IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received 17 May 2024, accepted 5 June 2024, date of publication 11 June 2024, date of current version 18 June 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3412758

TOPICAL REVIEW

Energy Consumption of Machine Learning Enhanced Open RAN: A Comprehensive Review

XUANYU LIANG, (Student Member, IEEE), QIAO WANG, (Member, IEEE), AHMED AL-TAHMEESSCHI, (Member, IEEE), SWARNA B. CHETTY, (Member, IEEE), DAVID GRACE, (Senior Member, IEEE), AND HAMED AHMADI, (Senior Member, IEEE)
School of Physics, Engineering and Technology, University of York, YO10 5DD York, U.K.

Corresponding author: Hamed Ahmadi (Hamed.ahmadi@york.ac.uk)

This work was supported in part by the Engineering and Physical Sciences Research Council United Kingdom (EPSRC), Impact Acceleration Accounts (IAA) (Green Secure and Privacy Aware Wireless Networks for Sustainable Future Connected and Autonomous Systems), under Grant EP/X525856/1; and in part by the Department of Science, Innovation and Technology, U.K., through Yorkshire Open-RAN (YO-RAN), under Grant TS/X013758/1.

ABSTRACT The Open Radio Access Network (RAN) emerges as a revolutionary architecture promising unprecedented levels of openness, flexibility, and intelligence within radio access networks. Central to this innovation is the integration of Machine Learning (ML) and Artificial Intelligence (AI) within the RAN Intelligent Controller (RIC), aimed at optimizing network operations and enhancing control mechanisms. This paper undertakes a thorough examination of Open RAN, particularly focusing on its energy consumption aspects, which are pivotal for ensuring the sustainability of future wireless networks. In this paper, we review and compare Open RAN architecture with previous network architectures. In particular we focus on O-RAN Alliance specifications. Additionally, we explore the deployment of ML across various facets of Open RAN and highlights how to estimate the energy consumption of ML models. Through constructing explicit energy consumption models for key O-RAN components, we provide a granular analysis of their energy profiles. Finally we compare the energy dynamics of O-RAN against traditional RAN architectures, delineating the impact of virtualization and disaggregation on energy efficiency.

INDEX TERMS Open radio access network (Open RAN), energy efficiency, machine learning, disaggregation.

I. INTRODUCTION

Fifth Generation (5G) cellular networks are engineered to greatly increase the speed and responsiveness of wireless networks. The services provided by the 5G cellular networks can be divided into three main categories, including enhanced Mobile Broadband (eMMB), Ultra-Reliable Low-Latency Communication (URLLC) and massive Machine-Type Communication (mMTC). Each service corresponds to a network slice and different requirements, and the standards for 5G networks need to meet these various demands. These advancements come with substantial energy requirements. By 2025, it is projected that the telecommunications industry will consume around 30% of global energy consumption [1]. The denser placement of 5G Base Station (BS), which are necessary to ensure network coverage and capacity, will result in a significant increase in energy consumption. In fact, approximately 80% of energy consumption in cellular mobile networks is attributed to BS [2]. A recent analysis conducted by the Global System for Mobile Communications

The associate editor coordinating the review of this manuscript and approving it for publication was Zesong Fei.

Association (GSMA) reveals that, on average, network operations are responsible for 90% of a mobile operator's energy consumption, with the RAN contributing to over 80% of this network-related energy usage [3]. Therefore, attaining optimization in energy efficiency constitutes a pivotal advancement in the evolution of the telecommunications industry. The energy efficiency-related requirements are even tighter in Sixth Generation of Mobile Networks (6G) [4].

Another critical concern to consider is that the current 5G cellular architecture is still far away from supporting multiple services and meeting their Quality of Service (QoS) due to its inherent design and operation. In the current stage of RAN development, the provision of RAN components, both hardware and software, is predominantly controlled by a single vendor. This arrangement significantly restricts the flexibility and adaptability of the system. Furthermore, the absence of standardized interfaces between various network nodes hampers their interactivity and interoperability. Compounding these issues, the black-box nature of the operating system further complicates efforts to reconfigure nodes to support diverse deployments and meet varying traffic demands in different scenarios. Consequently, in such an architecture, the task of achieving dynamic resource allocation and real-time energy efficiency optimization becomes a formidable challenge [5].

The evolution of cellular network architectures is an inevitable consequence of the ever-growing network service demands and emerging technologies. Among the proposed models for future RAN development, Open RAN stands out as a prominent approach. Open RAN introduces various concepts, including virtualization, disaggregation, intelligent and optimized management, and open interfaces [6], [7], to address the evolving requirements of the network. To work on this concepts, a world-wide community of mobile network operators, vendors, and research and academic institutions operating in the RAN industry formed O-RAN Alliance. The O-RAN Alliance defined specifications for next generation RAN which is known as O-RAN. In the rest of this paper, by the term "Open RAN" we refer to the general concept and in places where we use "O-RAN" we refer to the O-RAN Alliance one.

By leveraging virtualization technology, Open RAN enables flexible resource allocation and efficient utilization through the virtualization of network functions. Disaggregation breaks down the traditional monolithic approach by splitting the BS functionalities into Central Unit (CU), Distributed Unit (DU), and Radio Unit (RU) [6], allowing for the independent development and deployment of RAN components. Therefore, the characteristic of disaggregation leads to a distinct energy consumption model for Open-RAN compared to traditional Base Stations (BSs). This difference arises due to the virtualization of BS functions, with the DU and CU, and even parts of the RU, being deployed in the O-Cloud. Consequently, the energy consumption of DU and CU is generated by servers. In contrast, the energy consumption of RU can draw parallels to traditional BSs since they are physical entities. Moreover, the varied deployment locations of DU, CU, and RU also contribute to differences in energy consumption [8]. The integration of intelligent and optimized management techniques are supported by RAN Intelligent Controllers (RICs) [9], [10] empowering the network to dynamically adapt to changing conditions, optimize resource allocation, and enhance overall performance. Furthermore, the adoption of open interfaces fosters interoperability and allows for the integration of diverse components from multiple vendors, promoting innovation and avoiding vendor lock-in.

