Artificial Intelligence Control Plane for Deterministic Networks Proof-of-Concept

Alejandro Calvillo-Fernandez
acalvill@pa.uc3m.es
Universidad Carlos III de Madrid
Madrid, Spain

Matteo Ravalli
m.ravalli@nextworks.it
Nextworks
Pisa, Italy

Juan Brenes
j.brenes@nextworks.it
Nextworks
Pisa, Italy

Jose Luis Carcel
jose.carcel@eviden.com
Eviden
Valencia, Spain

David Rico-Menendez
drico@pa.uc3m.es
Universidad Carlos III de Madrid
Madrid, Spain

Fernando Agraz
fernando.agraz@upc.edu
Universitat Politècnica de Catalunya
Barcelona, Spain

Pietro G. Giardina
p.giardina@nextworks.it
Nextworks
Pisa, Italy

Salvatore Spadaro
salvatore.spadaro@upc.edu
Universitat Politècnica de Catalunya
Barcelona, Spain

Luis Velasco
luis.velasco@upc.edu
Universitat Politècnica de Catalunya
Barcelona, Spain

Abstract

This paper presents the design and implementation of an Artificial Intelligence Control Plane (AICP) for deterministic networks, emphasizing the Proof-of-Concept (PoC) demonstration. The AICP framework integrates AI, digital twin technology, and real-time telemetry to manage complex network environments, ensuring reliable and low-latency communication. The PoC showcases the practical viability of the AICP by dynamically adapting to varying network demands and maintaining stringent to the Key Performance Indicator (KPI)s of the service. Extensive testing and real-world simulations highlight the framework's potential to enhance the efficiency and resilience of industrial communication networks.

ACM MobiCom '24, November 18–22, 2024, Washington D.C., DC, USA

ACM ISBN 979-8-4007-0489-5/24/11
https://doi.org/10.1145/3636534.3696725

CCS Concepts

• **Networks → Network services**; *Network management*; Network architectures.

Keywords

Deterministic Network Controller, Network Management Services, Control Plane, Multi-Domain Networks

ACM Reference Format:
Alejandro Calvillo-Fernandez, Matteo Ravalli, Juan Brenes, Jose Luis Carcel, David Rico-Menendez, Fernando Agraz, Pietro G. Giardina, Salvatore Spadaro, and Luis Velasco. 2024. Artificial Intelligence Control Plane for Deterministic Networks Proof-of-Concept. In *The 30th Annual International Conference on Mobile Computing and Networking (ACM MobiCom '24), November 18–22, 2024, Washington D.C., DC, USA*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3636534.3696725

1 Introduction

The rise of machine-type communications (MTC) networks, driven by the need for reliable, low-latency transmissions, has emphasized the importance of deterministic networks. Traditional Time Sensitive Networking (TSN) has been central to industrial communications, offering predictable and efficient data flows [4]. To address these needs, we present the design and implementation of an AICP for deterministic networks, previously introduced in [3]. The AICP integrates AI, digital twins, and real-time telemetry to manage multi-domain networks, providing End-to-End (E2E) deterministic

