An SDN Control Plane for Multiband Networks Exploiting a PLI-aware Routing Engine

Ramon Casellas[1], **Evangelos Kosmatos**[2], **Andrew Lord**[3], **Chris Matrakidis**[2], **Ricardo Martínez**[1], **Dimitris Uzunidis**[2], **Ricard Vilalta**[1], **Alexandros Stavdas**[2], **Raül Muñoz**[1]
[1] *CTTC/CERCA, Spain* [2] *OpenLightComm Europe, Czech Republic* [3] *British Telecom, UK*
ramon.casellas@cttc.es

Abstract: We report the design of a TAPI SDN control plane for multi-band networks with externalized PLI-aware RMSA. We detail the architecture, data model extensions, algorithms and the implementation and validation in an emulated BT 22-ROADM network.

1. Introduction

Optical transmission exploiting multiple optical bands is the sole solution in the context of wavelength routed networks that increases network capacity without compromising network node connectivity and without exhausting the operator's deployed fiber reserves. A disaggregated, vendor-agnostic, optical multi-band (OMB) transportation ecosystem is necessary to ensure the cost-effective deployment of these systems, which are facing considerable challenges in physical network design and network planning: in an OMB system, physical layer phenomena impose additional and complex performance limitations. Additionally, highly desirable features are network automation and interoperability by means of open interfaces and an SDN-enabled control/management framework. Such interfaces and frameworks are not currently designed with OMB support, which poses additional challenges. In this context we present a modular SDN architecture where a physical layer impairment aware (PLA) routing engine is used for path computation relying on open interfaces and standard data models extended for PLA/OMB.

2. OMB Disaggregated Network architecture with TAPI-enabled SDN Control

This work assumes a disaggregated optical transport network, composed of Terminal Devices (TD), OMB In-Line Amplifiers (ILAs) and OMB ROADMs. An SDN control plane includes a controller for service provisioning and dedicated (sub)controller for path computation (Fig. 1.a) using open and standard interfaces.

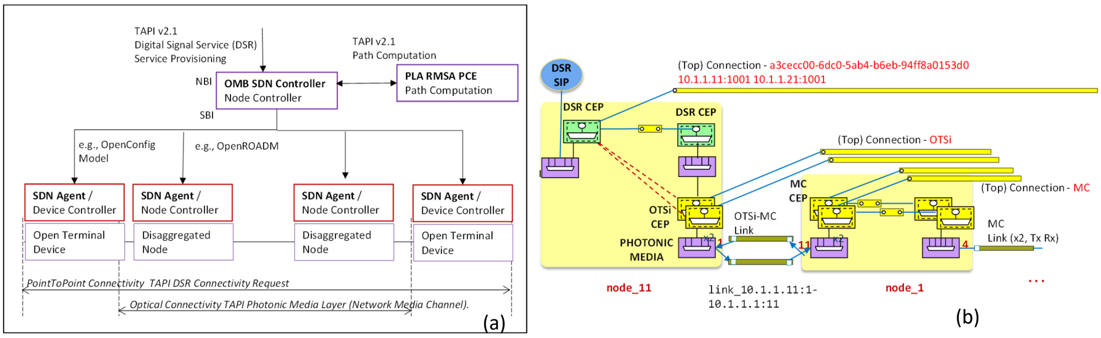


Fig. 1. a) System architecture: b) Sample TAPI context for a digital service carried over 2 OTSi and dedicated media channels exported by the SDN controller NBI

Transport API extensions for PLI aware multi-band network: We have addressed the extension of the current ONF Transport API (TAPI [1]) v2.1.3 photonic media layer models to support the dynamic provisioning of services in the context of PLA OMB networks. Such extensions are challenging, requiring addressing multiple aspects: i) identifying a transceiver's operational modes in terms of bit rates (100G-400G), maximum output power, modulation format, reach and required frequency slot width; ii) the PLA attributes of OMS links (e.g. distance km) per supported band; iii) the inclusion of ILA characteristics (for each supported band, gain, maximum output power, noise figure); iv) MB extensions in terms of different media channel (MC) pools and supportable spectrum frequency ranges and, internally, adequate path computation and resource allocation algorithms. These extensions address mostly the TAPI common, topology, connectivity and path computation modules. The information concerning topological entities as well as current active services and connections is managed by the SDN Controller in the TAPI context (as in Fig. 1.b) and can be queried by external elements.