In the context of O-RAN, the integration of AI and ML models is facilitated by two RICs, namely Near-RT RIC and Non-RT RIC, which enable enhanced operational capabilities throughout the O-RAN architecture. The Near-RT RIC encompasses xApps where trained ML models are deployed to control near-real-time network operations with time scales ranging from 10ms to 1s [11]. The Non-RT RIC utilizes AI/ML for non-real-time intelligence functions that operate on larger time scales, typically exceeding 1s [12]. These functions encompass predictive maintenance and capacity planning. The Non-RT RIC is particularly suitable for training AI/ML models based on longer data paths, allowing for the identification of data trends and behavioral patterns over time. Detailed discussions on the AI/ML workflow and ML-based O-RAN applications will be provided in subsequent sections. To meet the high energy efficiency goals of new generations, traditional numerical optimization schemes are limited due to their high computing power requirements and lack of adaptability to the dynamic and evolving network environment. In contrast, ML-based optimization methods offer a promising range of applications. These methods can handle real-time problems and achieve their objectives using simpler optimization models, thereby reducing computing power and energy consumption to a certain extent [13]. O-RAN provides an exceptionally conducive environment and framework for both the training and inference phases of ML models.As presented the suitability of O-RAN for incorporating ML in the new generations of mobile networks and its effect on energy efficiency motivated us for this review paper. The work presented in [8] reviews power consumption models and energy efficiency techniques for O-RAN, identifies challenges in optimizing enregy efficiency, and suggests future research directions. It also delves into the components and power consumption models of O-RAN. However, the introduction of some formulas for power consumption models is missing and does not visually represent the differences between different energy models. It also lacks a detailed description of the O-RAN structure. The paper [5] provides a detailed tutorial on O-RAN, discussing its architecture, interfaces, and challenges, emphasizing the importance of understanding O-RAN for the wireless community. It explores the experimental research platforms for testing O-RAN networks. But still not consider about the energy consumption of O-RAN components. Next we present our contributions and the organisation of the paper.

A. CONTRIBUTIONS

The emergence of the O-RAN architecture is driving a significant evolution of the RAN towards virtualization and intelligence. This transformative shift brings about heightened network flexibility, improved interactivity, and disrupts the existing vendor monopoly.

- This article focuses on exploring the technical specifications of O-RAN, elucidating its architectural components, and delving into the open interfaces that interconnect the various O-RAN nodes.
- This paper also presents the ML background and the current ML methods employed in O-RAN. Furthermore, the paper provides a comprehensive review of recent research on energy consumption generated by ML models during the training and inference phase.
- This study reviews the ML-based approaches adopted by previous generation RANs for improving energy efficiency. With previous RANs, paper also highlights how energy efficiency improvement can be achieved in O-RAN architecture.
- Our work delves into the specifics of power consumption associated with key O-RAN components, including the RU, CU, and DU. By dissecting these models, we identify the elements that significantly impact energy usage and how they interplay within the O-RAN structure.
- The paper features a practical case study by comparing the energy consumption of conventional BS and O-RAN architecture, shedding light on real-world energy consumption scenarios in O-RAN.

B. PAPER STRUCTURE

The paper is structured as follows to provide a comprehensive understanding of the subject matter. Section II presents an overview of the evolution of RAN, tracing its progression from Distributed-RAN (D-RAN) to Vritualized-RAN (v-RAN). This serves as a foundation for comprehending the innovations introduced by O-RAN, which are discussed in Section III. Moving forward Section IV and V introduce the architecture of both Near-RT and Non-RT RIC, delving into their respective functionalities and interfaces. Section VI provides an in-depth analysis of the background of ML, encompassing classic algorithms. In Section VII,energy consumption of ML model during training and inference and multiple measurement applications to estimate the energy consumption. In section VIII, AI/ML workflow is introduced and the paper also explores applications that combine ML techniques with O-RAN. In Section IX, the paper overviews energy efficiency approaches in previous RAN and in Section X it examines the approaches and strategies employed to optimize energy consumption in O-RAN. Following this, Section XI introduces a energy consumption model applicable to O-RAN, elaborating on its parameters and utility. Concluding the paper, Section XII features a case study, wherein we draw a direct comparison between the energy expenditures of traditional BSs and O-RAN systems, highlighting the distinctions and analyzing their implications.

II. OVERVIEW OF RAN EVOLUTION

This section will review the evolution of RAN architecture, starting from D-RAN to Cloud RAN (C-RAN) and then through v-RAN, for a better understanding of the structure of O-RAN and how some innovations came about in the next section.

A. DISTRIBUTED RAN (D-RAN)

In 2G networks, both baseband and radio signals were processed by the BSs, enabling digital voice transmission and basic data services for mobile devices. However, the evolution to 3G/4G networks introduced a new architecture called Distributed RAN (D-RAN) [14], which replaced the unified BS function of 2G RAN with separate components: the Base Band Unit (BBU) and the Remote Radio Unit (RRU) or Remote Radio Head (RRH). In D-RAN, each cell site is equipped with its own BBU, responsible for processing the baseband signals. The RRU, located at the cell site, connects to the mobile devices for receiving and transmitting signals under the control of the BBU. To ensure efficient communication between the BBU and RRU, a high-capacity fronthaul link, often implemented using technologies like optical fibers, is employed. This fronthaul link facilitates high data rates and low latency between the BBU and the RRU, enabling seamless communication.

B. CENTRALIZED RAN (C-RAN)

Unlike the traditional distributed cell site-based architecture, the C-RAN adopts a centralized approach where baseband processing and control protocol operations are consolidated in a BBU pool located at a central site. The C-RAN architecture offers several advantages, including significant cost and energy efficiency gains resulting from hardware consolidation [15]. The centralized approach enables cooperative radio resource management, leading to improved network performance and resource utilization. Additionally, C-RAN provides scalability and flexibility to accommodate fluctuations in network traffic and emerging technologies. However, the adoption of C-RAN also introduces challenges [16], particularly the requirement for high-capacity and low-latency fronthaul connections to ensure seamless communication between the RRHs and the centralized BBU pool [17]. These connections play a crucial role in maintaining the synchronization and timely delivery of data between the network elements. Overall, C-RAN represents a transformative network architecture that delivers cost savings, improved performance, and energy efficiency through centralized baseband processing and cooperative resource management. While it presents challenges related to fronthaul connectivity, C-RAN offers a promising solution for addressing the evolving needs of mobile networks.

C. VIRTUALIZED RAN (V-RAN)

To address the diverse requirements of 5G networks, the concept of v-RAN has been proposed, incorporating key technologies such as Network Function Virtualization (NFV) and Software Defined Network (SDN) [18]. NFV enables the decoupling of network functions from dedicated hardware, allowing them to be implemented as software applications on standard servers. SDN, on the other hand, separates the network control plane from the data plane, enabling centralized control and dynamic resource management. These technologies play a crucial role in the scalability, flexibility, and efficiency of V-RAN. In a V-RAN architecture [19], [20], the traditional BBU functions are virtualized and referred to as the V-BBU. These virtualized BBUs run as software on commodity servers within a data center environment. To ensure the necessary processing power, these servers are often organized in clusters, forming a cloud-based BBU pool or Cloud-RAN. The Cloud-RAN is connected to the cell sites (network edge) where the RRHs are located. The crucial link between the Cloud-RAN and the RRHs is established through Fiber Ethernet links. These fronthaul links are essential to support real-time processing requirements in mobile networks.

III. OPEN RAN KEY INNOVATIONS

Building on the section above, this section takes a closer look at key innovations in the Open RAN architecture to adapt to the ever more complex network environment including disaggregation, RAN Intelligent controllers and closed-Loop control, virtualisation and open interfaces.

