Highly-available SDN control of Flexi-grid Networks with Network Function Virtualization enabled replication (Invited)

Ramon Casellas, Ricard Vilalta, Ricardo Martínez, and Raül Muñoz

***Abstract*—New trends and emerging requirements have driven the development of extensions to the Path Computation Element (PCE) architecture beyond the computation of a set of constrained routes and associated resources between endpoints, given a network topology. Such extensions involve the use of a PCE for the control of network services, in which deploying a PCE as a centralized network controller facilitates the adoption of SDN principles while allowing a progressive migration of already existing deployments.**

A key requirement for the adoption of centralized control solutions is the ability to deploy a resilient, secure, dynamically configurable, adaptive and highly available (virtualized) infrastructure supporting end-to-end services, including critical and vertical ones. Part of this infrastructure are the control plane functional elements (e.g., controllers), and the use of Network Function Virtualization (NFV) is a enabler for the high-availability of such elements, while additionally reducing OPEX and CAPEX: NFV provides a feature-complete framework for the replication of software components, straightforward and commonly adopted approach to address the aforementioned requirement, but it implies the need for timely synchronization of databases between replicas.

In this paper [1] we present, implement and validate an architecture for PCE and SDN control high-availability, combining the virtualization of the control function by means of dynamic replication and the timely synchronization of their internal state, using the PCEP and BGP-LS protocols. We experimentally validate the approach with a testbed including a GMPLS/PCE control plane, and a replica management system implemented following the ETSI NFV framework, using the OpenStack cloud management software.

***Index Terms*—Software Defined Networking (SDN), Network Function Virtualization (NFV), Path Computation Element (PCE), Flexi-grid networks control and management, High-availability, Replication**

I. Introduction

THE *path computation function* is commonly accepted as an integral part of either a management or control plane (either centralized or distributed). As such, the Path Computation Element (PCE) architecture, developed within the IETF [**?**], defines a PCE as an entity capable of performing constrained path computation, along with a PCE communications protocol (PCEP, [**?**]). The PCEP protocol was initially specified to allow a path computation client (PCC) to request path computations, enabling a wide range of deployment scenarios and addressing specific problems such as path computation in multi-domain networks with limited topology visibility [**?**].

The ASON/GMPLS architecture remains a viable, mature approach for the provisioning of data channels benefiting of mature protocols, and the adoption of SDN is in part justified by the fact that business and application logic can be easily integrated into a control layer, while relegating e.g. the GMPLS control plane to an automation tool part of the provisioning process. With SDN, in view of programmability and the use of open interfaces, operators can provision new services efficiently.

Common deployments of PCEs are centralized, although this is not mandated by the architecture. This has driven the development of extensions to the PCE architecture beyond the original scope of computing constrained routes between end-points, given a network topology. Such extensions involve the use of a PCE for the control of network services, driving the actual provisioning processes. A PCE can ease the adoption of SDN principles while allowing progressive migration of already existing deployments, acting as a centralized entity where operator-defined algorithms and policies can be deployed, while still driving distributed MPLS/GMPLS transport networks and other technologies.

In particular, the Applications Based Network Operations (ABNO) [**?**] architecture defines a SDN-like approach that can be used for the control of transport optical networks, including a stateful PCE (a PCE that takes into account both the network topology and connections database to perform path computation). Lately, a PCE is becoming more and more functionally equivalent to a Software Defined Networking (SDN) controller and PCEP extensions are being developed to use a PCE with different south bound interfaces (SBIs) including the PCE driven control and instantiation of Label Switched Paths (LSPs) in MPLS/GMPLS

Manuscript received June 15, 2016.

The authors are with CTTC, Av. Carl Friedrich Gauss 7, PMT, Edifici B4, 08860 Castelldefels, Barcelona, Spain

e-mail: ramon.casellas@cttc.es

[1]This manuscript is an extended version of our previous work, published in OFC2016 [**?**]

[?], or its use in Segment Routing (SR) networks [?], where a source node can choose a path without relying on hop-by-hop signalling protocols such as RSVP-TE. Finally, efforts are ongoing to allow a PCE to have direct control over each node along the path, driving the setup and release of cross-connections and related forwarding operations [?].

Generically, a key requirement for the adoption of centralized control is the deployment of a resilient, secure, dynamically configurable, adaptive and highly available (virtualized) infrastructure supporting end-to-end services, including critical and vertical ones. For the particular case of the Path Computation Function, network operators need to be able to upgrade different components without disrupting existing network operation. This includes hot-swapping, software and hardware upgrades, policy changes, etc. Carrier class solutions require reliable software components, with flexible upgrade/update cycles, redesigning of active-standby deployments, as well as innovative approaches and mechanisms dealing with unprecedented system complexity and service criticality (i.e. including environments supporting multi-tenancy).

The use of Network Function Virtualization (NFV) [?], described later, partially addresses this requirement, additionally reducing OPEX and CAPEX. Its use for the deployment of control plane functions, including the PCE, has been recently considered [?], [?]. The use of replication for software components is a straightforward approach to high availability, but it implies the need for distributed network databases and their timely synchronization: one of the missing aspects of previous work is related to the synchronization of the PCE internal databases.

The GMPLS/PCE architecture conveys two main control plane databases: the Traffic Engineering Database (TED), and the Label Switched Path Database (LSPDB). While the use of general-purpose distributed databases is in scope, we still lack clear and standard information and data models to successfully model such databases, along with the actual reference points, protocol(s) and interfaces, specially in order to avoid vendor-specific solutions and scenarios, limiting interoperability. Alternatively, and as put forward in this work, the synchronization of the TED and LSPDB between dynamically instantiated replicas is carried out by using existing, mature, open and standard protocols, namely, PCEP and BGP-LS [?]. Consequently, the network (link and node) data and information models are implicit by the currently supported protocol information objects.

The paper is structured as follows: after this introduction, we briefly present the main concepts behind the ETSI NFV (Section II) in view of its applicability for the virtualization of PCE replicas. In Section III, we detail our proposed control plane architecture and proposed functional entities, message exchanges and workflows. In Section IV, we present the main components of our experimental testbed and in Section V we summarize the main results of our experimental evaluation. Finally, Section VI concludes the paper.

