Workgroup: Network Management Research Group
Internet-Draft:
draft-paillisse-nmrg-performance-digital-twin-00
Published: 24 October 2022
Intended Status: Informational
Expires: 27 April 2023

Authors: J. Paillisse UPC-BarcelonaTech
P. Almasan UPC-BarcelonaTech
M. Ferriol UPC-BarcelonaTech
P. Barlet UPC-BarcelonaTech
A. Cabellos UPC-BarcelonaTech
S. Xiao Huawei
X. Shi Huawei
X. Cheng Huawei
C. Janz Huawei
A. Guo Futurewei
D. Perino Telefonica I+D
D. Lopez Telefonica I+D
A. Pastor Telefonica I+D

Performance-Oriented Digital Twins for Packet and Optical Networks

Abstract

This draft introduces the concept of a Network Digital Twin (NDT) for performance evaluation, a so-called Performance-Oriented Digital Twin (PODT). Two types of PODTs are described. The first, referred to as a Network Performance Digital Twin (NPDT), produces performance estimates (delay, jitter, loss) for a packet network with a specified topology, traffic demand, and routing and scheduling configuration. The second, referred to as an Optical Performance Digital Twin (OPDT), produces transmission performance estimates of an optical network with specified optical service topologies and network equipment types, topology and status. This draft also discusses interfaces to these digital twins, how these digital twins relate to existing control plane elements, and describes use cases for these digital twins, as well as possible implementation options.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

A Digital Twin for computer networks is a virtual replica of an existing network with a behavior equivalent to that of the real one. The key advantage of a Network Digital Twin (NDT) is the ability to recreate the complexities and particularities of the network infrastructure without the deployment cost of a real network. Hence, network administrators can test, deploy and modify network configurations safely, without worrying about the impact on the real network. Once the administrator has found a configuration that fulfills the expected objectives, it can be deployed to the real network. The information provided by the NDT can also be used as part of a closed loop-based automated process. In addition, using a NDT is faster, safer and more cost-effective than interacting with the physical network. All these characteristics make NDT useful for different network management tasks ranging from network planning or troubleshooting to optimization.

The concept of a NDT has been proposed for different approaches: network management [I-D.draft-zhou-nmrg-digitaltwin-network-concepts], 5G networks [digital-twin-5G], Vehicular networks [digital-twin-vanets], artificial intelligence [digital-twin-AI], or Industry 4.0 [digital-twin-industry], among others.

This draft proposes Digital Twins with a focus on performance evaluation, for use in network management, network operations optimization, network operations automation and other contexts. Performance evaluation is examined in two contexts: with respect to a packet network, and with respect to an optical transmission network. In the packet network case, given several input parameters (topology, traffic matrix, etc.), a Network Performance Digital Twin (NPDT) predicts network performance metrics such as delay (per path or per link), jitter, or loss. In the optical transmission network case, given specified optical service topologies and network equipment types, topology and status, an Optical Performance Digital Twin (OPDT) estimates transmission performance metrics, such as optical channel terminal powers and margins, in the face of channel transmission noise and impairments [OPDT].

This draft defines the inputs and outputs of both types of Performance-Oriented Digital Twins, their associated interfaces to other modules in the network management or control plane or to network equipment and components, and also discusses use cases.

This draft further discusses possible implementation options for the NPDT, with a special emphasis on those based on Machine Learning. The aim of Section 7 (Implementation Challenges) is, in part, to describe the advantages and limitations of these techniques. For example, most Machine Learning technologies rely heavily on large amounts of data to achieve acceptable accuracy. Other considerations include adjusting the architecture of the Neural Network to successfully understand the structure of the input data. Challenges particular to OPDT implementation are also discussed.

In order to use a Performance-Oriented Digital Twin (PODT) in practical scenarios (c.f. Section 6), such as network optimization, it should meet certain requirements:

Fast: low delay when making predictions (in the order of milliseconds) to use it in optimization scenarios that need to test a large number of configuration variables (c.f. Section 6.2).

Accurate: the error of the prediction (vs the ground truth) has to be below a certain threshold to be deployable in real-world networks.

Scalable: support networks of arbitrarily large topologies

Variety of Inputs: accept a wide range of combinations of input variables, depending on network type, packet or optical:

*Routing configurations

*Scheduling configurations (FIFO, Weighted Fair Queueing, Deficit Round Robin, etc)

*Topologies

*Traffic matrices

*Traffic models (constant bitrate, Poisson, ON/OFF, etc)

*Optical network and/or service topologies and parameters

Accessible: despite the internal architecture of the PODT, it needs to be easy to use for network engineers and administrators. This includes, but is not limited to: interfaces to communicate with PODT that ideally are well-known in the networking community, metrics that are readily understood by network engineers, or confidence values of the estimations.

Note that the inputs and outputs described here are an example, but other inputs and outputs are possible depending on the specificities of each scenario.

2. Terminology

Digital Twin (DT):
A virtual replica of a physical system that supports accurate prediction of selected behaviours of the physical system.

Network Digital Twin (NDT):
A virtual replica a physical network that supports accurate prediction of selected behaviours of the physical network.

Network Performance Digital Twin (NPDT):
A NDT that can predict with accuracy several performance metrics of a physical packet network.

Optical Performance Digital Twin (OPDT):
A NDT that can predict with accuracy several transmission-related performance metrics of a physical optical transmission network.

Performance-Oriented Digital Twin (PODT):
A generic term for a NDT whose function is to accurately predict particular performance-oriented behaviours of a physical network. Two flavours of PODTs are considered in this draft: NPDTs and OPDTs.

Network Optimizer:
An algorithm capable of finding the optimal configuration parameters of a network, e.g. OSPF weights (packet network) or route, spectrum, modulation and coding assignments for optical transmission channels (optical network), given an optimization objective, e.g. latency below a certain threshold (packet network) or lowest aggregate spectral use (optical network).

Control Plane:
Any system, hardware or software, centralized or decentralized, in charge of controlling and managing a physical network. Examples are routing protocols, SDN controllers, etc. Some or all

of these functions may also be encompassed by what is referred to as a Management Plane.

3. Generic Architecture of a Performance-Oriented Digital Twin

Figure 1 presents an overview of the generic architecture of a Performance-Oriented Digital Twin (PODT).

```
                     |Administrator Interfaces,
                     |Service Demand Interfaces,
                     |Intent-Based Interfaces,
                     |Associated Application Interfaces, etc.
                     |
 +-------------------------------------------+
 |                                           |
 |                                           |
 |                                           |         +-------------+
 |                                           |   DT    |             |
 |                                           |Interface| Performance |
 |               Management Plane            |<------->|   Oriented  |
 |                                           |         |   Digital   |
 |            + Embedded Applications        |-------->|     Twin    |
 |                                           | Network |             |
 |                                           | Inform. +-------------+
 |                                           |Interface
 |                                           |
 +-------------------------------------------+
                  |                  |
     Measurement  |                  |  Configuration
       Interface  |                  |  Interface
                  |                  |
         +--------------------------------------+
         |                                      |
         |          Physical Network            |
         |                                      |
         +--------------------------------------+
```

Figure 1: Generic global architecture of a Performance-Oriented Digital Twin (PODT).