A. DISAGGREGATION

The first characteristic of RAN is disaggregation. It splits the BS into units with different functions. The structure corresponds to the New Radio (NR) Next Generation Node Bases (gNBs) defined by 3GPP. The gNB is split into CU, DU and RU. where CU is further divided into two distinct parts, Control Plane (CP) and User Plane (UP). This splitting method allows each part to exist independently of the other. Different functions can therefore be located in different parts of the network or on different hardware platforms, making the whole network more flexible and versatile. The different functions of those three parts are as follows. The interfaces of them will be given in later section.

1) OPEN-RADIO UNIT (O-RU)

RUs, typically constructed using Field Programmable Gate Arrays (FPGAs) and Application-specific Integrated Circuits (ASICs) without virtualization, are situated in proximity to the RF antennas, as highlighted by [5]. The O-RAN Alliance has undertaken comprehensive assessments of various 3GPP-endorsed RU and DU partitioning protocols, showing a particular inclination towards the 7.2x division [5], given it balances the time delay with inter-network traffic. This specific split entails a distribution of the physical layer across the RU and DU. In this arrangement, the RU manages solely the lower level PHY layer processing (PHY-low) functions, encompassing tasks such as Fast Fourier Transform (FFT)/ IFFT and cyclic prefix manipulation, aiming to mitigate deployment complexities and associated expenses. Beyond PHY-low operations, RU's architecture integrates Radio Frequency (RF) processing elements, including power amplifiers, beamformers, and transceivers, to constitute a comprehensive operational unit.

2) DISTRIBUTED UNIT (O-DU)

The DU is responsible for the remaining higher level physical layer processing (PHY-high) such as channel modulation, part of precoding, and mapping into physical resource blocks, the Medium Access Control (MAC) layer and the Radio Link Control (RLC) layer. These three layers are generally closely synchronized due to the MAC layer generate Transport Blocks for physical layer by using the data buffered in RLC layer.

3) CENTRAL UNIT (O-CU)

The CU being placed higher up the 3GPP stack [21], which takes care of Radio Resource Control (RRC) layer, Packet Data Convergence Protocol (PDCP) layer and Service Data Adaptation Protocol (SDAP) layer. O-CU in O-RAN performs a wide range of radio functions, including efficient resource management, signal processing, connection management, quality of service optimization, network slicing support, and interface coordination [6]. Its comprehensive role is vital for enhancing the performance, efficiency, and flexibility of the RAN within the O-RAN architecture.

B. RAN INTELLIGENT CONTROLLERS AND CLOSED-LOOP CONTROL

Open RAN's notable second innovation is the RIC, a programmable entity designed to navigate the increasing complexities within network frameworks, thereby enhancing both accuracy and operational efficiency. By serving as a central network abstraction, the RIC compiles comprehensive Key Performence Measurements (KPMs) data, reflecting various facets of network infrastructure status (for instance, user metrics, traffic burdens, and throughput capacities) and information from resources outside of RAN. This comprehensive data spectrum is analyzed through AI and ML methodologies, facilitating decision-making pertaining to RAN policies and operational strategies [22]. The non-Real Time (RT) RIC and near-RT RIC are the two primary modules of RIC that O-RAN Alliance has introduced [6]. The control loop with RAN components is carried out by a near-RT RIC with a time scale between 10ms and 1s, while a non-RT RIC is in charge of operations on a time scale larger than 1s, such as training AI and ML models as mentioned above. Figure 1 provides an overview of the closed-loop control implemented by the RICs across the disaggregated O-RAN infrastructure and the real-time extensions that need to be considered in the future. In the following paragraphs we will discuss the functionality of the different RICs and the closed-loop control associated with them.

1) NON-REAL-TIME RIC AND CONTROL LOOP

The non-RT RIC constitutes an integral segment of the Service Management and Orchestration (SMO) framework, maintaining a direct connection with the A1 interface to near-RT RIC, as illustrated in Figure 1. Rather than operating through mutual interfacing within the SMO, the non-RT RIC exists as a specialized subset, executing only select functions of the broader SMO capabilities [10]. This component monitors operations related to RAN constituents on a temporal scale exceeding 1 second and facilitates AI/ML workflows, encompassing aspects like model training and configuration. Furthermore, it provides guidance and enrichment information to the Near-RT RIC utilizing a non-real-time control loop. Its connection, both direct and collateral, with interfaces like A1, O1, and O2 through the SMO, empowers the non-RT RIC to administer and configure RAN elements efficiently. An extensive discussion regarding the non-RT RIC and SMO is deferred to Section V.

2) NEAR-REAL-TIME RIC AND CONTROL LOOP

The near-RT RIC, functioning as a specialized Open-RAN network function, orchestrates near real-time supervision and enhancement of E2 Node resources and services. This is achieved through fine-grained data gathering and subsequent operations executed via the E2 interface, with control loops operating within an expedited 10 ms to 1-second timeframe. The near-RT RIC consists of multiple applications to support network functions, called xApps. An xApp hosts multiple microservices such as mobility management, traffic steering, radio connection management, and QoS management. Normally, an xApp receives the data from RAN components (i.e. CUs and DUs) and xApp sends back the action result after computing. At the same time, near-RT RIC also provides a range of functionality to support xApp, which will be mentioned in section IV.

C. VIRTUALIZATION

The third characteristic of Open-RAN is the O-Cloud which is a cloud computing platform to manage and optimize the network infrastructure and operations, changing the network from edge system to virtualization platform. All the components mentioned in Figure 1, can be deployed in the O-Cloud Platform, which includes the following characteristics [23]:

- The O-Cloud platform combines hardware and software components that support cloud computing capabilities to execute RAN network functions.
- The cloud platform hardware is standardized to support the requirement of RAN functions to satisfy their performance objectives.
- The software in the O-Cloud platform exposes the open and well-defined Application Programming Interface (API) to manage and orchestrate life cycle of network functions and O-Cloud as well.
- The software is decoupled from the hardware, which means it is available from a variety of vendors.

The virtualization of O-RAN components and internal network functions plays a crucial role in energy conservation. Virtualization enables the flexible allocation of network resources based on user demands, optimizing the use of network functions and, consequently, reducing energy consumption [24]. Furthermore, through the application of closed-loop control, as discussed earlier, dynamic cell activation and deactivation are facilitated, contributing to additional reductions in power consumption [25].

D. OPEN INTERFACES

Lastly, the O-RAN Alliance in order to overcome the limitation of RAN has introduced well-defined specifications that uses open interfaces to connect different O-RAN elements. Figure 1 demonstrates the open interfaces defined by O-RAN and internal interfaces as defined by 3GPP. Open interfaces allow for diversity of options and operators can choose from a variety of vendors in different locations which breaks up vendor monopolies and increases market competitiveness.

As shown in Figure 1, F1 and E1 interfaces are defined from 3GPP. F1 interface builds the connection between DUs and CUs. E1 interface likes a bridge binding the user planes and control planes at CU. The rest interfaces are all defined by the O-RAN Alliance, the E2 interface connects near-RT RIC to RAN nodes. The DU and CU send measurements to the near-RT RIC through the E2 interface and then the configuration command back to the CU and DU to form a near-real-time control loop shown in Figure 1. The A1 interface [7] facilitating information exchange between the near-RT and non-RT RIC, which enables AI/ML related parameters or models deploy on the near-RT RIC. Besides, the non-RT RIC and SMO also connect to the O-Cloud via O2 interface for SMO to intelligently manage and operate each O-RAN node running on the top of the O-Cloud platform. Finally the O1 interface, which is standardized by the 3GPP [21], is used for management and optimization of the RAN nodes.