II. ETSI Network Function Virtualization

The ETSI Network Function Virtualization (NFV) Industry Specification Group (ISG) addresses the dynamic deployment and operation of common network functions, stored and executed in virtual computing instances, which are in turn typically running in commodity hardware. NFV defines the architecture and interfaces for the management and orchestration of such Virtualized Network Functions (VNFs) and, amongst relevant aspects, the initial documents recognized the need for the arbitrary and flexible composition of VNFs into graphs, potentially spanning multiple domains. An end-to-end ETSI network service (NS) can be described by a Network Function (NF) Forwarding Graph of interconnected Network Functions and end-points.

Notable functional elements of NFV Management and Orchestration (MANO) part are the NFV Orchestrator (NFV-O) – which manages the lifecycle of ETSI network services, global resource allocation and the validation and authorization of infrastructure resource requests – and the Virtualized Infrastructure Manager (VIM) – which controls and manages the compute, storage and network resources, within one operator infrastructure sub-domain. – Multiple VIMs can be orchestrated by the orchestrator (NFV-O).

The concept of domain within the NFV is manifold. The architecture defines, notably, the concepts of VNF domain, infrastructure domain and tenant domain, where multiple tenant domains can co-exist in a single infrastructure domain, separating domains associated with VNFs from domains associated with the NFV infrastructure (NFVI). Within the NFVI [?], the aspects of compute, hypervisors, and infrastructure networking are maintained as separate. Geographically speaking, a NFVI may have multiple points of presence (NFVI-PoP), defined as a single location with a set of deployed NFVI-Nodes. A given NFVI can be administratively split into NFVI domains, thus managed by one or more Virtual Infrastructure Managers or VIMs.

In this work, we are mostly concerned with a single VNF domain, potentially although not necessarily across multiple infrastructure domains. We consider PCE (or SDN controller) replicas as the VNFs, and it is thus the role of the NFV-O to orchestrate NFVI resources across one or multiple VIMs. We assume that a (private) NFVI is available for the network operator to deploy control plane functions. By operating this domain, multiple instances can be launched under the control of the operator (see Figure 1).

III. Control Plane architecture

In this section, we detail the major elements of the control plane architecture, focusing on the virtualization of PCE functions. PCE high-availability relies

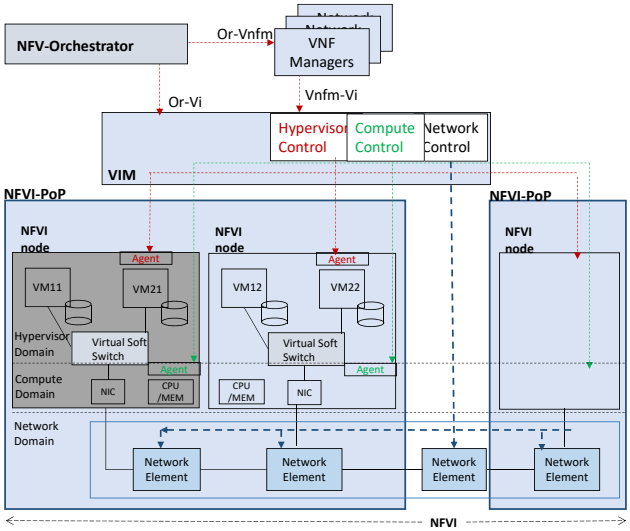


Figure 1. Simplified ETSI NFV architecture enabling the execution of virtualized PCE instances

on synchronized PCE *replicas*, and is enabled by the combined use of cloud computing architectures (with the actual coordination of PCE instances under the responsibility of dedicated cloud infrastructure controller) and entities that enable the database synchronization avoiding complex state machines.

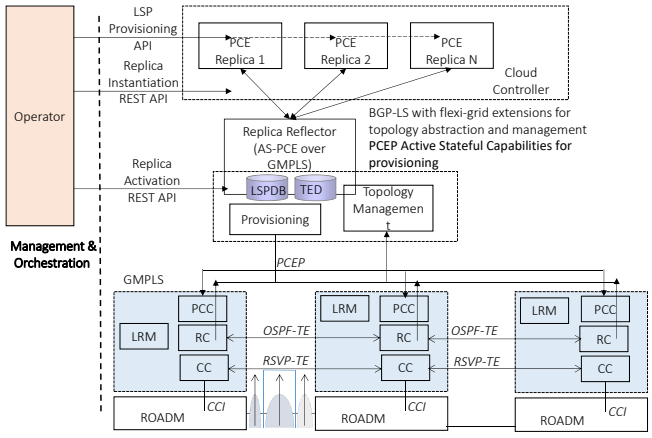


Figure 2. Proposed control plane and database synchronization architecture and main components: PCE replicas as virtualized SDN controllers (VNFs); Replica manager as NFV-O; Replica reflector for synchronization and GMPLS controlled flexi-grid network (with Link Resource Manager, LRM; Connection Controller, CC and Routing Controller, RC

The proposed architecture (see Figure 2) relies on the following main component concepts:

- A controlled transport network infrastructure. In this work, this network is assumed to be an optical transport network with flexi-grid optical spectrum switching, composed of flexi-grid ROADMs interconnected with optical fibers in an arbitrary mesh topology. For the experimental demonstration, this work assumes that PCEs act as SDN controllers, ultimately delegating establishment of LSPs to an underlying GMPLS control plane (without excluding other PCE southbound interfaces not requiring GMPLS, including e.g. OpenFlow [**?**], [**?**], or PCEP for forwarding configuration [**?**]).
- A private ETSI NFVI (implemented in terms of a cloud infrastructure and OpenStack deployment). This, in turn, enables the deployment of multiple, dynamically allocated PCE *replicas*, understood as different instances of the same functional entity, which are themselves synchronized by means of the PCEP and BGP-LS protocols.
- The use of a *replica reflector*, an entity acting conceptually as a BGP reflector [**?**] thus avoiding the full mesh between replicas and limiting control plane overhead. This PCEP/BGP-LS reflector acts as a bridge between the replicas and the underlying control plane, being a proxy for centralized LSP provisioning and path computation.
- The *replica manager* with Graphical User Interface (GUI) both interacting with the operator's operation and business support systems (OSS/BSS) and, at the same time, behaving like a NFV-O for the dynamic allocation of replicas and the coordination of the replica reflector.