Individual elements are discussed further in following sections of this draft. But overall, the PODT represented in Figure 1 is an information-oriented "utility". It works in concert with a Management Plane and the latter's "embedded" applications or components (i.e. applications or components comprising a part of the Management Plane implementation), as well as any "associated" applications (i.e. applications working in interaction with the Management Plane implementation).

The essential function of the PODT is to estimate - and then to present at the Digital Twin (DT) interface - particular, targeted performance-oriented behaviours of the physical network, assessed in scenarios that are defined by information presented at the DT Interface. Optionally, complementary information used by the behavioural models may be provided by the Network Information Interface. The DT Interface is a sort of "run-time" interface while the Network Information Interface, if present, serves as a conduit for e.g. measurement data streamed from the physical network to the management plane, or network or service topology information generated within the management plane itself. As a rule, the Network Information Interface, if present, provides to the NDT the broad set of changing information necessary to construct an accurate and up-to-date replica of the physical network for behavioural modeling purposes.

The Management Plane may have interfaces to administrator, service demand and/or intent generating systems, etc. It also has configuration or control interfaces to the physical network. Configuration or control inputs do not flow directly from the PODT to the network: rather, the PODT provides information to the Management Plane and the latter's embedded and associated applications, components and processes - potentially including closed loop-based automated processes - may generate network-facing configuration or control outputs. The information provided by the PODT may thus be used in evaluation, decision, etc. procedures that serve to optimize operational outcomes, whether or not such procedures form part of closed loop-based automated processes. The particular nature of such operations, decisions etc. of PODT outputs is what defines and distinguishes various use cases. Examples of use cases are considered in Section 6.

3.1. Architecture of the Network Performance Digital Twin

Figure 2 presents an overview of the architecture of a Network Performance Digital Twin (NPDT).

```
       Administrator Intent
                    |
                    |
                    |Intent-Based Interface
                    |
                    |
+-------------+-----------------------------+
|             |     |                       |
|             |   Intent-Based   Optimizer  |
|             |   Renderer                  |         +-------------+
|             |                             |   DTI   |   Network   |
| Management  |                             |Interface| Performance |
| Plane       |                             |<------->|   Digital   |
|             |                             |         |    Twin     |
|             |                             |         |             |
|             |   Measure        Configure  |         +-------------+
|             |  |                  |       |
+-------------+-----------------------------+
                 |                  |
                 |                  |
    Measurement  |                  |  Configuration
      Interface  |                  |  Interface
                 |                  |
        +--------------------------------------+
        |                                      |
        |       Physical Packet Network        |
        |                                      |
        +--------------------------------------+
```

Figure 2: Global architecture of the Network Performance DT

Each element is defined as:

Network Performance Digital Twin (NPDT): a system capable of generating performance estimates of a specific instance of a packet network.

Physical Network: a real-world network that can be configured via standard interfaces.

Management Plane: The set of hardware and software elements in charge of controlling the Physical Network. This ranges from routing processes, optimization algorithms, network controllers, visibility platforms, etc. The definition, organization and implementation of the elements within the management plane is outside of the scope of this document. In what follows, some elements of the management plane that are relevant to this document are described.

- *Optimizer: a network optimizer that can tune the configuration parameters of a network given one or more optimization objectives, e.g. do not exceed a latency threshold in all paths, minimize the load of the most used link, and avoid more than 10 Gbps of traffic at router R4 [DEFO].

- *Intent-Based Renderer: a system capable of understanding network intent, according to the definitions in [irtf-nmrg-ibn-concepts-definitions-09].

- *Measure: any system to measure the status and performance of a network, e.g. Netflow [RFC3954], streaming telemetry [streaming-telemetry], etc.

- *Configure: any system to apply configuration settings to the network devices, e.g. a NETCONF Manager or an end-to-end system to manage device configuration files [facebook-config].

And the functions of each interface are:

DT Interface (DTI): an interface to communicate with the Network Performance Digital Twin (NPDT). Inputs to the NPDT are a description of the network (topology, routing configuration, etc), and the outputs are performance metrics (delay, jitter, loss, c.f. Section 4).

Configuration Interface (CI): a standard interface to configure the physical network, such as NETCONF [RFC6241], YANG, OpenFlow [OFspec], LISP [RFC6830], etc.

Measurement Interface (MI): a standard interface to collect network status information, such as Netflow [RFC3954], SNMP, streaming telemetry [openconfig-rtgwg-gnmi-spec-01], etc.

Intent-Based Interface (IBI): an interface for the network administrator to define optimization objectives or run the NPDT to obtain performance estimates, among others.

3.2. Architecture of the Optical Performance Digital Twin

Figure 3 presents an overview of the Optical Performance Digital Twin (OPDT).

```
                    |Service Demand Interfaces,
                    |Intent-Based Interface,
                    |Associated Application Interfaces, etc.
                    |
+-------------------------------------------+
|                                           |
|
|                                           |   DT    +-------------+
|            Management Plane               |Interface|             |
|                                           |<------->|   Optical   |
|          + Embedded Applications          |         | Performance |
|                                           |         |   Digital   |
|                                           |-------->|    Twin     |
|                                           | Network |             |
|                                           | Inform. +-------------+
|                                           |Interface
+-------------------------------------------+
                 |                  |
                 |                  |
    Measurement  |                  |  Configuration
      Interface  |                  |  Interface
                 |                  |
        +--------------------------------------+
        |                                      |
        |      Physical Optcial Network        |
        |                                      |
        +--------------------------------------+
```

Figure 3: Global architecture of an Optical Performance Digital Twin

The elements shown are defined as follows:

Optical Performance Digital Twin (OPDT): A NDT that can predict with accuracy several transmission-related performance metrics of a physical optical transmission network.

Physical Network: a real-world optical transmission network that can be configured - and on which optical services may be configured and subsequently provisioned - via standard or non-standard interfaces.

Management Plane: The set of hardware and software elements and processes in charge of managing and controlling the optical physical network. This ranges from core processes such as optical service route, spectrum and other parametric computation and allocation, to embedded supporting applications and components such as optimization algorithms, network controllers, visibility platforms, etc. The definition, organization and implementation of the elements within the optical network management plane is outside of the scope of this document; however, associated applications are referred to in Section 6 dealing with use cases.

The functions of the respective interfaces are:

DT Interface (DTI): an interface to communicate with the Optical Performance Digital Twin (OPDT). This is a "run-time" interface whose inputs to the OPDT specify the scenario under which optical service transmission performance is to be evaluated, and whose outputs from the OPDT comprise key transmission performance metrics for the set of optical services present in that scenario, such as service terminal powers and noise margins. The particular scenario-defining inputs reflected in a given OPDT instance may be a function of particular use case requirements. Further, the information provided by these inputs may or may not replace or complement information presented to the Network Information Interface, which reflects in detail the actual configuration and status of network, services, equipment, performance, etc.

Network Information Interface (NII): an interface to communicate with the Optical Performance Digital Twin (OPDT). This is a unidirectional interface whose inputs to the OPDT reflect current attributes of the optical network and services configured and operating on it, such as network topology, equipment types and status, optical service topology, performance and other attributes, instrumentation-generated measurement data, etc. The source of this information may be the physical network itself, in which case the information flows first to the Management Plane over the so-called Measurement Interface; or, the source of the information may be the Management Plane itself. Information related to models used in performance analysis may also be transferred over this interface.