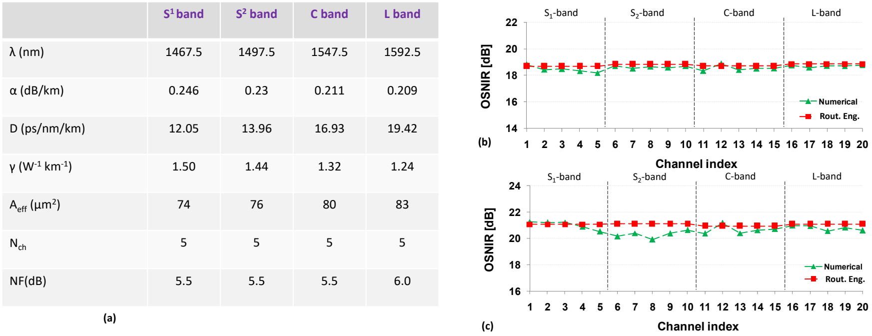
Control plane Provisioning Workflow: The operator requests a Digital Signal Rate (DSR) service between transceivers client ports. The request shall contain the identifiers of the Service Interface Points (SIPs, transceiver client ports); the requested bit rate (e.g. 200G) and applicable routing and topological constraints. The SDN controller delegates the computation to the externalized Routing, Modulation and Spectral Assignment (RMSA) engine which, upon successful computation, provides the number of required OTSi and, for each OTSi, the path in terms of links, the frequency slot and the selected parameters. The SDN controller instantiates the DSR, OTSi and MC layers connection and connection end point (CEPs) objects, configures the transceivers' operational modes and the media channels in the ROADM devices via the SBI interface.

3. A PLA OMB Routing Computation Engine

The details of the OMB routing engine are reported in [2]. The routing engine consists of a RMSA algorithm that takes into account spontaneous effects, like ASE accumulation, induced effects like inter-band effects (SRS) and intra-band effects like self and cross-channel interference (SCI, XPI, respectively). For an OMB transmission system that engages the S, C and L bands, the combined physical layer degradation in BT's network depicted in Fig. 3 is estimated by means of the Optical Signal-to-(Noise plus Interference)-Ratio (OSNIR) defined as

$$OSNR = \frac{P_{ch}\prod_{i=1}^{N_s} G_{SRS,i}}{P_{ASE}+P_{NL}} \tag{1}$$

where N_s is the number of fiber spans a channel is traversing; P_{ch} denotes the power at link ingress for the channel under observation, $G_{SRS,i}$ calculates the SRS Gain/Loss effect for the i^{th} fibre span. The computation complexity of the PLA-RMSA is considerable, and to alleviate this, the impact of fibre non-linearity is calculated by means of closed-form approximations as detailed in [2]. To account for the multiband effects, in the derivation of the corresponding formulae the wavelength dependence of the most important fibre propagation parameters like $\alpha(\lambda)$, $D(\lambda)$, $\gamma(\lambda)$, $A_{eff}(\lambda)$ is included and indicative values are listed in Fig. 2.a. Moreover, the RMSA algorithm has been modified to optionally allocate flows to wavelength channels that are spaced widely apart under lighter loads. Since the requirements for higher node connectivity and link capacity upgrades have to be met simultaneously, the OSNIR performance of all channels across the activated bands has to be as homogeneous as possible. To this end, we have proposed [3], [4] a method to estimate the optimal launch power ($P_{ch,opt}$), where, assuming fully loaded links, the channels in each band may swing between the ASE-limited and NL-limited regimes to achieve uniform OSNIR performance across the spectrum of interest. Here, we extend the optimization method that is applied to links with an arbitrary load of channels. As such, the $P_{ch,opt}$ is re-estimated based on the approach of [4] whenever a quanta of 5 channels is introduced in the network. Here we showcase the effectiveness of this approach by selecting two paths from the network in Fig. 3 as follows: In Fig. 2.b the OSNIR performance is shown for a 750 km path with OMB system with sparsely populated 100G PD-QPSK channels (32 Gsymbols/s) as estimated from our routing engine and from an independent numerical simulation tool. Fig. 2.c is showing the corresponding results for a link 450 km in length populated with spectrally consecutive channels.



