On the Integration of AI/ML-based scaling operations in the 5Growth platform

On the Integration of AI/ML-based scaling operations in the 5Growth platform

J. Baranda*, J. Mangues-Bafalluy*, Engin Zeydan*, L. Vettori*, R. Martínez*, Xi Li¶, A. Garcia-Saavedra¶, C.F. Chiasserini⋆, C. Casetti⋆, K. Tomakh‡, O. Kolodiazhnyi‡, C. J. Bernardos&
*Centre Tecnològic de Telecomunicacions de Catalunya (CTTC/CERCA), Spain; ¶NEC Laboratories Europe, Germany;
⋆Politecnico di Torino, Italy; ‡ Mirantis, Ukraine; &University Carlos III, Spain

***Abstract*—The automated assurance of vertical service level agreements (SLA) is a challenge in 5G networks. The EU 5Growth project designs and develops a 5G End-to-End service platform that integrates Artificial Intelligence (AI) and Machine Learning (ML) techniques for any decision-making process in the management and orchestration (MANO) stack. This paper presents the detailed architecture and first prototype of the 5Growth platform taking AI/ML-based network service auto-scaling decisions. This also includes the modification of the ETSI network service descriptors for requesting AI/ML-based decisions for orchestration problems and the integration of a data engineering pipeline for real-time data gathering and model execution. Our evaluation shows that AI/ML-related service handling operations (1-2 s.) are well below instantiation/termination procedures (80/60 s., respectively). Furthermore, online classification can be performed in the order of hundreds of milliseconds (600 ms).**

***Index Terms*—AI/ML, Scaling, NFV/SDN, Automated network management, End-to-End Service Orchestration**

I. Introduction

5G has opened the door to vertical industries transformation, enabling the advent of innovative digital use cases through Network Function Virtualization (NFV), Software Defined Networking (SDN) and Network Slicing. These allow simultaneous support of vertical services over a shared infrastructure. However, the automated assurance of vertical service quality under the dynamics of the available infrastructure resources and service demands is still a challenge in practice.

To this aim, the EU 5Growth (5Gr) project [1] extends the 5G End-to-End service platform developed in the EU 5G-Transformer (5GT) project [2] towards an AI-driven automated network management platform. With the addition of the Artificial Intelligence (AI) and Machine Learning (ML) platform (5Gr-AIMLP), and its integration with the different building blocks of the 5Gr platform, a closed-loop automation and zero-touch service and network management system is being developed to: (*i*) manage vertical service life cycle and Service Level Agreement (SLA) assurance; (*ii*) adapt to dynamic changes and/or anomalies in the infrastructure; and, (*iii*) control radio access, transport, core and cloud/edge resources, across multiple technologies, vendors and domains.

In this paper, we present the detailed operation and a first operational prototype of the high-level architecture design first introduced in [3]. It features, (i) an extension to the ETSI NFV-IFA 014 [4] network service descriptor (NSD) to express the need and the required metrics to perform AI/ML-based decisions for a given network management problem (e.g., scaling), (ii) the integration of an AI/ML-based solution to handle the scaling of an NFV-Network Service (NFV-NS), (iii) the integration of a data engineering pipeline to collect, ingest, and process/analyze data, which can be used for both offline model generation from training data and online (real-time) classification over streaming data, (iv) automated use of such pipeline with a classification model to decide the instantiation level (IL) to which the service should scale in/out.

For validation purposes, we built an experimental proof-of-concept using a virtual Content Delivery Network (vCDN) NFV-NS to profile the impact of AI/ML-related operations during the NFV-NS life cycle management operations.

In brief, this paper is aligned with the main architectural concepts behind some of the previous work explained in section II, and it represents a first realization and quantitative evaluation of these ideas that fully integrates a monitoring platform, a AI/ML model execution, and a MANO stack for automated scaling decisions.

II. Background and related work

The 5Growth project aims at developing a modular platform that is fully automated and yet highly flexible. Through a data-driven approach and leveraging AI/ML algorithms, the platform creates, re-configures, and manages end-to-end slices to fulfill the service requirements, while minimizing the consumption of network, computing, and storage resources.