The open interface characteristic of Open-RAN notably enhances its adaptability, permitting a flexible arrangement of components within the O-RAN architecture. This flexibility eliminates the need for a set location for network elements, enabling a configuration that is tailor-made to the specific demands of the network. For instance, the work in [26] positions the CUs and DUs at the network edge, with the RUs situated at the cell sites. Alternatively, there exists a scenario where the DUs and RUs are jointly situated at cell sites [27], leaving only the CUs at the edge, specifically to accommodate services sensitive to delays.

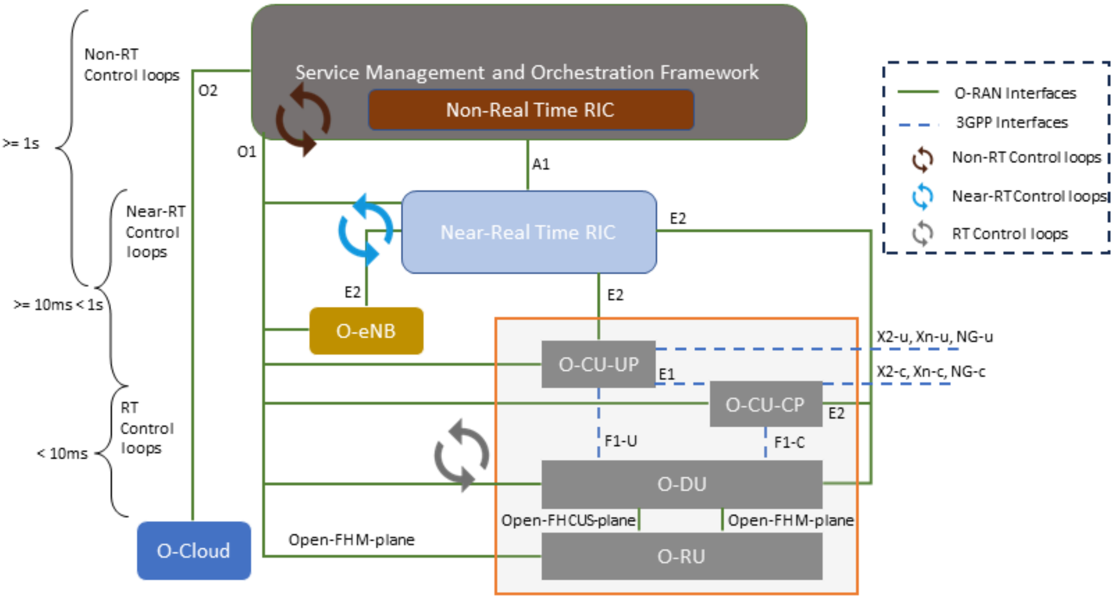


FIGURE 1. O-RAN architecture, with components and interfaces from O-RAN and 3GPP. O-RAN interfaces are drawn as solid lines, 3GPP ones as dashed lines. And three different control loops.

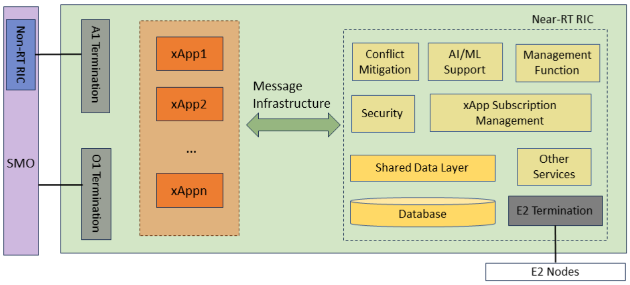


FIGURE 2. Near-RT RIC internal architecture.

IV. NEAR-RT RIC

The near-RT RIC is the core for control and optimization of the RAN through the E2 interface. It is illustrated in Figure 2 which is showing the whole internal function of near-RT RIC. The xApps deployed on the near-RT RIC are the significant part to run intelligent control of the near-real-time control loop. xApps could be provided by the third parties to support a variety of functions. According to O-RAN Alliance specification [28], an xApp consists of xApp descriptor and xApp software image which includes a set of files to support deployment of xApp. The xApp descriptor includes necessary information about xApps enables management of xApps, such as health management (e.g. autoscaling the policy when xApps are under excessive loads or unhealthy situations), deletion and update information. In addition, the xApp descriptor shall contain the data type generated or consumed by the xApps, in order to support control capabilities. In next subsection, based on O-RAN Alliance specification [9], [28], the internal functions and E2 interface of Near-RT RIC will be discussed.

A. NEAR-RT RIC INTERNAL FUNCTIONS

1) NETWORK INFORMATION BASE DATABASE AND SHARED DATA LAYER API

There are two types of Network Information Base (NIB) database, UE-NIB database and R-NIB database respectively. The former stores a list of UEs and UE identity. The UE identity is a really important and sensitive information in the RIC, because it allows UE-specific control, but at the same time it can expose sensitive information on the users. Therefore, UE-NIB database keeps tracking and correlation of the UE identities with their connected E2 nodes. The R-NIB database contains information (e.g. configurations and real-time information) on E2 nodes and the mapping between them. Shared Data Layer (SDL) API makes it possible to expose information from UE-NIB, R-NIB and other specific use case to each other, which facilities the development of a white-box system.

2) MESSAGING INFRASTRUCTURE

The internal messaging infrastructure in O-RAN specification [9] connects every components in near-RT RIC with low latency message delivery. It supports registration, discovery and deletion of endpoints (i.e. xApps and internal RIC components). And it also provides the APIs to transmit and receive messages in point-to-point mode or public/subscribe mode. It also provides routing and to avoid the data loss.