	S^1 band	S^2 band	C band	L band
λ (nm)	1467.5	1497.5	1547.5	1592.5
α (dB/km)	0.246	0.23	0.211	0.209
D (ps/nm/km)	12.05	13.96	16.93	19.42
γ (W^{-1} km^{-1})	1.50	1.44	1.32	1.24
A_{eff} (μm^2)	74	76	80	83
N_{ch}	5	5	5	5
NF(dB)	5.5	5.5	5.5	6.0

(a)

Fig. 2. a) Wavelength-dependent parameters and the OSNIR performance for b) a 750 km link with sparsely allocated channels; c) 450 km link with spectrally consecutive channels in the BT network.

4. Experimental Integration and Control Plane Latency

BT core Emulated Network Scenario: the experimental testbed has been set-up at CTTC laboratories. The emulated flexi-grid network encompasses 22 ROADMs (Fig. 3). The PLA computation engine (PCE) is located at OLC-E premises interconnected via a VPN connection. The SDN controller, extending [5], exports a TAPI NBI. To validate the system, we request a DSR service between transceivers at different source destination pairs. During the provisioning, the SDN controller delegates path computation to the PCE which, upon request, retrieves (if needed) the network topology and list of active services/connections from the controller. Subsequently, the PCE executes the RMSA algorithm on the received context and replies with the list of routes for the request.

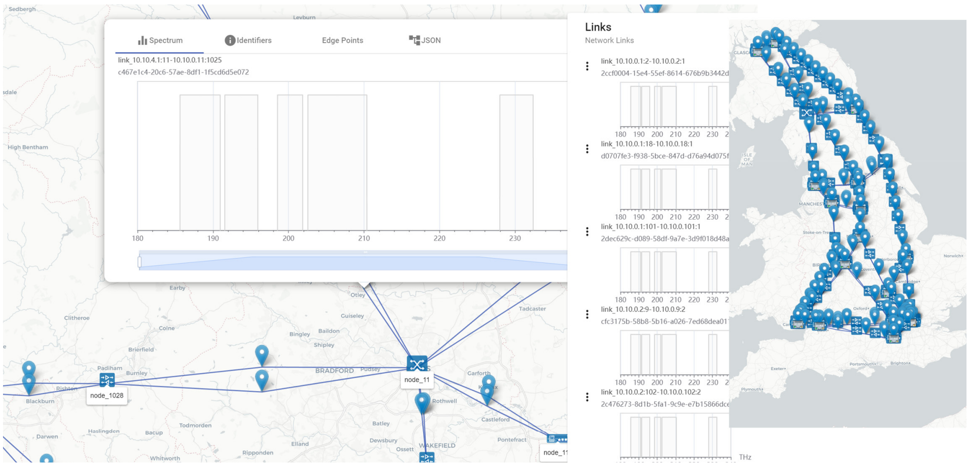
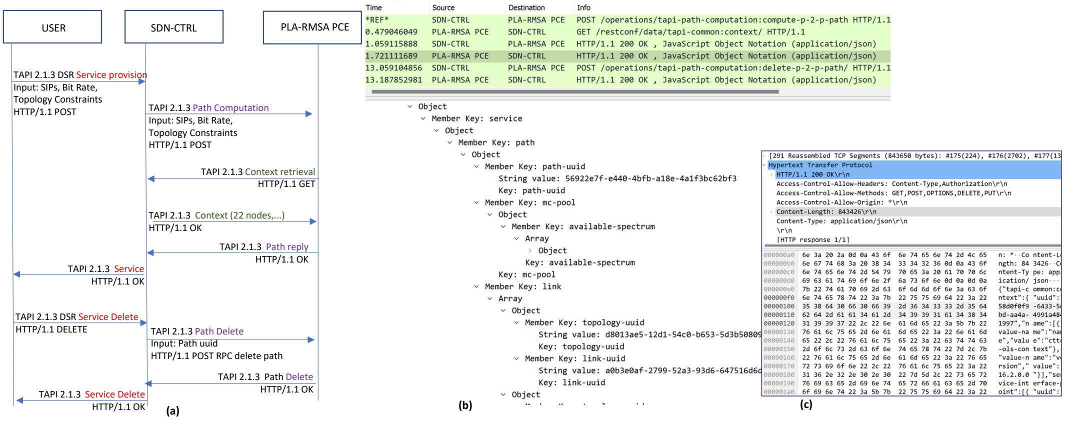