The combination of the NII and a flexibly-defined DTI, enables optical transmission performance to be assessed in respect of any scenario, ranging from: wholly defined by the actual (or, some historical) state and status of the physical network and services; to, an entirely hypothetical network and service state and status; or, to any scenario between these extremes. This enables the PODT to be used in a wide variety of use cases, which are discussed in detail in Section 6.

Configuration Interface (CI): an interface to configure the physical optical network and the optical services deployed on it. This interface may or may not be standards-based.

Measurement Interface (MI): an interface or set of interfaces to collect network and service status and other information, including device status, service performance, instrumentation data, etc. Information related to models used in performance analysis may also be transferred over this interface. This (these) interface(s) may or may not be either partly or wholly standards-based.

Service Demand Interface: a standards-based interface used to provide optical service requirements to the Management Plane.

Intent-Based Interface: an (ideally) standards-based interface used to specify, to the Management Plane, optical service requirements and/or other attributes or constraints related to service delivery or network operation.

Associated Application Interfaces: interfaces connecting the Management Plane to externally-implemented applications, such as network planning tools or other. These interfaces may or may not be standards-based.

4. Interfaces

4.1. Network Performance Digital Twin

4.1.1. Administrator

This interface can be a simple CLI or a state-of-the-art GUI, depending on the final product. In summary, it has to offer the network administrator the following options/features:

*Predict the performance of one or more network scenarios, defined by the administrator. Several use-cases related to this option are detailed in Section 6.

*Define network optimization objectives and run the network optimizer.

*Apply the optimized configuration to the physical network.

4.1.2. Configuration Interface

This interface is used to configure the Physical Network with the configuration parameters obtained from the optimizer. It can be composed of one or more IETF protocols for network configuration, a non-exhaustive list is: NETCONF [RFC6241], RESTCONF/YANG [RFC8040], PCE [RFC4655], OVSDB [RFC7047], or LISP [RFC6830]. It is also possible to use other standards defined outside the IETF that allow the configuration of elements in the forwarding plane, e.g. OpenFlow [OFspec] or P4 Runtime [P4Rspec].

4.1.3. Digital Twin Interface (DTI)

This interface can be defined with any widespread data format, such as CSV files or JSON objects. There are two groups of data. We are assuming a network with N nodes.

Inputs: data sent to the NPDT to calculate the performance estimates:

- *Topology: description of the network topology in graph format, eg. NetworkX [NetworkXlib].

- *Routing configuration: a matrix of size N*N. Each cell contains the path from source N(i) to destination N(j) as a series of nodes of the topology. Note that not all source-destination pairs may have a path. Since the NPDT only needs a sequence of nodes to define a route, it supports different routing protocols, from OSPF, IS-IS or BGP, to SRv6, LISP, etc.

- *Traffic Demands: a definition of the traffic that is injected into the network. It can be specified with different granularities, ranging from a list of 5-tuple flows and their associated traffic intensity, to a N*N matrix defining the traffic intensity for each source-destination pair. Some source-destination pairs may have zero traffic intensity. The traffic intensity defines parameters of the traffic: bits per second, number of packets, average packet size, etc.

- *Traffic Model: the statistical properties of the input traffic, e.g. Video on Demand, backup, VoIP traffic, etc. It can be defined globally for the whole network or individually for each flow in the Traffic Demands.

- *Scheduling configuration: attributes associated to the nodes of the topology graph describing the scheduling configuration of the network, that is (1) scheduling policy (e.g. FIFO, WFQ, DRR, etc), and (2) number of queues per output port.

Outputs: performance estimates of the NPDT: three matrices of size N*N containing the delay, jitter and loss for all the paths in the input topology.

Note that this is an example of the inputs/outputs of a performance NPDT, but other inputs and outputs are possible depending on the specificities of each scenario.

4.2. Optical Performance Digital Twin

Per Figure 3 in Section 3.2, an Optical Performance Digital Twin (OPDT) interfaces with a Management Plane to obtain the information related to the physical network and the scenario for which optical service performance information is sought. Again, such information is either sent to the OPDT at scenario assessment run-time through the DT Interface (DTI), or is made available to the OPDT through the Network Information Interface (NII) as stable (if evolving) configuration, state, instrumentation and other data. As discussed in Section 6, the best partitioning of information presentation between DTI and NII is to some degree use case-specific. Categories of such information include:

*Traffic-engineered (TE) topology and physical network topology configuration, which includes customized TE topologies, TE policies, profiles, and administrative routing constraints, etc.

*Operational state of the TE topology and network topology, which contains critical information such as spectrum allocation status and optical impairment characteristics of the optical components, such as the modulation, error correction capabilities, launching power and receiving power margins of the optical transponders.

*Wavelength-based optical service configuration, which includes descriptions about the source, destination, path, protection and restoration configurations, and other possible routing and administrative configurations associated with the optical services.

*Optical status of all the optical services within the network.

*Device configurations to various components such as fibers, amplifiers, wavelength add-drop switches, transceivers, etc.

*Network and device level telemetry including alarm and performance monitoring (PM) and instrumentation data.

*Historical configuration, state, telemetry and other data per the preceding. This allows the OPDT to accommodate scenarios that reflect network and service circumstances and status at prior times. This is useful or necessary in some use cases.

A good part of the interface specifications needed to support these information categories are available today in the IETF. For example, the ACTN framework defines a hierarchical control framework, which coupled with the various models defined in the TEAS, CCAMP, OPSAWG and other working groups, can provide the configuration and state data related to TE topology and services. The following is at least a partial list of available models applicable to the DTI and NII interfaces:

- [RFC8345]:A YANG data model for network topologies

- [RFC8795]: YANG data model for traffic-engineering topologies

- [RFC9094]: A YANG data model for wavelength switched optical networks

- [I-D.ietf-ccamp-flexigrid-yang]: YANG data model for flexi-grid optical networks

- [I-D.ietf-ccamp-optical-impairment-topology-yang]: YANG data model for optical impairment-aware topology

- [I-D.ietf-teas-yang-te]: YANG model for TE tunnels

- [I-D.ietf-ccamp-wson-tunnel-model]: A Yang data model for WSON tunnel

- [I-D.ietf-ccamp-flexigrid-tunnel-yang]: A YANG data model for Flexi-Grid tunnels

Data models for management configuration and optical device configurations, on the other hand, are mostly not available and need to be developed in the IETF. As a starting point, the following draft could potentially be extended to support OPDT functional requirements:

- [I-D.yg3bp-ccamp-network-inventory-yang]: A YANG data model for Network Hardware Inventory

Management models developed in other standard organizations such as TM Forum and OpenConfig, might also be used by the OPDT. Applicable instrumentation and measurement telemetry models are for further study.

5. Mapping to the Network Digital Twin Architecture

Since the PODT is a type of Network Digital Twin, its elements can be mapped to the reference architecture of a NDT described in [I-D.draft-zhou-nmrg-digitaltwin-network-concepts]. Table 1 maps the elements of the NDT reference architecture to those of the PODT. Note that the Physical Network is the same for both architectures.

NDT Reference Architecture		This draft
Application Layer		e.g. Intent-Based Interface
		e.g. Optimizer
Digital Twin Layer	Management	Management Plane
	Service Mapping Models	Performance-Oriented Digital Twin
	Data Repository	Optional in production deployments
Physical Network	Data Collection	Measurement Interface
	Control	Configuration Interface

Table 1: Mapping of NDT reference architecture elements to the architecture of the Performance-Oriented DT.

