IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received March 19, 2022, accepted April 5, 2022, date of publication April 11, 2022, date of current version April 18, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3166160

Supporting Intelligence in Disaggregated Open Radio Access Networks: Architectural Principles, AI/ML Workflow, and Use Cases

ANASTASIOS GIANNOPOULOS[1], SOTIRIOS SPANTIDEAS[1], NIKOLAOS KAPSALIS[1], PANAGIOTIS GKONIS[2], LAMBROS SARAKIS[2], (Member, IEEE), CHRISTOS CAPSALIS[3], MASSIMO VECCHIO[4], AND PANAGIOTIS TRAKADAS[1]

[1]Department of Ports Management and Shipping, National and Kapodistrian University of Athens, 34400 Euboea, Greece
[2]Department of Digital Industry Technologies, National and Kapodistrian University of Athens, 34400 Euboea, Greece
[3]School of Electrical and Computer Engineering, National Technical University of Athens, 15780 Athens, Greece
[4]Fondazione Bruno Kessler, 38123 Trento, Italy

Corresponding author: Anastasios Giannopoulos (angianno_8@hotmail.com)

This work was supported in part by the European Commission through the Affordable5G Project under Grant H2020-ICT-2020-1, and in part by the Horizon 2020 and 5G-PPP Programs under Grant 957317.

ABSTRACT Driven by the emerging trend for transparent, open and programmable communications, Open Radio Access Network (O-RAN) constitutes the dominant architectural approach for deploying the future wireless networks. Towards standardizing and specifying the building blocks and principles of O-RAN, a coordinated global effort has been observed, mainly comprised of the O-RAN Alliance, the operators and several research activities. This paper presents the architectural aspects and the current status of O-RAN deployments, integrating both existing and ongoing activities from the O-RAN enablers. Furthermore, since the Artificial Intelligence and Machine Learning (AI/ML) act as key pillars for realizing O-RANs, a comprehensive view on the AI/ML functionality is provided as well. Additionally, a Network Telemetry (NT) architecture is also proposed to ensure end-to-end data collection and real-time analytics. To concretely illustrate the O-RAN supporting mechanisms for hosting AI/ML, we implemented two realistic ML algorithms: (i) a Supervised Learning (SL) based algorithm for cell traffic prediction using the training data of an open dataset and (ii) a Deep Reinforcement Learning (DRL) based algorithm for energy-efficiency maximization using a 5G-compliant simulator to obtain RAN measurements. We schematically demonstrate the AI/ML workflow for both ML-assisted algorithms through the usage of xApps running on the Radio Intelligent Controller (RIC), as well as we outline the role of the O-RAN components involved in the AI/ML loop. Combining the high-level architectural descriptions with a detailed presentation of ML-empowered resource allocation schemes, the paper discusses and summarizes the O-RAN disaggregation principles and the role of AI/ML embedded in future O-RAN deployments.

INDEX TERMS 5G, B5G, O-RAN, AI/ML, radio intelligent controller, resource allocation, supervised learning, reinforcement learning.

I. INTRODUCTION

The next-generation wireless networks are envisioned to act as the cornerstone technology towards initiating the fourth industrial revolution [1]. In this context, synergetic actions across multiple operators, organizations and working groups are continuously taken with the goal of establishing the fifth-generation (5G) and beyond (B5G) cellular networks [2]. To drive this evolution, three key objectives have been set, each one defining a novel type of 5G/B5G service [3]: (i) enhanced mobile broadband (eMBB), (ii) ultra-reliable and low-latency communications (uRLLC) and (iii) massive machine type communications (mMTC). To effectively support and handle these services, a significant reshape of the existing 5G architectures is required, mainly targeting to offer flexibility, configurability and intelligence. To that end, Artificial Intelligence and Machine Learning (AI/ML) principles must be embedded in the modern 5G/B5G

The associate editor coordinating the review of this manuscript and approving it for publication was Wai-Keung Fung.

architectures, ensuring dynamic network configuration, service heterogeneity and multi-connectivity across multi-vendor environments [4].

Open radio access network (O-RAN) Alliance has already started to participate in the recent 5G architectures, primarily embracing the concepts of transparency, openness, programmability and intelligence [5]. Key pillars in the deployment of O-RAN building blocks offer the opportunity to the mobile operators to independently define and design open hardware, supporting both centralized and distributed cognitive radio control across all the radio access network (RAN) components [6]. Towards achieving these goals, a concerted global effort and working groups (e.g. C-RAN alliance, xRAN forum, ITU-T Focus Group on Autonomous Networks, and the recently announced group RAN Intelligence & Automation-RIA) have been formed to collaborate in the definition of requirements, use cases and architectural aspects that an O-RAN setup has to be compliant with [7], [8].

In addition to RAN programmability and network openness, one of the main objectives of O-RAN is to transform the current architectures into more intelligent and autonomous versions [9]. The O-RAN solutions are provisioned to host advanced data science tools, leveraging AI/ML functionalities to significantly automate the control-flow process and enhance the RAN performance. Introducing such an intelligence level requires a well-defined architectural design in order to support algorithms that continuously gather and exploit the network data in an efficient manner [10]. RAN functionalities and building blocks must be re-defined beyond the traditional manually-programmed methods in order to host AI/ML capabilities and handle the ever-increasing complexity of the envisioned networks. The flexible system disaggregation, along with open interfaces, have been reported as the key concept according to which operators can isolate RAN components from different vendors, while providing multi-vendor interoperability [6], [10]. These disaggregated and open RAN solutions are the first step before AI/ML algorithms can take over the management of advanced RAN functions, especially with regards to AI algorithms placement and the introduction of agent-based distributed techniques in solving complex problems requiring high security and trust.

According to O-RAN Alliance specifications describing the AI/ML workflow and requirements [11], two main entities, namely the non-real-time RAN intelligent controller (non-RT RIC) and the near-real-time RAN intelligent controller (near-RT RIC), will play a critical role in the AI/ML assistance and control loops, determining the optimization rationale of the O-RAN deployments according to the decision time-scale each of them handles (e.g., the near-RT RIC operates in order of ms, while the non-RT RIC decides above 500 ms) [12]. Besides the controllers' functionality, the O-RAN Alliance also specifies open interfaces and disaggregates the near-RT functions from central unit (CU) and distributed unit (DU), partially relaxing the hardware requirements at the cell site. This is coupled with the introduction of applications (xApps/rApps), which can potentially be hosted either on the near-RT RIC for near real-time control or on the non-RT RIC, depending on the time sensitivity of the control process. AI/ML algorithms are considered to run as on-top of the near-RT RIC (xApps, for resource optimization) or the non-RT RICs applications (rApps for policies and orchestration) and can be re-configured by the mobile operators depending on their specific needs [13].

A. TOWARD NON-PUBLIC NETWORKS

Probably the most critical target of the O-RAN is the realization of intent-based management. This idea mainly implies an intelligent configuration performed at the RAN targeting at setting technical parameters (e.g. handover thresholds between cells), scheduling and prioritizing across users and services in an intent-driven manner [6]. The key enabling objective coupled with the realization of intent-based management in the frame of O-RAN can be identified as: given that in O-RAN multiple vendors can share, deploy and program the same open network environment (e.g. all vendors have open access to a common ML catalog that contains pretrained models), a particular vendor can exploit an AI/ML model that has been deployed by another vendor in the past. Network administrators can define a request or business objective (i.e. the 'intent') and the network's software can reply on how to achieve that goal, based on existing ML catalogues. Consequently, intent-based management is able to replace the manual routine processes and enable multi-vendor sharing or model exchange. As part of the Service and Management Orchestration (SMO) module, the non-RT RIC will have critical engagement in this concept, at least during the initial phase of the O-RAN deployment.

The idea of O-RAN triggered industries and business enterprises to establish the concept of 5G Private Networks or Non-Public Networks (NPNs) [14]. Contrary to public (and usually vendor lock-in) networks, a 5G NPN allows for enterprise-specific (capital and operational) cost reduction and flexibility (each enterprise deploys virtual network functions (VNFs) depending on its particular requirements), since the network deployment is open across multiple vendors and costs are distributed among operators [15]. This shared-among-vendors architectural approach introduces service heterogeneity, while reinforces also the need for Network Slicing capabilities in order to support different (parallel) network functionalities under the same network infrastructure [16]. For example, depending on the type of work that is conducted in one site/location of the enterprise, the system might be able to support a slice addressing mMTC requirements, while in another enterprise site, the business needs might require spectrum sharing among public and private 5G spectrum in several bands to cover eMBB services.

B. OVERVIEW OF O-RAN EVOLUTION

O-RAN evolution relies on the combined principles of Cloud-RANs (C-RANs) and Virtualized-RANs (V-RANs), extending their functionalities to incorporate openness and interoperability [17]. Traditional deployment of cellular networks involves an inflexible, monolithic and 'black-box' infrastructure, incapable of decoupling the hardware and software of underlying network infrastructure. This vendor lock-in approach is now unable to handle the design requirements of 5G/B5G, with the latter being characterized by vast number of available resources, network parameters, real-time traffic conditions and optimal network configuration [2].

C-RANs were the first solution to eliminate these constraints, partially exploiting the computational abilities of the Cloud [18]. Specifically, C-RAN architecture consists of three building blocks, namely the baseband units (BBUs, located in the cloud), the remote radio heads (RRHs, acting as remote antenna elements) and the fronthaul links to interconnect the BBUs with RRHs. In C-RAN, the Base Stations (BSs) are decoupled into two parts: distributed RRHs and BBUs clustered into a pool. The pool is centrally-placed at the Cloud, allowing resource sharing, real-time and flexible scheduling in a centralized manner, while enabling the radio resource sharing amongst different BBUs and meeting the dynamic user demands.

Following the technologies of the Software-Defined Network (SDN) and Network Function Virtualization (NFV), V-RANs extended the C-RAN concept [19]. By decoupling hardware and software, V-RAN facilitates the creation of logically-isolated instances over the physical infrastructure, leveraging the wireless and BBU resources to be shared among RRHs, based on the time-varying traffic conditions. This inevitably imposed new cloud requirements for the virtualization, orchestration and network resource scaling. The dominant technologies to meet these requirements are the *hypervisor-based* (a VM runs a guest operating system, e.g. OpenStack) and the *container-based* (executes a specific software app in isolated system environments called containers. e.g. Docker) virtualization schemes.

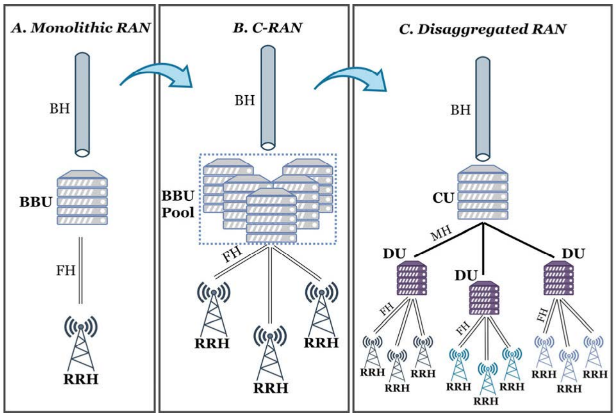

FIGURE 1. Evolution of RAN architectures. (A) Closed system monolithic approaches offer little or no flexibility (vendor lock-in). (B) Cloud or Centralized RAN where baseband unit is located either in a data center or a far-edge location and remote radio heads are connected through a high-bandwidth front-haul. (C) O-RAN disaggregation to enable multi-access edge computing and distributed network monitoring.