Fig. 3. BT emulated network with 22 ROADMs showing a link with different frequency bands

Control plane latency results The system is evaluated in terms of functionality and performance. Fig. 4.b shows the message exchange. We measure the delay between the path request and the response. Two scenarios are considered: a) the PCE database is not up to date, therefore it needs to retrieve the TAPI context; b) the PCE database is up to date, no need to retrieve it. A set of experiments are executed for both scenarios and the final results are analysed and reported. In the first scenario, a delay of 1.7 s is measured, while for the second scenarios 1.3s. The JSON encoded TAPI context is 843426 bytes (in 291 TCP segments) (Fig. 4.c) and the TAPI path computation response is 2128 bytes (in 2 TCP segments). Clearly, the maintenance of an updated database in the PCE further decreases latency and the number of messages exchanged between the two entities.



```
Time          Source         Destination    Info
*REF*         SDN-CTRL       PLA-RMSA PCE   POST /operations/tapi-path-computation:compute-p-2-p-path HTTP/1.1
0.479046049   PLA-RMSA PCE   SDN-CTRL       GET /restconf/data/tapi-common:context/ HTTP/1.1
1.059115888   SDN-CTRL       PLA-RMSA PCE   HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
1.721111689   PLA-RMSA PCE   SDN-CTRL       HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
13.059104856  SDN-CTRL       PLA-RMSA PCE   POST /operations/tapi-path-computation:delete-p-2-p-path/ HTTP/1.1
13.187852981  PLA-RMSA PCE   SDN-CTRL       HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
```

```
⌄ Object
  ⌄ Member Key: service
    ⌄ Object
      ⌄ Member Key: path
        ⌄ Object
          ⌄ Member Key: path-uuid
              String value: 56922e7f-e440-4bfb-a18e-4a1f3bc62bf3
              Key: path-uuid
          ⌄ Member Key: mc-pool
            ⌄ Object
              ⌄ Member Key: available-spectrum
                ⌄ Array
                  › Object
                  Key: available-spectrum
              Key: mc-pool
          ⌄ Member Key: link
            ⌄ Array
              ⌄ Object
                ⌄ Member Key: topology-uuid
                    String value: d8013ae5-12d1-54c0-b653-5d3b50809
                    Key: topology-uuid
                ⌄ Member Key: link-uuid
                    String value: a0b3e0af-2799-52a3-93d6-647516d6d
                    Key: link-uuid
              ⌄ Object
```

```
› [291 Reassembled TCP Segments (843650 bytes): #175(224), #176(2702), #177(13
⌄ Hypertext Transfer Protocol
  › HTTP/1.1 200 OK\r\n
    Access-Control-Allow-Headers: Content-Type,Authorization\r\n
    Access-Control-Allow-Methods: GET,POST,OPTIONS,DELETE,PUT\r\n