6. Use Cases

6.1. Network planning

6.1.1. Packet Network Planning

The size and traffic of networks has doubled every year [network-capacity]. To accommodate this growth in users and network applications, networks need periodical upgrades. For example, ISPs might be willing to increase certain link capacities or add new connections to alleviate the burden on the existing infrastructure. This is typically a cumbersome process that relies on expert knowledge. Furthermore, modern networks are becoming larger and more complex, thus exacerbating the difficulty of existing solutions to scale to larger networks [planning-scalability].

Since the NPDT models large infrastructures and can produce accurate and fast performance estimates, it can help in different tasks related to network capacity and planning:

- Estimating when an existing network will run out of resources, assuming a given growth in users.

- Use performance estimates to plan the optimal upgrade that can cope with user growth. Network operators can leverage the NPDT to make better planning decisions and anticipate network upgrades.

- Find unconventional topologies: in some networking scenarios, especially datacenter networks, some topologies are well-known to offer high performance [Google-Clos]. However, it is also possible to search for new topologies that optimize performance with the help of algorithms. On one hand, the algorithm explores different topologies and, on the other hand, the NPDT provides fast performance estimations to the algorithm. Hence, the NPDT guides the optimization algorithm towards the topologies with better performance [auto-dc-topology].

6.1.2. Optical Network Planning

In an optical service planning exercise, the optical network topology and equipment map are presumed fixed while the set of provisioned optical services may be altered. In an optical network planning exercise, not only the optical service map (or some component of it) but also the network topology and deployed equipment map may be augmented or changed. Optical network planning may thus be viewed as a superset of the steps and processes associated with optical service planning. The goal of optical network planning is usually to accommodate some particular set of services or, more fundamentally, some particular transmitted traffic requirements, using the least amount or least (total or incremental) cost of equipment. Optimization is thus generally a process component of optical network planning; again, this is discussed in Section 6.2.2.

In general, the role of an OPDT in support of optical network planning is the same as the one described in section Section 6.4 supporting optical services planning: to verify that all postulated optical services would operate within acceptable performance bounds, when deployed on a postulated new network topology and detailed equipment and fibre map. As in the optical services planning case, beyond identifying scenarios in which one or more optical services would fail to operate, the identification of undesirably low or unnecessarily high optical service margins could serve as a trigger to explore alternative conjoint optical network and service plans.

For this use case, the appropriate input DT Interface specifies the topology and other characteristics of postulated optical services, and also the postulated new or modified network topology and map of deployed equipment. Specific such postulated scenarios are conceived by the Management Plane-embedded or -associated applications and processes that are responsible for optical network planning. These same applications and processes make use of the performance information provided by the OPDT.

In the case of brown field optical network planning, the physical network "twinned" is partly real and partly hypothetical. In the case of green field planning the physical network is entirely hypothetical. In practice, this means that in optical network planning, the Network Information Interface can supply to the OPDT's behavioural models only part - at best - of the information it would supply in other cases. Such information "gaps" must be filled by other means, e.g. using generic rather than specific equipment- and fibre-characterizing information.

6.2. Network Optimization

6.2.1. Packet Network Optimization

Since the DT can provide performance estimates in short timescales, it is possible to pair it with a network optimizer (Figure 4). The network administrator defines one or more optimization objectives e.g. maximum average delay for all paths in the network. The optimizer can be implemented with a classical optimization algorithm, like Constraint Programming [DEFO], or Local Search [LS], or a Machine-Learning one, such as Deep Neural Networks [DNN-TM], or Multi-Agent Reinforcement Learning [MARL-TE]. Regardless of the implementation, the optimizer tests various configurations to find the network configuration parameters that satisfy the optimization objectives. In order to know the performance of a specific network configuration, the optimizer sends such configuration to the NPDT, that predicts the performance metrics of such configuration.

```
                  +------------+   Candidate        +-------------+
                  |            |   Network Config.  |   Network   |
Optimization----> | Network    |-------------------->| Performance |
objectives        | Optimizer  |                    |   Digital   |
                  |            |<-------------------|    Twin     |
                  +------------+    Estimated       +-------------+
                        |           Performance
                        |
                        |
                        v
          Optimized Network Configuration
```

Figure 4: Using a NPDT as a network model for an optimizer.

An example of optimization use case would be multi-objective optimization scenarios: commonly, the network administrator defines a set of optimization goals that must be concurrently met [DEFO], for example:

*Bound the latency of all links to a maximum.

*Do not exceed a link utilization of 80%, but for only a sub-set of all the links.

*Route all flows of type B through node 10.

*Avoid more than 35 Gbps of traffic to router R5.

- *Minimize the routing cost, that is, the number of flow to re-route [ReRoute-Cost].

6.2.2. Optical Services and Network Optimization

As suggested in Section 6.4 and Section 6.1.2, optimization - finding the "best" solution as determined by some quantitative criterion that is assessed for each candidate solution - is generally an intrinsic component of optical network planning. This is because such planning usually involves trying to find e.g. a lowest total or incremental cost-of-equipment network plan. Where optical services planning considers new service demands in batches or permits re-configuration of some or all existing services, optimization of new and/or modified batches of optical services may be sought as part of the planning solution. Such optimization could involve, e.g. seeking to maximize the overall spectral efficiency of the total optical services. Such an optimization maximizes, in effect, the unused optical network capacity that remains available for further service deployment.

The functions of the OPDT in these cases are essentially those described in Section 6.4 and Section 6.1.2. First, the OPDT is used to verify that all postulated optical services would operate within acceptable performance bounds, when deployed on the existing or on a postulated new network topology and detailed equipment map. Second, the optical service margin information generated by the OPDT may flag candidate solutions that feature a large number of unnecessarily high optical service margins. Such findings reflect a general inefficiency of the candidate solutions and may be used to indicate that e.g. more spectrally efficient solutions are available and should be sought.

The use of the OPDT with and within optimization process architectures may be represented in ways qualitatively similar to what Figure 4 depicts in respect of NPDTs in packet network optimization use cases.

6.3. Optical Service (Re-)Provisioning

Optical service (re-)provisioning presents operational challenges and risks. Optical service power levels and - by extension - their optical noise and other impairment characteristics, are coupled by optical amplifiers, which act collectively on transiting optical services. Optical service add, drop and change operations can thus have deleterious and non-obvious impacts across optical services, particularly in ring and mesh optical network topologies and potentially resulting in failure of added, changed or unchanged optical services.

An OPDT can be used to assess the optical service performances that would result from prospective optical service (re-)provisioning operations. Such information could then be used by Management Plane-embedded or -associated applications seeking to e.g. optimize add/drop/change batching and sequencing operations, or to determine optimized optical service launch powers.

For this use case, the appropriate input DT Interface specifies the topology of optical services postulated for the post-(re)provisioning scenario, as well as the launch powers and other characteristics (modulation, coding, spectral characteristics, etc.) defining those services. Specific scenarios thus postulated for performance assessment are conceived and determined by the applications referred to above.

6.4. Optical Service Planning

Before optical service provisioning is attempted, proposed routes (topology) and other characteristics - launch powers, spectral allocations, modulation, coding, baud rates, etc. - must be planned for new optical services to accommodate new traffic, as must any changes to or deletions of existing optical services that may be suggested by shifting transmission traffic loads.

