SDN-enabled Sliceable Multi-dimensional (Spectral and Spatial) Transceiver Controlled with YANG/NETCONF

R. Muñoz1, N. Yoshikane2, R. Casellas1, J. M. Fábrega1, R. Vilaalta1, M. Svalto Moreolo1, L. Nadal1,
R. Martínez1, D. Soma2, Y. Wakayama2, S. Beppu2, S. Sumita2, T. Tsuritani2, I. Morita2
1Centre Tecnològic de Telecomunicacions de Catalunya (CTTC/CERCA), Castelldefels (Barcelona), Spain.
2KDDI Research, Inc., Saitama, Japan.
raul.munoz@cttc.es, yoshikane@kddi-research.jp

Abstract: We demonstrate the first SDN-enabled sliceable SDM-WDM transceiver providing multiple spectral-spatial super-channels spanning different cores and modes over a 11-km 6-mode 19-core fiber. We define an open API based on YANG/NETCONF for disaggregated optical networks.

OCIS codes: (060.4250) Networks; (060.4253) Networks, circuit-switched.

1. Introduction
Spatial Division Multiplexing (SDM) has been proposed as the key technology to overcome the capacity crunch that the conventional optical single-mode fibers (SMFs) are facing to accommodate the increasing bandwidth demands forecasted for 5G mobile communications, Internet of Things (IoT), and Big Data technologies. Even though the simplest way to make use of the spatial dimension is to deploy bundles of SMFs, the main target is exploiting the spatial dimensions of the optical fiber (i.e., modes and/or cores), having parallel propagation in the same fiber using multicore fibers (MCF), multimode fibers (MMF), or combining cores and modes in few-mode multicore fibers (FM-MCFs). The combination of SDM with flexi-grid DWDM enables to exploit both dimensions and to provide spectral-spatial super-channels (SSSChs) [1] using SDM-WDM transceivers. An SSSCh can be defined as the association of several flexi-grid DWDM channels (spectral) that can be jointly allocated in different modes and/or cores (spatial) in order to create a (logical) channel with the desired capacity. The use of SSSChs enables simplified switching, joint digital signal processing (DSP), and DSP complexity reduction techniques [2].

On the other hand, there is a general trend towards the disaggregation of the optical networks. In brief, it aims at separating the various hardware elements used in the transmission layer (ROADMs, transceivers, amplifiers, line cards, etc.) as well as the management/control software in order to provide open hardware supporting open application programming interfaces (APIs) that can be accessible and controlled by open source software (e.g., SDN controllers). Disaggregated hardware (i.e., white boxes made of generic, off-the-shelf hardware -bare metal) is not a necessary condition for disaggregated optical networks. The main requirement is to provide an open API with a generic configuration data model. It allows any vendor to contribute with new hardware and software to the system.

This paper presents and experimentally demonstrates the first SDN-enabled sliceable spectral-spatial transceiver (S-SST) that can provide multiple independent SSSChs (slices). Previous works in the literature are focused on transceivers for flexi-grid WDM networks [3]. We define a configuration data model based on YANG, deploy SDN Tx and Rx agents, and define an open API based on NETCONF protocol to interact with the SDN controller.

2. SDN-enabled sliceable spectral-spatial transceiver architecture
The proposed SDN-enabled S-SST (composed of a transmitter -Tx- and receiver -Rx) architecture (Fig.1.a) is based on a modular approach composed of sliceable spectral transceivers (S-STs), mode muxes/demuxes, MCF fan-in/fan-out devices, Tx/Rx DSP, and two SDN agents (for the Tx and Rx). The S-ST provides multiple optical (spectral) channels with bandwidth adaptability (for variable bitrate/distance data flows). The S-STs can be implemented by an array of \( N \) bandwidth variable transponder (BVT) modules as presented by the authors in [4]. The BVTs are connected to programmable spectrum selective switches (SSSs) whose bandwidth occupation, central optical carrier frequency and power/attenuation per port can be adaptively tuned by the SDN Tx/Rx agents. An array of \( F \) S-STs

Fig.1. a) Proposed SDN-enabled sliceable spectral-spatial transceiver architecture. b) Example of spectral-spatial super-channels.
Fig. 2. a) YANG model in-tree, b) <edit-config> XML message, c) NETCONF message exchange

(module: sliceable-transceiver-sdm
  +--rw transceiver
    +--rw slice(s) [discard]
    +--rw slice
      +--rw optical-channel* [opticalchannelid]
      |    +--rw optical-channel
      |       +--rw sliceid
      |       +--rw opticalchannelid
      |       +--rw frequency-slot(n,m)
      |       +--rw equalization
      |       +--rw core-id
      |       +--rw num-taps
      |       +--rw fec
      |       +--rw bandwidth
      |       +--rw constellation
      |       |   +--rw opticalchannelid
      |       |     +--rw slice
      |       |     +--rw opticalsignal
      |       +--rw slice:transceiver-sdm
      +--rw optical-signal* [opticalchannelid]
        |    +--rw ncf
        |      +--rw frequency-slot
        |      +--rw mode-id
        |      +--rw core-id
        |      +--rw opticalchannelid
        |      +--rw sliceid
        |      +--rw num-taps
        |      +--rw equalization
        |      +--rw fec
        |      +--rw bandwidth
        |      +--rw constellation
        |      |   +--rw opticalchannelid
        |      |     +--rw slice
        |      |     +--rw opticalsignal
        |      +--rw slice:transceiver-sdm
      +--rw sliceable-transceiver-sdm)