Standardization groups, such as the ETSI Zero touch network & Service Management (ZSM) [5] and the Experiential Networked Intelligence (ENI) [6], are currently working along the same directions, although no complete working solution is yet available. In particular, ETSI ZSM envisions new solutions to realize an agile, fully automated system to support new business opportunities enabled by network slicing, while ETSI ENI focuses on the use of AI for the management of network slices as well as of security issues.

O-RAN specifications [7] are also relevant to our work. They envision the use of RAN-related data to learn and improve radio and network performance. O-RAN defines two ML hosts, one executing the ML model training and evaluation, the other performing inference. Their location in the network architecture depends on the type of the adopted ML technique

This work has been partially funded by the EU H2020 5Growth Project (grant no. 856709), by MINECO grant TEC2017-88373-R (5G-REFINE) and Generalitat de Catalunya grant 2017 SGR 1195.

(e.g., supervised/unsupervised learning or reinforcement learning) and on the time scale of the decision process.

Similar research issues are also explored by other EU projects, such as 5G-CLARITY [8], SELFNET [9], and 5G-ZORRO [10]. In particular, [8] is developing an AI-based network management system that provides an intent-based interface for network configuration. SELFNET instead targets self-organizing network management mechanisms leveraging the NFV/SDN paradigms jointly with AI/ML technologies. Finally, 5G-ZORRO focuses on zero-touch management in a multi-stakeholder scenario, using distributed ledger technologies, with particular emphasis to security and trust.

The scope of 5Growth is slightly different in some aspect from other projects like 5G-CLARITY, since the goal is to exploit AI/ML capabilities for automated network management through the MANO stack itself and not to expose them to human operators through intent-based interfaces. Nevertheless, at a high-level, the work in this paper is similar in scope to some of the above architectural concepts. In this sense, it presents a working solution/prototype that goes all the way through the data engineering pipeline from monitored data generation to decision making. It also represents a preliminary instantiation of O-RAN architectural concepts, though not the exact interfaces (still under definition). That is, it includes the split between training and model execution, though not just focused on the RAN but with an end-to-end scope. Specifically, the 5Gr-AIMLP performs the model training, and other building blocks of the 5Gr architecture, like the Service Orchestrator (5Gr-SO), performs the inference.

III. Operation and Implementation

This section presents the evolution introduced by the 5Gr platform with respect to the 5GT architecture to integrate AI/ML-based scaling operations. First, we present the tools used to integrate the data pipeline management and then, we describe the integration of these tools and the additional changes throughout the 5Gr platform. Last, we describe the workflow to perform AI/ML-based scaling operations.

A. Data pipeline management

In a data engineering pipeline creation process, a series of transformations and actions are performed. In a traditional setting, first, data is consumed by data ingestion frameworks. Then, data processing and analysis steps are executed. After that, the analyzed data is stored and later it can be visualized by data storage and visualization frameworks.

The tools used to execute the data pipeline in the current 5Growth prototype are as follows. Apache Kafka [11] is used for data ingestion purposes and Apache Spark [12] is used for both offline model generation from training data and online (real-time) classification over streaming data. Apache Livy [13] is used as a REST service to submit/terminate Apache Spark jobs via REST-API from the 5Growth platform.

During the training phase, a model is created in the 5Gr-AIMLP. To create a model inside Apache Spark, a pipeline object is created for a better internal parallelism and resource utilization [14]. For this work, we created an offline training application that generates a random forest classifier model (via the *.fit()* function of Spark API). This model will be used by the Apache Spark instance running at the inference host to perform online classification during NFV-NS runtime, when the streaming application consumes data directly from Apache Kafka topics (via the *.transform()* function of Spark API).

