Deployment of Multi-Agent System Pipelines for Near-Real-Time Operation of 6G Network Services

Pol González, Sima Barzegar, Marc Ruiz, and Luis Velasco*

Optical Communications Group (GCO), Universitat Politècnica de Catalunya (UPC), Barcelona, Spain

**e-mail: luis.velasco@upc.edu*

ABSTRACT

A Machine Learning Function Orchestrator deploying and reconfiguring Multi-Agent Systems (MAS) pipelines to support near-real-time control of network services is showcased. An architecture to orchestrate the dynamic deployment of agents has been defined by integrating different components. Near-real-time operation of an optical system controlled by a distributed MAS fed with telemetry is presented as use case.

Keywords: Distributed intelligence, Multi-agent systems, Telemetry, Orchestration

1. INTRODUCTION

Near-real-time autonomous network operation is required to deal with the expected large traffic dynamicity and provide the stringent performance required by beyond 5G and 6G network services (NS) [1]. Solutions for autonomous operation running in a centralized controller have the potential to greatly reduce costs, but they might lead to inefficient resource utilization because of their long response times. To minimize response time, control algorithms (agents) might be executed as close as possible to the network devices [2]. Near-real-time control autonomous network operation, e.g., based on Machine Learning (ML) algorithms, is limited by the centralized nature of software-defined networking (SDN). Therefore, new approaches need to be considered, evaluated, and assessed to control highly dynamic NSs to provide the required stringent performance with limited overprovisioning. One of these approaches consists in delegating the near-real-time decision-making to agents deployed close to the data plane to minimize response times, while providing the required overall supervision of the process [3]. However, this approach needs the definition of procedures to deploy the multi-agent systems (MAS) pipeline connecting the agents, as well as to solve the security issues related to such distributed approach.

This paper describes the deployment of a MAS pipeline consisting of: *i*) a set of intelligent agents deployed in distant locations that coordinate among them for the near-real-time control of a NS; and *ii*) the required communication infrastructure. We target the control of a 6G NS requiring both point-to-point (P2P) and point-to-multipoint (P2MP) packet connectivity, supported by an optical P2MP connection based on Digital Subcarrier Multiplexing (DSCM) [4]. Specifically, this paper will show the operation of: *i*) a ML Function Orchestrator (MLFO) [5] coordinating the deployment of agents and their configuration with the help of a service management and orchestration system (SMO); and *ii*) a distributed telemetry processing [6] and data exchange among the agents.

This work describes a control and orchestration system capable of deploying MAS pipelines connecting distributed agents for the near-real-time control of highly dynamic NS. Such MAS pipelines are created per NS; agents participating in the control need to be deployed dynamically in the network infrastructure together with the required connectivity. The main contributions of this paper are: present the *i*) autonomous operation of agents making near-real-time decisions based on measurements collected and processed in different locations and supervised by a centralized entity; and describe the *ii*) deployment of MAS pipelines, including agents and communication, which optimal design is computed, once the NS has been deployed. This work extends the ideas and topics presented in [1] and [3].

2. ARCHITECTURE AND SETUP

This paper aims to show the deployment of a MAS pipeline for the near-real-time control of a NS. For illustrative purposes, we assume that the NS requires P2P and P2MP communications. The authors in [7] showed that services requiring P2P and/or P2MP can take advantage of optical P2MP connectivity based on DSCM (see Fig. 1). A set of Nyquist subcarriers (SC) contiguous in the optical spectrum is assigned to each leaf optical transponder (TpA and TpB in Fig. 1), so each one can communicate with the hub (TpZ) independently of the others. For explanatory purposes, optical telemetry data being collected and processed by the agent on TpZ is used to estimate the pre-FEC BER of each of the SCs. As in [1] for P2P, leaf Tps configure each SC individually in terms of modulation format and bit rate as a function of the input traffic; in this paper they use the estimated BER

The research leading to these results has received funding from the Smart Networks and Services Joint Undertaking under the European Commission Horizon Europe SEASON project (G.A. 101096120), by the AEI through the IBON project (PID2020-114135RB-I00), and by the ICREA institution.

979-8-3503-7732-3/24/$31.00 ©2024 IEEE

to decide which configuration fits better to each SC for the current optical connection. Finally, note that SCs are configured based on the expected input traffic from the virtualized network function (VNF) connected to the leaf Tp. To reduce overprovisioning and increase the accuracy of the prediction, such traffic estimation is performed for the short term, which is the reason why near-real-time operation is strictly needed.