3) CONFLICT MITIGATION

This function addresses issues arising from conflict between different xApps. Such mitigation is essential since distinct xApps, executing diverse network functions, may initiate

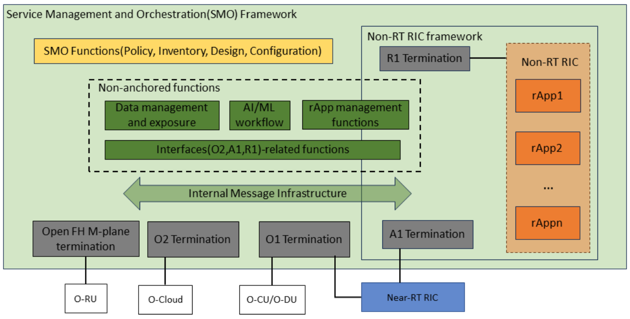

Service Management and Orchestration(SMO) Framework
SMO Functions(Policy, Inventory, Design, Configuration)
Non-RT RIC framework
R1 Termination
Non-RT RIC
rApp1
rApp2
...
rAppn
Non-anchored functions
Data management and exposure
AI/ML workflow
rApp management functions
Interfaces(O2,A1,R1)-related functions
Internal Message Infrastructure
Open FH M-plane termination
O2 Termination
O1 Termination
A1 Termination
O-RU
O-Cloud
O-CU/O-DU
Near-RT RIC

FIGURE 3. Non-RT RIC and SMO internal architecture.

configurations that clash, leading to a decline in overall performance. Conflicts may target various elements such as a cell, a bearer, or a UE and can involve any actions in radio resource control management. The O-RAN Alliance specifications categorize conflict control into three types: direct, indirect, and implicit conflicts. Direct conflicts are those that the mitigation function can immediately identify. These occur, for example, when two or more xApps set differing parameters for a single control target or when there is a discrepancy in the active configurations among xApps. On the other hand, neither indirect nor implicit conflicts are directly observable. However, indirect conflicts can be discerned through the interrelations between different xApps. An instance of this would be configurations that enhance certain users' performance while concurrently causing a non-apparent decline in others. Resolution for these issues might involve post-action verification methods, such as reverting specific xApps' actions based on comprehensive system observations. Implicit conflicts, regarded as the most challenging, are conflicts where the source is difficult to pinpoint. Mitigating these conflicts may necessitate the deployment of ML strategies to diminish the likelihood of their emergence.

4) SUBSCRIPTION MANAGEMENT

The subscription management enables the subscription from xApps to E2 nodes. It also authorizes xApps to access E2 messages, and can merge multiple identical subscription from different xApps to a single subscription request to the E2 node.

5) SECURITY

The security function to Near-RT RIC is going to prevent malicious xApps from leaking network information to unauthorized external systems or from overusing the RAN functions, affecting RAN performance. The detail of this components are still need further studies [29].

B. E2 INTERFACE

The E2 interface is an open interface connects the near-RT RIC and E2 nodes (i.e. DUs, CUs, and O-RAN-compliant LTE eNBs) supplied by the different vendors. The main function of E2 interface is to telemetry the information on E2 nodes (e.g. configuration information, network measurements, etc.) and then forward them to the near-RT RIC. Therefore, The E2 interface enables the near-RT RIC to control the E2 nodes. According to the O-RAN specifications [30], there are four categories RIC services provided by the E2 nodes and E2 interface.

1) REPORT

The report service is the E2 nodes send RIC Indication message includes the data, information or the measurements from E2 node to the near-RT RIC. But the E2 nodes are required to send a REPORT message contains the event trigger to the near-RT RIC by the near-RT RIC, before the RIC indication message is sent out. And the relevant produce continues when the event is occurred.

2) INSERT

The insert service is when the E2 nodes send an INSERT message to the near-RT RIC for the purpose of RIC subscription and the near-RT RIC suspends all relevant produces after the pre-defined event trigger is occurred. After the time wait timer has expired. The near-RT RIC will decide whether to stop or continue the relevant produces.

3) CONTROL

The control service is the near-RT RIC sends a RIC control request message to the E2 nodes. Then, the E2 nodes should delete the current produce or resume the setting of previous produce.

4) POLICY

The policy service is the near-RT RIC sends specific policy that the E2 nodes need to apply on ongoing produces when the relevant event trigger is occurred. But the E2 nodes should send the report message to the near-RT RIC to do the RIC subscription before the policy service happens.

V. SERVICE MANAGEMENT AND ORCHESTRATION (SMO) FRAMEWORKS

The second key element of the O-RAN architecture is the SMO framework. This component is in charge of handling all orchestration, management and automation produces to monitor and control RAN components. The SMO perform these services through four interfaces as shown in Figure 1 to the RAN elements [6]. A1 interface between the non-RT RIC in the SMO and the near-RT RIC for RAN optimization. O1 interface between the SMO and the O-RAN Network Functions for FCAPS support. Open Fronthaul M-Plane interface between SMO and O-RU for FCAPS support. O2 interface between the SMO and the O-Cloud to provide platform resources and workload management.The non-RT RIC is the functionality internal to the SMO framework as we mentioned in the Section II, and provides policy-based guidance, enrichment information, and AI/ML model management to the near-RT RIC via the A1 interface.

The architecture of the SMO illustrated in Figure 3, which includes the main functionalities and will be detailed in the later of this section. According to O-RAN Alliance specification, there is no specific standard to separate non-RT RIC and SMO functions. However, the specifications group divides such functionalities into three categories. The first set is identified such interfaces and functionalities are anchored inside the non-RT RIC framework. The second set is the opposite of the first one, which identifies the functionalities anchored outside the non-RT RIC. The last set is the non-anchored functions refer to those components or functionalities that are not exclusively tied to either the Non-RT RIC or the SMO's core management capabilities. The goal of next subsection is going to introduce these functionalities and interfaces of non-RT RICs.

A. NON-RT RIC

Similarly to the near-RT RIC, the non-RT RIC is one of the core components of the Open RAN architecture, it enables the closed-loop control of the RAN with the time scale larger than 1s and it also supports the execution of third-party applications, i.e. rApps, which are similar to the xApp, to provide RAN optimization and operations, including policy guidance enrichment information, configuration management and data analytic.

As shown in Figure 3. The non-RT RIC hosts the R1 termination, which allows rApps could exchange the message with the non-RT RIC framework. This allows the rApps enable access to data management and exposure service, AI/ML functionalities as well as A1, O1, O2 interfaces. It is worth mentioning that the rApp can not only support the same control functionalities provided by the xApps at the large timescale, but can realize management and orchestration at a higher level and affect a large number of users and nodes, including frequency and interface management, RAN sharing, end to end Service Level Agreement (SLA) assurance and network slicing.

According to the [31], the non-RT RIC architecture could be described by using two different versions, a functional approach architecture and a service-based approach architecture. The functional architecture represents all required functionalities of combination of non-RT RIC and SMO framework, which is similar to the architecture of Figure 3. All functions are fixed and are divided into three categories based on the position of the functional blocks. In functional approach, logic entities are well-defined and connect with interfaces. However in order to provide a more flexible architecture that the behaviors of logic entities or network components are not based on the fixed functions but can be changed in real time and make service as the central point of the architecture. In this [5] paper,the author summarized two intelligent services for the non-RT RIC. The first is intent-based network management, which allows operators could use human-machine interface to express their intents to the non-RT RIC. Then, the new configurations will be deployed on the rApps and xApps after the procession of the non-RT RIC. The intelligence orchestration is another non-RT RIC high-level service. The growth in complexity of network control is inevitable with the development of the O-RAN. This calls for the solutions enable orchestration of all xApps and rApps. The non-RT RIC is charge of coordinating selected applications to make sure that each application operate in an orderly manner that meets the requirements of the operators.