B. 5Growth platform evolution

The evolution proposed by 5Growth to support AI/ML-based decisions with respect to the 5GT platform affects all layers and components of the stack. In this work, among these components, the most evolved one is the 5Gr-Service Orchestrator (5Gr-SO), which coordinates the end-to-end orchestration and lifecycle management of NFV-NSs. Next, we present the changes introduced in the 5Growth platform, paying special attention to the ones introduced in the 5Gr-SO to support the AI/ML-based scaling operation.

```
"aimlRules": [
    { "ruleId": "aiml_rule1",
      "problem": "scaling",
      "nsMonitoringParamRef": ["mp1", "mp2"]} ]
```

Listing 1: New IE defined for AI/ML-based operations

Prior to starting with this description, it is worth mentioning that in this work, we propose a new AI/ML information element (IE) (see Listing 1 for an example) extending the ETSI NFV-IFA 014 [4] NSD template. This new IE expresses the need of AI/ML-based decisions for a given MANO problem ("scaling") and specifies the metrics out of the ones already defined in the NSD field *"monitoredInfo"* required by this AI/ML problem (*"mp1"* and *"mp2"*) to perform its decisions.

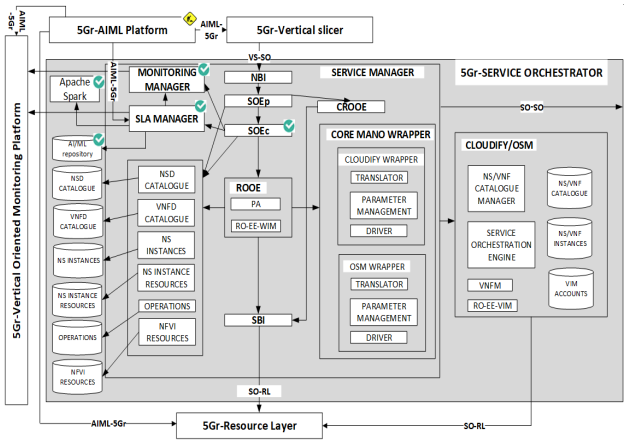

Fig. 1: 5Gr-Service Orchestrator architecture highlighting changes to support AI/ML-based scaling operations

Fig. 1 shows the 5Growth stack with a focus on the architecture of the 5Gr-SO. The marked submodules are the ones that have been modified/added with respect to the 5GT-SO [15] architecture to support AI/ML-based scaling operations:

•*SLA Manager*: This submodule is responsible for handling the NFV-NS SLA compliance and triggering the scaling process in case of SLA violation. Initially, the detection was based upon an alerting system configured through the 5Gr-Vertical

oriented Monitoring System (5Gr-VoMS). Now, it has been extended to orchestrate all the operations to handle the AI/ML-based scaling operation when including the described IE in the NSD. For this purpose, it interacts mainly with the Monitoring Manager submodule, the 5Gr-AIMLP, and with Apache Spark. As mentioned in Section II, the inference operation is done in the 5Gr-SO, in line with the O-RAN specifications, using Apache Spark and orchestrated by the SLA Manager.

•*Monitoring Manager*: This submodule interacts with the 5Gr-VoMS to configure the collection of performance metrics expressed in the NSD and configuration of visualization dashboards. Now, its interaction with the 5Gr-VoMS has been extended to configure dedicated Apache Kafka topics where the monitoring data expressed in the AI/ML IE of the NSD is inserted to be processed by the Apache Spark job.

•*Service Orchestrator Engine child (SOEc)*: this submodule of the SOE, handling the orchestration of regular (i.e., non-composite) NFV-NSs, has been extended to interact with the new capabilities of the SLA Manager module.

Furthermore, the 5Gr-SO architecture includes an instance of Apache Spark and the *AI/ML repository*, which is a new submodule where the SLA Manager stores the requested AI/ML models and processing routines obtained from the 5Gr-AIMLP. Some aspects of the 5Gr-AIMLP and of the formal definition of the interface with the 5Growth MANO stack are still under development. These models and processing routines are required to launch the Apache Spark jobs in charge of checking the SLA compliance and deciding on the appropriate instantiation level of the NFV-NS for given network conditions.

Finally, the 5Gr-VoMS has also been extended to support the AI/ML-based scaling operation. In addition, to host the Apache Kafka platform, the 5Gr-VoMS adds a REST-API to create *"data scraper"* elements. These elements, upon the request orchestrated by the SLA Manager, filter out the collected monitoring data for an NFV-NS by the Prometheus platform used in the 5Gr-VoMS and insert them in the requested Kafka topic to be ingested by the Apache Spark streaming process.