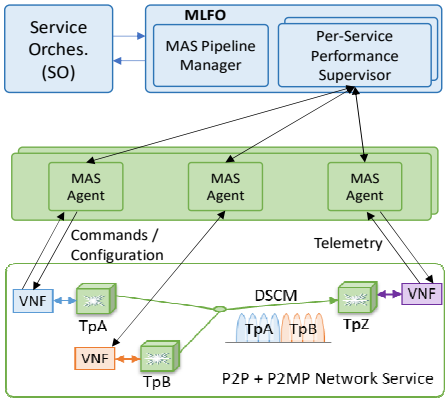

Fig. 1 MAS pipeline to control a NS

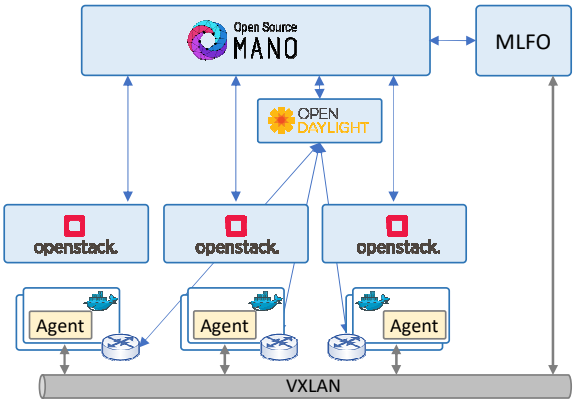

Fig. 2 Overall architecture of the setup

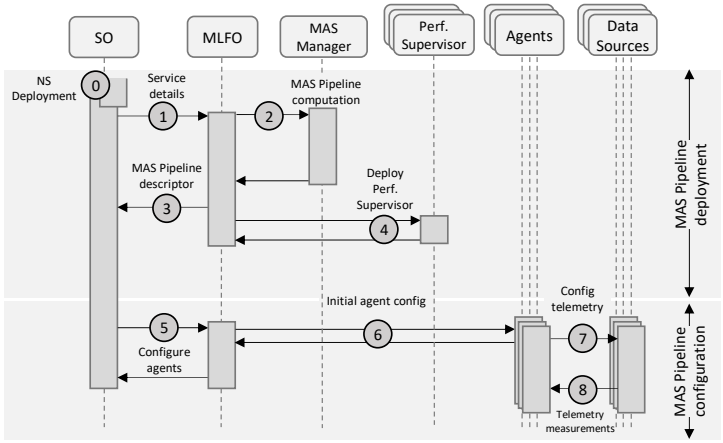

Fig. 3 MAS Pipeline deployment workflow

Fig. 1 shows the proposed architecture of a MAS pipeline in control of a network service. The MLFO is part of the Intelligent Distributed Control system and acts as an independent orchestrator that manages MAS pipelines by placing agents in different locations in the network and connecting them to create an overlay system for the near-real-time control of single services. The MLFO is composed by two main components: i) the MAS Pipeline Manager and ii) a Per-Service Performance Supervisor. The MAS Pipeline Manager receives requests from the Service Orchestrator, computes the optimal deployment, and issues back the deployment or reconfiguration of the network service. A Performance Supervisor is deployed per-service and is in charge of receiving data from the MAS agents; this data may include commands, telemetry or petitions to add or remove resources from the existing MAS pipeline.

Fig. 2 presents the setup for this demonstration where, for the sake of clarity, the control of the optical network is not represented. Three different locations are considered where TpA, TpB, and TpZ are installed. Each location includes computing resources, so a local virtualized infrastructure manager (OpenStack) is in charge of automating the deployment of VNFs. OpenDaylight (ODL) SDN controller is on top of the packet network and used to create the connectivity for the ML pipeline. Opensource MANO (OSM) is the selected orchestration system in charge of the deployment of NSs. A MLFO decides the locations where agents need to be deployed and how they need to be connected.

3. MAS DEPLOYMENT AND OPERATION

Two workflows are sketched in Fig. 3 and Fig. 4, the first one describing the initial deployment process of a fresh network service and the second one the reconfiguration process of an existing one.

