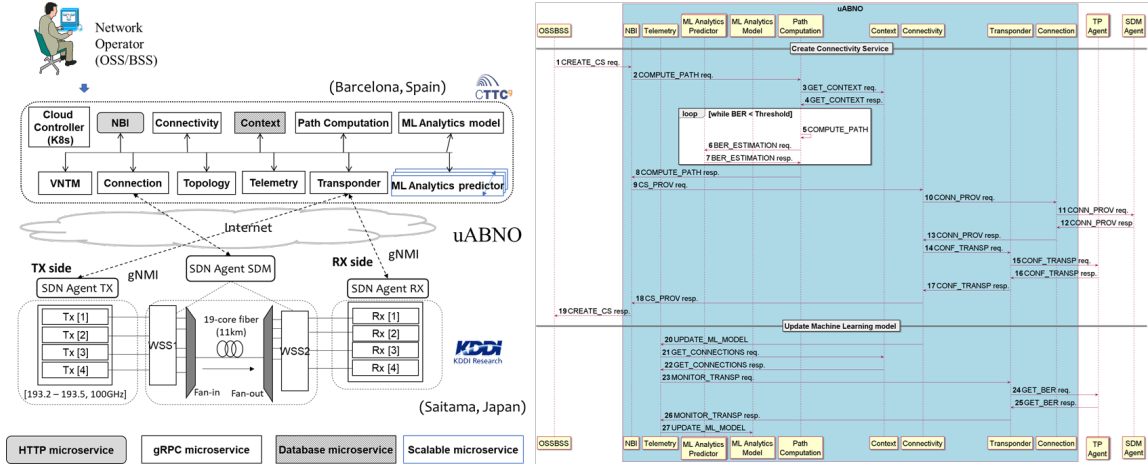


Fig. 1: a) Architecture (left) and b) Provisioning workflow (right)

The machine learning QoT predictor for this controller is divided into two micro-services: the Machine Learning Analytics Model (MLAM) and the Machine Learning Analytics Predictor (MLAP). The MLAM deals with the creation and updating of the mathematical models that will be used by the MLAP to make predictions. The MLAM only interacts with two modules to maximize its decoupling from the rest of the controller, the Telemetry module [2] and the MLAP. The Telemetry module is responsible for harvesting the necessary data from the Transponder module and handing it over to the MLAM to update it. The MLAP is responsible for making predictions with the MLAM supported models and it is also the scalable micro-service demonstrated in section 3. The Kubernetes orchestrator, through Horizontal Pod Autoscaler (HPA), can monitor different metrics such as CPU/RAM consumption or number of queued incoming requests of the current deployed MLAP. In case it detects that a certain threshold is reached, new replicas of the MLAP micro-service are deployed and the QoT prediction requests are load balanced to the new replicas. This allows to scale up depending on the current workload.

As a proof-of-concept to validate this architecture, a straightforward QoT predictor has been implemented using a Deep Neural Network (DNN). The input layer receives the frequency spectrum, full path, capacity, and modulation format for the requested and already provisioned connectivity services. The workflow in Fig. 1(b) depicts the necessary steps and connections between micro-services to provision a QoT-enabled connectivity service. After a connectivity service between two nodes has been requested (step 1), the NBI calls the Path Computation micro-service to compute the path along the network (step 2). It gets the state of the network from the Context micro-service (steps 3, 4) and computes a possible path (step 5). It then asks the MLAP to get the estimated Bit-Error Rate (BER) for the computed path, (steps 6, 7) and if the estimated BER is below a certain threshold, steps 5, 6, 7 are repeated. Steps 9-19 show the actual provisioning and configuration of the connection to the SDM and Transponder agents and the response to the user. A concurrent process to the main workflow is called after step 18 from the Connectivity micro-service (step 20). It signals the Telemetry micro-service that there has been a new connection established, and after getting the state of the other connections established at the moment and gathering the BER by means of the Transponder micro-service (steps 21-26), it feeds the MLAM with the necessary information to update the model (step 27). This constant gather of data by means of the telemetry micro-service enables an incremental learning approach, where new data can be fed into the model whenever it is available to continually improve the predictions.

### 3. Experimental validation

The experimental setup is based on a cloud-native SDN controller deployed at CTTC in Barcelona (Spain), and data plane hardware and SDN agents deployed at KDDI Research in Saitama (Japan) connected using OpenVPN tunnels across the Internet. The experimental setup consisted of a WDM/SDM network domain, which comprised optical switches, a 11-km SDM transmission line (i.e. 19-core fiber [3]) with a fan-in device, a fan-out device, and SDN agents. In the transmitter side, four transponders (ADVA FSP3000) operated at from 193.2 THz to 193.5 THz following the 100 GHz ITU grid were connected to the WDM/SDM domain. Each transponder was equipped with a C-band tunable wavelength 200-Gb/s optical interface and modulation format of the optical signal was 16-QAM. The transponders were controlled by the SDN controller via NETCONF.