The source code of the 5Gr-SO (and the rest of 5Gr platform) is available as open source under the Apache v2.0 license on GitHub [16].

C. 5Growth AI/ML-based scaling workflow

Fig. 2 presents the workflow followed by the 5Gr-SO to configure the AI/ML-based scaling operation. This workflow takes as starting point the last step of the instantiation process (after VNFs have been allocated and their interconnections and monitoring jobs have been configured), when the SOEc contacts the SLA Manager. The workflow is as follows:

1) The SLA Manager checks the existence of an AI/ML IE in the NSD for a *scaling* problem. The next steps happen upon a positive confirmation.

2) The SLA Manager contacts the Monitoring Manager to configure a dedicated data topic in the Apache Kafka instance run by the 5Gr-VoMS to insert the required monitoring information expressed to handle the AI/ML-based scaling operation.

3) The SLA Manager, through the Monitoring Manager, creates the "data scrapers" elements at the 5Gr-VoMS to filter out the monitoring data specified at the AI/ML IE.

4) The SLA Manager would contact the 5Gr-AIMLP (under development) to download the required model and its associated streaming application and stores them in the AI/ML repository.

5) The SLA Manager launches the Apache Spark streaming job. In this work, the SLA manager publishes the current NFV-NS instantiation level (IL) in the dedicated Apache Kafka topic created in step 2) to give the appropriate context to the Apache Spark application.

6) Finally, the SLA Manager saves the AI/ML-based information (Apache Kafka topic, data scrapper, Apache Spark job references) in the NFV-NS instance database.

7) From this point on, periodically, the Apache Spark job ingests the data requested in step 3) from the Kafka topic, performs online classification, and notifies the result (i.e., the best IL given the current context) to the SLA Manager.

8) The SLA Manager checks the notification, and if the received IL differs from the current IL, it stops the Apache Spark Job and triggers the scaling operation through the northbound interface of the 5Gr-SO.

In case of scaling, the 5Gr-SO proceeds to create/terminate the required VNFs instances as indicated by the new IL, updating the interconnections between VNFs and the performance monitoring jobs accordingly, as described in [2]. As the last step of the scaling procedure, added by the integration of AI/ML-based operations, the SOEc contacts the SLA Manager, which, when retrieving the information from the NFV-NS instance database, repeats steps from 5) to 7) to close the loop.

The 5Growth architecture could also accommodate other AI/ML-based problems specified in the presented IE. For that, we would need to evolve the submodules in charge of orchestrating the corresponding instantiation operation to support the interaction with the 5Gr-AIML platform and the streaming functions of Apache Spark, similarly to the approach herein described for the scaling operation use case.

IV. Experimental Proof of Concept Validation

This section presents the validation of the AI/ML-based close-loop scaling operation presented in the previous section. For this validation, we use a virtual Content Delivery Network (vCDN) NFV-NS to profile the associated time of the added AI/ML operations during instantiation, run-time, and termination of the NFV-NS using the 5Growth platform. In this initial work, we are running a simple offline-trained AI/ML model that takes the vCPU consumption of the critical VNFs of the deployed NFV-NS as well as the current Instantiation Level (IL) to decide on the best available IL described in its NSD.

A. Scenario Setup

The experimental evaluation has been performed using an instance of the 5Growth platform made up of the 5Gr-Vertical Slicer (5Gr-VS), the 5Gr-Service Orchestrator (5Gr-SO), the 5Gr-Resource Layer (5Gr-RL) and the 5Gr-Vertical oriented Monitoring Service (5Gr-VoMS). This instance of the