O-RAN constitutes the logical derivative of the C-RAN and V-RAN, primarily adding the concept of *open-and-intelligent* network configuration. By introducing open interfaces and open-source software, O-RAN offers the opportunity to the small vendors to gradually add novel services according to their own requirements. It also promotes a fast, competitive and efficient network deployment, while preserving the backward compatibility with legacy systems [20]. This is achieved by an extreme level of network disaggregation to allow handling among multi-vendors, but, unfavorably, increasing the orchestration and radio resource management complexity. To account for this software splitting and densification, AI/ML is inevitably coupled with O-RAN design, in order to assist and automate the network management and provide self-organizing capabilities [6].

C. KEY FEATURES OF O-RAN

Apart from the C-RAN/V-RAN capabilities offered by O-RAN, there are also well-known or novel technologies that have to be included in a complete 5G/B5G system architecture, such as Self-Organizing Network (SON), Multi-access Edge Computing (MEC), Network Slicing (NS), Neutral Hosting (NH) and Network Telemetry (NT) [20]. Although most of those features show interdependencies, being partly complementary and synergetic, some functions developed within one can be beneficially reused by others. However, their direct integration requires the identification of functional block redundancy, because the latter directly affects the overall system delay and performance. Notwithstanding the foregoing, several key enabling concepts are identified in the O-RAN based 5G/B5G architectures, summarized as follows:

1) SELF-ORGANIZING NETWORK

SON principles include the automation, intelligence and self-configuration presented by the O-RAN. Newly-deployed nodes and alarm-triggered reconfiguration of existing nodes have to be supported by SONs, whereas self-optimization functions are essential parts of SON modules. Those include coverage, capacity, handover, QoS satisfaction, energy efficiency and interference control. There is no specific documentation by 3GPPP for SON architecture, but its functionality is primarily supported by the O-RAN intelligent controllers deployed as xApps or rApps.

2) MULTI-ACCESS EDGE COMPUTING

Nowadays, a considerable amount of data is generated at the network edge (i.e., manufacturing environments) rather than the network core. Therefore, over the last years, the necessity to process data on the edge has emerged, as a valuable tool to reduce network latency, minimize backhaul network congestion and ensure high service availability. To this end, the concept of MEC enables cloud-computing capabilities at the edge of the network, thus minimizing network congestion and allowing flexible deployment of applications and services [21].

3) NETWORK SLING

NS refers to the mechanisms needed to parallelize the network infrastructure without increasing the costs [22].

Based on service level agreements (SLAs), it is possible to assign different responsibilities in diverse logical networks (slices), specialized to cover particular services. For example, the three types of 5G services (mMTC, eMBB and URLLC) may be delivered by the same physical infrastructure divided into three isolated, end-to-end slices tailored to fulfil the service-specific requirements.

4) NEUTRAL HOSTING

NH enables a cost-efficient wireless infrastructure shared among multiple operators to increase network densification [14]. It is used to provide services to end-users with subscriptions to several different hosted operators. With the introduction of open APIs, open interfaces, open-source software and SDN/NFV, O-RAN can leverage the cost efficiency of deploying 5G services by co-operating with a neutral host service provider. This would allow differentiated services blended with services offered by operators to maintain continuity within the coverage area of the neutral host.

5) NETWORK TELEMETRY

Network Telemetry (NT) refers to real-time data collection, in which devices or other network entities push data to a centralized location [23]. Telemetry metrics are generated from enterprise resources, such as switches, routers, wireless infrastructure and IoT systems, and are used by business and technology applications to monitor trends and help IT respond to threats or react to changing network conditions. A major trend nowadays is the provision of NT to AI/ML algorithms, for holistic network monitoring and reconfiguration when necessary. The finer granularity and higher frequency of data available through telemetry enables better performance monitoring and therefore, better troubleshooting. It helps a more service-efficient bandwidth utilization, link utilization, risk assessment and control, remote monitoring and scalability.

D. CONTRIBUTIONS

This paper summarizes the design principles underlying the O-RAN potential to encapsulate the concept of a cognitive management-oriented SON, especially for purposes of Radio Resource Management (RRM). Firstly, we present the well-documented parts of the O-RAN architecture with respect to AI/ML support in a generic manner, outlining possible deep learning (DL) and deep reinforcement learning (DRL) algorithms implemented within O-RAN deployments. Secondly, we illustrate how the intelligence loop, engaging the cognitive controller, can be unfolded within the O-RAN architecture, highlighting the critical components involved in the design, training, inference and evaluation phases of the AI/ML models. We also concretely describe the AI/ML workflow, responsible for an end-to-end delivery of both supervised and reinforcement learning-based models in a unique manner. As practical use case implementation, we show how the O-RAN can (i) support predictive capabilities (using a cell load prediction paradigm from an open dataset [23]) and (ii) ensure enhanced energy-efficient power transmissions (using multiple energy efficiency-targeted agents). To generate realistic network measurements, we also developed a general-purpose 5G simulator (publicly available soon), following 3GPP-compliant channel models [24] and supporting user mobility patterns. Finally, to support data collection across the entire 5G network components, a high-level description of a network-wide telemetry architecture is also provided, highlighting how the network analytics spanning from the radio access part to the core network functions can be extracted.

Combining the high-level architectural descriptions with a detailed presentation of ML-empowered resource allocation schemes, the paper discusses and summarizes the O-RAN disaggregation principles and the role of AI/ML embedded in future O-RAN deployments.

II. O-RAN ARCHITECTURE AND INTELLIGENCE

In this section, the O-RAN architecture along with the enabled AI/ML capabilities is described. As previously mentioned, the support of open interfaces following the O-RAN initiative is expected to allow flexible and cost-efficient 5G deployments in private and enterprise networks. This solution will not only facilitate the deployment of a 5G RAN using components from multiple vendors, but it will also simplify network management leading to reduced operational costs. This will become possible by embedding intelligence using emerging deep learning techniques at both component and network level of the RAN architecture [5]. In combination with the standardized southbound interfaces, AI-optimized closed-loop automation is achievable and is expected to enable a new era for network operations. Therefore, in the following subsections, after the description of the O-RAN architecture, the key-enabling technologies that support the deployment of AI/ML frameworks in O-RAN are described as well.

A. GENERAL O-RAN ARCHITECTURE

The general approach of the O-RAN architecture is depicted in Fig. 2. Primarily, it can be decomposed into two layers, namely the Service, Management and Orchestration (SMO) module, as well as the radio access site. The latter consists of all the radio access entities and functionalities, including the near-RT RIC, the vertically-split control (CP) and user (UP) planes of the central units (CU), the distributed units (DU), as well as open interfaces interconnecting the O-RAN nodes. The non-RT RIC is located in the SMO layer and communicates with the near-RT RIC via the A1 interface. By placing the non-RT RIC in the SMO, a combined usage of RAN metrics and contextual (usually external) data can be realized, allowing for priority-aware or condition-specific RAN optimization. In this context, open A1 interface is dedicated to provide intent-based policies, especially for the O-RAN optimization, acting as the bridge for policy communication between the near-RT and non-RT RICs [12]. This interface enables also vendor-agnostic policy guidance to the underlying RAN elements, making the management policies to be acknowledged in the radio access side.

The O1 interface connects the SMO to the RAN-managed elements. Through this interface, metrics associated with the performance of the RAN nodes can be collected to the SMO and, additionally, the SMO can control and even apply configuration changes to the RAN (e.g. reallocation of slice resources). The corresponding E2 interface establishes communication among the lower RAN modules (CU, DU and RU) and the near-RT RIC and is intended to serve similar utilities to O1 for time-sensitive control of the RAN components. Finally, the E1 interface connects the control plane (CP) of the O-CU with the corresponding user plane (UP), while the A1 interface connecting the non-RT RIC and near-RT RIC is employed for policy management and coordination.

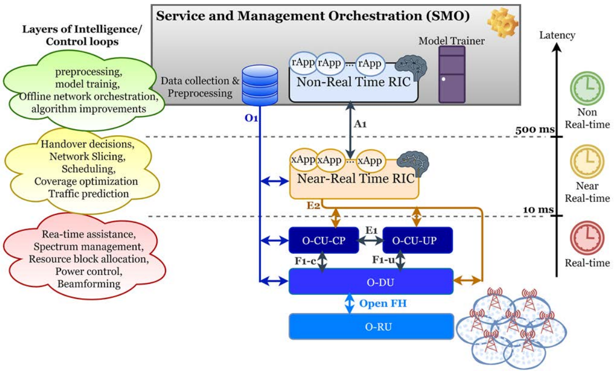
Layers of Intelligence/ Control loops
preprocessing, model trainig, Offline network orchestration, algorithm improvements
Handover decisions, Network Slicing, Scheduling, Coverage optimization Traffic prediction
Rea-time assistance, Spectrum management, Resource block allocation, Power control, Beamforming
Service and Management Orchestration (SMO)
Data collection & Preprocessing
rApp rApp ... rApp
Non-Real Time RIC
Model Trainer
O1
A1
xApp xApp ... xApp
Near-Real Time RIC
E2
O-CU-CP
E1
O-CU-UP
F1-c
F1-u
O-DU
Open FH
O-RU
Latency
500 ms
10 ms
Non Real-time
Near Real-time
Real-time

FIGURE 2. General O-RAN architecture. A cross-layer intelligence scheme can be supported, including the management examples mentioned in the colored clouds (on the left side). A closed-loop top-down management is feasible, with intelligence functions running on top of the O-RAN controllers as rApps or xApps. A bottoms-up data collection and performance monitoring is also achieved through O1 open interface.

B. AI/ML OVERVIEW

The ultimate goal of AI/ML algorithms is to provide alarms, predictions or suggested actions on unknown network states (e.g. traffic prediction, cell congestion alarms, power configuration towards throughput maximization). In other words, an AI/ML model targets to find a mapping between inputs (or features) and outputs (or dependent variables) in order to guarantee a specific objective. Depending on the approach followed to find this 'mapping', three broad AI/ML categories are identified, namely Supervised Learning (SL), Unsupervised Learning (USL) and Reinforcement Learning (RL) [25]. The first two AI/ML branches totally rely on historically collected samples that relate the input with outputs. Outputs can be either numerical values (i.e. regression problems [26]) or categorical values (i.e. classification problems [27]). SL uses the labels (i.e. the desired outputs) to fit a mapping function between features and outputs on the training dataset (i.e. the historical data). On the contrary, USL does not use labels, thus it is appropriate to extract hidden patterns in the training dataset (e.g. clustering, dimensionality reduction) [28]. Following a different approach during training, RL suits for decision-making problems, since it includes an agent that aims to maximize collective future rewards through trial-and-error interactions with a well-defined environment [29]. Fig. 3 summarizes the AI/ML branches, along with the most frequent algorithms associated with each category.