There are several important considerations to note. First, although the straightforward implementation of the concept relies on homogeneous software images, diversity is not precluded (for example, to cover migration or software upgrades), as long as the different software images implement the synchronization protocols. Second, the use of a reflector raises the issue of high-availability of the reflector itself. Even if the reflector is significantly simpler than the actual PCEs and not subject to updates, upgrades and life-cycles, multiple reflectors can potentially be deployed (e.g., in pairs and clustering) thus fulfilling high-availability requirements (see Figure 3).

A. Dynamic operation and procedures

We summarize here the main, simplified workflows and message exchanges for the system, with the help of Figure 4: The NMS/Replica manager (NFV-O) uses the cloud controller (VIM) exported REST API that enables the on-demand dynamic instantiation and deallocation of customized PCE instances, with varying capabilities in terms of memory, CPU and deployed algorithms and policies (in the Figure, *Nova Instance Launch*), retrieving its dynamic IP address. Once a new replica is instantiated, the reflector establishes both a BGP-LS and a PCEP session towards the new replica upon request from the manager (in the Figure, *Replica activation*). After the PCEP and BGP-LS handshakes, the sessions are kept active for the purpose of continuous and dynamic synchronization.

The activation and de-activation of a replica is assumed to happen at a longer time-scale than the on-demand provisioning of flexi-grid optical connections.

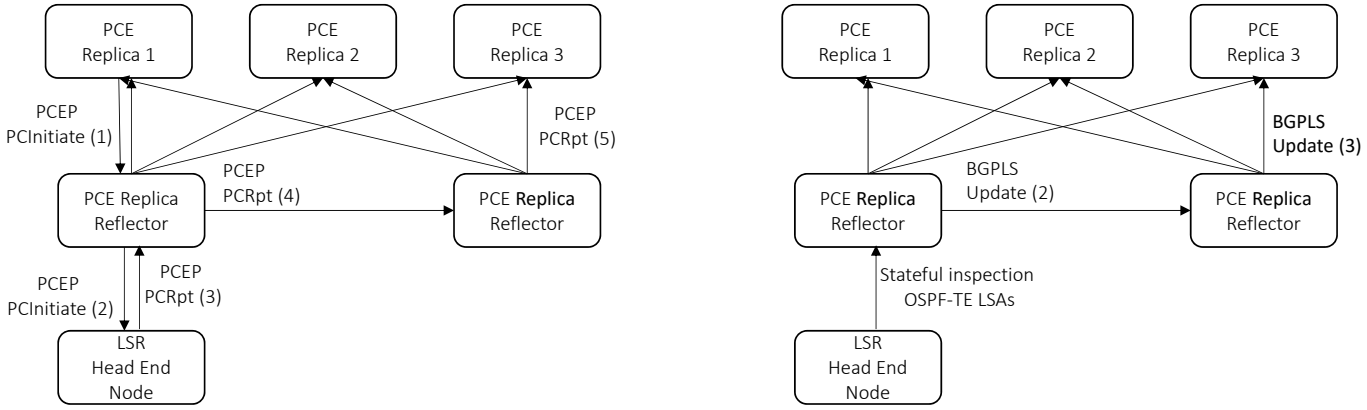


Figure 3. Basic message flow procedure to synchronize state. LSPDB synchronization is done mainly by means of the PCRpt messages, that are forwarded after a PCInit-driven successful provisioning of an LSP. TED synchronization is done by forwarding BGP-LS update messages (mapping OSPF-TE inspected Link State Advertisements, LSAs.

This involves consuming the north bound interface (NBI) defined of any replica (in practice this can be accomplished by the use of floating IP addresses or DNS round robin). When an instance receives an instantiation request, it sends a PCEP Path Initiate message (PCInit) to the reflector, which forwards it to the corresponding head end-node. Upon the successful establishment of the LSP, the head end node sends a PCEP Path Computation Report (PCRpt), which is forwarded to all the replicas. Figure 3 shows the architecture and the simplified flow of messages.

B. LSPDB synchronization

The synchronization of the LSP database (LSPDB), that is, the set of active LSPs and their attributes, is done mainly by means of the PCEP Stateful Capabilities with Instantiation protocol extensions and, in particular, the use of the PC Initiate (PCInit) and PC Report (PCRpt) messages. The PCInit message specifies that an LSP is to be instantiated (or released) in the network. The PCInit message includes, notably, the endpoint nodes, the path to use (in terms of Explicit Route Object, or ERO) and related objects to uniquely identify the LSP in the scope of the control domain (such as the LSP object and/or LSP symbolic name). For flexi-grid networks, additional parameters are included (such as the optical spectrum needed, and allocated frequency slot). Likewise, The PC Report (PCRpt) message, used to advertise the status of an LSP upon initiation, or modification (commonly sent by the ingress PCC upon completion of the establishment procedure). It conveys the LSP operational status, LSP identifiers and mapping with the GMPLS control plane constructs and other relevant information (such as the detailed route and resources used). Consequently, forwarding or relaying the same PCRpt messages to multiple instances or replicas is an effective means to synchronize the LSPDB.

C. TED Synchronization

Topology synchronization happens at two different levels. At the lowest level, the PCEP/BGP-LS reflector is able to obtain an up-to-date, detailed view of the topology (TED) by passive inspection of OSPF-TE Link State Advertisements (LSAs). The TED can later be exported, since, at the highest level, the synchronization between the reflector and the PCE replicas is done by means of the BGP-LS protocol with extensions for flexi-grid. In short, BGP-LS refers to the extensions done to the well-known BGP protocol to support the exchange of link-state (topological) information between entities and it is used to relay TE information, directly mapping OSPF-TE information objects to BGP-LS ones. From the perspective of protocol operation, the synchronization happens after the BGP-LS session has been established, where a BGP-LS peer can send UPDATE messages including the MP_REACH attribute. The Network Layer Reachability Information (NLRI) contains the attributes of the network nodes and links: for a node, this is reflected in terms of IPv4 router ID, Autonomous System (AS) identifiers, Routing Area and other related properties. For a TE link it means its source and destination node and the TE attributes. For this, the protocol uses the IPv4 addresses of the nodes, Local Node Descriptors and Remote Node Descriptors. Additionally, in a flexi-grid network, unnumbered interfaces of the links as well as the maximum, unreserved, reservable bandwidths, the TE default metric, SRLGs and a new bitmap reflecting the status of the different nominal central frequencies are also included. For further details, please see, for example, [**?**].