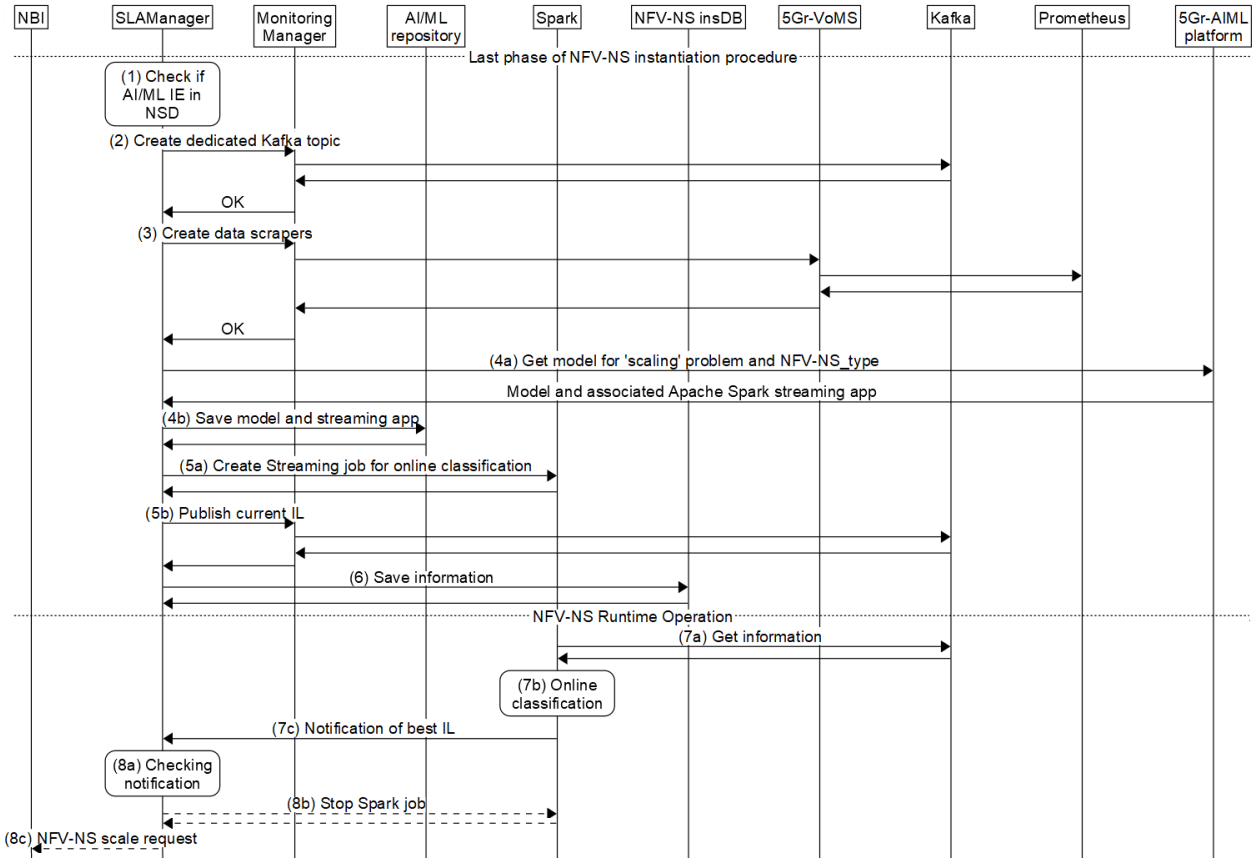

Fig. 2: 5Growth workflow to configure AI/ML-based scaling operation

5Growth platform is completed with: (*i*) an instance of OSM Release 6 and an instance of Apache Spark (version 2.4.0) paired with the 5Gr-SO, and (*ii*) two OpenStack-based Virtual Infrastructure Managers (VIMs) and an ABNO-based WAN Infrastructure Manager (WIM) controlling a multi-technology (wireless, optical) transport network under the control of the 5Gr-RL. The NFV-NS under evaluation is a vCDN, which, in its initial IL, consists of three different VNFs: a webserver, a cache server, and an origin server [2]. This NFV-NS presents another IL, where a new instance of the cache server is deployed to avoid degradation in the NFV-NS performance with an excessive number of users.