B. A1 INTERFACE

The A1 interface makes the direct connection between the non-RT RIC and near-RT RIC. As mentioned earlier, the SMO layer is charged with high-level orchestration or optimization and AI/ML workflow. Therefore, the main function of A1 interface [32] is to provide policy-based guidance, ML model management and enrichment information from the non-RT RIC to the near-RT RIC so that the RAN components can get the proper configurations. The A1 interface is applied on the A1 Application Protocol (A1AP) which is defined by the 3GPP service framework for network functions [33]. The A1AP contains APIs over Hypertext Transfer Protocol (HTTP) for the A1 interface services [5].In each service, the A1AP contains service provider and service consumer. Both of them have a HTTP client and HTTP server, which are used to support the services [33].

1) A1 POLICY MANAGEMENT SERVICE

The non-RT RIC defines or modifies policies that are transmitted to the near-RT RIC via the A1 interface based on the intents from RAN components and the measurement data from O1 interface. A1 polices are charged with ensuring the RAN performance meets the RAN intents. On top of this, the non-RT RIC modifies policies based on the A1 policy feedback and the result from O1 interface. Besides, the non-RT RIC can provide enrichment information to strength the policy in the near-RT RIC, which I will discuss in the next section.

2) A1 ENRICHMENT INFORMATION SERVICE

The A1 enrichment information service is in order to improve the performance of RAN tasks by providing information from external and internal sources of O-RAN. When the SMO collects information from those sources and then the non-RT RIC provide new configurations and policies to the near-RT RIC after processing. Note that the A1 interface is responsible for enrichment information from internal sources normally. But A1 interface could be used for discover external information which is authorised by the non-RT RIC.

3) A1 ML MODEL MANAGEMENT SERVICE

According to the O-RAN specification [34], the ML model is trained in the SMO layer with its internal functions but can be executed in the different places (i.e. non-RT and near-RT RIC). When the ML model is used by the near-RT RIC to improve the performance of RAN, the A1 interface is charged with providing training data to the SMO layer and also receiving the feedback or enrichment information from SMO layer after its processing.

VI. MACHINE LEARNING BACKGROUND

Before delving into the Open RAN energy efficiency using ML, we will have a brief overview of ML; for more detail about any of the mentioned ML approaches please check the references cited. ML is a collection of methods that allows computers to be able to learn based on huge data sets or the previous experience and then achieves optimisation [35]. ML is the key feature of the O-RAN to achieve automation and intelligent radio resources management. The non-RT or near-RT RIC enable extension of their functionalities based on using different ML approaches. ML is traditionally divided into three main categories Supervised Learning (SL), Unsupervised Learning (UL), Reinforcement Learning (RL) and other secondary categories like the Deep Learning (DL) which is typically chosen to address 4G/5G RAN or O-RAN issues. In the remainig of this section, we will discuss each of the ML categories.

A. SUPERVISED LEARNING

Supervised learning algorithms leverage labeled datasets, consisting of input features and corresponding outputs, to develop models capable of inferring unseen data points. These datasets, comprising distinctive features and training examples, are strategically partitioned into subsets for training and validation purposes. The primary objective of such a model is the accurate prediction of outputs, approximating as closely as possible the actual values.

SL plays a crucial role, particularly in intelligent network management and optimization, by facilitating the estimation, prediction, and classification of diverse variables. Fundamental to SL are regression and classification techniques, pivotal for understanding complex data relations [36]. Classification pertains to the categorization of new observations within predefined classes, utilizing a labeled training set to discern patterns indicative of various class distinctions. This methodology encompasses a spectrum of algorithms, including decision trees [37], support vector machines [38], and neural networks [39]. Conversely, regression analyzes the correlation between dependent and independent predictors, aspiring to forecast or elucidate variations in a continuous outcome variable. This statistical approach employs several techniques [40], such as linear and logistic regression, alongside random forests, asserting the predictive relationship by fitting the data within the model.

B. UNSUPERVISED LEARNING

Unsupervised learning encompasses computational methods that operate on datasets without labeled responses, typically discovering hidden patterns or intrinsic structures within the input data. It plays a crucial role in various academic and research domains, especially where the data lack explicit annotations. The algorithms in [41] involved in unsupervised learning achieve this by discerning and exploiting data distributions and properties. Techniques such as clustering and dimensionality reduction are paramount [42]. Clustering, including methods like k-means [43] and hierarchical clustering, involves grouping data points into subsets or clusters, such that items in the same cluster are more similar to each other than to those in different clusters. Dimensionality reduction, illustrated by methods like Principal Component Analysis (PCA) [44] and t-distributed Stochastic Neighbor Embedding (t-SNE), reduces the number of random variables under consideration, deriving a set of principal variables.

C. REINFORCEMENT LEARNING

Unlike the SL and UL, the RL needs to find the optimal solution or achieve a certain goal through autonomous interacting with the environment. The goal of the RL is to take action to maximize expected reward. The agent in RL acts the learner or decision maker to map observed states of the environment to corresponding actions. The function which maps the particular state to specific action is known as the policy function. It is updated through an trial-and-error process in which various actions are tried in each state. Each state of environment responds rewards to different actions with numerical value. Another important function is the value function which is used to determine if the reward is good in the current state. In contrast to the reward function, the value function reflects the total cumulative expected reward rather than an immediate return [45].

In RL, the agent always faces the problem of trade-off between exploration and exploitation. There are two options: the first is called greedy action which means the agent selects the action that receives the greatest reward among previous actions. In this case, the agent is exploiting the knowledge it knows. In another case, the agent try to explore new possibilities for better actions and higher rewards. RL identifies the problem as Markov Decision Process (MDP) [46] $\{\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma\}$, where the $\mathcal{S}$ is a collection of states of the environment $\mathcal{S} = \{ s_1, s_2, \ldots, s_m\}$. $\mathcal{A}$ is a collection of possible actions $\mathcal{A} = \{ a_1, a_2, \ldots, a_m\}$. Then the transition function, $\mathcal{T}(s' \mid s, a)$, identifies the probability to the state s' based on the current state and action. $\mathcal{R} = (s, a)$ is reward function which indicates the feedback performance of algorithm for the current state-action pair, and $0 \leq \gamma \leq 1$ is discount factor.

D. DEEP LEARNING

Deep learning, a specialized segment of machine learning, characterizes a class of advanced algorithms that intuitively mimic the mechanism of the human brain, facilitating the modeling of complex, hierarchical representations of data [47]. It is anchored in the use of artificial neural networks with multiple processing layers (hence the term "deep"), which autonomously extract features from raw input, transitioning from simple to increasingly abstract and complex concepts. -Deep learning excels with high-dimensional, unstructured data, offering state-of-the-art performance in tasks such as image and speech recognition, natural language processing, and audio analysis, among others. Architectures such as Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), and more recently, Transformer models, represent deep learning's application versatility, each with structural attributes catering to specific types of data input.

Deep Neural Network (DNN) is the foundation of deep learning and consists of an input layer, an output layer and multiple hidden layers. DNNs are sometimes called Multiple-layer Perceptron (MLP) [48]. Each layer has multiple units perform the nonlinear processing of the input data called neurons. Each neuron that is an input has a corresponding weight to a different neuron in the next layer. After the weighted summation of the inputs, the bias value is added. The activation function, e.g., a sigmode function is then used to finally process the data and produce the output.