Deep neural networks (DNNs) have been incorporated in every AI/ML branch, since they show impressive capabilities in mapping both linear and non-linear (multi-feature and multi-variate) functions [27]. DNNs can be used as SL or USL regressors, classifiers, or even function approximators in Deep RL (DRL). For example, DRL agents use DNNs to approximate the 'quality' of being in a given state and performing a specific action (the so-called Q-function), instead of using memory-inefficient and computationally-intensive Q-tables [30]. In SL, DNNs shows enhanced performance in fitting complex relationships by adding multiple levels of abstraction (hidden layers) and gradient descent operations, thus allowing to decipher non-linear patterns in the data and overcoming the, usually simplified, assumptions of other classical AI/ML models, such as linear regression, logistic regression and random forests [31].

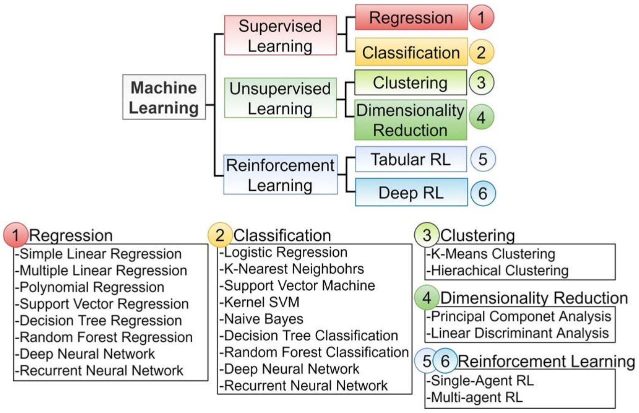

FIGURE 3. Machine learning broad categorization (upper panel) and widely-used examples per AI/ML branch (lower panel).

In the frame of O-RAN architecture, AI/ML is provisioned to play a critical role in several cross-layer topics. For example, autonomous radio resource management (or RRM) should operate at the Transmission Time Interval (TTI) time scale, thus requiring an AI/ML model that is hosted at an edge O-RAN component (e.g. O-DU). Such models could be responsible for regulating O-RU parameters, including the power levels and/or bandwidth allocation in resource blocks, and could be trained following SL, USL or RL principles. On the other hand, the relatively time-insensitive and computationally-expensive operations, such as beamforming parameter configuration, dynamic resource assignment in networks slices, placement of virtual network functions, could be hosted in higher layer (e.g. in Non-RT RIC/SMO).

C. AI/ML FUNCTIONALITIES IN O-RAN

In the vision of O-RAN architectures, three control loops and AI/ML-dedicated nodes have been standardized to host automated and intelligent management functionalities [11]. SMO, Non-RT and Near-RT RIC, along with their interconnection interfaces, constitute the crucial O-RAN components to host AI/ML functionality across network domains, attending to support both offline/online training and inference. Offline training refers to the time-consuming processes required for training a model. Online training refers to real-time agents that learn by interacting with the environment through trials-and-errors. The offline training support is essential in O-RANs because time-sensitive decisions have to exploit already pre-trained models. In this respect, this approach mainly relies on the following design principles: (i) an offline learning module is by default essential to train SL, USL and RL algorithms based on historical data, (ii) offline training refers to a pre-trained model that may be inferred during the online operation of the network, (iii) online training refers to the concept of real-time learners (e.g. RL), as the model is trained through interaction with the network and (iv) completely untrained models cannot be directly deployed in the network without prior training and testing.

In O-RAN, two critical building blocks will be responsible for the execution of the ML workflow [12]: (i) the Near-RT RIC is a logical function that enables near-real-time control and optimization of RAN elements and resources via fine-grained data collection and actions over E2 interface, as depicted in Fig. 4, and (ii) the Non-RT RIC, which is a logical function within SMO that enables non-real-time control and optimization of RAN elements and resources, AI/ML workflow including model training, inference and updates, and policy-based guidance of applications/features in Near-RT RIC.

It should be noted that in the deployment of SL and USL algorithms, the ML training host is essentially located in the Non-RT RIC, while the ML model host/actor can be located either in the Non-RT RIC or in the Near-RT RIC. On the contrary, in the framework of reinforcement learning, both the ML training host and the ML inference host/actor shall be co-located as part of Non-RT RIC or Near-RT RIC [11].

There are three types of control loops defined in O-RAN depending on the time sensitivity of the required decision-making process:

- *Control Loop 1* deals with per TTI msec level scheduling and operates at a time scale of the TTI or above.
- *Control Loop 2* operates in the Near-RT RIC and is responsible for decisions within the range of 10-500 msec and above (resource optimization).
- *Control Loop 3* operates in the Non-RT RIC at greater than 500 msec (policies, orchestration). It is not expected that these loops are hierarchical but can instead run in parallel in a heterogeneous or synergetic manner.

Fig. 4 shows the mapping between the O-RAN modules and the AI/ML phases (training, inference, controlled entity) in the three O-RAN control loops.

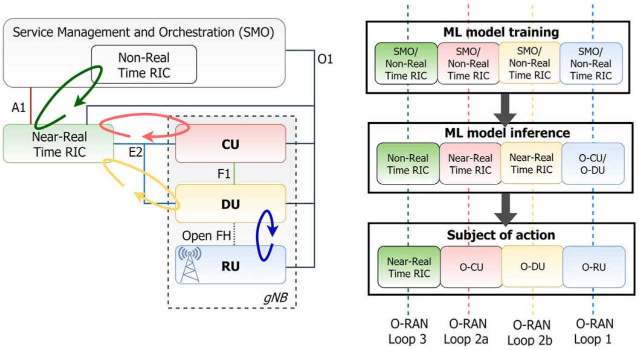

FIGURE 4. Schematic representation of O-RAN control loops. The AI/ML-dedicated modules and interfaces of O-RAN are depicted (left), along with involved modules for the training-inference-action sequence of each control loop (right).

Graphically, Loop 1 refers to the case that the model inference is hosted in the O-DU and the exact configuration of this loop is under consideration. Loops 2 and 3 are clearly defined as the loops that host the ML training at the Non-RT RIC, while the ML inference is typically running in Near-RT RIC and Non-RT RIC, respectively. In general, control loop 1 is justified to host the pretrained model and the inference data in O-DU, targeting at supporting intelligent operations at the edge (in the time scale of TTI). Thus, in the case of control loop 1, the O-DU collects the inference data and provides predictions or corrective actions to O-RU (actor). In addition, the related interfaces engaged in control loops 2 and 3 are well-specified [5], whereas the interface between O-RU and O-DU (i.e. the Open Fronthaul) for control loop 1 is relatively understudied with respect to AI/ML model or parameters exchange.

D. NETWORK-WIDE DATA COLLECTION

As previously mentioned, NT can be viewed as a real time data collection from various components of the architectural layers within a 5G network. In this context, two significant components are the NWDAF (Network Data Analytics Function) and the C-MDAF (Centralized Management Data Analytics Function), shown in Fig. 5, where the former is a well-specified component of the 5G Core network, according to 3GPP specifications [32]. The NWDAF collects data from Core Network Functions and provides network data analytics services to the 5GC Network Functions (NFs) subscribed as NWDAF consumers [32]. This approach encourages and allows multivendor deployments and facilitates customization to suit individual service needs by the utilization of 3GPP compliant implementation.

The C-MDAF is provisioned with all the centralized telemetry capabilities, located at the SMO layer [33]. In this context, a particular NF can subscribe to the C-MDAF as a consumer in order to collect or provide management data for forecasting or resource information purposes. Furthermore, the Telemetry Data Collector, which is integrated with the C-MDAF, includes a monitoring server (e.g. Prometheus) for collecting performance measurement data from the NWDAF, the virtualized infrastructure and the Transport Network elements (TN-EMS). Performance measurement data are also collected from the O-RAN VNFs using the O1 Virtual Event Streaming (VES) collector [11].

The NWDAF incorporates the necessary interfaces to collect data from different types of data sources, notably 5G Core NFs. These data are made available to a Prometheus monitoring server residing in the NWDAF. In this context, the NWDAF offers two services (called Nnwdaf services) [32]. The first one is the Nnwdaf_EventsSubscription service, which enables the NF service consumers (like PCF, NSSF, OAM etc.) to subscribe to and unsubscribe from different analytics events provided by the NWDAF, and subsequently enables the NWDAF to notify the NF consumers about subscribed events. The second Nnwdaf service is the Nnwdaf_AnalyticsInfo service, which enables the NF consumers to request and get specific analytics from the NWDAF. Both the NWDAF and the C-MDAF provide data analytics; in this respect, ML-based data analytics are provided by the ML analytics module (that can be regarded as part of the AI/ML framework) shown in Fig. 5. In addition, the Northbound Interface (NBI) is used for the connection between the NWDAF with external systems like ML frameworks. Finally, a graphical dashboard (e.g. Grafana) provides charts and notifications representing the current operational status of each one of the monitoring sources.

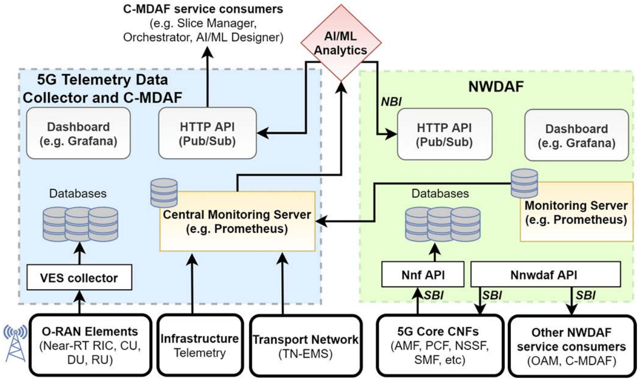

FIGURE 5. Network-wide data collection scheme in O-RAN based architecture. In the left part, telemetry collection supports data gathering from O-RAN (via VES collector), Infrastructure layer and Transport Network (via Prometheus exporters). In the right part, the core network function metrics are collected via NWDAF functionalities to host core network telemetry.

III. SUPERVISED LEARNING SCENARIO

To illustrate a practical implementation of network AI/ML forecasting, we firstly considered a SL use case. Historically-gathered cell load time-series were used to train a Recurrent Neural Network (RNN). To eliminate the vanishing gradient problem which is usually met in RNNs, we used a Long-Short Term Memory (LSTM) neural network to predict future values of the cell loads, based on a specified previous window of the cell time-course [34]. The cell load prediction example was selected as a fundamental paradigm in the network optimization, as the predictive traffic alarms can significantly impact several network decisions for scaling up/down resources, such as slice configuration, channel allocation, power regulation and virtual resource placement [35]. This scenario concerns the Control Loop 2 of the O-RAN architecture, since the trained cell-specific load predictors can be deployed as xApps (within the Near-RT RIC) to provide cell load alarms either in a group of cells (controlled by an O-CU) or in a particular cell (controlled by an O-DU). The detailed LSTM technical description and implementation is described below.