<edit-config>
<target><running/></target>
<config>
<top>
  <transceiver>
    <slice>
      <optical-channel>
        <opticalchannelid>1</opticalchannelid>
        <frequency-slot>1</frequency-slot>
        <mode-id>1</mode-id>
        <core-id>1</core-id>
        <equalization>
          <num-taps>500</num-taps>
          <mimo>true</mimo>
          <fec>sd-fec</fec>
          <bandwidth>12000000000</bandwidth>
          <constellation>nyquist</constellation>
        </equalization>
        <opticalsignal>
          <slot-width>1</slot-width>
          <equalization>
            <num-taps>500</num-taps>
            <mimo>true</mimo>
            <fec>sd-fec</fec>
            <bandwidth>12000000000</bandwidth>
            <constellation>nyquist</constellation>
          </equalization>
        </opticalsignal>
      </optical-channel>
    </slice>
  </transceiver>
</top>
</config></edit-config>

Fig.2. a) YANG model in-tree, b) <edit-config> XML message, c) NETCONF message exchange

(where \( F \) is the number of supported modes) are mode-multiplexed/demultiplexed. A programmable Tx and Rx DSP module yielding different transmission schemes is available for each array of S-STs in order to perform joint DSP for all the modes (e.g., MIMO equalization). Both the Tx and Rx DSPs are configured by the SDN agents in order to define the optical signal parameters associated to the optical channels. This block is repeated for all the cores supported \((M)\). There is no joint DSP among the cores since it is assumed that inter-core crosstalk can be reduced to negligible values provided that the cores are carefully designed. Finally, the mode muxes/demuxes are connected to the individual cores of a MCF by means of a fan-in/fan-out. This modular approach enables to support MCF, MMF and FM-MCF by adding/removing S-STs, mode muxes/demuxes and fan-in/fan-out devices.

3. YANG data model and Open API based on NETCONF protocol

By means of two SDN agents, the S-SST can be programmed to (re)-configure independent SSSChs composed of multiple optical channels allocated in different modes and cores, transmitting flows at variable bitrate/distance and different transmission formats. In particular, the SDN-enabled S-SST can support four different kinds of SSSChs according to the allocation of the spectral channels. They can be allocated: i) in the same mode and same core (Flexi-grid DWDM channels - Slice #1 in Fig.1.b); ii) in several modes in one core (SDM multimode channels – Slice #2 in Fig.1.b); iii) in several cores in one mode (SDM multicore channels – Slice #3 in Fig.1.b); iv) in different modes and several cores (SDM few-mode multicore channels - slice #4 in Fig.1.b). Disaggregated optical networks requires to define a data model for the S-SST and agreeing on the so called SDN controller south-bound interface (SBI) - with the corresponding message formats and encodings - towards the SDN agents. In this paper, we consider NETCONF protocol standardized by the IETF for network control and management. NETCONF relies on YANG as modelling language, used for defining the configurable parameters and state information in of the S-SST in XML format. Thus, NETCONF and YANG provide a standard way to offer an open API for the SDN controller’s SBI.

Fig. 2.a shows the YANG tree describing the model of the S-SST published online on a public repository [5]. In particular, the SDN-enabled S-SST encompasses a list of slices (each S-SST can support the configuration of multiple slices). Each slice is composed of a slice ID, a list of optical-channels and a list of optical-signal parameters (associated to the optical-channels). The data that can be configured for the optical-channel list is the optical-channel-id, frequency-slot(n,m), mode-id and core-id. Regarding the optical-signal list, the configured data is the optical-channel-id (for associating the signal parameters to an optical channel), constellation (e.g. Nyquist-shaped DP-16QAM), bandwidth (e.g 12GHz), FEC (e.g. SD-FEC), and equalization: equalizer (e.g., LMS 12x12), MIMO (e.g. True/False), and num-taps (e.g. 500). State data of the optical-signal list is the BER in the monitor field.

Fig. 2.b shows the sequence of NETCONF messages employed for provisioning one slice and requesting the BER. First, there is an exchange of <hello> messages between the SDN controller and the SDN Tx agent and SDN Rx agent in order to discover the peer’s supported capabilities (e.g. netconf1.0, netconf1.1). Once the session is open, the SDN controller can request the provisioning of one slice or multiple slices through the exchange of <rpc> and <rpc-reply> messages. In particular, the SDN controller sends a <edit-config> message (Fig.2.c) to both the SDN Tx and Rx agents, and the agents reply with <ok> or <rpc-error>. Once provisioned, the SDN controller can request to the SDN Tx and Rx agents the BER of the optical channels of the slice by sending a <get> message. The SDN Tx and Rx agents reply with a <rpc-reply> message with the state data requested. The SDN controller can remove any slice by sending a <delete-config> message to the SDN Tx and Rx agents.