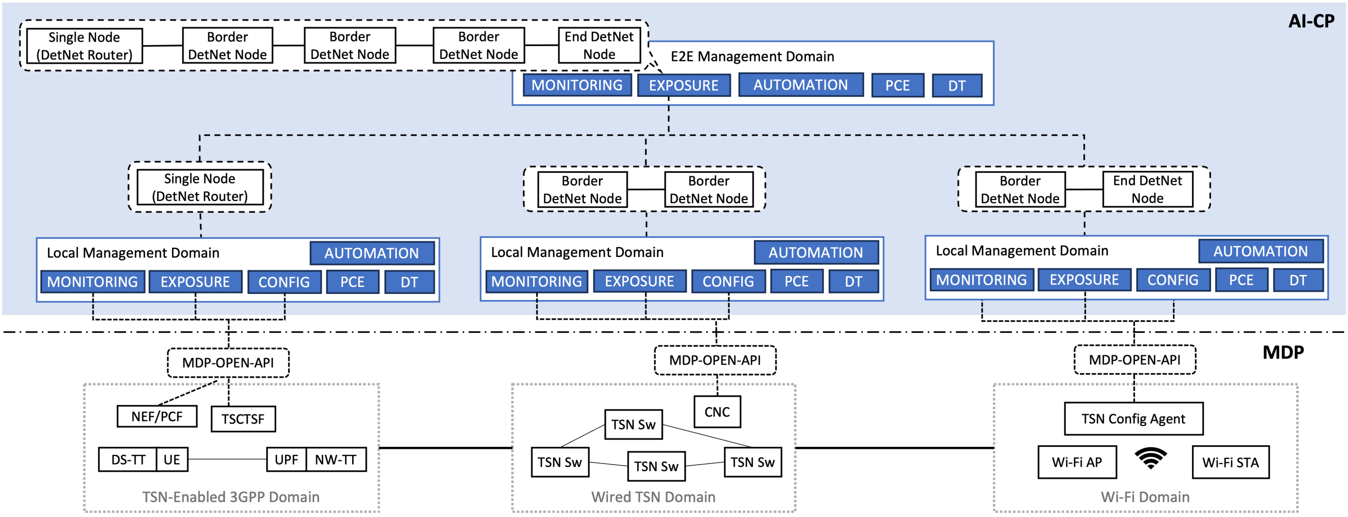

Figure 1: High-level control architecture and abstraction modelling.

services. The main challenge lies in coordinating diverse transport technologies. For this AICP, we considered: 3GPP, and WiFi, which are currently being extended to support determinism; and wired TSN. Each technological domain enables its control and management by exposing a specific API (MDP-OPEN-API in the figure) Our PoC demonstrates the AICP's capability to dynamically adapt to varying network demands while maintaining strict KPI requirements. Figure 1 illustrates the general architecture of the AICP and highlights the multi-domain, multi-technology data plane abstraction, which enables E2E service provisioning.

The AICP aims to control different transport technologies in a unified manner to provide E2E deterministic services, focusing on 3GPP, WiFi, and wired Time-Sensitive Networking (TSN). Each domain exposes an API (MDP-OPEN-API) that provides data plane topology, resource configuration, and monitoring functionalities. The AICP consists of two levels: the **Local Management Domain** level, composed of technology-specific control entities, and the **E2E Management Domain** level, which orchestrates the local entities to ensure E2E connectivity. Local management domain entities share functionalities known as Management Services (MS), which include *Exposure*, *Configuration*, *Path Computation*, KPI assessment via a Digital Twin (DT), *Service Automation*, and *Monitoring*. Each domain provides an abstracted view of its underlying technology to the E2E management domain, enabling unified control across the E2E scenario using IETF standards-based models.

As shown in Figure 1, local management domains expose their topology and capabilities as IETF Deterministic Networking (DetNet) elements. This allows the E2E exposure MS to build an abstracted view of the E2E scenario enriched with deterministic capabilities. When an E2E service request is received, the automation MS contacts the E2E PCE, which computes a domain sequence that meets the service requirements. The E2E DT then assesses the feasibility of this sequence, and the automation MS configures each Local Management Domain entity involved in the service provisioning. This process is detailed in Section 8.

This paper describes the AICP and its operation via a PoC implementation, demonstrating its potential to transform industrial communication networks by dynamically adapting to network demands while ensuring high performance and reliability.

2 E2E IETF Deterministic Networking integration

The DetNet framework establishes deterministic data paths over Layer-2 bridged and Layer-3 routed networks, guaranteeing bounded latency, loss, and jitter for high reliability. By constructing a Layer-3 overlay (IP/UDP with MPLS [7]), DetNet enables deterministic properties in a multi-domain scenario, acting as the "glue" between the inherent deterministic capabilities of each technological domain.

The service sub-layer includes an S-label to identify flows and a DetNet Control Word (d-CW) to sequence packets. The forwarding sub-layer manages explicit routes and allocates resources to meet Quality of Service (QoS) requirements. Traffic flows are encapsulated into DetNet tunnels at the entrance of the first domain, forwarded across domains, and decapsulated at the endpoint.

In this architecture, DetNet services are integrated at the borders of different domains, functioning as gateways (Border DetNet Nodes in Fig. 1). The E2E Modules of the AICP perceive DetNet nodes as encapsulating each domain from a black-box perspective. MPLS over UDP/IP handles tunnelling flows, with MPLS Labels identifying flows across domains. DetNet performs three main functions: (i) *Forwarding*, transmitting flows and performing packet corrections, (ii) *Encapsulation*, and (iii) *Decapsulation*, serving as system endpoints. S-labels ensure proper flow routing and handling at the inter-domain level.