Note that a new replica can be instantiated at any time so, in addition to the continuous updating via the reflector to active replicas, a new instantiated replica will receive a ”dump” of the system status upon successful completion of a BGP-LS and PCEP handshake. At this point, there will be as many PCRpt

as active LSPs and typically as many BGP-LS Updates as topology elements (networks and nodes) as shown in Figure 4. This is the part that may present a bigger spike in control plane overhead, as we will see in Section V.

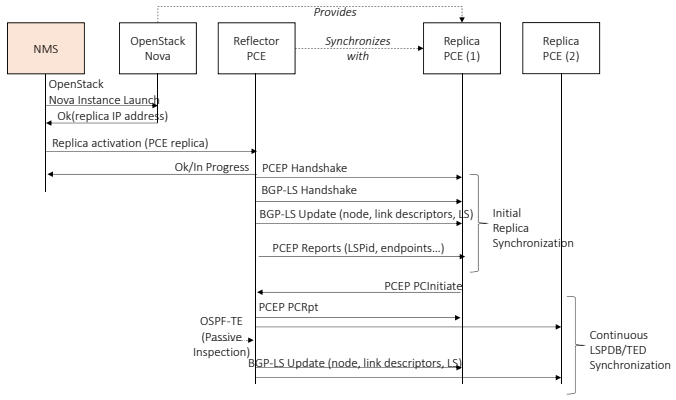


Figure 4. Summary of the Workflow and message exchange showing the NMS/App and the different architecture components

IV. Testbed Deployment

In order to validate the approach and architecture, we have deployed an experimental control plane testbed. In this section, we detail its different elements (see Figure 5).

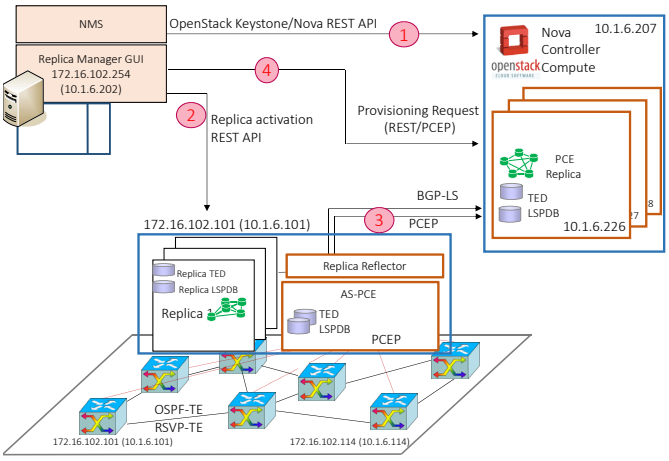


Figure 5. Deployed testbed overview for the performance evaluation of PCE replication

At the lowest level of the testbed there is an optical transport network, featuring a GMPLS control plane. The PCE acts as the centralized element behaving as a SDN controller, with or without delegation to GMPLS. The control plane protocols PCEP, BGP-LS, OSPF-TE and RSVP-TE have been extended to convey attributes such as the per-link nominal central frequencies (NCFs) and the availability of sliceable bandwidth variable transceivers (S-BVTs) [**?**].

The replica manager application is implemented in Python and executed in a dedicated GNU/Linux PC, presenting the system status to the operator (see Figure 6). It is responsible for coordinating, authorizing, and reserving NFVI resources (in our case within one single NFVI-PoP), consuming the north bound APIs offered by the OpenStack controller. Other functions of the replica manager are to keep track of how many PCE replica instances have been allocated and their attributes and notifying the reflector of a new instantiated replica, triggering the hand-shake and initial state synchronization. The consumed APIs concern the OpenStack keystone, glance and nova services, for identity, image and computing resources management, respectively. There are no requirements for advanced networking services (that is, all instances are launched in the same private subnetwork that is mapped to a physical network, providing connectivity between the replicas and physical systems). The management and orchestration application resides at host 10.1.6.202.

The deployed OpenStack setup, includes an OpenStack controller node and one compute node, located at 10.1.6.207. Virtualized PCE instances (replicas) are all instantiated in the same administrative project, each one configured with only one network interface card (NIC), dynamically obtaining an IP address via an OpenStack managed DHCP agent, from a single, common, and private IP address pool (10.1.6.226, 227,...). The requirements of our implemented PCE/SDN controller are reasonable, so each instance can be launched using an OpenStack *small* flavor, which requires a single CPU core, 1GB of RAM and a small hard disk (20 Gb). Finally, the GMPLS controllers in the aforementioned topology have a private IP address for management (from 10.1.6.101 to 10.1.6.128) while the controllers themselves use the node identifiers 172.16.102.101 to 172.16.102.114, respectively. The PCE reflector is itself located at the node 10.1.6.101. The reflectors and GMPLS controllers run on Intel Core 2 Duo CPU @ 2.66GHz with 2-4 GB RAM.

In this paper, we have focused on the highly-available features of the proposed architecture when a replica instance fails, and how the PCEP and BGP-LS protocols are used to synchronize state between replicas. It is worth noting the high-availability requirements have further implications, such as accounting for the failure of compute nodes. While this can be, in part, mitigated by the operator by instantiating replicas in different compute nodes and different availability zones (making use of the cloud management capabilities), macroscopically we leverage on the benefits of NFVI/cloud approach, and existing mechanisms to move VMs and re-instantiate active ones in other compute hosts.

For our specific scenarios, the emulated data plane flexi-grid network topology (Figure 7) represents a core Spanish network with 14 flexi-grid ROADMs and 14 client nodes, attached to a each core node via a dedicated link and having a sliceable transceiver.