The deployment workflow consists of two phases: i) MAS pipeline deployment; and ii) MAS pipeline configuration. The SMO initiates the workflow after the NS is deployed (0). The SMO starts with the MAS pipeline deployment and requests the definition of a MAS pipeline for the NS and provides its details, including the location of the VNFs (1). Based on the NS details and the requirements of the MAS pipeline in terms of delay and throughput among the agents, as well as the required IT resources of the agents, the MLFO computes a graph

with the resources in the network infrastructure that meet the requirements and can be used to support the MAS pipeline (2). With such data, the MLFO computes the optimal MAS pipeline design and sends back a descriptor containing the location where the agents need to be deployed, the VM image to be installed and the connectivity to be created. A list of iterations is generated that includes the communication of OSM with the OpenStack

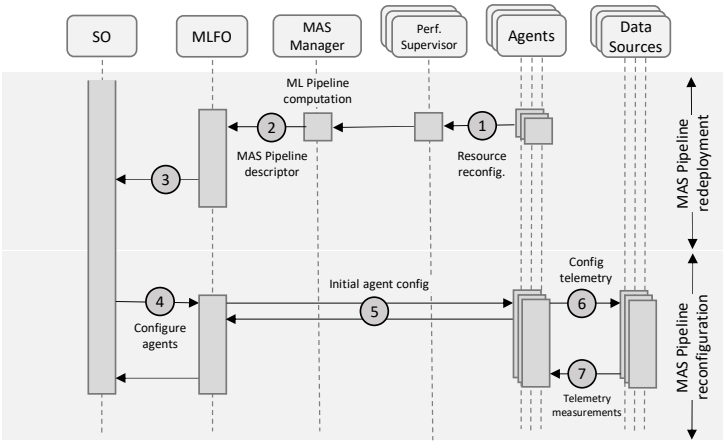

Fig. 4 MAS pipeline reconfiguration deployment

managers for the deployment of the agents encapsulated into virtual machines (VM) (3), and with the SDN controller for managing the connectivity (not shown). The MLFO deploys the performance supervisor for this particular MAS pipeline (4). Once the agents are running and the connectivity is available, the MAS pipeline is deployed and the configuration phase starts (5). When the MLFO receives the request to configure the agents, it sends the initial configuration that includes the addresses of the VNFs and that of the other agents, as well as the algorithms that every agent runs (6). At this time, the deployment of the NS ends from the viewpoint of OSM.

The agents send a request to the device they have been assigned to start sending telemetry measurements (7). Devices start to stream telemetry with a specific configuration including the metrics being exported and the telemetry period (8).

The reconfiguration workflow also consists of two phases: i) MAS pipeline redeployment; and ii) MAS pipeline reconfiguration. One of the MAS pipeline agents sends a request to the particular Performance Supervisor of that MAS pipeline asking for more resources (1). The request is processed and forwarded to the MAS Manager, where a new MAS pipeline is computed based on the resources asked (2). This new petition could lead to the deployment of an extra agent or the redeployment of any of the existing ones. The MAS pipeline descriptor is sent to OSM and a series of iterations is generated that includes the communication of OSM with the OpenStack managers for the deployment of the agents encapsulated into virtual machines (VM) (3), and with the SDN controller for managing the connectivity (not shown). Once this process is completed the rest of the workflow follows the same procedure as the deployment workflow.

4. MAS FOR NEAR-REAL-TIME NETWORK OPERATION

Once deployed, MAS agents can be used for controlling or monitoring data plane components and improving the performance of centralized solutions by reducing the response time. Control loops can be fed with telemetry measurements and decisions can be taken autonomously based on this metrics. Being able to collect and process measurements from closer devices could also result in reducing the amount of traffic used to convey that data to centralized systems for further analysis. In addition, delegating the control of the devices to the agents could help reduce the complexity of centralized systems and distribute the required network and compute resources [8]. NSs demanding a high dynamicity can benefit from these architectures due to their short response time and their on-demand dynamic reconfiguration and scalability.

Dynamic flow routing is a use case of near-real-time network operation that requires the interaction between telemetry agents and flow agents in charge of making routing decisions, which eventually leads to the deployment of a distributed MAS. Among different routing policies, multi-path routing introduces flexibility in the design and operation of the network by allowing operators to split traffic demands into multiple streams that are routed independently of each other to the destination. A possible routing policy is to evenly split each traffic flow among all available routes. However, such a strategy might fail under dynamic network conditions that can generate congestion in some routes and consequently, lead to flows QoS degradation. Moreover, routes might have different utilization costs, and hence, the percentage of traffic sent through each route is a complex decision that needs to be dynamically tuned in order to meet robust QoS performance with overall minimum cost.

In [3], the authors showcased the near-real-time operation of an optical packet network controlled by a distributed MAS. The MAS combines: i) pervasive INT agents supported on P4-based components; and ii) multi-

flow routing agents that are used to dynamically adjust multi-path flow routing policies in the packet nodes with the objective to guarantee the target QoS performance. Hence, flow routing operation is controlled by a set of heterogeneous agents that are fed with telemetry data collected from P4 switches. This work aimed at experimentally assessing the feasibility of a distributed MAS fed with telemetry measurements to perform near-real-time flow routing operation. For illustrative purposes, Fig. 5 shows an example where several traffic flows (Fi to Fk), each following a multi-path routing strategy, enter and leave a network at different packet nodes. In the example, traffic flow Fi (from R1 to R5) can follow three different routes, where p1 and p2 are multi-hop paths on the packet network, whilst p3 uses an optical bypass connecting R1 and R5 through the underlying optical network.