The AICP's E2E Path Computation module calculates the E2E path that flows will traverse, adjusting the forwarding functions of nodes to follow the determined paths. It dynamically enables or disables functionalities such as Packet Replication, Elimination, and Ordering Functions (PREOF), ensuring precise and deterministic flow handling across the multi-domain network. The path is computed based on service requirements and domain capabilities, and the service sub-layer can implement techniques to increase reliability by replicating packets, eliminating duplicates, and ordering them before exiting the system, ensuring a robust and reliable E2E data path.

3 Service Automation

The Service Automation module ensures closed-loop automation of deterministic E2E services across heterogeneous infrastructures. It manages three key lifecycle stages:

· **Service provisioning:** Configures the technology domain to meet service requirements.

· **Service assurance:** Continuously monitors and adjusts services during their lifecycle.

· **Service termination:** Removes services after expiration or upon request.

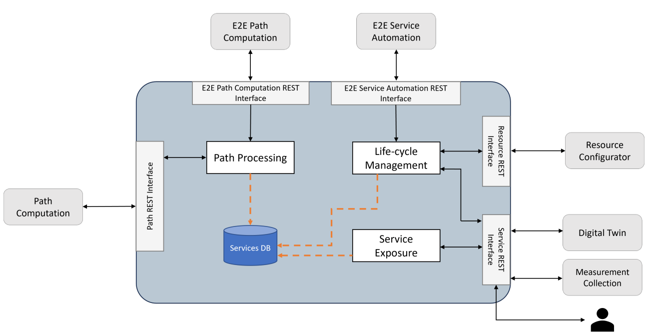

Figure 2: MD Service Automation SW Design

The module operates across two main Management Service (MS): the E2E Service Automation MS, which manages service implementation across all domains, and the MD Service Automation MS, which ensures service automation within each domain. Closed-loop automation is achieved by interacting with the Path Computation, Network Digital Twin, and Monitoring modules. Artificial Inteligence (AI) in the Network Digital Twin predicts service KPIs during path computation, while the module also exposes service details to other MSs or users via the E2E Service Exposure and MD Service Exposure functionalities.

The Service Automation module's software design integrates functions, interfaces, and databases to ensure closed-loop automation at both E2E and MD levels, as illustrated in Figure **??**. In this PoC, only the MD Service Automation SW implementation is described.

· **Path Processing Function** manages path allocation by receiving data from the E2E Path Computation module and storing it in the Services DB.

· **Lifecycle Management Function** manages service provisioning and decommissioning, resource allocation via the Resource Configurator, and coordinates with the DT for measurement collection.

· **Service Exposure Function** provides information on local services within the MD scope by processing exposure requests and forwarding the retrieved data.

· **Services DB** a relational database storing all information about MD services, including service characteristics and links between E2E services, paths, and domains. It also supports the E2E Service Automation module.

REST interfaces handle interactions between modules:

· **E2E Path Computation REST Interface** receives path computation requests and forwards notifications.

· **E2E Service Automation REST Interface** manages provisioning/decommissioning requests and notifications.

·**Service REST Interface** notifies DT MS about service status, manages measurement collection, and exposes service data to external users.

· **Resource REST Interface** allocates service resources and retrieves configuration results.

· **Path REST Interface** allocates service resources and retrieves configuration results.

4 Path Computation

The Path Computation module establish deterministic services with bounded temporal KPIs. To provision a flow, the route from source to destination must be calculated before configuring the systems that enable data transfer. Path selection is essential to guarantee the desired KPIs, considering parameters like network topology, data plane technology, and resource status. Within the AICP, path computation occurs at two levels: E2E and the technological domain.

E2E Path Computation determines the sequence of technological domains (E2E path) through which the service is routed. This requires collecting abstracted topological information of the multi-domain data plane, represented as a connected graph of the abstracted views of different domains. The view includes capabilities information used to compute the E2E path in line with the KPIs of the service request. Path computation at the domain level focuses on a low-level path, considering the underlying data plane technology. It uses a weighted graph that factors in link lengths and estimated occupation from prior requests, generating a list of paths likely to meet the deterministic requirements while reducing the computational load on the DT predictions.