An OPDT, per the description in Section 6.3, can be used directly in support of an optical service planning application, which is presumed to operate as part of, or in conjunction with the Management Plane. For example, prospective new optical service plans can be validated as functional - i.e. that all services would operate within acceptable performance bounds - by the OPDT. Beyond identifying scenarios in which one or more optical services would fail to operate, the identification of undesirably low or unnecessarily high optical service margins could serve as a trigger to explore alternative plans. This suggests a linkage to optimization processes, which are discussed in Section 6.2.2

6.5. Optical Network Risk Mapping

The OPDT can be used to assess in advance the impacts on optical services that can be expected should various "risk" scenarios materialize [OPDT]. For example, the OPDT may be used to predict the impacts on transmission performances of optical services that would be indirectly affected by particular fibre cuts. This is important, as while services that transit a cut fibre link will be interrupted, other optical services - those that co-transited uncut links along with the services that interrupted by the cut(s) - may experience changes in terminal powers and margins due to amplifier-based coupling. Where unacceptable event-driven risks to optical service performances are identified by OPDT-based analysis, solutions may be proactively sought. For example, optical service planning may be undertaken to find more resilient optical service solutions for the at-risk service instances identified.

6.5.1. Optical Network Dynamic Restoration Planning

An important specific use of risk mapping is in the assessment of optical service dynamic restoration solutions [OPDT]. Dynamic restoration involves pre-computing a set of failure scenario-based restoration responses. Resources are not reserved a priori for each active service; rather, restoration services are delivered as needed from a "pool" of resources. This is a more efficient restoration modality than dedicated protection, as the size of the resource pool is limited by an assumption that only a limited number of failures may happen at once. However, dynamic restoration requires ongoing re-planning of restoration solutions, as optical service maps and equipment and fibre conditions may both evolve over time, affecting restoration service performance. Risk mapping as described in Section 6.5 may be used on an ongoing basis to identify service risks corresponding to planned failure-restoration scenarios. When such performance risks are found, a search for new dynamic restoration plans may be triggered, with new candidate restoration solutions checked for predicted performance integrity using the OPDT.

6.6. Troubleshooting

There are many factors that cause network failures (e.g., invalid network configurations, unexpected protocol interactions). Debugging modern networks is complex and time consuming. Currently, troubleshooting is typically done by human experts with years of experience using networking tools.

Network operators can leverage a PODT to reproduce previous network failures, in order to find the source of service disruptions. Specifically, network operators can replicate past network failure scenarios and analyze their impact on network performance, making it easier to find specific configuration errors. In addition, the PODT helps in finding more robust network configurations that prevent service disruptions in the future.

6.7. Anomaly detection

Since the PODT models the behaviour of a real-world network, network operators have access to an estimation of the expected network behaviour. When the real-world network behaviour deviates from the PODT's behaviour, it can act as an indicator of an anomaly in the real-world network. Such anomalies can appear at different places in a network (e.g. core, edge, IoT), and different data sources can be used to detect such anomalies. Further, trial-and-error modifications to the scenario assessed by the PODT - e.g. to network or component configuration or status - can assist with anomaly troubleshooting if convergence between predicted and observed performance can be thus obtained. The detailed differences between the scenario found to produce such convergence, and the actual physical network scenario, may help to indicate the source of the anomaly.

6.8. What-if scenarios

The PODT is a unique tool for performing what-if analysis; that is, for analyzing the impact of potential scenarios and configurations safely without any impact on the real network. In this context, the PODT acts as a safe sandbox wherein, different configurations and scenarios may be presented to the PODT in order to understand their prospective impacts on physicalnetwork behaviours. It might be said that the role of the PODT is always to perform a what-if behavioural analysis, as its intrinsic function is to assess aspects of network or service performance in scenarios that may be partly or entirely hypothetical. Nevertheless, a variety of uses of PODTs beyond the specific use cases already covered, and which fit the what-if scenario analysis description, may be imagined. Some examples of such cases include:

*What is the impact on my network performance if we acquire company ACME and we incorporate all its employees?

*When will the network run out of capacity if we have an organic growth of users?

*What is the optimal network hardware upgrade given a budget?

*We need to update this path. What is the impact on the performance of the other flows?

*A particular day has a spike of 10% in traffic intensity. How much loss will it introduce? Can we reduce this loss if we rate-limit another flow?

*How many links can fail until the SLA is degraded?

*What happens if link B fails? Is the network able to process the current traffic load? Or, on an optical transmission network, what will be the performance of surviving optical services? (Per the risk map use case described in Section 6.5).

*

6.9. Training

As discussed before, the PODT can be understood as a safe playground where misconfigurations do not affect the real-world system performance. In this context, the PODT can play an important role in improving the education and certification process of network professionals, both in basic networking training and advanced scenarios. For example:

*In basic network training, understand how routing modifications impact delay.

*In more advanced studies, showcase the impact of scheduling configuration on flow performance, and how to use them to optimize SLAs.

*In cybersecurity scenarios, evaluate the effects of network attacks and possible counter-measures.

7. Implementation Challenges

7.1. Network Performance Digital Twin Implementation Challenges

This section presents different technologies that can be used to build a NPDT, and details the advantages and disadvantages of using them to implement a NPDT. It takes into account how they perform with respect to the requirements of accuracy, speed, and scale of the NPDT predictions.

7.1.1. Simulation

Packet-level simulators, such as OMNET++ [OMNET] and NS-3 [ns-3] simulate network events. In a nutshell, they simulate the operation of a network by processing a series of events, such as the transmission of a packet, enqueuing and dequeuing packets in the router, etc. Hence, they offer excellent accuracy when predicting network performance metrics (delay, jitter and loss), but they take a significant amount of time to run the simulation. They scale linearly with number of packets to simulate.

In fact, the simulation time depends on the number of events to process [limitations-net-sim]. This limits the scalability of simulators, even if the topology does not change: increasing traffic intensities will take longer to simulate because more packets enter the network per unit of time. Conversely, simulating the same traffic intensity in larger topologies will also increase the simulation time. For example, consider a simulator that takes 11 hours to process 4 billion events (these values are obtained from an actual simulation). Although 4 billion events may appear a large figure, consider:

- A 1 Gbps ethernet link, transmitting regular frames with the maximum of 1518 bytes.

- This translates to approx. 82k packets crossing the link per second.

- Assuming a network with 50 links, and that the transmission of a packet over a link equals to a single event a in the simulator, such network translates to 82k packets/s/link * 50 links * 1 event/packet ~ 4 million events to simulate one second of network activity.

- Then, with a budget of 4 billion events, it takes 11 hours to simulate only 16 minutes of network activity.

These figures show that, despite the high accuracy of network simulators, they take too much time to calculate performance estimations.

7.1.2. Emulation

Network emulators run the original network software in a virtualized environment. This makes them easy to deploy, and depending on the emulation hardware, they can produce reasonably fast estimations. However, for large scale networks their speed will eventually decrease because they are not using specific hardware built for networking. For fully-virtualized networks, emulating a network requires as many resources as the real one, which is not cost-effective.

In addition, some studies have reported variable accuracy depending on the emulation conditions, both the parameters and underlying hardware and OS configurations [emulation-perf]. Hence, emulators show some limitations if we want to build a fast and scalable NPDT. However, emulators are useful in other use cases, for example in training, debugging, or testing new features.