B. Profiling the integration of AI/ML operations

This section presents a profiling of the added operations in the 5Gr-SO to perform AI/ML-based scaling operations. This profiling aims at studying the impact of such operations during the instantiation, run-time, and termination procedures of the vCDN NFV-NS presented previously. Fig. 3 shows the statistical behaviour for such operations. The box stretches from the 20th to the 80th percentiles, including the median value. The whiskers represent the maximum and minimum values. Each experiment is repeated 10 times.

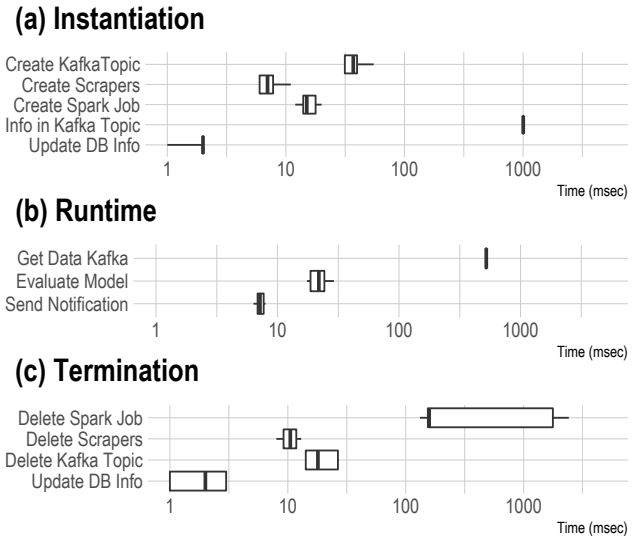

Fig. 3: Profiling of the impact of AI/ML operations in: (a) instantiation; (b) runtime; (c) termination procedures

Fig. 3(a) shows the operations related to the configuration of the data-pipeline tools during the NFV-NS instantiation phase as in steps 2 to 6 of the workflow presented in Section III-C. The time needed to configure them is in the order of tens of milliseconds, being the creation of the Apache Kafka topic the most time consuming operation (37.8 ms on average). This can be partly explained by the fact that the Apache Kafka is owned by the 5Gr-VoMS platform, which is located in another computer. After the Apache Spark streaming job is created, the 5Gr-SO writes additional information in the Kafka topic to inform the Apache Spark job of the current NFV-NS IL (step 5). This information will be updated after every scaling action. This is the most time consuming action of all included in Fig. 3(a), presenting an average value of 1 second. Nevertheless, the sum of all these operations has a limited impact in the total NFV-NS instantiation time, which is in the order of 80 seconds, as analyzed in [2]. As a final remark for the instantiation procedure, it is worth mentioning that this graph does not include the interaction of the 5Gr-SO, working as Inference Host, with the 5Gr-AIMLP because it is still under definition. As explained in step 4) of the presented workflow, the 5Gr-SO would download the AI/ML model and the associated file to run the Apache Spark streaming job. In particular, for this evaluation, we generated offline a random forest classifier model using Apache Spark. This generated model and its associated streaming application, which have a size of 270KB, are placed manually in the AI/ML repository. Thus, the time to exchange such files between the 5Gr-SO and the 5Gr-AIMLP would be in the order of milliseconds as well.

Fig. 3(b) presents the main operations performed by the Apache Spark streaming job to determine the required NFV-NS IL based on the information available in the dedicated Apache Kafka topic (step 7 of the presented workflow). These operations are: (*i*) the reading of the required monitored information from the created Apache Kafka topic, (*ii*) the evaluation of the AI/ML model based on this data and the knowledge of current IL, and, (*iii*) the interaction between Apache Spark and the 5Gr-SO to communicate the decision. While operations (*ii*) and (*iii*) are in the order of tens of ms, operation (*i*) presents an average of 522 ms. The interactions with Apache Kafka to consume/produce data in their topics are the most time consuming profiled operations, confirming the trend discovered in the instantiation phase.