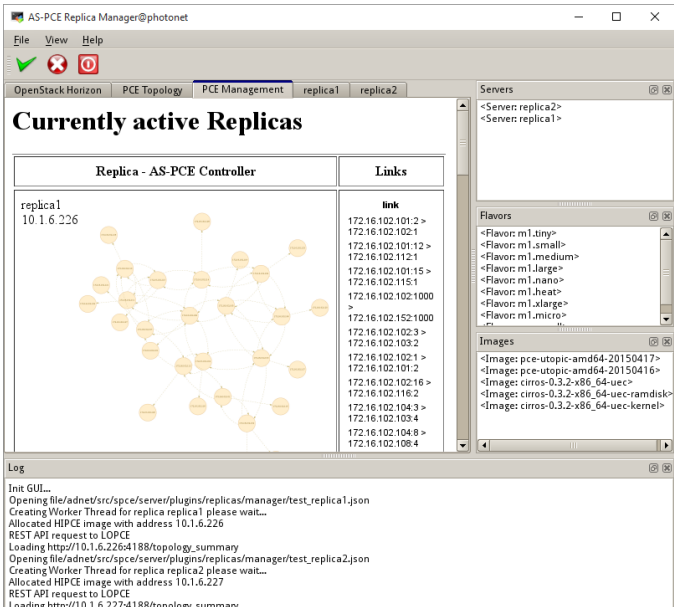


Figure 6. Replica manager GUI application showing the topology seen by replica in 10.1.6.226

There are 22 inter-ROADM bidirectional links and 14 attachment links. Each link has 128 nominal central frequencies (NFCs). The ROADMs are supposed colorless, directionless and contentionless, being able to switch any frequency slot from any port to any port.

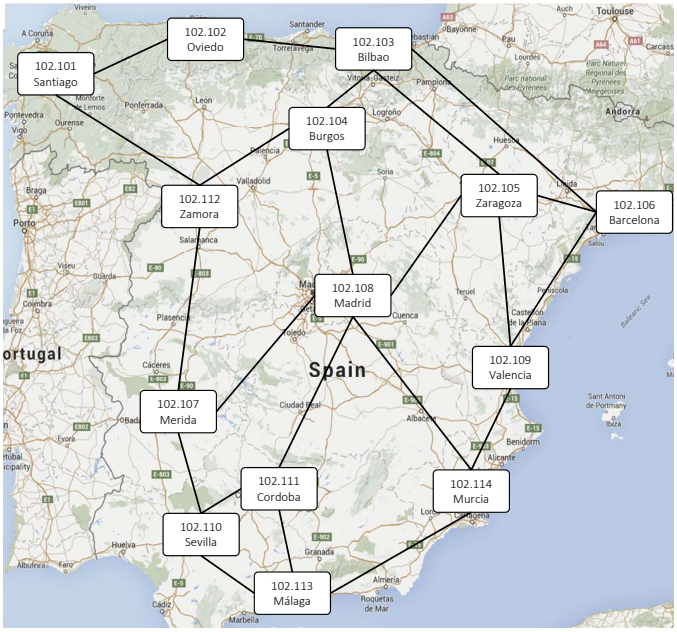


Figure 7. Spanish topology with 14 core nodes and 22 links (client nodes not drawn). Each link has 128 NCFs.

V. Experimental performance evaluation

To carry out the performance evaluation, illustrate the main procedures and obtain some meaningful performance indicators, we proceed with different experiments, instantiating up to two replicas (details of the replicas can be seen from the OpenStack Horizon web interface, Figure 8).

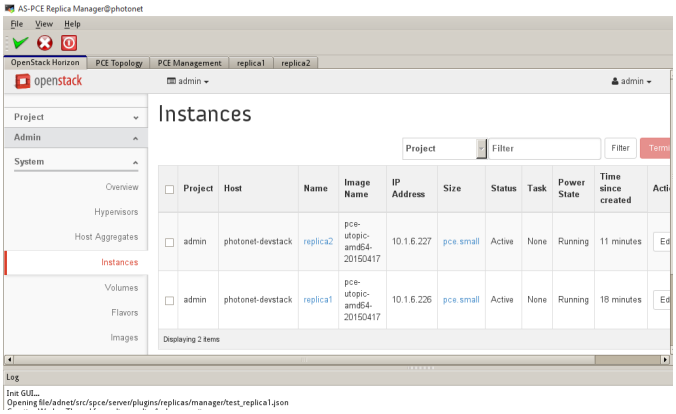


Figure 8. Replica Manager GUI embedding OpenStack Horizon interface showing the two instantiated PCE replicas in a OpenStack compute node. Replicas have IP addresses 10.1.6.226(replica1) and 10.1.6.227(replica2) and have been instantiated with small flavor (1 Gb RAM, 20 Gb HDD)

A first quantitative result involves the time it takes to instantiate a Virtual Machine (VM) for an image containing the PCE software. In short, the latency for instantiating a replica depends on several factors. First, the capabilities of hosting nodes (compute nodes for OpenStack) which can be quite diverse in terms of processing power and memory, including the fact whether the CPU has instructions supporting virtualization. Second, the parameters associated to the VM request itself, such as the VM image size (commonly a qcow file) and the requested memory and CPU for the VM. As a guideline, with PCE software running on an Ubuntu GNU/Linux OS below 3 GB, a given PCE replica is typically operative in between 10-60s, measured since the use of the REST interface to allocate VMs, until the replica manager is able to retrieve the IP address allocated to the replica by actively polling for its state.

A second performance indicator is strongly tied to the initial synchronization. Even if a given replica can be instantiated when there are no active LSPs (empty LSPDB), the initial synchronization of the TED will always be required. In this case, it is also dependent on the actual TCP implementation (the BGP-LS protocol is implemented over TCP) and different options (MTU, loss rate) that define the TCP application throughput. In our specific case, where components run in a dedicated LAN, the initial TED synchronization between the reflector and replica 1 (address 10.1.6.226) is carried out in a few seconds (1.15s in the iteration for which we show the Wireshark capture in Figure 9). This includes not only the Update messages (which can be packed in one or multiple TCP segments) but also the BGP handshake (including the Open and KeepAlive messages).