5 The Network Digital Twin

KPI estimation for requested and deployed non-TS flows is based on emulating a partition of the network scenario where the flow will be deployed. The emulation involves four components: *generators*, *queues*, *links*, and *sinks*, which accurately reproduce traffic and network conditions with fine granularity. Generators produce synthetic traffic at two levels: i) at the macroscopic level, traffic intensity is generated periodically (e.g., daily) with coarse resolution (e.g., one value per hour); and ii) at the microscopic level, fine resolution calculations (e.g., μs scale over 1 to 10 seconds) generate traffic flows based on probability distributions for inter-arrival time, packet size, and burst size. These bursts are aggregated for the desired number of users using distributions like Power Law or Weibull. Queue models, based on time-dependent models from [6], emulate TS-capable interfaces, pre-empting service rates for TS flows based on their duration and availability, with remaining time allocated to non-TS flows. Links simulate transmission delays, while sinks serve as flow endpoints for KPI evaluation.

The generated traffic propagates through the queuing system, producing metrics such as queued traffic, which are used to compute flow KPIs like E2E delay. The NDT provides two types of KPI estimations: i) E2E for non-TS requests, and ii) variations Δ, representing the KPI change for each flow if a new request is deployed along the same path. The NDT maintains two databases (DB): i) the network DB, which stores the current network topology and flow details, and ii) the monitoring DB, which holds real E2E performance measurements (delay, throughput, etc.) for existing flows. Fine-grained, segmented measurements are crucial for accurate KPI estimations produced by the DT.

6 Resource Configuration

The Resource Configurator module manages the network resources, ensuring efficient and dynamic allocation of network flows, VLANs, and traffic classification across PREDICT-6G domain (Ethernet, WiFi, and 3GPP). Each domain has its dedicated Resource Configurator to meet unique requirements and stringent KPIs.

A key feature is handling worst-case scenarios through Worst-Case Traversal Time (WCTT) analysis, ensuring the network meets KPIs even under maximum load conditions. This involves calculating the WCTT for each flow, considering all potential delays, to ensure the system operates within defined constraints in demanding situations. This includes considering factors such as network congestion, frame preemption, and the specific QoS mechanism in place (e.g., traffic shaping, priority queuing).

The WCTT analysis process includes: (1) **Network Simulation**: Simulates network traffic, including all flows and priorities, seeking for a configuration. (2) **KPIs Measurement**: Measures KPI metrics under worst-case conditions. (3) **Performance Evaluation**: Compares measurements against KPI constraints. (4) **Configuration Adjustment**: Adjust network configuration. If requirements are not complied, the process starts again.

Incorporating WCTT analysis ensures the network reliably meets performance requirements for critical data flows under worst-case conditions. The Resource Configurator can also integrate real-time updates and reconfigure network resources dynamically, using APIs for seamless integration across different domains.

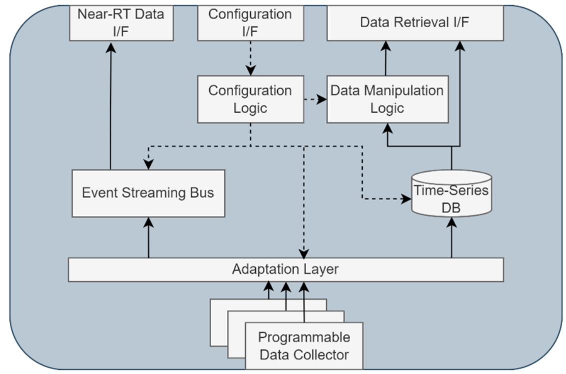

Figure 3: Data Collection and Management module design

7 Monitoring

The continuous monitoring of targeted metrics is fundamental to enabling the predictability of deterministic services. The Data Collection and Management (DCM) platform, as shown in Figure 3, is the AICP component designed for this purpose. It gathers metrics from heterogeneous data sources, normalizes them into a common format, and provides this data in near real-time via an Event Streaming Bus, storing it in a Time Series DB. Aggregated metrics such as minimum, mean, maximum values, and standard deviation are produced. These data are consumed by AI/ML analytic processes, the DT, other AICP components, local applications, and human users.

The DCM module uses a plugin-based mechanism to retrieve data from different sources and perform post-processing. A northbound interface (Configuration interface (I/F)) allows the creation and termination of Programmable Data Collectors on demand and manages data consumers. In a local Management Domain (MD), the module provides an API to configure plugins by establishing the KPIs targeted by the service. Data sources can be configured at runtime with adjustable sampling frequency, supporting both active (polling) and passive (push) monitoring. Telegraf instances handle data collection and the adaptation layer.

The AICP DCM module uses a plugin-based mechanism to retrieve data from different sources and perform post-processing. It exposes a northbound interface (Configuration I/F) to create and terminate Programmable Data Collectors on demand and manage data consumers. In a local MD, the module exposes an API to configure plugins by establishing the KPIs targeted by the service. Specific data sources and configurations for TSN can be found in Section 8.