A. DATASET

To illustrate the training process and the AI/ML workflow in an SL scenario, we used the dataset extracted in [23] (can be found at https://github.com/sevgicansalih/nwdaf_data). This dataset contains labeled data for 5G cellular networks, generated according to 5G specifications. Considering a topology of 5 partially-overlapped 5G cells with varying:

(i) *network area information* (i.e., cell identifiers of a group of associated UEs: ID 1, 2, 3, 4 and 5),

(ii) *subscription categories* (i.e., the policy of a group of subscribed UEs: platinum, gold and silver) and

(iii) *personal equipment devices* (i.e., device type information of a UE: IoT device, vehicle, cell phone, smartwatch and tablet), the dataset contains the data rate (in bytes per 15-min period) of each category and device type. Apart from confidential/privacy issues raised by operators in sharing network measurements, the main idea of using this type of information in our example relies on NT node abilities (e.g. NWDAF) to collect (i) abnormal behavior for a group of UEs of a single UE and (ii) network load performance in an area of interest. Data generation follows realistic assumptions of the traffic pattern, accounting for spatiotemporal variations during simulations, such as handovers, UE mobility/velocity, subscription prioritization, cell adjacency and device-specific characteristics. Traffic data have been recorded for 6 months with a sampling frequency of 1 sample per 15 minutes. Since there are 5 cells, 3 subscription classes and 5 device types, the dataset contains 75 measures per 15 minutes, resulting into 6(months) × 30(days) × 24(hours) × 4(quarters) × 75(measures) = 1296000 total samples. The total load of each cell was computed as the summed data rate across its associated subscribers and devices. To build the LSTM model for predicting the total cell load, we split the whole dataset into training and testing sets. The training set included the cell load measures during the first 5 months plus the first 3 weeks of the 6th month, whereas the last week of the 6th month comprised the testing set. Fig. 6 shows the dataset organization and splitting, the 5-cell topology and the cell-specific LSTM predictors under consideration.

B. LSTM REGRESSION

Before LSTM creation and training, all cell load values of the training set were scaled in the range of [0, 1] to reduce the contamination of outlier data and eliminated the skewed distribution effects. The scaled load values of cell i

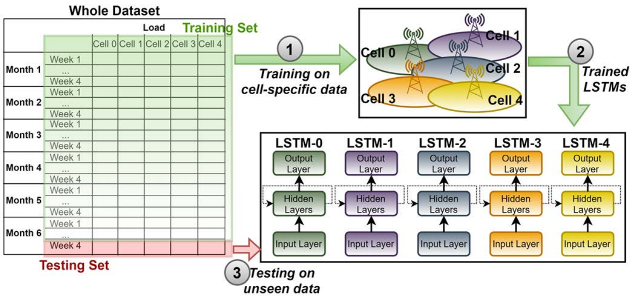

FIGURE 6. **Cell-specific load prediction outline. The time-series of the cell load was used to train the LSTM models, exploiting 23-week load data for training and 1-week load data for inference purposes. Cell load values are captured every 15 minutes, corresponding to 672 data points per week.**

($i = 1, \ldots, 5$) at time t ($t = 1, \ldots, 16608$) were computed as follows:

$$L_{i,t}^{norm} = \frac{L_{i,t} - \min\{\boldsymbol{L_i}\}}{\max\{\boldsymbol{L_i}\} - \min\{\boldsymbol{L_i}\}} \tag{1}$$

where $\boldsymbol{L_i}$ is the training load vector of cell i for each training point $t \in [1, 1608]$. Normalization was applied only in the training set to avoid information leakage (testing set should be unknown to the neural network), whereas the inverse scaler was applied during the LSTM inference phase.

The feature space was defined by the 2-week previous (exact) load values (i.e. $2 \times 7 \times 24 \times 4 = 1344$ samples), as opposed to window-averaged previous values used in [23]. Thus, the LSTM regressor is in charge of predicting the load value L_t at time t, given the feature vector $[L_{t-1344}, \ldots, L_{t-1}]$. Fig. 7 shows the architecture of the LSTM model in rolled and unrolled versions. A Dropout rate of 20% was also established to eliminate overfitting, whereas 4 stacked hidden layers (each one with 50 units) were architected in the model. The loss function used to update the neural network weights during backpropagation was the Mean Squared Error (MSE). Hence, whenever a batch of 64 training samples has passed the LSTM layers, the MSE was computed as the sum of squared differences between the actual and predicted load values. Stabilization of the LSTM hyperparameters is shown in ''Simulation Results'' section.

IV. REINFORCEMENT LEARNING SCENARIO

In this section, we consider the implementation of a practical energy-efficiency (EE) optimization algorithm [36]. To illustrate how O-RAN could support distributed intelligence, we trained a multi-agent DRL model. The O-RAN is jointly optimized in terms of (i) experienced throughput and (ii) power consumption. Specifically, given the UE measurement reports (collected from the O-RAN), the algorithm aims to maximize the network EE by providing a power allocation scheme for all Physical Resource Blocks (PRBs) of all active Radio Units (RUs). To that end, three interacted DRL agents are trained on simulated network measurements obtained from mobile users inside a three-cell area. The implementation framework starts with a simulation environment in order to (i) train the DRL agents in synthetic and 5G-compliant network conditions and (ii) test and validate the deployed algorithm by using the simulated data for model inference purposes.

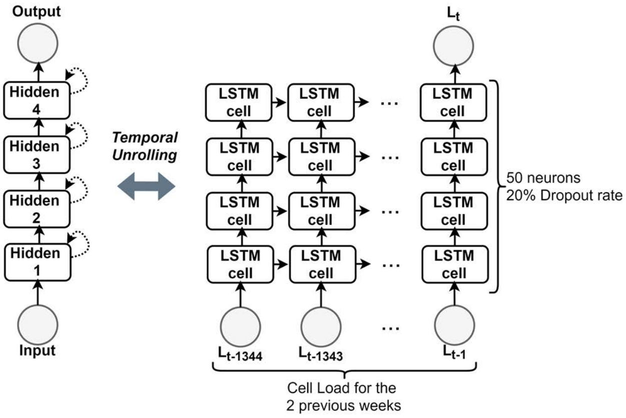

FIGURE 7. **LSTM neural network architecture in rolled (left) and unrolled (right) forms. The feature vector for predicting one target sample at time *t* (L_t) consists of the cell load values of the previous 2 weeks. Four hidden layers were stacked, each one having 50 hidden LSTM units and 20% Dropout rate for eliminating overfitting.**

Key motivations in using DRL to solve the EE optimization problem include:

(i) Suitability for non-convex optimization problems, such as EE optimization, given that EE-targeted power control attempts to jointly ensure adequate throughput and low power consumption, under station-specific power and user-specific throughput constraints.

(ii) Effortless model inference and deployment of DRL agents, once they are pre-trained.

(iii) No necessity for training data, since DRL training is based on trial-and-error interactions between a software agent and a real or simulated environment.

A. RAN MEASUREMENT GENERATOR

As shown in the left part of Fig. 8, a 5G network generator is built to provide O-RAN measurements. A network area consisting of three 5G urban macro-cells (UMa) is considered, following the specification of UMa cells detailed in [24]. Each RU has 12 available PRBs to transmit data using the OFDM modulation scheme. Inside the network area, a set of mobile users is established, with each individual UE experiencing a specific throughput value expressed in Mbps. At each time instance, each UE follows a random-walk model with a pedestrian speed of 1 m/s. The algorithm supports time-varying number of users, given that UEs may enter or exit the network area, according to their trajectories. In addition, the network simulator is aware of possible handovers, since a UE may be served from different RUs depending on which is the best server for a given time point. The key functionality of the 5G traffic generator includes the interference calculations for each UE by not only considering the channel losses but also the accumulated interference

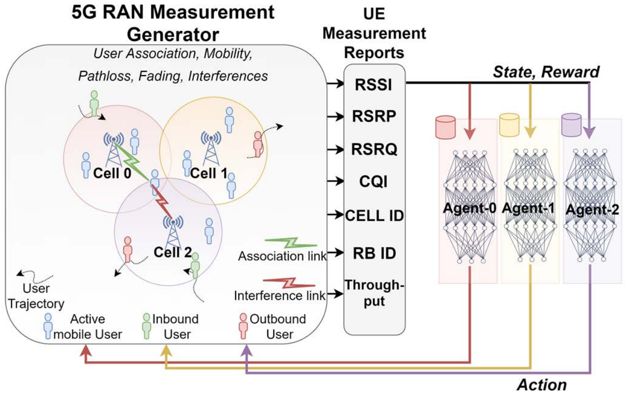

FIGURE 8. The interaction cycle between the DRL agents and the considered RAN environment. The 5G RAN Generator includes the internal functionality for user mobility, inter-cell interference calculations, user association and the pathloss/fading estimations according to 5G-compliant channel models. The telecom environment acknowledges the state and reward to the agents, before the next DRL action.

caused by the non-servers, as specified by the Shannon's formula. Specifically, the signal-to-interference-plus-noise ratio is computed for each pair of associated UE-RU, taking into account the 5G-compliant UMa channel models for multipath losses [24]. Furthermore, the association scheme is implemented according to the maximum-throughput criterion, meaning that each UE occupies the PRB from which it receives the best throughput.

In summary, the traffic generator produces the following output UE measurement reports:

- *Received Strength Signal Indicator (RSSI):* measures the average total received power observed only in OFDM symbols containing reference symbols in the measurement bandwidth over 12 resource blocks.
- *Reference Signal Received Power (RSRP):* RSRP is an RSSI type of measurement, proportional to the power of the LTE Reference Signals spread over the full bandwidth and narrowband.
- *Reference Signal Received Quality (RSRQ):* Quality considering also RSSI and the number of used Resource Blocks measured over the same bandwidth. RSRQ is a C/I type of measurement. The RSRQ measurement provides additional information when RSRP is not sufficient to make a reliable handover or cell reselection decision.
- *Channel Quality Indicator (CQI):* ranges from 1-15 depending on the quality of the received signal and the experienced throughput.
- *Cell ID:* an index representing the serving cell according to maximum-throughput criterion
- *PRB ID:* a number indicating the associated PRB of the serving cell according to maximum-throughput criterion
- *Throughput:* a value in Mbps reflecting the experienced throughput from the associated Cell-PRB.