7.1.3. Analytical Modelling

Queueing Theory (QT) is an analytical tool that models computer networks as a series of queues. The key advantage of QT is its speed, because the calculations rely on mathematical equations. QT is arguably the most popular modeling technique, where networks are represented as interconnected queues that are evaluated analytically. This represents a well-established framework that can model complex and large networks.

However, the main limitation of QT is the traffic model: although it offers high accuracy for Poisson traffic models, it presents poor accuracy under realistic traffic models [qt-precision]. Internet traffic has been extensively analyzed in the past two decades, and despite the community has not agreed on a universal model, there is consensus that in general aggregated traffic shows strong autocorrelation and a heavy-tail [inet-traffic].

7.1.4. Neural Networks

Finally, Neural Networks (NN) and other Machine Learning (ML) tools are as fast as QT (in the order of milliseconds), and can provide similar accuracy to that of packet-level simulators. They represent an interesting alternative, but have two key limitations. First, they require training the NN with a large amount of data from a wide range of network scenarios: different routings, topologies, scheduling configurations, as well as link failures and network congestion. This dataset may not be always accessible, or easy to produce in a production network (see Section 7.1.4.6). Second, in order to scale to larger topologies and keep the accuracy, not all NN provide sufficient accuracy, therefore, some use cases need custom NN architectures.

7.1.4.1. MultiLayer Perceptron

A MultiLayer Perceptron [MLP] is a basic kind of NN from the family of feedforward NN. In short, input data is propagated unidirectionally from the input layer of neurons through the output. There may be an arbitrary number of hidden layers between the input and output layer. They are widely used for basic ML applications, such as regression.

7.1.4.2. Recurrent Neural Networks

Recurrent Neural Networks [RNN] are a more advanced type of NN because they connect some layers to the previous ones, which gives them the ability to store state. They are mostly used to process sequential data, such as handwriting, text, or audio. They have been used extensively in speech processing [RNN-speech], and in general, Natural Language Processing applications [NLP].

7.1.4.3. Convolutional Neural Networks

Convolutional Neural Networks (CNN), are a Deep Learning NN designed to process structured arrays of data such as images. CNNs are highly performant when detecting patterns in the input data. This makes them widely used in computer vision tasks, and have become the state of the art for many visual applications, such as image classification [CNN-images]. Hence, their current design presents limited applicability to computer networks.

7.1.4.4. Graph Neural Networks

Graph Neural Networks [GNN] are a type of neural network designed to work with graph-structured data. A relevant type of GNN with interesting characteristics for computer networks are Message Passing Neural Networks (MPNN). In a nutshell, MPNN exchanges a set of messages between the graph nodes in order to understand the relationship between the input graph and the expected outputs of the training dataset. They are composed of three functions, that are repeated several iterations, depending on the size of the graph:

*Message: encodes information about the relationship of two contiguous elements of the graph in a message (an n-element array).

*Aggregation: combines the different messages received on a particular node. It is typically an element-wise summation. The result is an array of constant length, independently of the number of received messages.

*Update: combines the hidden states of a node with the aggregated message. The result of this function is used as input to the next message-passing iteration.

Note that the internal architecture of a MPNN is re-build for each input graph.

Such ability to understand graph-structured data naturally renders them interesting for a Network Performance Digital Twin. Since computer networks are fundamentally graphs, they have the potential to take as input a graph of the network, and produce as output performance estimations of such the input network [qt-precision].

7.1.4.5. NN Comparison

Figure 5 presents a comparison of different types of NN that predict the delay of a given input network. We use a dataset of the performance of different network topologies, created with simulation data (i.e, ground truth) from OMNET++. We measure the error relative to the delay of the simulation data. In order to evaluate how well the different NN deal with different network topologies, we train each NN in three different scenarios:

*Same topology: the training and testing datasets contain the same network topologies.

*Different topology: the training and testing datasets contain different sets of network topologies. The objective is determining if the NN keeps the same performance if we show it a topology it has never seen.

*Link failures: here we remove a random link from the topology.

Mean Average Percentage Error of the delay prediction			
Scenario	MLP	RNN	GNN
Same topology	0.123	0.1	0.020
Different topology	11.5	0.305	0.019
Link failures	1.15	0.638	0.042

Figure 5: Performance comparison of different NN architectures

We can see that all NNs predict with excellent accuracy the network delay if we don't change the topology used during training. However, when it comes to new topologies, the error of the MLP is unacceptable (1150 %), as well as the RNN, around 30%. On the other hand, the GNN can understand new topologies, with an error below 2%. Similarly, if a link fails, the RNN has difficulties offering accurate predictions (60% error), while the GNN maintains the accuracy (4.2%). These results show the potential of GNNs to build a Network Performance Digital Twin.

7.1.4.6. Training of ML-based Digital Twins

In the context of Digital Twins based on Machine Learning, they require a training process before they can be deployed. Commonly, the training process makes use of a dataset of inputs and expected outputs, that guides the training process to adjust the internal architecture of e.g. the neural network. There are some caveats regarding the training process:

*In order to obtain sufficient accuracy, the training dataset needs to be representative, that is, contain samples of a wide range of possible inputs and outputs. In networks, this translates to samples of a congested network, with a link failure, etc. Otherwise, the resulting algorithm cannot predict such situations.

*Taking the latter into account, this means that some kind of samples, e.g. those of a congested or disrupted network are difficult to obtain from a production network.

*A way to acquire those samples is in a testbed, although it may not be possible for some networks, especially those of large scale. A possible solution in this situation is developing Neural

Networks that are invariant to some of the metrics of the graph, e.g. number of nodes. That is, the NN does not lose accuracy if the number of nodes increases. This makes it possible to train the NN in a testbed, and then deploy it in a network that is larger than the testbed without losing accuracy.

7.2. Optical Performance Digital Twin Implementation Challenges

Significant challenges to OPDT implementation, deployment and use relate to e.g. models and instrumentation.

Optical transmission performance is difficult to model accurately because the different impairments and other factors that determine performance are not easy to model with high accuracy and system specificity. For example, optical amplifiers are a key determinant of transmission behaviours and limits, but their gain and noise characteristics are complicated functions of optical service input power and spectral profiles, operational set points, etc. In addition, they vary considerably among amplifier designs, types and even instances. Yet, transmission systems may contain long chains of amplifiers, so that accurate end-to-end service modeling requires highly accurate individual amplifier models. The development of models that deliver sufficiently accurate performance predictions across operational circumstances and potentially also amplifier vendors, types etc., represents a significant challenge. Nonetheless, promising solution paths have been developed [EDFA1], [EDFA2].

Transmission performance prediction accuracy may be improved when the necessary scope of modeling can be reduced through enhancements in direct measurement of relevant parameters on the physical network. For example, if optical signal-to-noise ratio and other impairments can be measured directly on operating services, the available margins on those optical services is yielded directly. Although significant advances have been made in this area it will take time before such improved instrumentation features become widely deployed, and both usable and susceptible to standardization (e.g. of Measurement Interfaces).

8. IANA Considerations

This memo includes no request to IANA.

9. Security Considerations

An attacker can alter the software image of the PODT. This could produce inaccurate performance estimations, that could result in network misconfigurations, disruptions or outages. Hence, in order to prevent the accidental deployment of a malicious PODT, the software image of the PODT MUST be digitally signed by the vendor.