E. FEDERATED LEARNING

Federated learning is one of state-of-the-art distributed ML approach which allows multiple devices or servers to collaboratively train a shared model without sharing raw data. Each individual device only updates local model with central server after training with its own data. The central server aggregates these updates to improve the global model, which is then distributed back to the devices for the further training. The whole process does not end until the model converges. Federated learning helps to address the problem of data security and privacy due the raw data remains on local devices. However, with a training paradigm based on federated learning, the cost and capacity of communication [49] and energy consumption is a serious issue as the number of user devices, network slices increases.

In O-RAN architecture [34], the Non-RT RIC can act as a central server and distributes the AI/ML model to the Near-RT RICs. Both Non-RT and Near-RT RIC can upload and download updated model through A1/O1 interface. In [50], the authors use federated learning to train two different RL model in RAN slicing, which has the better network performance than typical RL.

VII. ENERGY CONSUMPTION OF MACHINE LEARNING

A pivotal innovation within the Open RAN architecture lies in its capacity to configure and enhance network performance through the deployment of two RICs. Furthermore, a central tenet of Open RAN's evolution is the intelligent orchestration of the network, achieved by training and implementing ML models within the RICs. However, it is critical to acknowledge the energy implications of these ML models, a factor often sidelined in current research to prioritize the models' accuracy. This oversight underscores the need for a nuanced understanding of the energy expenditure these models incur. Consequently, this part is dedicated to introducing diverse methodologies for estimating the energy consumption attributable to ML models, thereby proposing a more holistic view of their operational impact.

Basically, ML models such as deep neural networks are consisted of two phase of computation, training phase and inference phase respectively. The pursuit of deeper and more precise models necessitates a substantial training dataset. Presently, the training of these models is predominantly conducted on desktops or servers, implicating specific components in energy consumption. Notably, the CPU, GPU, and DRAM emerge as the principal consumers of energy in the training process [51]. The energy consumption of each component within the system comprises both static and dynamic components. Static energy consumption refers to the energy utilized by the circuit when it is idle and not actively processing information. In contrast, dynamic energy consumption pertains to the energy expended during the operation of the circuit's capacitors, which occurs when the circuit is actively engaged in processing tasks, as formulated below [52]:

$$P_d = \alpha \cdot C \cdot V_{dd}^2 \cdot f. \tag{1}$$

In (1), α represents the activity factor, which signifies the circuit's load, namely, the proportion of the circuit that is being utilized. C denotes the capacitance, V_{dd} the supply voltage, and f the clock frequency. Consequently, accurately determining the activity factor of these components while an application is executing is essential for calculating the dynamic energy consumption. A prevalent approach for determining the activity coefficients involves the utilization of performance counters (PMCs), which are hardware-based counters that are present in most modern microprocessors to measure the performance of various aspects of the processor's operation. These counters provide a low-overhead, high-resolution method for collecting measurements about the processor's behavior and the system's overall performance. Subsequently, the aggregate energy consumption is computed by deriving the power weights corresponding to each PMC through methodologies such as linear regression, as illustrated below [53]:

$$P_{tot} = (\sum_{i=1}^{n_{component}} AR_i \cdot \omega_i) + P_{static}, \tag{2}$$

where AR_i is the activity ratio of component i, ω_i is the power weight corresponded to component i, and P_{static} indicates static energy consumption of all components.

Initial approaches to calculating the energy consumption of neural network models focused on tallying the Multiply-Accumulate operations, representing the count of floating-point computations on CPUs or GPUs. This method also involved enumerating the weights to simulate the primary memory access events for a pre-configured model, serving as energy proxies [54]. Considering the significant energy demands associated with loading weights from DRAM, especially relative to Multiply-Accumulate

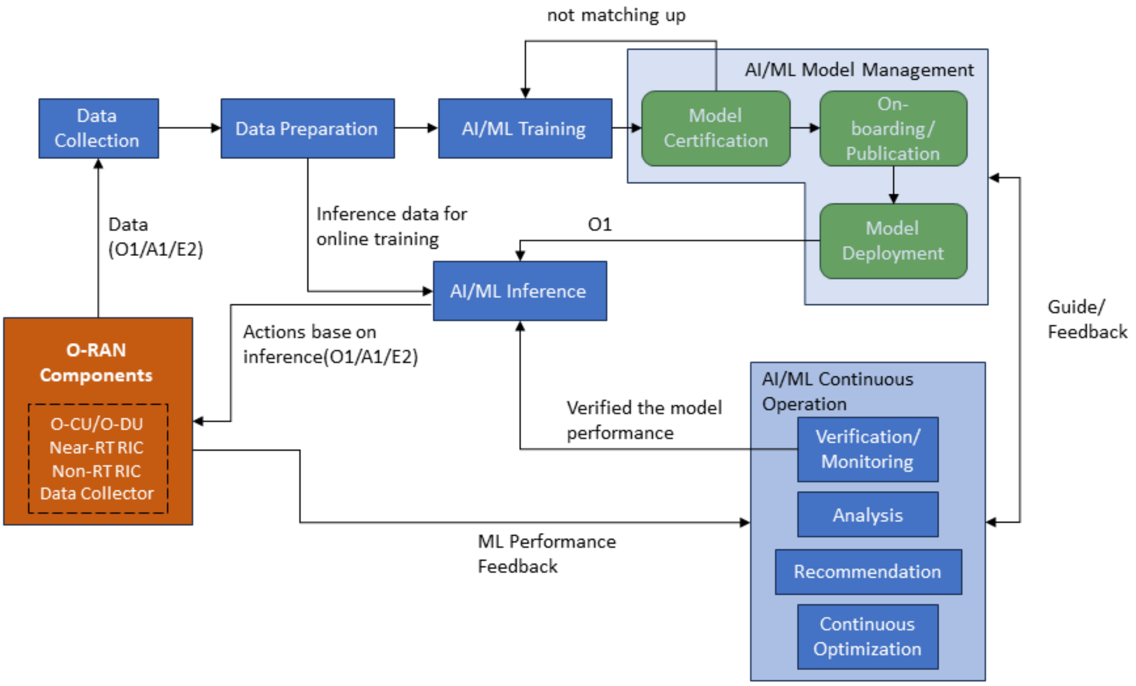


FIGURE 4. O-RAN AI/ML general procedure.

operations. Numerous optimization strategies, including pruning [55], have emerged.