B. DEEP REINFORCEMENT LEARNING AGENTS

In general, DRL models are appropriate for real-time decision-making problems in complex environments, where the knowledge of beneficial future actions is gathered through trials-and-errors [30]. The agent initially observes the wireless environment through the state space S and then randomly performs an action selected from the available action space A, leading to a new state of the environment. Depending on whether the outcome of the performed action was beneficial towards the optimization goal (i.e. towards EE maximization), the agent receives a positive, negative or null reward r and registers this experience tuple (state, action, new state, reward) in its memory. The agent continues to experiment with the environment, targeting to ideally perform all available actions from A to all possible states of the environment S and quantifying the profitability of each action from a given state in the registered experience tuples. The policy learning process entails the training of the agent to gradually perform the actions (or a series of actions) that will return the maximum reward. The quality of each action, or Q-value, is calculated according to the Bellman equation [29]:

$$Q_t(s_t, a_t) = (1-\alpha)\cdot Q_{t-1}(s_t, a_t) + \alpha \cdot (r(s_t, a_t) + \gamma \cdot \{Q(s_{t+1}, a')\}) \tag{2}$$

where the parameter α is used to balance between previous and learned Q-values and γ is the discount factor that is used for trade-off between immediate and long-term rewards. The fundamental principle of DRL (specifically we use the Deep Q-Learning method or DQL) is the use of neural networks to approximate the quality of each action from a given system state [36].

The framework of decentralized DRL was adopted for establishing the training process. According to this scheme, three different DRL agents were deployed and trained for EE optimization, each one located in the respective cell area. Note that, the system-level EE (in Mbps/Watt) is computed as the sum of the total allocated data rate divided by the total transmitted power of the RUs. The algorithm solves the EE optimization by taking into account a maximum power constraint for each RU ($P_{\max}$) and a minimum guaranteed data-rate per user (1 Mbps for Voice over IP). The simplified DRL scheme included a software agent per cell that, after observing the underlying environment, decides the proper action leading to positive rewards. Thus, the objective function is inherently embodied in the rewarding system of the DRL to guide the agent's behavior.

1) STATE SPACE

Each agent located in a respective cell area partially gathers information from the environment, i.e. observes only the users that have been associated to his PRBs. To this end, the state space of each agent is an array with dimension equal to the PRB number (6 in our case but this can be generalized depending on the selected 5G numerology) that contains the channel quality index (CQI) value of the user that is currently connected to the respective PRB or null otherwise.

2) ACTION SPACE

Each agent selects a local action, based on its own policy. The available actions for each agent involve the selection of a single PRB and either the preservation, increase or decrease of its power level by a predetermined constant power step P_s. The individual actions selected by each agent are then combined to form a global action vector, i.e. the power values of all PRBs in the three cells.

C. REWARD FUNCTION

The reward is calculated globally and is defined to reflect the optimization target of the algorithm, i.e. the percentage increase in the current system EE relative to the previous system EE. Quantitatively, the reward received at episode $\boldsymbol{t}$ is equal to the EE increment $\mathbf{100} \times [(\boldsymbol{EE_t} - \boldsymbol{EE_{t-1}})/\boldsymbol{EE_{t-1}}]$ across all RUs and users in the network area. In this manner, although each individual agent has partial observability of its environment, it is able to "sense" the network-wide environment by receiving the system-level EE increment (i.e. global reward). For instance, when a cell agent selects a selfish action based on the observability of its users, it only contributes in some terms of the global EE formula. By taking into account the throughput of each user and the power level of all cells, a global reward is returned to every cell agent, giving insights to each of them about how globally good were their local actions.

V. SIMULATION RESULTS

In this section, the outcomes of several simulations are firstly demonstrated to quantify the performance of the implemented AI/ML models. Those results are given both for the training and testing phases of the algorithms. Then, a general-purpose workflow of both supervised and DRL-based scenarios is proposed to illustrate how diverse types of AI/ML models can be executed and supported across O-RAN deployments. We also discuss how the workflow unfolds to deliver AI/ML functionalities and suggest possible extensions towards a complete AI/ML integration in the O-RAN architecture.

All the presented simulations were conducted in Python 3.8, whereas the libraries TensorFlow (version 2.3), Keras and Scikit-Learn were used for constructing and training the AI/ML models. Coding scripts ran on a personal PC (CPU i7-8700; 3.2 GHz; RAM 8 GB; no GPU usage).

Noteworthy, for the purposes of this paper, the training of the models has been conducted offline. In O-RAN, training could be also performed in the Non-RT RIC. Then, the models are dockerized and deployed as xApps in a real commercial Near-RT RIC (control loop 2 of O-RAN). Thus, the model inference takes place at the Near-RT RIC, with the predictions/corrective actions being published in the Near-RT databus for further exploitation.

A. CELL LOAD PREDICTION WITH LSTMs

In the first part of the simulations, the crucial learning hyper-parameters of the LSTM models are stabilized. Based on the similarities in the spatiotemporal fluctuations of the load traffic between all the neighboring cells considered in the network area, the learning rate (a) and the window length (W) hyper-parameters were fine-tuned for one single cell, and the resulted optimal values were inherited by the rest of the LSTM models. The configuration of the LSTM model used for the training process is summarized in Table 1. Specifically, all training simulations used three hidden layers, 50 LSTM units per layer and a Dropout rate of 20% to mitigate overfitting effects. Stochastic Gradient Descent performed by the Adam optimizer was selected to update the neural network weights during the back-propagation iterations, while the MSE loss function was selected to estimate the training errors of the LSTM regressor. Initial simulations proved that 20 epochs are sufficient for low-valued convergence ($\sim 10^{-3}$) of the loss function.

Observing that different values of the learning rate significantly affect the LSTM performance, different models were derived with varying values (a=0.1, a=0.01, a=0.001). Fig. 9A depicts the MSE loss convergence among the training epochs, with the value of $a = 0.01$ showing the optimal (i.e. faster and minimum) MSE loss. Moreover, the window length, which considerably influence the dimensionality and memory requirement of the LSTM models (directly proportional to the input layer size), was varied for three training setups (2 weeks or $W = 1344$, 1 week or $W = 672$, 1 day or $W = 96$ samples). To quantify the optimal value of W, Fig. 9B shows the MSE loss curves, revealing the optimal window length for $W = 96$ samples.

After setting the learning rate and window length to their optimal values, five LSTM models were trained on the cell-specific load time-series.

To evaluate the performance of the trained models, inference samples were derived for the testing set (unseen 1-week data for each cell). For a given testing sample, the input of the LSTM is the previous 96 cell load values, whereas the output is the predicted load in the next 15-min. Fig. 10 illustrates the predicted and the actual load values for each cell during the last (testing) week. Evidently, all LSTM model showed enhanced performance in predicting the periodic traffic variations. The enhanced performance in all cells might be explained by the high spatiotemporal traffic similarity across neighboring network areas, as well as the identical probabilistic assumptions drawn for the traffic distribution and mobility profile.

It is also worth noting that LSTM models do not perfectly fit the peak traffic values (2Gbps maximum deviation from the actual values), however they accurately predict the traffic trends, especially the upward and downward slopes of the cells load.

B. ENERGY-EFFICIENCY WITH DRL

In this section, the training and validation phase of the multi-agent DRL for EE enhancement is presented. The ultimate goal of EE maximization is to reduce the power consumption, without significantly affecting the experienced

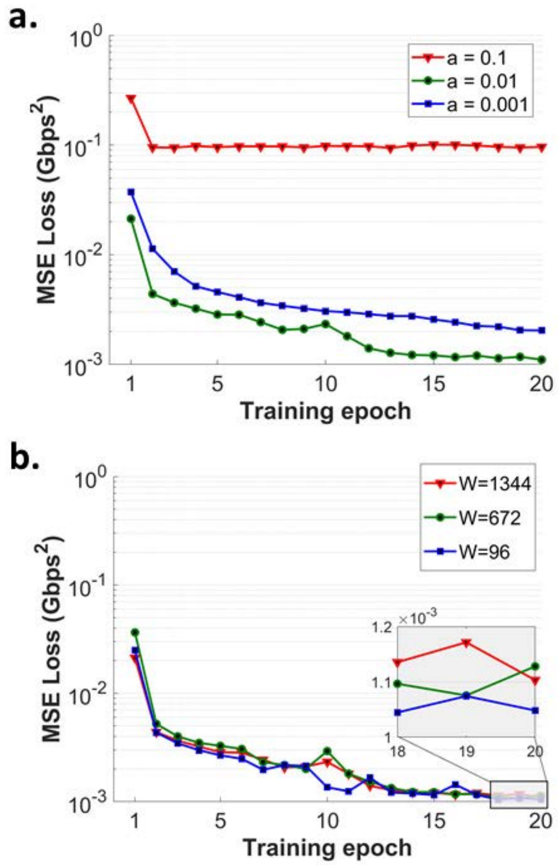

FIGURE 9. Mean squared error (MSE) loss curve as a function of the training epochs for different values of learning rate *a* (panel a) and window length *W* (panel b). Y-axes are in logarithmic scale.

TABLE 1. Architecture of long-short term memory network.

Parameter	Value
LSTM regression model	Sequential (TensorFlow Keras 2.3)
Input Neurons	24×4 = 96
Output Neurons	1
Number of hidden layers	4
Hidden units per layer	50
Dropout Rate	20%
Optimizer	Adam
Learning rate	0.01
Training set samples	(6×30×24×4)-(8×24×4) = 16512
Testing set samples	1×7×24×4 = 672
Feature scaling	MinMaxScaler (sklearn library)
Training epochs	20
Batch size	64
Loss function	Mean Squared Error

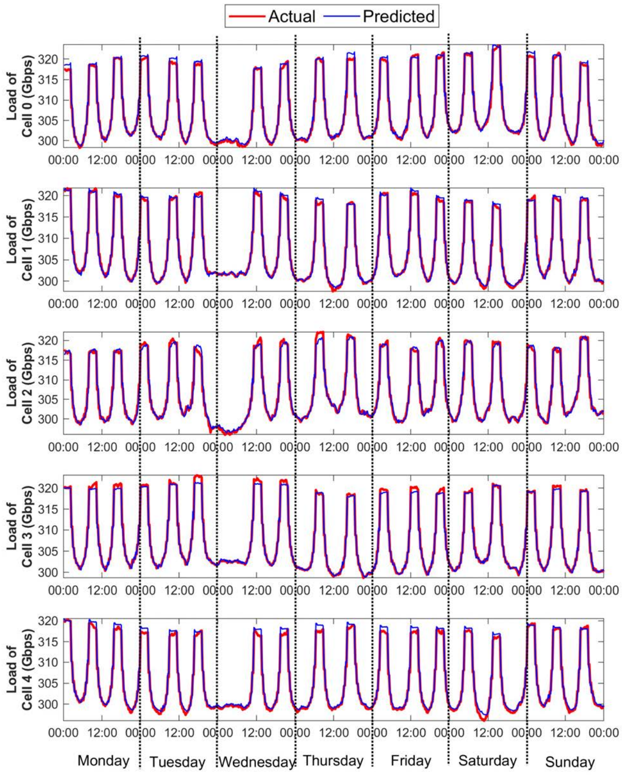

FIGURE 10. Actual and predicted load (Gbps) for each cell-specific LSTM model. Validation data refers to 1-week total load values used for the testing set.