10. References

10.1. Normative References

10.2. Informative References

[OMNET] "https://omnetpp.org/", 2022.

[ns-3] "https://www.nsnam.org/", 2022.

[P4Rspec] "https://p4.org/p4-spec/p4runtime/main/P4Runtime-Spec.html", 2021.

[OFspec] "TS-025: OpenFlow Switch Specification https://opennetworking.org/wp-content/uploads/2014/10/openflow-switch-v1.5.1.pdf", 2015.

[NetworkXlib] "https://networkx.org/", 2022.

[openconfig-rtgwg-gnmi-spec-01]
Shakir, R., Shaikh, A., Borman, P., Hines, M., Lebsack, C., and C. Morrow, "gRPC Network Management Interface (gNMI)", March 2018, <https://datatracker.ietf.org/doc/html/draft-openconfig-rtgwg-gnmi-spec-01>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, DOI 10.17487/RFC6830, January 2013, <https://www.rfc-editor.org/info/rfc6830>.

[RFC4655] Farrel, A., Vasseur, J.-P., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, DOI 10.17487/RFC4655, August 2006, <https://www.rfc-editor.org/info/rfc4655>.

[RFC7047] Pfaff, B. and B. Davie, Ed., "The Open vSwitch Database Management Protocol", RFC 7047, DOI 10.17487/RFC7047, December 2013, <https://www.rfc-editor.org/info/rfc7047>.

[RFC3954] Claise, B., Ed., "Cisco Systems NetFlow Services Export Version 9", RFC 3954, DOI 10.17487/RFC3954, October 2004, <https://www.rfc-editor.org/info/rfc3954>.

[I-D.draft-zhou-nmrg-digitaltwin-network-concepts] Zhou, C., Yang, H., Duana, X., Lopez, D., Pastor, A., Wu, Q., Boucadir, M., and C. Jacquenet, "Digital Twin Network: Concepts and Reference Architecture", Work in Progress, Internet-Draft, draft-zhou-nmrg-digitaltwin-network-concepts-06, 2 December 2021, <https://datatracker.ietf.org/doc/html/draft-zhou-nmrg-digitaltwin-network-concepts-06>.

[irtf-nmrg-ibn-concepts-definitions-09] Clemm, A., Ciavaglia, L., Granville, L. Z., and J. Tantsura, "Intent-Based Networking - Concepts and Definitions", March 2022, <https://datatracker.ietf.org/doc/html/draft-irtf-nmrg-ibn-concepts-definitions-09>.

[digital-twin-5G] Nguyen, H. X., Trestian, R., To, D., and M. Tatipamula, "Digital Twin for 5G and Beyond", 2021, <https://doi.org/10.1109/MCOM.001.2000343>.

[digital-twin-vanets] Zhao, L., Han, G., Li, Z., and L. Shu, "Intelligent Digital Twin-Based Software-Defined Vehicular Networks", 2020, <https://doi.org/10.1109/MNET.011.1900587>.

[digital-twin-industry] Groshev, M., Guimarães, C., Martín-Pérez, J., and A. D. L. Oliva, "Toward Intelligent Cyber-Physical Systems: Digital Twin Meets Artificial Intelligence", 2021, <https://doi.org/10.1109/MCOM.001.2001237>.

[streaming-telemetry] Gupta, A., Harrison, R., Canini, M., Feamster, N., Rexford, J., and W. Willinger, "Sonata: Query-Driven Streaming Network Telemetry", 2018, <https://doi.org/10.1145/3230543.3230555>.

[network-capacity] Ellis, A. D., Suibhne, N. M., Saad, D., and D. N. Payne, "Communication networks beyond the capacity crunch", 2016, <https://royalsocietypublishing.org/doi/abs/10.1098/rsta.2015.0191>.

[planning-scalability] Zhu, H., Gupta, V., Ahuja, S. S., Tian, Y., Zhang, Y., and X. Jin, "Network Planning with Deep Reinforcement Learning", 2021, <https://doi.org/10.1145/3452296.3472902>.

[limitations-net-sim] Rampfl, S., "Network simulation and its limitations", 2013, <https://doi.org/10.2313/NET-2013-08-1_08>.

[emulation-perf] Jurgelionis, A., Laulajainen, J., Hirvonen, M., and A. I. Wang, "An Empirical Study of NetEm Network Emulation Functionalities", 2011, <https://doi.org/10.1109/ICCCN.2011.6005933>.

[qt-precision] Ferriol-Galmés, M., Rusek, K., Suárez-Varela, J., Xiao, S., Cheng, X., Barlet-Ros, P., and A. Cabellos-Aparicio, "RouteNet-Erlang: A Graph Neural Network for Network Performance Evaluation", 2022, <https://arxiv.org/abs/2202.13956>.

[inet-traffic] Popoola, J. and R. Ipinyomi, "Empirical Performance of Weibull Self-Similar Tele-traffic Model", 2017.

[MLP] Pal, S. and S. Mitra, "Multilayer perceptron, fuzzy sets, and classification", 1992, <https://doi.org/10.1109/72.159058>.

[RNN] Hochreiter, S. and J. Schmidhuber, "Long Short-Term Memory", 1997, <https://doi.org/10.1162/neco.1997.9.8.1735>.

[RNN-speech] Mikolov, T., Kombrink, S., Burget, L., Černocký, J., and S. Khudanpur, "Extensions of recurrent neural network language model", 2011, <https://doi.org/10.1109/ICASSP.2011.5947611>.

[GNN] Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., and G. Monfardini, "The Graph Neural Network Model", 2009, <https://doi.org/10.1109/TNN.2008.2005605>.

[DEFO] Hartert, R., Vissicchio, S., Schaus, P., Bonaventure, O., Filsfils, C., Telkamp, T., and P. Francois, "A Declarative and Expressive Approach to Control Forwarding

Paths in Carrier-Grade Networks", 2015, <https://doi.org/10.1145/2785956.2787495>.

[facebook-config] Sung, Y. E., Tie, X., Wong, S. H., and H. Zeng, "Robotron: Top-down Network Management at Facebook Scale", 2016, <https://doi.org/10.1145/2934872.2934874>.

[auto-dc-topology] Salman, S., Streiffer, C., Chen, H., Benson, T., and A. Kadav, "DeepConf: Automating Data Center Network Topologies Management with Machine Learning", 2018, <https://doi.org/10.1145/3229543.3229554>.

[CNN-images] Krizhevsky, A., Sutskever, I., and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", 2012, <https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf>.

[MARL-TE] Bernárdez, G., Suárez-Varela, J., López, A., Wu, B., Xiao, S., Cheng, X., Barlet-Ros, P., and A. Cabellos-Aparicio, "Is Machine Learning Ready for Traffic Engineering Optimization?", 2021, <https://doi.org/10.1109/ICNP52444.2021.9651930>.

[LS] Gay, S., Hartert, R., and S. Vissicchio, "Expect the unexpected: Sub-second optimization for segment routing", 2017, <https://doi.org/10.1109/INFOCOM.2017.8056971>.

[DNN-TM] Valadarsky, A., Schapira, M., Shahaf, D., and A. Tamar, "Learning to Route", 2017, <https://doi.org/10.1145/3152434.3152441>.