4. Experimental Setup and validation

The experimental setup is based on an SDN controller deployed at CTTIC in Barcelona (Spain), and an SDN-enabled S-SST deployed at KDDI Research in Saitama (Japan) connected using OpenVPN tunnels on top of internet, as depicted in Fig.3.a. The S-SST is based on the SDM transceiver presented by the authors in [6]. It can generate 1824 optical channels through a 6-mode 19-core fiber (Fig. 3.a). In the Tx, a tunable laser with a narrow spectral linewidth (<30 kHz) and high frequency stability is used for a measured channel. The measured channel and other 15 dummy channels are independently modulated using a 2-channel AWG operated at 60 GSamples/s for I (in-phase) and Q (quadrature) components and two IQ modulators, driven by 12-Gbaud Nyquist-shaped electrical multi-level
signal (8 levels) for 64QAM. The amplified (by EDFA) signals are combined with 12.5 GHz spacing and polarization-multiplexed with a delay of 87 ns. After the 16 channels are power-equalized with an SSS, 16-channel WDM Nyquist-shaped DP-64QAM signals are obtained. The WDM signals are split into two branches for the measured core and other cores. After that, the measured signal is divided into 6 paths with the relative delay of 200 ns between subsequent paths for the decorrelation. These signals are 6-mode-multiplexed using a mode multiplexer based on a multi-plane light conversion. Then the mode-multiplexed signals are launched into the measured core of 6-mode and 19-core fiber through a lens-coupled type fan-in device. After 11 km transmission, the measured SDM channel is demultiplexed with the fan-out device and mode demux. The 6 signals at the output of the mode demux pass through optical bandpass filters, then they are detected by the 6 coherent receivers (heterodyne detection with a free-running oscillator). The received electrical signals are digitized at 80 Gsample/s using three synchronized real-time oscilloscopes. The stored samples are processed offline. First, the samples are down-converted to base band. After the rectangular-shaping, the samples for all-modes are simultaneously processed by a half-symbol spaced 12x12 MIMO equalizer. The tap size is set to be around 200 for all cores, and the tap coefficients are updated based on a least-mean square (LMS) algorithm. After that, the BER is measured. On top of the Tx and Rx, we deploy two SDN agents in order to map the high-level operations coming from the SDN controller for setting up the SSSChs into low-level, hardware-dependent operations of the 16 optical channels, SSS, oscillator, and bandpass filters.

The proof-of-concept is based on sequentially provisioning three slices and experimentally measuring the BER of the optical channels of slice #1 in all modes in the reference core in order to monitor the impact due to the provisioning of additional slices. The 16 available optical channels (1 measured + 15 dummy) are transmitted in the 6 modes and the 19 cores (16x6x19=1824 channels). For slice #1, we transmit 114 channels (measured channel – Fig 3.b- x 6 modes x 19 cores). For slice #2, we transmit 912 channels (8 dummy channels - Fig 3.b- x 6 modes x 19 cores), and finally, for slice #3 we transmit 798 channels (7 dummy channels - Fig 3.b- x 6 modes x 19 cores). We sequentially provision the three slices using the NETCONF message exchange between the SDN controller and the SDN Tx and Rx agents, as defined in the previous section. The SDN Rx agent sends the <ok> message after 3-4 minutes, and the SDN Tx agent after 3 minutes (including hardware configuration in both cases). Thus, the overall provisioning time of a slice is between 6-7 minutes. Fig 3.c shows the optical spectrum at the output of the SSS (Tx) after provisioning the slice #1, #2 and #3. The table in Fig 3.d shows the BER of the measured channel of slice#1 for the six modes in core #19 after provisioning slice#1, after provisioning slice #2, and after provisioning slice #2 and #3. The BER limit used is 2.7e-2 (20% overhead SD-FEC). In the case of slice#1, since the number of optical channels was one, the channel power was relatively higher than other cases (EDFA: automatic current control mode). In the case of “slice#1 with slice#2”, the performance was slightly worse than “slice#1 with slice #2 and #3” case. The reason is that the bandwidth of the measured channel was severely limited by the SSS.

5. Conclusions
We have experimentally assessed the first SDN-enabled sliceable spectral-spatial transceiver with an open YANG/NETCONF API by dynamically provisioning multiples slices and measuring the real impact on the BER due to the fact of provisioning independent (virtual) slices that share the same (physical) transceiver.

5. Acknowledgments
Work supported by EC H2020 BLUESPACE (762055) and the Spanish DESTELLO (TEC2015-69256-R) projects.

6. References