data-rates [37]. The network topology corresponds to that described in section IV.A. The operating 5G band used in the simulations was at 6 GHz with a channel bandwidth of 20 MHz. In addition, the channelization scheme was based on 5G numerology $\mu = 4$. This means that each available PRB had a bandwidth of 2.88 MHz (consisting of 12 subcarriers with spacing $2^{\mu} \times 15 = 240$ kHz), whereas the Guardband bandwidth was 1360 kHz. Similar to the previous section, we firstly experimented with varying learning rates on the DRL model. Note that, here the learning rate (a) refers to the Bellman equation and is used to balance between the current and future rewards (see section IV.B). To this end, Fig. 11 depicts the learning curves during the training phase of the DRL agents, showing that the reward (i.e. EE enhancements) incrementally reaches the values of 72.3%, when the learning rate is $a = 0.0001$. Following the rewarding definition, this final reward represents the accumulated % increment in EE relative to the initial network state of each episode. This means that, if we infer the DRL agents for purposes of maximizing the system EE, a gain of 72.3% will be achieved relative to the initial system EE (without DRL assistance). The DRL EE increment is relative to the initial system EE, called ‘‘non-DRL’’. The initial power configuration is the ‘average’ scheme (all PRBs of all RUs operate with the fixed average power levels). The selection criterion for this initial network state relies on the fact that an ‘average’ power configuration scheme provides a reasonable three-fold balance between power consumption (there are no maximum power levels), achieved throughput (there are no minimum power levels) and interference mitigation. The impact of discount factor was negligible in the reward convergence, thereby it was constantly set to 0.9, whereas the power step was selected at 5 Watt following similar simulations as in [36].

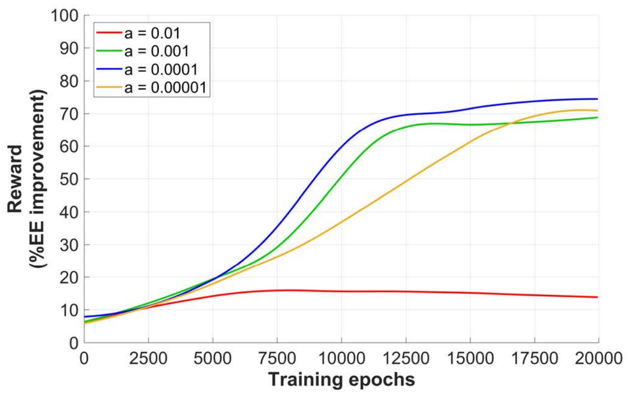

FIGURE 11. Reward convergence of the multi-agent DRL scheme for different learning rates (a) as a function of the training episodes.

Before validating the pre-trained DRL schemes, Table 2 summarizes the final (optimal) parameters included in the Deep Q-Networks.

TABLE 2. Architecture of multi-agent deep Q-networks.

PARAMETER	Value
Number of DRL agents	3
Input Neurons	12
Output Neurons	24
Number of hidden layers	3
Hidden units per layer	96, 72, 48 (1st, 2nd and 3rd layer)
Activation function	ReLU (for hidden layers), Linear (for output layer)
Replay Memory Size	1000
Batch size	64
Optimizer	Adam
Loss Function	Huber loss
Update Target Frequency	100
Learning rate	0.0001
Discount Factor	0.9
ε-greedy policy	Linear decay (up to 10k episodes)
Training Episodes	20000
Testing Episodes	100
Power Step (Watt)	5

The DRL performance was assessed by inferring the trained agents in 100 different episodes. Each episode was configured with random number of users, user positioning and user mobility speed (1 m/s for pedestrians, 15 m/s for vehicles). Fig. 12 shows the achieved DRL performance in the 100 scenarios, including (i) the total consumed power (summed across PRBs and RU power levels), (ii) the total allocated throughput (summed across users' data-rates) and (iii) the final achieved EE. Also, the bar-plots quantify both the mean consumed power and achieved throughput (averaged across the validation scenarios). Overall, the DRL-assisted EE optimization can significantly reduce the power consumption of the system (∼50 Watt or 40% power savings), while ensuring slightly enhanced throughput allocation (∼4Mbps), compared with the initial system state. By dividing the throughput by the consumed power curves and taking the mean across the 100 scenarios, DRL-assisted solution provides an average EE of 1.31 Mbps/Watt, whereas the non-DRL power allocation shows an EE of 0.66 Mbps/Watt.

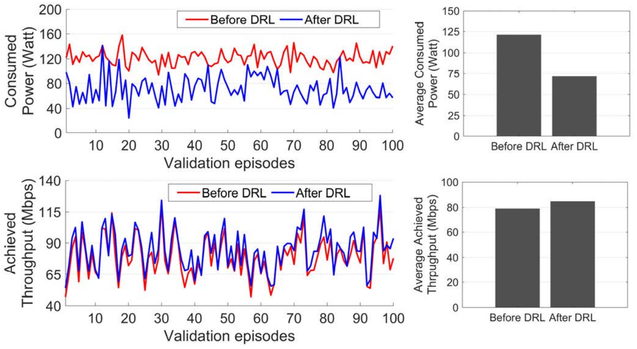

FIGURE 12. Consumed power (upper panel) and allocated throughput (lower panel) resulted from 100 validation episodes. Bar plots depict the average (across the 100 validation scenarios) allocated power and throughput of the non-DRL-assisted and DRL-assisted power regulation.

C. WORKFLOW OF O-RAN INTELLIGENCE

In this section, we describe in detail the workflow of AI/ML in O-RAN-based 5G system architectures. This general-purpose illustration targets to elucidate the involved architectural blocks (boxes) and the respective interlinking actions (arrows) required to support AI/ML functionalities. A unique UML diagram is suggested in Fig. 13, aiming to host both SL-based and RL-based models. Without loss of generality, the proposed intelligence loop refers to the control loop 2 of O-RAN specifications, given that the Near-RT RIC is the key intelligence actor. The workflow is divided into 6 discrete stages:

1) MODEL CONSTRUCTION

The process starts with the model construction phase, including an ML developer initiating a programming environment and formulating an ML model. This procedure may be accomplished using widely established development toolkits, such as Python libraries (Keras, Tensorflow, PyTorch) or AI/ML platforms (e.g. AcumosAI, Airflow). At the end of the design process, the Orchestrator located in SMO is acknowledged for the upcoming model training via a description file. Notably, this process may support the intent-based management, meaning that the Orchestrator can check whether the described objective-specific model already exists in the ML catalogue.

2) MODEL TRAINING

Afterwards, the Orchestrator sends the description file to the ML Model Trainer in order to initialize the training process. Then, a particular subset of the data gathered by the Data Collector is selected from the ML Model Trainer to represent the training samples. The training procedure exploits the description file to construct the AI/ML model with the reported architectural parameters (dimensionality, number of

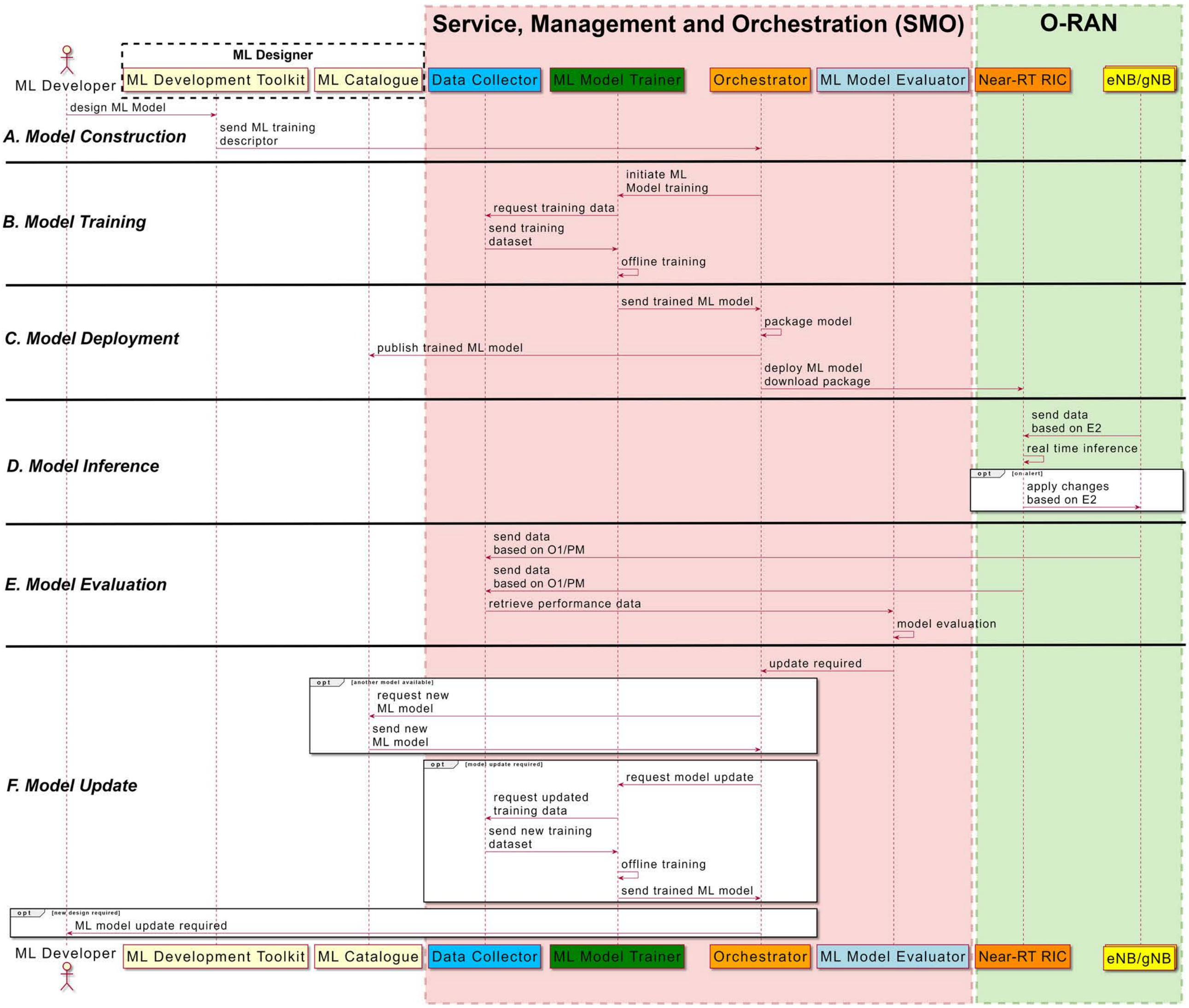

FIGURE 13. General-purpose UML diagram of the AI/ML delivery cycle in O-RAN based 5G system architectures. The workflow is divided in six (A–F) discrete stages, illustrating the complete AI/ML provision, spanning from the model construction to the model evaluation/update stages. Background colored components index their respective location (red: SMO, green: O-RAN), whereas the white "opt" boxes depict "optional" or "conditional" entries in the pipeline.

hidden layers, loss function, etc.). The hyperparameter fine-tuning takes place in the ML Model Trainer using widely-known tools (e.g. Tensorboard).