Finally, Fig. 3(c) shows the operations involved during the NFV-NS termination phase. These operations are performed in a reverse order with respect to the instantiation phase. As it can be observed, the time needed to delete the associated Apache Kafka topic and *data scrapers* is in the order of tens of ms. In this phase, the most time consuming operation is the deletion of the Apache Spark job, which presents a high variability. This variability is due to the moment when the deletion request arrives. If this request arrives while the Apache Spark job is being executed, the time required to delete it is around 2 seconds. If the deletion request arrives when the Apache Spark job is not being executed, the time to complete this operation is around 150 ms. Adding the contribution of all the deletion operations, they would span from around 200 ms to 2200 ms. This represents a small impact with respect to the overall NFV-NS termination procedure, which in this evaluation, presented an average value of 62 seconds.

V. Conclusions and Future Work

This paper presents a first prototype of our work on integrating closed-loop AI/ML-based decision-making for automated network management in the 5Growth MANO stack (including the data engineering pipeline). In this paper, it is used for scaling in/out network services, but it can be applied to all automated management decisions of the 5Growth platform (e.g., AI/ML-based resource management at the resource layer). The experimental results present the time profiling to configure the AI/ML-based scaling operation for different life cycle operations of a NFV-NS (instantiation, run-time and termination). It is shown that the interaction with Apache Kafka to create, consume, produce, and delete data in their topics are the most time consuming operations, yet no significant impact is measured compared to the total instantiation or termination times (in the order of tens of seconds). As future work, we will fully integrate the solution with a full-fledged 5Gr-AIMLP, which is still under development, to enable online training of the AI models in run time (e.g., reinforcement learning). Besides, we aim to enhance the scaling solution with enhanced AI/ML-based scaling algorithms, and by also including prediction algorithms for resource forecasting.

References

[1] EU 5G-PPP 5Growth Project: 5G-enabled Growth in Vertical Industries, http://5growth.eu/ [Accessed in August 2020].

[2] X. Li et al, "Automating Vertical Services Deployments over the 5GT Platform", IEEE Comms Mag., July 2020.

[3] C. Papagiani, J. Mangues-Bafalluy, et. al., "5Growth: AI-driven 5G for Automation in Vertical Industries", in Proc. of EuCNC 2020, June 2020.

[4] ETSI ISG NFV-IFA 014, Network Functions Virtualisation (NFV); Management and Orchestration; Network Service Templates Specification, September 2019.

[5] ETSI Zero touch network & Service Management (ZSM), https://www.etsi.org/committee/zsm [Accessed in August 2020].

[6] ETSI Experiential Networked Intelligence (ENI), https://www.etsi.org/committee-activity/eni [Accessed in August 2020].

[7] O-RAN Working Group 2: AI/ML workflow description and requirements, *Tech. Rep. O-RAN.WG2.AIML-v01.01*.

[8] EU 5G-PPP 5G-CLARITY Project: Beyond 5G multi-tenant private networks integrating Cellular, Wi-Fi, and LiFi, Powered by Artificial Intelligence and Intent Based Policy, https://www.5gclarity.com/index.php/projects [Accessed in August 2020].

[9] EU 5G-PPP SELFNET: Framework for Self-organized Network Management in Virtualized and Software Defined Networks, hhttps://selfnet-5g.eu [Accessed in August 2020].

[10] EU 5G-PPP 5G-ZORRO: Zero-touch Security and Trust for Ubiquitous Computing and Connectivity in 5G Networks, https://www.5gzorro.eu [Accessed in August 2020].

[11] Apache Kafka, "A distributed streaming platform", https://kafka.apache.org/, [Accessed in August 2020].

[12] Apache Spark, "Apache Spark™ - Unified Analytics Engine for Big Data," https://spark.apache.org/, [Accessed in August 2020].

[13] Apache Livy, "A REST Service for Apache Spark", https://livy.apache.org/, [Accessed in August 2020].

[14] Apache Spark, "Overview: estimators, transformers and pipelines - spark.ml", http://spark.apache.org/docs/2.4.0/ml-pipeline.html [Accessed in July 2020].

[15] J. Mangues et al, "5G-TRANSFORMER Service Orchestrator: Design Implementation and Evaluation", in Procs. of EUCNC 2019, June 2019.

[16] 5Growth project code repository, https://github.com/5growth, [Accessed in August 2020].