Access to collected data is managed through authentication and authorization mechanisms, providing per-metric granularity. The DCM module dynamically creates/removes data consumers and configures access credentials via the Configuration I/F. Operations are supported by Apache Kafka for the Event Streaming Bus and InfluxDB for the Time Series DB, enabling data consumers to sign in and access both near real-time and historical data.

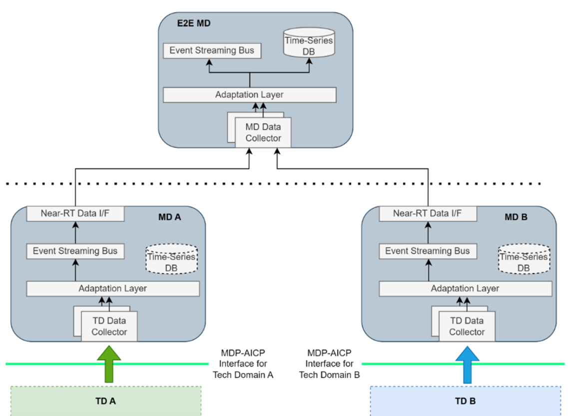

Figure 4: Hierarchical configuration of multiple instances of the Data Collection and Management module

In services spanning multiple technological domains, the DCM module instances are structured hierarchically (Figure 4). At the local MD level, metrics are collected from the technological domain through an Multi-domain Data Plane (MDP)-AICP interface, adapted to a specific format, and stored locally in a Time Series DB. The E2E DCM module aggregates data from these domain-specific instances, abstracts them into common data models, and publishes the data in the E2E level bus and Time Series DB. This setup supports the automatic management of E2E services across multiple domains.

The DCM module software is available at the PREDICT-6G GitLab repo [1].

8 Proof of Concept setup and results

The Proof of Concept (PoC) of the AICP integrates the components described previously to enable the automated provisioning of deterministic networking services over an Ethernet TSN Domain. The workflow, shown in Figure 5, begins when a service provisioning request is triggered. In Listing 1, the REST request used in the PoC follows the standard API for all MDPs.

```
{
    "e2e_service_id": <id>,
    "td_nodes": [
        ...
    ],
    "qos_characteristics": {
        "priority": 1,
        "td_reliability": 100,
        "td_delay": 400,
        "td_jitter": 42,
        "burst_arrival_time_window": {
            ...
        },
        "burst_completion_time_window": {
            ...
        }
    },
    "traffic_characteristics": {
        "direction": "bidirectional",
        "periodicity": false,
        "period": 0,
        "burst_size": 20,
        "maximum_flow_bitrate": 5000
    }
}
```

Listing 1: Service Provisioning Request

Upon receiving the request, the MD Service Automation component processes it in three phases: determining the path for service flows (steps 2-6 in Figure 5), allocating the service flow with the required deterministic characteristics (steps 7-10), and configuring the monitoring metrics (steps 11-13).

Determining the specific path to be allocated .
Service Automation MD processes the request (step 1) and forwards it to Path Computation MD (step 2) via the Path Processing submodule. Path Computation MD retrieves QoS characteristics and topology information (steps 3 and 4), computes the path (step 5), and forwards the result back to Service Automation MD (step 6), which stores it in the Services DB.

Requesting the Allocation of the service .
Service Automation MD requests flow allocation from Resource Configuration MD (step 7). In this domain, IEEE

[1] https://gitlab.netcom.it.uc3m.es/predict-6g/aicp/monitoring-and-data-collection.git