Next, we proceed with experiments varying the offered traffic load, requesting LSP connections follow-

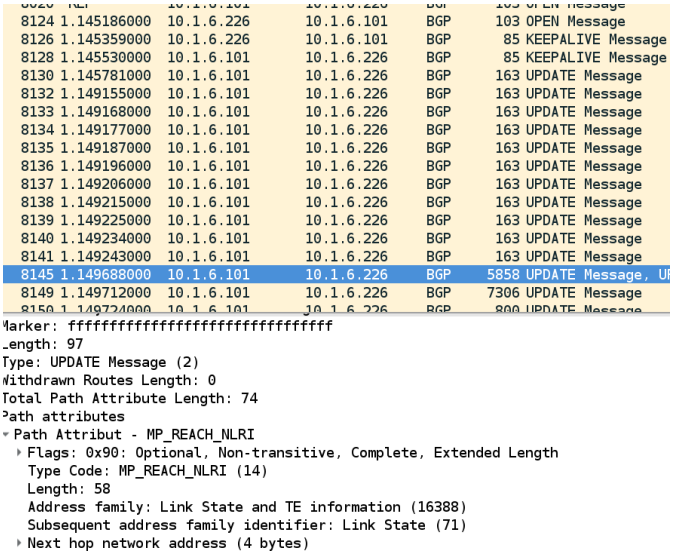


Figure 9. Wireshark capture of the initial synchronization of the TED, showing the different BGP-LS Update messages. Several Update messages can be included in a single TCP segment and the initial synchronization between the reflector and a replica takes 1.15, including the handshake

ing a Poisson arrival process with inter-arrival time set to negative exponential of average 3s and varying holding time depending on the desired Traffic Load. Other relevant parameters for the connections include the random selection of source and destination endpoints and uniformly chosen amongst distinct transceiver pairs (from the set of client facing interfaces). Each connection requested client data rate (bandwidth parameter) is selected randomly between 100, 200 to 500 Gbps, each PCE performing routing and spectrum assignment (RSA) allocating the required optical frequency slot parameters (see [**?**]). For the dynamic setup and release of connections, the average provisioning time, as seen from the NMS that performs the request, is 155ms, with values ranging from a minimum of 89ms to a maximum of 360ms.

At a given time, we instantiate a second PCE replica, measuring the time it takes to synchronize databases. The sync time roughly increases with the number of active LSPs, up to approximately 2.05s which is obtained at 30 Erlangs. Macroscopically, it is easy to see that this LSPDB latency will be determined by the maximum number of LSPs that can be active at a given time. In our specific case, it was easy to give this maximum since we have a limited number of usable transceivers (14), which is limiting the number of active LSPs. Figure 10 shows the LSPDB of the replica at a given time.

Finally, a new experimentation removing the transceiver limitation (just provisioning flexi-grid media channels, without interacting with transceivers) is run, theoretically having a larger potential number of concurrently active LSPs and allowing us to better measure control plane overhead. In this case, we were just dealing with requests for optical spectrum, requesting values for m=1..5 resulting in m * 12.5 GHz.

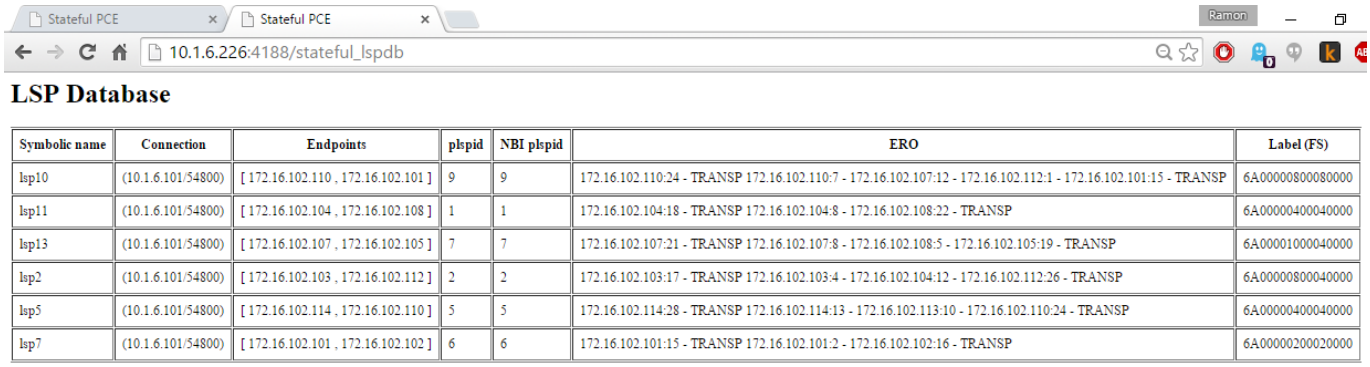
The main parameters that impact control plane overhead in a replication enabled scenario are, a priori, i) the number of active replicas, ii) the redundancy in terms of reflectors and iii) the traffic arrival rate. The first one means e.g. that a given reflector will need to forward as many copies of a topology or LSPDB update to as many replicas, in order to keep synchronization between replicas. The second factor, the number of reflectors (in case a reflector fails), when deployed in simple yet inefficient approaches will also in increase the number of individual messages linearly since the reflectors will forward copies that may have already been received by each replica. Finally, the traffic pattern itself will determine the arrival and departures of LSPs (thus generating PCRpts accordingly and topology changes for at least the number of traversed links that change state.

To provide some numerical values, with 100 Erlangs, the replica 2 initial synchronization of approximately 90 LSPs happened in around 2.8s, with 39 captured packets with average packet size 845 bytes, thus requiring a throughput of 0.113 Mbit/s. Note that, in practice, synchronization delay is not necessarily linear with the number of active LSPs since the Linux kernel is able to pack multiple PCEP PCRpt and BGP-LS messages into a single TCP segment. As a main guideline, in dynamic operation close to expected production systems and real operation, the main factor to control plane overhead will be synchronizing the TED, since a single LSP generates multiple OSPF-TE LSAs (per each crossed link) that are mapped into a BGP-LS Update messages (in our measurements, of around 248-348 bytes). In the case that this presents a scalability problem, it can be mitigated applying thresholds and policies, at the expenses of slightly outdated TED.

Finally, another experiment is set up to stress the system: we deploy two replicas, wait until the system has converged after the initial TED synchronization and launch sequentially 100 LSPs (a new LSP is set up when the previous one has been acknowledged as established) monitoring the real time synchronization with the two replicas. In total, the LSPs are setup and the TED/LSPDBs are synchronized to the new state in less than 12 seconds, requiring on average 0.49 Mbit/s, as seen from the reflector (see Figure 11).