3) MODEL DEPLOYMENT

Once the model is stabilized with the optimal hyperparameters, it is sent back to the Orchestrator, where the packaging/dockerization (accompanied with the associated license and/or metadata) is achieved by the standardized tools (e.g. Docker, MLOps). Then, the trained model is stored in the ML Catalogue, being available for intent-based multi-vendor exploitation. Finally, the Orchestrator is also responsible for deploying the packaged trained model to the Near-RT RIC, hosting the model inference in the O-RAN part of the 5G network.

4) MODEL INFERENCE

Near-real inference iterations are continuously performed during the O-RAN operation, with the Near-RT RIC collecting data from the O-RAN elements (O-RU, O-DU, O-CU and/or eNB nodes for non-standalone deployments). For instance, the collected data for the presented LSTM scenario would be the 96 previous values of the total cell load in the operating cells and the inference outcome would be the next (after 15 min) predicted traffic in Gbps. In the DRL scenario, the collected data would represent the current time slot CQI values of each active PRB and cell, and after inferring the multi-agent model, the Near-RT RIC applies the suggestive power regulation actions to the O-RAN nodes. Note that, the corrective action in the DRL case is directly provided by the inference outcome, whereas the LSTM model simply provides a predicted traffic value to the Near-RT RIC.

In this case, an internal policy could be predefined in the Near-RT RIC to create alerts on extreme traffic conditions, along with the respective corrective actions. Finally, the Near-RT RIC extracts evaluation metrics resulted from the applied action (e.g. the EE increment in the DRL scenario).

5) MODEL EVALUATION

For performance monitoring (PM) purposes, the collected data from the O-RAN elements are also sent to the Data Collector via the O1/PM interface, along with performance data from the Near-RT RIC. Then, the ML Model Evaluator entity exploits the PM data to quantify whether a model re-training is needed or not. This can be achieved by either a threshold-based policy (update is required when PM data exceeds a predefined value) or a trend analysis investigation (update is required when a negative-going PM curve is observed).

6) MODEL UPDATE

In case that a retraining alarm is triggered, the ML Model Evaluator notifies the Orchestrator for the required model update. The Orchestrator then selects between two options, depending on whether an appropriate high-performance model is already available in the ML Catalogue (potentially provided by other vendors) or not. In the latter case, a new ML training cycle is initiated, with the ML Model Trainer reinitiating the training procedure using an extended training dataset collapsed with recently gathered data. Finally, a completely different model designing is also supported in case that the previous two options do not suffice.

Noteworthy, although here we presented the near-RT RIC engagement in the control of the network (control loop 2), the described AI/ML workflow can be effortlessly extended to support the other O-RAN control loops as well, following similar steps. For instance, the control loop 1 (involving CU/DU as the inference host) could be easily adapted, especially to serve DRL algorithms. In that case, the model will be deployed directly at the CU/DU component and the inference data will be collected through the Open-FH interface. This interface will be also utilized for applying the final corrective actions.

Moreover, the AI/ML workflow can be also extended with minor changes to host the control loop 3, especially in the SL case. For example, to include the non-RT RIC in the considered SL intelligence loop, the model could be directly deployed in the SMO, whereas the traffic prediction could be exploited by another component/subject of action (e.g. Orchestrator). This module can then take corrective actions (e.g. threshold-based alarm triggering) and apply them to the O-RAN network entities (e.g. different allocation between network slices) through the O1 interface. Finally, various collaborative models can simultaneously run in the different control loops, meaning that a model running in control loop 3 can also supervise control loop 2 models.

Finally, some key limitations in using synthetic or simulated data for training the AI/ML models should be identified. Real operators' data are difficult to be obtained, given the privacy and confidential issues in publishing/sharing real traffic data. Based on the above considerations, in this study, we relied on (i) a realistic-but-synthetic dataset for training the LSTM models and (ii) a 5G network measurement generator for training the DRL agents. The selection criteria for the LSTM dataset were primarily (i) the compliance of the spatiotemporal traffic patterns for 5G networks with 3GPP standards, using the fields (Data rate, Network area information, Subscription categories, Personal equipment ID), and (ii) the realistic assumptions in traffic properties, such as the mean handover ratios according to the time of the day [23]. On the other hand, DRL model training relies on the channel estimations provided by the 5G measurement generator. Channel imperfections are expected to be relatively reduced, given that path losses are calculated based on the empirical 3GPP standards for UMa cells, as well as the agent is trained on discretized (and not the exact) channel coefficients (i.e. the CQI values). In any case, the training of models on real network data may result into model performance variations, thus a soft retraining on some real data is always suggested prior deploying the proposed models in real network environments.

VI. CONCLUSION

In summary, O-RAN currently comprises the most attractive solution for deploying the next-generation multi-vendor networks, embracing the ideas of open, programmable, collaborative and intelligent communications. In this context, this paper gave an overview of the key architectural principles underlying the O-RANs, especially focusing on the AI/ML components involved in the architecture. A high-level overview of the network-wide data collection functionalities was also proposed to ensure collection of training data both in the radio access and core network parts. To give concrete use cases on how the AI/ML can be efficiently supported in the O-RAN based 5G systems, two optimization scenarios were presented, namely (i) a cell load predictive model using LSTMs and (ii) an energy efficiency-targeted model using distributed DRL agents. After quantitatively training and validating the AI/ML models, we presented a general-purpose workflow for the AI/ML model construction, delivery and evaluation. Several modifications in the described workflow can be adopted in the future, depending on operator-oriented requirements, technical preferences or architectural differences, however the key steps of AI/ML pipeline are discussed. Overall, this paper provides evidence on how an open and disaggregated RAN deployment can support predictive and optimization objectives, useful both for researchers and practitioners who drive the O-RAN and B5G evolution.

ACKNOWLEDGMENT

The authors would like to thank the Affordable5G's partnership for their valuable contribution in the technical aspects of this article. Special thanks to Simon Pryor for his contribution to Near-RT RIC architecture.

REFERENCES

[1] M. Gundall, M. Strufe, H. D. Schotten, P. Rost, C. Markwart, R. Blunk, A. Neumann, J. Griebbach, M. Aleksy, and D. Wubben, "Introduction of a 5G-enabled architecture for the realization of industry 4.0 use cases," *IEEE Access*, vol. 9, pp. 25508–25521, 2021.

[2] C. Benzaid and T. Taleb, "AI-driven zero touch network and service management in 5G and beyond: Challenges and research directions," *IEEE Netw.*, vol. 34, no. 2, pp. 186–194, Mar. 2020.

[3] J. Jankovic, Z. Ilic, A. Oracevic, S. M. A. Kazmi, and R. Hussain, "Effects of differentiated 5G services on computational and radio resource allocation performance," *IEEE Trans. Netw. Service Manage.*, vol. 18, no. 2, pp. 2226–2241, Jun. 2021.

[4] M. Bagaa, T. Taleb, J. Riekki, and J. Song, "Collaborative cross system AI: Toward 5G system and beyond," *IEEE Netw.*, vol. 35, no. 4, pp. 286–294, Jul. 2021.

[5] *O-RAN Architecture Description—V2.00*, O-RAN Working Group 1, Alfter, Germany, 2020.

[6] L. Bonati, M. Polese, S. D'Oro, S. Basagni, and T. Melodia, "Open, programmable, and virtualized 5G networks: State-of-the-art and the road ahead," *Comput. Netw.*, vol. 182, Dec. 2020, Art. no. 107516.

[7] A. Garcia-Saavedra and X. Costa-Perez, "O-RAN: Disrupting the virtualized RAN ecosystem," *IEEE Commun. Standards Mag.*, vol. 5, no. 4, pp. 96–103, Dec. 2021.

[8] J. K. Chaudhary, A. Kumar, J. Bartelt, and G. Fettweis, "C-RAN employing xRAN functional split: Complexity analysis for 5G NR remote radio unit," in *Proc. Eur. Conf. Netw. Commun. (EuCNC)*, Jun. 2019, pp. 580–585.

[9] *O-RAN: Towards an Open and Smart RAN*, O-RAN Alliance, Alfter, Germany, Oct. 2018.

[10] D. Harutyunyan and R. Riggio, "Flex5G: Flexible functional split in 5G networks," *IEEE Trans. Netw. Service Manag.*, vol. 15, no. 3, pp. 961–975, Sep. 2018.

[11] *O-RAN AI/ML Workflow Description and Requirements Version—01.01*, O-RAN Alliance, Alfter, Germany, 2020.

[12] *O-RAN Near-Real-Time RAN Intelligent Controller Architecture & E2 General Aspects and Principles—V1.01*, O-RAN Working Group 3, Alfter, Germany, 2020.

[13] H. Lee, J. Cha, D. Kwon, M. Jeong, and I. Park, "Hosting AI/ML workflows on O-RAN RIC platform," in *Proc. IEEE Globecom Workshops (GC Wkshps*, Dec. 2020, pp. 1–6.

[14] P. Trakadas, L. Sarakis, A. Giannopoulos, S. Spantideas, N. Capsalis, P. Gkonis, P. Karkazis, G. Rigazzi, A. Antonopoulos, M. A. Cambeiro, S. Gonzalez-Diaz, and L. Conceição, "A cost-efficient 5G non-public network architectural approach: Key concepts and enablers, building blocks and potential use cases," *Sensors*, vol. 21, no. 16, p. 5578, Aug. 2021.

[15] A. Gabilondo, Z. Fernandez, A. Martin, R. Viola, M. Zorrilla, P. Angueira, and J. Montalban, "5G SA multi-vendor network interoperability assessment," in *Proc. IEEE Int. Symp. Broadband Multimedia Syst. Broadcast. (BMSB)*, Aug. 2021, pp. 1–6.

[16] M. Chahbar, G. Diaz, A. Dandoush, C. Cerin, and K. Ghoumid, "A comprehensive survey on the E2E 5G network slicing model," *IEEE Trans. Netw. Service Manage.*, vol. 18, no. 1, pp. 49–62, Mar. 2021.

[17] S. K. Singh, R. Singh, and B. Kumbhani, "The evolution of radio access network towards open-RAN: Challenges and opportunities," in *Proc. IEEE Wireless Commun. Netw. Conf. Workshops (WCNCW)*, Apr. 2020, pp. 1–6.

[18] M. Khan, Z. H. Fakhri, and H. S. Al-Raweshidy, "Semistatic cell differentiation and integration with dynamic BBU-RRH mapping in cloud radio access network," *IEEE Trans. Netw. Service Manage.*, vol. 15, no. 1, pp. 289–303, Mar. 2018.

[19] K. Nguyen, P. L. Nguyen, Z. Li, and H. Sekiya, "Empowering 5G mobile devices with network softwarization," *IEEE Trans. Netw. Service Manage.*, vol. 18, no. 3, pp. 2492–2501, Sep. 2021.