The optical switches were based on wavelength-selective switches (WSS) with EDFAs and had multiple inputs/outputs and supported flexible-grid in the C-band. Input ports of the optical switch were connected the fan-in device in order to input optical signals into the SDM transmission line. After the SDM fiber transmission, output ports of the fan-out device were connected to the optical switch. In the receiver side, four transponders were connected to the SDM domain and were also controlled by the SDN controller via NETCONF.

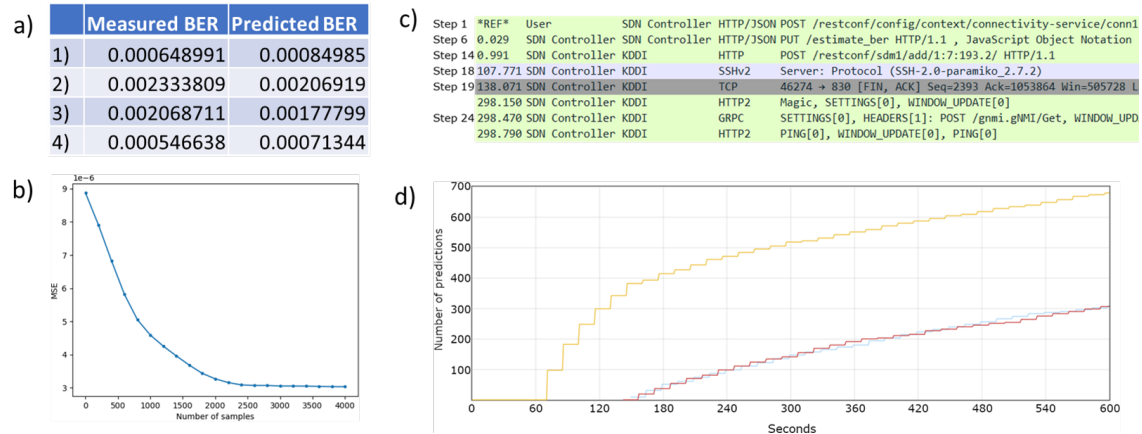


Fig. 2: Experimental results for the machine learning algorithm and micro-service scalability. a) Example BER measurements and predictions. b) MSE depending on number of samples. c) Wireshark capture and relation to workflow. d) Spawn of replicas.

For the experimental validation, only 5 of the 19 SDM cores were considered. To assess the effectiveness of the DNN model, different measurements were considered after the full training. Fig. 2(a) shows an example of different BER measurements and their respective predictions. In 1), TP1 is tuned at 193.5 THz through core #3, other TPs are disabled. In 2), TP1 is tuned at 193.5 THz running through core #3 and TP2 tuned at 193.3 THz through core #3. Comparing between 1) and 2), degradation of BER among optical channels within the same core is seen. This degradation occurs because the EDFA has a fixed power output that is shared among all optical channels. Examples 3) and 4) show TP1 tuned at 193.2 THz through core #4 and #7 respectively, as an example of how choosing between different SDM cores can impact the QoT and be effectively anticipated.

To represent why an incremental learning approach would be valuable, fig. 2(b) shows the accuracy and number of samples needed for the convergence of the model, around a Mean Squared Error (MSE) of  $3 \times 10^{-6}$  with 2500 samples. This strategy could help with recommendations instead of hard actions until a certain accuracy value has reached or to improve the accuracy over time if changes in the environment have affected the validity of the model.

Fig. 2(c) shows a Wireshark capture of the workflow presented in section 2. The first packet corresponds to the user requesting the creation of a connectivity service between two transponder nodes (step 1). Then, it shows how the Path Computation micro-service asks the MLAP the estimation of the BER for a certain connection before its provisioning (step 6). The next three packets show the provisioning of the SDM optical switches via REST (step 14) and the transponder using NETCONF over SSH (steps 18, 19). The final three packets show how the SDN controller gets the BER from the transponder agent using gRPC Network Management Interface (gNMI) (step 24).

A demonstration of the scalability is shown on fig. 2 (d). One connectivity service per second is requested to the SDN controller around second 70, triggering the subsequent requests to the MLAP to predict the BER for the incoming requests. It is configured to scale when the utilization of the requested CPU (50% of a core) or RAM (512 MiB) is exceeded and it does so around second 140, deploying two replicas and balancing the load to them to be below the specified amount.

#### 4. Conclusions and future work

This paper has presented and validated an scalable cloud-native architecture for SDN controllers with an integrated machine learning module. Further research is advised to assess the architecture with more complex and a higher number of cognitive functions and also to test the scalability features in a more real-like environment.

#### Acknowledgments

Work supported by the EC H2020 TeraFlow (101015857) and Spanish AURORAS (RTI2018-099178-I00) and Ministry of Internal Affairs and Communications, Japan grant number JP MI00316.

#### References

1. R. Vilalta *et al.*, “uabno: A cloud-native architecture for optical sdn controllers,” in *OFC*, (2020).
2. R. Vilalta *et al.*, “Telemetry-enabled cloud-native transport sdn controller for real-time monitoring of optical transponders using gnmi,” in *ECOC*, (2020).
3. D. Soma *et al.*, “10.16 peta-bit/s dense sdm/wdm transmission over low-dmd 6-mode 19-core fibre across c+ l band,” in *ECOC*, (2017).