VI. Conclusions

The successful deployment of centralized control plane functions (SDN controllers or specific functions such as a PCE) is constrained by stringent requirements regarding not only dynamicity, performance and cost efficiency but also high-availability, robustness and fault tolerance. The ultimate adoption of this technology, by carriers and operators, is conditioned



Symbolic name	Connection	Endpoints	plspid	NBI plspid	ERO	Label (FS)
lsp10	(10.1.6.101/54800)	[172.16.102.110 , 172.16.102.101]	9	9	172.16.102.110:24 - TRANSP 172.16.102.110:7 - 172.16.102.107:12 - 172.16.102.112:1 - 172.16.102.101:15 - TRANSP	6A00000800080000
lsp11	(10.1.6.101/54800)	[172.16.102.104 , 172.16.102.108]	1	1	172.16.102.104:18 - TRANSP 172.16.102.104:8 - 172.16.102.108:22 - TRANSP	6A00000400040000
lsp13	(10.1.6.101/54800)	[172.16.102.107 , 172.16.102.105]	7	7	172.16.102.107:21 - TRANSP 172.16.102.107:8 - 172.16.102.108:5 - 172.16.102.105:19 - TRANSP	6A00001000040000
lsp2	(10.1.6.101/54800)	[172.16.102.103 , 172.16.102.112]	2	2	172.16.102.103:17 - TRANSP 172.16.102.103:4 - 172.16.102.104:12 - 172.16.102.112:26 - TRANSP	6A00000800040000
lsp5	(10.1.6.101/54800)	[172.16.102.114 , 172.16.102.110]	5	5	172.16.102.114:28 - TRANSP 172.16.102.114:13 - 172.16.102.113:10 - 172.16.102.110:24 - TRANSP	6A00000400040000
lsp7	(10.1.6.101/54800)	[172.16.102.101 , 172.16.102.102]	6	6	172.16.102.101:15 - TRANSP 172.16.102.101:2 - 172.16.102.102:16 - TRANSP	6A00000200020000

Figure 10. Capture of the LSPDB at a given time for replica at 10.1.6.226. For each LSP we see the endpoints, Explicit Route Object (ERO) including opaque transponder objects and the allocated label (which conveys frequency slot center frequency and width, n and m parameters).

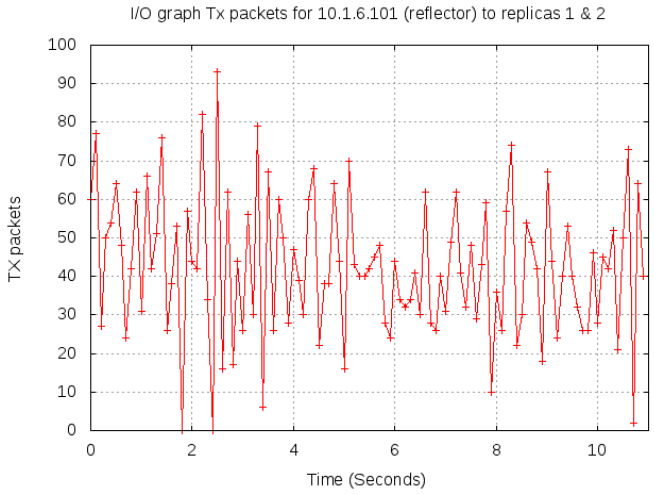


Figure 11. Packet I/O: Wireshark capture of transmitted IP packets as seen by the reflector at 10.1.6.101, for two replicas and stressing the system with a batch of 100 consecutive LSP requests.

by the availability of ”carrier class” solutions and infrastructures that meet such requirements while still delivering the benefits associated to SDN. In this scope, the use of transport-NFV concepts can fulfill such requirements, enabling much wanted features such as in-operation modifications, software image or policy upgrades and hot-swapping.

While the concept and use of functional replication for high-availability is quite well understood, the need for synchronization between databases is an issue to solve. On the one hand, the associated information and data models need to be clearly defined and, on the other hand, deployed solutions should not be exposed to vendor lock-in or proprietary products, for it should be possible to use implementations from different vendors and open solutions. We proposed an architecture and the use of existing open and standard PCEP and BGP-LS protocols for the synchronization of the main considered databases, namely, the traffic engineering (network topology) one and the LSP database (keeping state of active connections) thus avoiding the aforementioned vendor lock-in.

The main performance considerations are related to the synchronization delays and control plane overhead. This performance indicators need to be addressed keeping in mind the initial assumptions related to i) the dynamicity and associated timescales of traffic, which is the main source of database changes and ii) the availability of a deployed and dedicated control plane and management network in which control plane links have fairly consistent bandwidth and processing/transmission delays, along with the ability to deploy operator private clouds for the deployment of internal NFV services. Our experimental tests show that synchronization between replicas is of the order of a few seconds for the initial sync, and the order of milliseconds for subsequent updates, with reasonable control plane overhead for the targeted deployment scenarios. Further work is still needed in heavily constrained scenarios in which the data communications network that supports the control plane may limit performance.

Acknowledgments

This work has been partially funded by the European Commission H2020 project 5G-Crosshaul project under grant agreement H2020-671598 as well as by the Spanish Ministry of Economy and Competitiveness (MINECO) project DESTELLO (TEC2015-69256-R).

References

[1] R. Casellas, R. Vilalta, R. Martinez, and R. Muñoz, “Active Stateful PCE high-availability for the control of Flexigrid Networks with Network Function Virtualization enabled replication,” in *in Proc. of Optical Fiber Communication Conf. and Expo. (OFC)*, Anaheim, California, March 2016.

[2] A. Farrel, J.-P. Vasseur, and J. Ash, “A Path Computation Element (PCE)-Based Architecture,” RFC 4655 (Informational), Internet Engineering Task Force, Aug. 2006. [Online]. Available: http://www.ietf.org/rfc/rfc4655.txt