[ReRoute-Cost] Zheng, J., Xu, Y., Wang, L., Dai, H., and G. Chen, "Online Joint Optimization on Traffic Engineering and Network Update in Software-defined WANs", 2021, <https://doi.org/10.1109/INFOCOM42981.2021.9488837>.

[NLP] Chowdhary, K. R., "Natural Language Processing", 2020, <https://doi.org/10.1007/978-81-322-3972-7_19>.

[Google-Clos]
Singh, A., Ong, J., Agarwal, A., Anderson, G., Armistead, A., Bannon, R., Boving, S., Desai, G., Felderman, B., Germano, P., Kanagala, A., Provost, J., Simmons, J., Tanda, E., Wanderer, J., H\"{o}lzle, U., Stuart, S., and A. Vahdat, "Jupiter Rising: A Decade of Clos Topologies and Centralized Control in Google's Datacenter Network", 2015, <https://doi.org/10.1145/2785956.2787508>.

[digital-twin-AI] Mozo, A., Karamchandani, A., Gómez-Canaval, S., Sanz, M., Moreno, J. I., and A. Pastor, "B5GEMINI: AI-

Driven Network Digital Twin", 2022, <https://www.mdpi.com/1424-8220/22/11/4106>.

[OPDT] Janz, C., You, Y., Hemmati, M., Jiang, Z., Javadtalab, A., and J. Mitra, "Digital Twin for the Optical Network: Key Technologies and Enabled Automation Applications", 2022, <https://doi.org/10.1109/NOMS54207.2022.9789844>.

[EDFA1] You, Y., Jiang, Z., and C. Janz, "Machine Learning-Based EDFA Gain Model", 2018, <https://doi.org/10.1109/ECOC.2018.8535397>.

[EDFA2] You, Y., Jiang, Z., and C. Janz, "OSNR prediction using machine learning-based EDFA models", 2019, <https://doi.org/10.1049/cp.2019.1044>.

[RFC8345] Clemm, A., Medved, J., Varga, R., Bahadur, N., Ananthakrishnan, H., and X. Liu, "A YANG Data Model for Network Topologies", RFC 8345, DOI 10.17487/RFC8345, March 2018, <https://www.rfc-editor.org/info/rfc8345>.

[RFC8795] Liu, X., Bryskin, I., Beeram, V., Saad, T., Shah, H., and O. Gonzalez de Dios, "YANG Data Model for Traffic Engineering (TE) Topologies", RFC 8795, DOI 10.17487/RFC8795, August 2020, <https://www.rfc-editor.org/info/rfc8795>.

[RFC9094] Zheng, H., Lee, Y., Guo, A., Lopez, V., and D. King, "A YANG Data Model for Wavelength Switched Optical Networks (WSONs)", RFC 9094, DOI 10.17487/RFC9094, August 2021, <https://www.rfc-editor.org/info/rfc9094>.

[I-D.ietf-ccamp-flexigrid-yang] Jorge Lopez de Vergara Mendez, E., Burrero, D. P., King, D., Lee, Y., and H. Zheng, "A YANG Data Model for Flexi-Grid Optical Networks", Work in Progress, Internet-Draft, draft-ietf-ccamp-flexigrid-yang-13, 10 July 2022, <https://www.ietf.org/archive/id/draft-ietf-ccamp-flexigrid-yang-13.txt>.

[I-D.ietf-ccamp-optical-impairment-topology-yang] Beller, D., Rouzic, E. L., Belotti, S., Galimberti, G., and I. Busi, "A YANG Data Model for Optical Impairment-aware Topology", Work in Progress, Internet-Draft, draft-ietf-ccamp-optical-impairment-topology-yang-10, 11 July 2022, <https://www.ietf.org/archive/id/draft-ietf-ccamp-optical-impairment-topology-yang-10.txt>.

[I-D.ietf-teas-yang-te] Saad, T., Gandhi, R., Liu, X., Beeram, V. P., Bryskin, I., and O. G. D. Dios, "A YANG Data Model for Traffic Engineering Tunnels, Label Switched Paths and

Interfaces", Work in Progress, Internet-Draft, draft-ietf-teas-yang-te-30, 11 July 2022, <https://www.ietf.org/archive/id/draft-ietf-teas-yang-te-30.txt>.

[I-D.ietf-ccamp-wson-tunnel-model]
Lee, Y., Zheng, H., Guo, A., Lopez, V., King, D., Yoon, B. Y., and R. Vilalta, "A Yang Data Model for WSON Tunnel", Work in Progress, Internet-Draft, draft-ietf-ccamp-wson-tunnel-model-07, 11 July 2022, <https://www.ietf.org/archive/id/draft-ietf-ccamp-wson-tunnel-model-07.txt>.

[I-D.ietf-ccamp-flexigrid-tunnel-yang]
Jorge Lopez de Vergara Mendez, E., Burrero, D. P., King, D., Lopez, V., Busi, I., Belotti, S., and G. Galimberti, "A YANG Data Model for Flexi-Grid Tunnels", Work in Progress, Internet-Draft, draft-ietf-ccamp-flexigrid-tunnel-yang-01, 11 July 2022, <https://www.ietf.org/archive/id/draft-ietf-ccamp-flexigrid-tunnel-yang-01.txt>.

[I-D.yg3bp-ccamp-network-inventory-yang]
Yu, C., Busi, I., Guo, A., Belotti, S., Bouquier, J., Peruzzini, F., Dios, O. G. D., and V. Lopez, "A YANG Data Model for Network Hardware Inventory", Work in Progress, Internet-Draft, draft-yg3bp-ccamp-network-inventory-yang-02, 24 October 2022, <https://datatracker.ietf.org/api/v1/doc/document/draft-yg3bp-ccamp-network-inventory-yang/>.

Acknowledgements

TBD

Authors' Addresses

Jordi Paillisse
UPC-BarcelonaTech
c/ Jordi Girona 1-3
08034 Barcelona Catalonia
Spain

Email: jordi.paillisse@upc.edu

Paul Almasan
UPC-BarcelonaTech
c/ Jordi Girona 1-3
08034 Barcelona Catalonia
Spain

Email: felician.paul.almasan@upc.edu

Miquel Ferriol
UPC-BarcelonaTech
c/ Jordi Girona 1-3
08034 Barcelona Catalonia
Spain

Email: miquel.ferriol@upc.edu

Pere Barlet
UPC-BarcelonaTech
c/ Jordi Girona 1-3
08034 Barcelona Catalonia
Spain

Email: pere.barlet@upc.edu

Albert Cabellos
UPC-BarcelonaTech
c/ Jordi Girona 1-3
08034 Barcelona Catalonia
Spain

Email: alberto.cabellos@upc.edu

Shihan Xiao
Huawei
China

Email: xiaoshihan@huawei.com

Xiang Shi
Huawei
China

Email: shixiang16@huawei.com

Xiangle Cheng
Huawei
China

Email: chengxiangle1@huawei.com

Christopher Janz
Huawei
Canada

Email: christopher.janz@huawei.com

Aihua Guo
Futurewei
United States of America

Email: aihua.guo@futurewei.com

Diego Perino
Telefonica I+D
Barcelona
Spain

Email: diego.perino@telefonica.com

Diego Lopez
Telefonica I+D
Seville
Spain

Email: diego.r.lopez@telefonica.com

Antonio Pastor
Telefonica I+D
Madrid
Spain

Email: antonio.pastorperales@telefonica.com