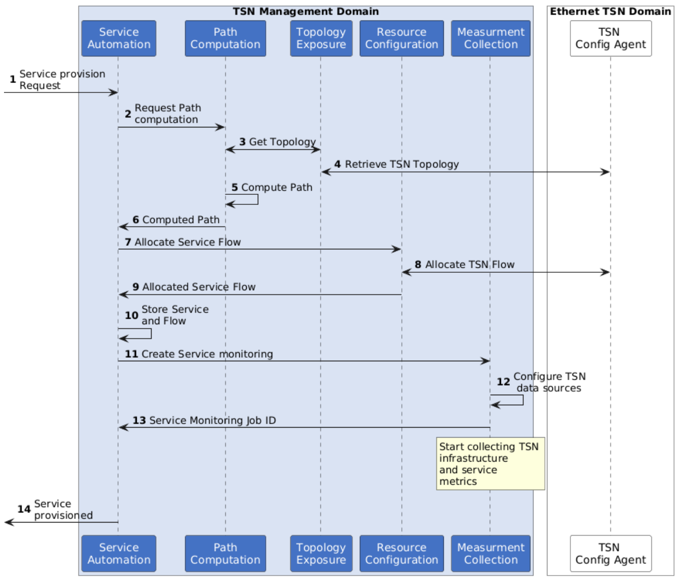

Figure 5: PoC Service Provisioning Workflow

802.11Qbv manages traffic scheduling to ensure priority traffic within designated windows. The Ethernet TSN Resource Configurator, using the Pegase library [2], creates the node topology and configures communication paths. This process runs in a separate thread for efficiency. Flow configurations are handled via REST API, which receives JSON requests from Service Automation MD and processes them to generate network configurations, with WCTT analysis, including scheduling and traffic class settings. Validated configurations are exported as Yang XML files and uploaded via the NetConf protocol [5] (step 8). The configurations are sent back to Service Automation MD (step 9), and MD Service information is updated in the Services DB. Resource Configurator manages traffic mapping and S-labels by communicating with DetNet border routers (step 10).

Configuring the monitoring jobs .
After the MD Service information is updated, Service Automation triggers service monitoring in step 11 by contacting the Measurement Collection MS via the Service REST Interface. This request includes the service ID and the KPIs of interest. The Measurement Collection MS creates the TSN data source in step 12 by setting up a Data Collector that polls a RESTful server implementing the monitoring API for the Ethernet TSN domain, based on the 3GPP TS 29.522 specification [1]. The data is translated into the Influx Line Protocol to be reported through both Apache Kafka and InfluxDB. Once the data source is configured, the Measurement Collection MS sends the Service Monitoring ID back to the Service Automation MS in step 13 through the Service REST Interface to confirm the activation of monitoring.

Upon confirmation, the Service Automation MS finalizes the provisioning of the MD service in step 14 via the E2E Service Automation REST Interface.

9 Conclusions

In this paper, we showcased the development and implementation of an AICP for deterministic networks, integrating AI, digital twins, and telemetry to manage complex network environments effectively. The AICP framework, complemented by IETF DetNet, ensures low-latency and reliable communication. Our PoC showcases the practical viability of this approach, highlighting its potential to transform industrial communication networks by dynamically adapting to varying network demands.

The hierarchical structure for real-time data collection and analysis enables proactive performance adjustments, ensuring continuous network reliability. The dynamic and flexible configuration process underscores the framework's adaptability and scalability. This work lays a solid foundation for future advancements in intelligent network management, promoting more efficient and resilient network infrastructures capable of meeting stringent KPIs requirements.

Acknowledgments

This work has been partially funded by the European Commission Horizon Europe SNS JU PREDICT-6G (GA 101095890) Project and by the Spanish Ministry of Economic Affairs and Digital Transformation and the European Union-Next Generation EU through the UNICO 5G I+D 6G-EDGEDT-4 (TSI-063000-2021-124).

References

[1] 3GPP. [n. d.]. 3GPP TS 29.522 V18.4.0; 5G System; Network Exposure Function Northbound APIs. https://www.3gpp.org/ftp/Specs/archive/29_series/29.522/
[2] RealTime at Work. 2024. RTAW-Pegase. https://www.realtimeatwork.com/rtaw-pegase/. Accessed: 2024-07-16.
[3] Pietro G Giardina et al. 2023. A hierarchical AI-based control plane solution for multi-technology deterministic networks. In *Proceedings of the Twenty-fourth International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing*. 316–321.
[4] Howon Lee et al. 2023. Towards 6G Hyper-Connectivity: Vision, Challenges, and Key Enabling Technologies. arXiv:2301.11111 [cs.IT] https://arxiv.org/abs/2301.11111
[5] ns et al. 2011. Network Configuration Protocol (NETCONF). RFC 6241. https://doi.org/10.17487/RFC6241
[6] Marc Ruiz et al. 2018. CURSA-SQ: A methodology for service-centric traffic flow analysis. *Journal of optical communications and networking* 10, 9 (2018), 773–784.
[7] Balazs Varga et al. 2024. Deterministic Networking (DetNet) Packet Replication, Elimination, and Ordering Functions (PREOF) via MPLS over UDP/IP. RFC 9566. https://doi.org/10.17487/RFC9566