[3] J. Vasseur and J. L. Roux, "Path Computation Element (PCE) Communication Protocol (PCEP)," RFC 5440 (Proposed Standard), Internet Engineering Task Force, Mar. 2009. [Online]. Available: http://www.ietf.org/rfc/rfc5440.txt
[4] J. Vasseur, R. Zhang, N. Bitar, and J. L. Roux, "A Backward-Recursive PCE-Based Computation (BRPC) Procedure to Compute Shortest Constrained Inter-Domain Traffic Engineering Label Switched Paths," RFC 5441 (Proposed Standard), Internet Engineering Task Force, Apr. 2009. [Online]. Available: http://www.ietf.org/rfc/rfc5441.txt
[5] D. King and A. Farrel, "A PCE-based Architecture for Application-based Network Operations," IETF RFC 7491, 2015.
[6] E. Crabbe, I. Minei, and S. S. R. Varga, "PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model," Internet Engineering Task Force, October 2015.
[7] S. Sivabalan, J. Medved, C. Filsfils, R. Raszuk, V. Lopez, J. Tantsura, W. Henderickx, and J. Hardwick, "PCEP Extensions for Segment Routing," Internet Engineering Task Force, March 2016.
[8] Q. Z. et al, "The Use Cases for Using PCE as the Central Controller(PCECC) of LSPs," Internet Engineering Task Force, March 2017.
[9] ETSI Group Specification, "Network function virtualization (nfv): Architectural framework," *ETSI GS NFV 002 v.1.1.1,*, 2013.
[10] R. Vilalta et al., "Transport PCE Network Function Virtualization," in *in Proc. of ECOC2014*, Cannes, France, Sept 2014.
[11] R. Muñoz et al., "Integrated SDN/NFV management and orchestration architecture for dynamic deployment of virtual SDN control instances for virtual tenant networks," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 7, no. 11, November 2015.
[12] H. Gredler, J. Medved, S. Previdi, A. Farrel, and S. Ray, "North-Bound Distribution of Link-State and TE Information using BGP," Internet Engineering Task Force, May 2014.
[13] ETSI Group Specification, "Network functions virtualisation (nfv); infrastructure overview," *ETSI GS NFV-INF 001 v.1.1.1,*, 2015.
[14] Open Networking Foundation (ONF), "OpenFlow Switch Specification, version 1.4 (Wire protocol 0x5)," Open Networking Foundation, October 2013. [Online]. Available: https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-spec-v1.4.0.pdf
[15] R. Casellas, R. Martínez, R. Muñoz, R. Vilalta, L. Liu, T. Tsuritani, and I. Morita, "Control and management of flexi-grid optical networks with an integrated stateful pce and openflow controller," *Journal of Optical Communications and Networking, doi: 10.1364/JOCN.5.000A57*, vol. 5, no. 10, 2013.
[16] T. Bates, et al., "BGP Route Reflection - An Alternative to Full Mesh IBGP," RFC 4456, Internet Engineering Task Force, Apr. 2006. [Online]. Available: http://www.ietf.org/rfc/rfc4456.txt
[17] O. González de Dios, R. Casellas, R. Morro, F. Paolucci, V. López, R. Martínez, R. Muñoz, R. Vilalta, and P. Castoldi, "Multipartner Demonstration of BGP-LS-Enabled Multidomain EON Control and Instantiation With H-PCE [Invited]," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 7, no. 12, December 2015.
[18] O. González de Dios, R. Casellas, F. Paolucci, A. Napoli, L. Gifre, A. Dupas, E. Hugues-Salas, R. Morro, S. Belotti, G. Meloni, T. Rahman, V. López, R. Martínez, F. Fresi, M. Bohn, S. Yan, L. Velasco, P. Layec, J.-P. Fernández Palacios, "Experimental Demonstration of Multivendor and Multidomain EON With Data and Control Interoperability Over a Pan-European Test Bed," *IEEE/OSA Journal of Lightwave Technologies (JLT)*, vol. 7, no. 34, April 2016.
[19] R. Martínez, R. Casellas, R. Vilalta, and R. Muñoz, "Gmpls/pce-controlled multi-flow optical transponders in elastic optical networks," *Journal of Optical Communications and Networking*, vol. 7, no. 11, November 2015.

Ramon Casellas (IEEE SM'12) graduated in telecommunications engineering in 1999 by both the UPC-BarcelonaTech university and ENST Telecom Paristech, within an Erasmus/Socrates double degree program. After working as an undergraduate researcher at both France Telecom R&D and British Telecom Labs, he completed a Ph.D. degree in 2002. He worked as an associate professor at the networks and computer science department of the ENST (Paris) and joined the CTTC Optical Networking Area in 2006, with a Torres Quevedo research grant. He currently is a senior research associate and the coordinator of the ADRENALINE testbed. He has been involved in several international R&D and technology transfer projects. His research interests include network control and management, the GMPLS/PCE architecture and protocols, software defined networking and traffic engineering mechanisms. He contributes to IETF standardization within the CCAMP and PCE working groups. He is a member of the IEEE Communications Society and a member of the Internet Society.

Ricard Vilalta graduated in telecommunications engineering in 2007 and received a Ph.D. degree in telecommunications in 2013, both from the Universitat Politècnica de Catalunya (UPC), Spain. He also has studied Audiovisual Communication at UOC (Open University of Catalonia) and is a master degree on Technology-based business innovation and administration at Barcelona University (UB). Since 2010, Ricard Vilalta is a researcher at CTTC, in the Optical Networks and Systems Department. His research is focussed on Optical Network Virtualization and Optical Openflow. He is currently a Research Associate at Open Networking Foundation.

Ricardo Martínez (IEEE SM'14) graduated and PhD in telecommunications engineering by the UPC-BarcelonaTech university in 2002 and 2007, respectively. He has been actively involved in several public-funded (national and EU) R&D as well as industrial technology transfer projects. Since 2013, he is Senior Researcher of the Communication Networks Division (CND) at CTTC. His research interests include control and network management architectures, protocols and traffic engineering mechanisms for next-generation packet and optical transport networks within aggregation/metro and core segments.

Raül Muñoz (IEEE SM'12) graduated in telecommunications engineering in 2001 and received a Ph.D. degree in telecommunications in 2005, both from the Universitat Politècnica de Catalunya (UPC), Spain. After working as undergraduate researcher at Telecom Italia Lab (Turin, Italy) in 2000, and as assistant professor at the UPC in 2001, he joined the Centre Tecnològic de Telecomunicacions de Catalunya (CTTC) in 2002. Currently, he is Senior Researcher, Head of the Optical Network and System Department, and Manager of the Communication Networks Division. Since 2000, he has participated in several R&D projects funded by the European Commission's Framework Programmes (FP7 FP6 and FP5) and the Spanish Ministries, as well as technology transfer projects. He has led several Spanish research projects, and currently leads the European Consortium of the EU-Japan project STRAUSS. His research interests include control and management architectures, protocols and traffic engineering algorithms for future optical transport networks.