As in [9], we assume that traffic flows are splitable, i.e., they consist of a large number of sub-flows that can be routed independently. The objective is to find the flow routing policy that balances the incoming traffic of the flows among the available paths, so to ensure per-flow QoS (specifically e2e delay). Such routing policy varies as a function of the incoming traffic of the flow and the network conditions, i.e., the traffic of the rest of the traffic flows in the network. Therefore, the routing policy decision making process is continuously carried out based on the incoming traffic and the e2e delay measurements that allow evaluating the quality of the decision making. Note that the state of the network is known, and it is indirectly represented by e2e delay measurements for the traffic flow. In this paper, the authors rely on the INT functionality provided by the P4 switches to measure packet delay. Specifically, a P4 collector that collects, aggregates, and provides statistics of the delay measured by the switches supporting the traffic flows is showcased. Once the P4 collector preprocesses the QoS measurements, they are sent to a telemetry agent, which is in charge of producing flow telemetry statistics that are sent to the flow agent deployed at the source node, where flow routing policy decisions are made.

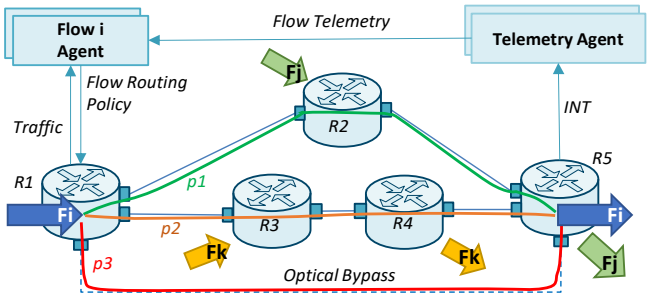

Fig. 5 MAS for dynamic flow routing scenario

The distributed system consists of several interconnected software components. Specifically, the telemetry agent, which is part of the distributed telemetry architecture, includes: i) a telemetry collector that periodically receives telemetry data from the P4 collector; and ii) a per-flow telemetry processor. The role of these telemetry components is to compute the required measurements and statistics that characterize the current QoS of the traffic flow, from the received INT telemetry. In addition, flow agents are grouped in a single module per location named multi-flow agent. In this case, the multi-flow agent at R1 includes flow agents for both f1 and f2. Note that one single flow agent makes routing decisions for each traffic flow. The manager running inside multi-flow agents has the role of collecting and distributing flow telemetry data and local input traffic data to the flow agents, as well as to push the flow routing policies computed by flow agents to the P4 switch.

5. CONCLUSIONS

An architecture for deploying MAS pipelines has been described, along with the components composing the MLFO. Two workflows have been stated, the first describing the initial deployment of a MAS pipeline and the second describing the reconfiguration of an existing one. A use case for the control of a network service using a MAS pipeline is presented, including network reconfiguration based on delay and traffic measurements being collected.

REFERENCES

[1] P. González *et al.,* "Deployment of Secure ML Pipelines for Near-Real-Time Control of 6G NS", OFC, 2024.
[2] L. Velasco *et al.,* "Autonomous and Energy Efficient Lightpath Operation based on DSCM," JSAC, 2021.
[3] P. González *et al.,* "Distributed MAS fed with Telemetry Data for Near-Real-Time Service Operation", OFC, 2024.
[4] H. Shakespear-Miles *et al.,* "Centralized and Distributed Approaches to Control Optical P2MP Systems Near-Real-Time," JOCN, 2024.
[5] A. Wassington *et al*., "Implementing a Machine Learning Function Orchestration," ECOC, 2021.
[6] L. Velasco *et al*., "Distributed Intelligence for Pervasive Optical Network Telemetry," JOCN, 2023.
[7] M. Iqbal *et al*., "Supporting Heterogenous Traffic on top of Point-to-Multipoint Light-Trees," MDPI Sensors, 2023.
[8] L. Velasco *et al*., "Monitoring and Data Analytics for Optical Networking: Benefits, Architectures, and Use Cases," IEEE Network, 2019.
[9] S. Barzegar *et al*., "Autonomous Flow Routing for Near Real-Time Quality of Service Assurance," TNSM, 2024.