[20] I. Chih-Lin, S. Kuklinski, T. Chen, and L. Ladid, "A perspective of O-RAN integration with MEC, SON, and network slicing in the 5G era," *IEEE Netw.*, vol. 34, no. 6, pp. 3–4, Nov. 2020.

[21] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta, and D. Sabella, "On multi-access edge computing: A survey of the emerging 5G network edge cloud architecture and orchestration," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1657–1681, 3rd Quart., 2017.

[22] X. Foukas, G. Patounas, A. Elmokashfi, and M. K. Marina, "Network slicing in 5G: Survey and challenges," *IEEE Commun. Mag.*, vol. 55, no. 5, pp. 94–100, May 2017.

[23] S. Sevgican, M. Turan, K. Gokarslan, H. B. Yilmaz, and T. Tugcu, "Intelligent network data analytics function in 5G cellular networks using machine learning," *J. Commun. Netw.*, vol. 22, no. 3, pp. 269–280, Jun. 2020.

[24] *Study on Channel Model for Frequencies From 0.5 to 100 GHz*, document 3GPP TR 38.901 Version 14.0.0 Release 14, 2017.

[25] Y. Sun, M. Peng, Y. Zhou, Y. Huang, and S. Mao, "Application of machine learning in wireless networks: Key techniques and open issues," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3108–3072, 4th Quart., 2019.

[26] C. Jiang, H. Zhang, Y. Ren, Z. Han, K.-C. Chen, and L. Hanzo, "Machine learning paradigms for next-generation wireless networks," *IEEE Wireless Commun.*, vol. 24, no. 2, pp. 98–105, Apr. 2017.

[27] M. Chen, U. Challita, W. Saad, C. Yin, and M. Debbah, "Artificial neural networks-based machine learning for wireless networks: A tutorial," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3039–3071, 4th Quart., 2019.

[28] M. E. Morocho-Cayamcela, H. Lee, and W. Lim, "Machine learning for 5G/B5G mobile and wireless communications: Potential, limitations, and future directions," *IEEE Access*, vol. 7, pp. 137184–137206, 2019.

[29] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 2018.

[30] A. Giannopoulos, S. Spantideas, C. Tsinos, and P. Trakadas, "Power control in 5G heterogeneous cells considering user demands using deep reinforcement learning," in *IFIP Int. Conf. Artif. Intell. Appl. Innov.*, vol. 628, 2021, pp. 95–105.

[31] A. Kaloxylos, A. Gavras, D. C. Mur, M. Ghoraishi, and H. Hrasnica, "AI and ML—Enablers for beyond 5G networks," Zenodo, Honolulu, HI, USA, Tech. Rep., 2020, doi: 10.5281/zenodo.4299895.

[32] *Architecture Enhancements for 5G System (5GS) to Support Network Data Analytics Services*, document TS 23.288, 3rd Generation Partnership Project (3GPP), Version 16.1.0, Sep. 2019. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3579

[33] *5G; Management and Orchestration; Architecture Framework*, document ETSI TS 128 533, 3GPP TS 28.533, Version 15.2.0, Release 15, 2019.

[34] Y. Hua, Z. Zhao, R. Li, X. Chen, Z. Liu, and H. Zhang, "Deep learning with long short-term memory for time series prediction," *IEEE Commun. Mag.*, vol. 57, no. 6, pp. 114–119, Jun. 2019.

[35] P. Alemany, A. Roman, R. Vilalta, A. Pol, J. Bonnet, E. Kapassa, M. Touloupou, D. Kyriazis, P. Karkazis, P. Trakadas, J. Martrat, R. Casellas, R. Martinez, and R. Munoz, "A KPI-enabled NFV MANO architecture for network slicing with QoS," *IEEE Commun. Mag.*, vol. 59, no. 7, pp. 44–50, Jul. 2021.

[36] A. Giannopoulos, S. Spantideas, N. Kapsalis, P. Karkazis, and P. Trakadas, "Deep reinforcement learning for energy-efficient multi-channel transmissions in 5G cognitive HetNets: Centralized, decentralized and transfer learning based solutions," *IEEE Access*, vol. 9, pp. 129358–129374, 2021.

[37] S. T. Spantideas, A. E. Giannopoulos, N. C. Kapsalis, A. Kalafatelis, C. N. Capsalis, and P. Trakadas, "Joint energy-efficient and throughput-sufficient transmissions in 5G cells with deep *Q*-learning," in *Proc. IEEE Int. Medit. Conf. Commun. Netw. (MeditCom)*, Sep. 2021, pp. 7–10.

ANASTASIOS GIANNOPOULOS received the Diploma of Electrical and Computer Engineering and Master of Engineering (M.Eng.) degrees from the National Technical University of Athens (NTUA), in 2018, where he is currently pursuing the Ph.D. degree with the Wireless and Long Distance Communications Laboratory (WLDCL). He is also with the National and Kapodistrian University of Athens. He has authored more than 12 articles in international scientific journals and conferences. He is included in the team composition of six European projects, among those an ongoing European 5G project, and he has participated in one 5GPPP white paper from the technical board. His research interests include advanced optimization techniques for wireless systems, ML-assisted resource allocation, cognitive radios, and multidimensional data analysis.

SOTIRIOS SPANTIDEAS received the Diploma of Electrical and Computer Engineering degree from the Polytechnic School of the University of Patras, in 2010, with the thesis "Design of Meander Printed Antennas in Portable Terminal Devices," the M.Sc. degree from the KTH Royal Institute of Technology (KTH), in 2013, under the Master's Program "Electrophysics," and the Ph.D. degree from the National Technical University of Athens (NTUA), in 2018, with a doctoral dissertation "Development of Methods for Obtaining DC and Low Frequency AC Magnetic Cleanliness in Space Missions." He is currently a Postdoctoral Research Associate with the Wireless and Long Distance Communications Laboratory, NTUA. He is also with the National and Kapodistrian University of Athens. His research interests include electromagnetic compatibility, machine learning for wireless networks, magnetic cleanliness for space missions, and optimization algorithms for computational electromagnetics.

NIKOLAOS KAPSALIS received the B.Sc. degree in computer science from the Department of Informatics, Athens University of Economics and Business, in 2008, and the M.B.A. degree in engineering economic systems and the Ph.D. degree from the School of Electrical and Computer Engineering, NTUA, in 2011 and 2015, respectively. He has authored articles in international scientific journals and conferences, and he is also a coauthor in a published book's chapter. His research interests include advanced signal processing, radio resource management, and optimization methods for wireless networks.

PANAGIOTIS GKONIS received the Diploma degree in electrical and computer engineering from the School of Electrical and Computer Engineering, National Technical University of Athens (NTUA), Greece, in 2005, the M.Sc. degree in engineering-economic systems program from NTUA, in 2009, and the Ph.D. degree from the School of Electrical and Computer Engineering, NTUA, in 2009. From 2010 to 2015, he worked as a Postdoctoral Researcher with the Intelligent Communications and Broadband Networks Laboratory (ICBNET), NTUA. He has also served as a Scientific/Laboratory Associate for the former Technological Educational Institute of Sterea Ellada, in the Departments of Electrical and Aircraft Engineering. In December 2015, he was appointed as an Electrical Engineer at Hellenic Distribution Network Operator (HEDNO), and in April 2019, he resigned to serve as an Assistant Professor with the National and Kapodistrian University of Athens (NKUA). He is an Assistant Professor with the General Department of NKUA. As a researcher, he has participated in various national and European funded projects. He is an author/coauthor of more than 50 scientific publications in international journals, scientific conferences, and book chapters, in the areas of wireless networks and broadband communications as well as computational modelling of cellular networks.

LAMBROS SARAKIS (Member, IEEE) received the Diploma degree in electrical and computer engineering from the National Technical University of Athens (NTUA), in 1996, the M.Sc. degree in communications and signal processing from Imperial College London, in 1997, and the Ph.D. degree in computer engineering from NTUA, in 2002. From 2004 to 2010, he served as an Assistant Researcher with the National Centre for Scientific Research "Demokritos," Institute of Informatics and Telecommunications, Greece. He is an Associate Professor with the Department of Digital Industry Technologies, National and Kapodistrian University of Athens. His research interests include 5G networks, sensor, mobile *ad-hoc* and vehicular networks, communication systems for smart grids, and mobility management in heterogeneous wireless networks. He has over 60 publications in edited books, journals, and conference proceedings in the above areas. He has participated in several research projects at national and international scale in research and management capacities, and has served on the technical program or organizing committee of several conferences and workshops.

CHRISTOS CAPSALIS received the bachelor's degree in economics from the National and Kapodistrian University of Athens, Athens, Greece, and the Diploma of Electrical Engineering, M.E., and Ph.D. degrees in electrical engineering from the National Technical University of Athens (NTUA), Athens, in 1979 and 1985, respectively. In 1986, he was a Lecturer with the Electrical Engineering Department, NTUA. In 1989, he was an Assistant Professor, and in 1993, he was an Associate Professor. Since 1995, he has been a Professor in electrical engineering. He has authored five books in the field of electromagnetic radiation and modelling. His research interests include electromagnetic fields, scattering and propagation of millimeter waves and optical wavelengths, and mobile telecommunication systems. His research interests also include next-generation wireless networks, radio resource management, and inter-country frequency coordination.

MASSIMO VECCHIO received the M.Sc. degree *(magna cum laude)* in information engineering from the University of Pisa, Pisa, Italy, in 2005, and the Ph.D. degree in computer science and engineering from the IMT School for Advanced Studies Lucca, Lucca, Italy, in 2009. Since 2015, he has been an Associate Professor with eCampus University. In 2017, he joined FBK CREATE-NET, Trento, Italy, to coordinate the research activities of the OpenIoT Research Unit. He is the Project Coordinator of AGILE, a project co-founded by the Horizon 2020 Program of the European Union. His current research interests include computational intelligence and soft computing techniques, the Internet of Things paradigm, and effective engineering design and solutions for constrained and embedded devices. Regarding his most recent editorial activity, he is a member of the Editorial Board of the journal of *Applied Soft Computing* and of the newborn *IEEE Internet of Things Magazine*, besides being the Managing Editor of the IEEE IoT Newsletters. He received the Doctor Europaeus mention from the IMT School for Advanced Studies Lucca.

PANAGIOTIS TRAKADAS received the Diploma and Master of Engineering degrees in electrical and computer engineering and the Ph.D. degree from the National Technical University of Athens. In the past, he was with the Technological Educational Institute of Athens as an Assistant Professor, and the Technological Educational Institute of Chalkida and the National Technical University of Athens as a Senior Researcher. He is an Associate Professor with the Department of Ports Management and Shipping, National and Kapodistrian University of Athens. He has published more than 130 papers in magazines, journals, books, and conferences. He is a reviewer in several journals and TPC in conferences. He has been actively involved in many research projects. His main research interests include 5G technologies, such as NFV and NOMA, wireless sensor networks, signal processing, and routing protocols.