    Access-Control-Allow-Origin: *\r\n
  › Content-Length: 843426\r\n
    Content-Type: application/json\r\n
    \r\n
    [HTTP response 1/1]
000000a0  6e 3a 20 2a 0d 0a 43 6f  6e 74 65 6e 74 2d 4c 65   n: *··Co ntent-L
000000b0  6e 67 74 68 3a 20 38 34  33 34 32 36 0d 0a 43 6f   ngth: 84 3426··C
000000c0  6e 74 65 6e 74 2d 54 79  70 65 3a 20 61 70 70 6c   ntent-Ty pe: app
000000d0  69 63 61 74 69 6f 6e 2f  6a 73 6f 6e 0d 0a 0d 0a   ication/ json···
000000e0  7b 22 74 61 70 69 2d 63  6f 6d 6d 6f 6e 3a 63 6f   {"tapi-c ommon:c
000000f0  6e 74 65 78 74 22 3a 7b  22 75 75 69 64 22 3a 22   ntext":{ "uuid":
00000100  35 38 64 30 66 30 66 39  2d 36 34 33 33 2d 35 64   58d0f0f9 -6433-5
00000110  62 64 2d 61 61 34 61 2d  34 39 39 31 61 34 38 34   bd-aa4a- 4991a48
00000120  31 39 39 37 22 2c 22 6e  61 6d 65 22 3a 5b 7b 22   1997","n ame":[{
00000130  76 61 6c 75 65 2d 6e 61  6d 65 22 3a 22 6e 61 6d   value-na me":"na
00000140  65 22 2c 22 76 61 6c 75  65 22 3a 22 63 74 74 63   e","valu e":"ctt
00000150  2d 6f 6c 73 2d 63 6f 6e  74 65 78 74 22 7d 2c 7b   -ols-con text"},
00000160  22 76 61 6c 75 65 2d 6e  61 6d 65 22 3a 22 76 65   "value-n ame":"v
00000170  72 73 69 6f 6e 22 2c 22  76 61 6c 75 65 22 3a 22   rsion"," value":
00000180  31 36 2e 32 2e 30 2e 30  22 7d 5d 2c 22 73 65 72   16.2.0.0 "}],"se
00000190  76 69 63 65 2d 69 6e 74  65 72 66 61 63 65 2d 70   vice-int erface-
000001a0  6f 69 6e 74 22 3a 5b 7b  22 75 75 69 64 22 3a 22   oint":[{ "uuid":
```

Fig. 4. a) Control plane workflow between entities; b) Wireshark capture showing the detailed payload c) TCP analysis and context size

5. Conclusions

We have designed and implemented an SDN control plane for multi-band networks, extending the current T-API models and accounting for physical layer impairments in support of externalized path computation. We have validated the approach and evaluated the control plane latency overhead and latency.

Acknowledgements Work supported by the EC H2020 B5G-OPEN (101016663) and Spanish AURORAS (RTI2018-099178-I00) projects.

References

1. ONF Transport API data models https://github.com/OpenNetworkingFoundation/TAPI
2. D. Uzunidis et al, “Strategies for Upgrading an Operator’s Backbone Network Beyond the C-Band: Towards Multi-Band Optical Networks,” in *IEEE Photonics Journal*, vol. 13, no. 2, pp. 1–18, April 2021.
3. D. Uzunidis, et al., “Power Optimization Strategy for Multi-Band Optical Systems,” in *ECOC*, 2020.
4. D. Uzunidis, et al., “Connectivity Challenges in E, S, C and L Optical Multi-Band Systems,” in *ECOC*, 2021.
5. R. Casellas, et al., “An OLS Controller for Hybrid Disaggregated Networks with Open Interfaces” in *OFC*, 2020.