However, the study in [56] posits that for DNNs, the predominant factor in energy consumption is attributable to data movement, superseding the actual computations involved. This is particularly evident in fully connected layers, where the absence of convolutional reuse leads to energy expenditure primarily for weight handling. Interestingly, the paper underscores that deeper networks with fewer weights do not inherently guarantee lower energy consumption compared to their shallower, more heavily-weighted counterparts. Introducing an innovative strategy known as Energy-Aware Pruning (EAP), the methodology leverages estimated energy metrics to navigate a systematic, layer-specific pruning process. This technique involves a hierarchical assessment of layers based on their energy profiles, prioritizing the pruning of higher-consuming layers. Unique to EAP is its nuanced approach to weight pruning, which assesses the collective impact of weights on feature maps, acknowledging their interdependencies. This holistic view facilitates a more aggressive pruning strategy while preserving the integrity of the model's accuracy, presenting a marked improvement over conventional magnitude-based methods. Furthermore, the method introduced accommodates the nuanced influences of bitwidth and sparsity on the energy dynamics. It recognizes a linear relationship between computation energy and input bitwidth, and a quadratic relationship when considering the bitwidths of interactive components. Additionally, the approach capitalizes on sparsity within feature maps and filters, bypassing certain multiplications contingent upon zero input, thereby enhancing energy efficiency.

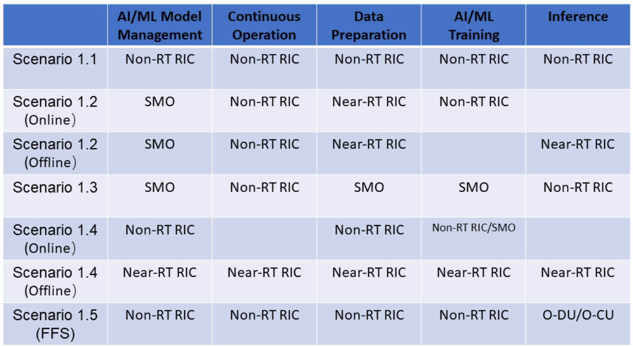
The energy measurement model proposed in [57] is called SyNERGY. It integrates the Caffe deep learning framework and vendor-specific tools such as ARM Streamline Performance Analyzer to measure and predict the energy consumption of CNN on the Nvidia Jetson TX1 embedded platform. The model focuses on fine-grained energy measurement, specifically at the per-layer level of the neural network. It utilizes the onboard power monitoring sensor (TI-INA3221x) available on the Jetson TX1 to measure power consumption during single image inferences on the CPU using an optimized OpenBLAS library. The authors provide a breakdown of the energy consumed in each layer of the CNN models. They use Streamline's annotation library to mark the beginning and end of each layer's execution and collect power samples using a power sampler script. The energy consumption is calculated by summing the rectangular areas of the power samples over the duration of the inference. To predict the energy consumption, the authors develop a regression model based on hardware performance counters such as Single Instruction, Multiple Data (SIMD) instruction executed and bus accesses. They train the model using a set of CNN models and their corresponding energy measurements. The regression model captures the relationship between the performance counters and energy consumption at the per-layer level.

The authors of [58] introduced NeuralPower, a predictive framework that employs sparse polynomial regression to estimate the energy consumption of CNNs across various GPU platforms, by layer by layer means. This method facilitates precise predictions and detailed analyses of power and execution time for all CNN layers, encompassing convolutional, pooling, and fully-connected layers. The research utilizes power metrics derived from the nvidia-smi tool [59], gathered from an NVidia GTX1070 GPU. Rather than adopting hardware-level specifics, the model incorporates application-level characteristics, such as kernel size and layer count, as determinants in the power estimation. Notably, to maintain controlled conditions, the influence of voltage and frequency variations was neutralized by sustaining a consistent GPU state. Furthermore, the study pioneers the ''energy-precision ratio'' as a novel metric, guiding practitioners towards more energy-conservative CNN configurations. Comparative analyses reveal that NeuralPower significantly surpasses its contemporaries in predictive accuracy, a testament to its efficacy underscored through rigorous testing on multiple GPU platforms and deep learning applications.

VIII. APPLICATIONS OF ML IN OPEN RAN

The integration of AI/ML into Open RAN architecture aims to transform RAN into more intelligent, efficient and adaptable system and even some of the theoretical ideas can be implemented in reality. By automating network resource allocation, optimizing traffic steering, enhancing security, etc. AI/ML indeed facilitate management of increasingly complex wireless networks. This section is going to provide an overview of AI/ML procedure and AI/ML deployment scenarios, which are being standardized by O-RAN alliance in [34].

Figure 4 illustrates the AI/ML general workflow of O-RAN architecture. First, O-RAN infrastructure provides data through O-RAN interfaces (O1, A1 and E2) to the data collection block. This could be the network usage, throughput, latency, channel quality information and other relevant parameters, since different AI/ML solutions might require different data collection. The data is then structured and transformed as needed for specific used AI/ML models in the data preparation block. All AI/ML solutions that have collected data require to be trained before the deployment [34] while ensuring the accuracy and reliability to avoid outages or inefficiencies in the network. Then trained models are going through the validation phase in AI/ML model management section to determine if the models perform as required when executing different network tasks. After that, the validated models are published in the SMO/Non-RT RIC catalogue. If the validation fails, they have to be re-trained or re-designed until pass the validation. The trained model in the AI/ML marketplace can be selected and deployed via containerized image to the execution node which refers to the inference host. Based on the output of the deployed ML model, actor in AI/ML assisted solution is informed to perform some control tasks including policy delivering (over A1 and E2 interfaces), configuration management (over O1 interface) and control action (over E2 interface). The last segment is AI/ML continuous operation, which has a crucial role in the overall workflow. It has the ability to detect and analyse intelligent models deployed throughout the network. In the event that models are not up to expectation in terms of reliability, accuracy or efficiency, it has the power to make those poorly performing models change their parameters and retrain them until they regain their original functionalities. In the 5G networks, it is impossible to handle all the problems in one solution. Therefore, O-RAN alliance has defined five different deployment scenarios which are shown in Figure 5, to support wide range of user cases and applications by placing the AI/ML workflow components in different places. While most practical cases can be covered by these deployment options, in practice they still need to be tailored to the specific requirements of the operator [5].



	AI/ML Model Management	Continuous Operation	Data Preparation	AI/ML Training	Inference
Scenario 1.1	Non-RT RIC	Non-RT RIC	Non-RT RIC	Non-RT RIC	Non-RT RIC
Scenario 1.2 (Online)	SMO	Non-RT RIC	Near-RT RIC	Non-RT RIC	
Scenario 1.2 (Offline)	SMO	Non-RT RIC	Near-RT RIC		Near-RT RIC
Scenario 1.3	SMO	Non-RT RIC	SMO	SMO	Non-RT RIC
Scenario 1.4 (Online)	Non-RT RIC		Non-RT RIC	Non-RT RIC/SMO	
Scenario 1.4 (Offline)	Near-RT RIC	Near-RT RIC	Near-RT RIC	Near-RT RIC	Near-RT RIC
Scenario 1.5 (FFS)	Non-RT RIC	Non-RT RIC	Non-RT RIC	Non-RT RIC	O-DU/O-CU

FIGURE 5. O-RAN AI/ML deployment scenarios.

The subsequent section elucidates the integration of ML with O-RAN for the implementation and optimization of network functions, thereby enhancing network performance.

A. NETWORK FUNCTION RELOCATION AND DU-CU PLACEMENT

The placement of DU and CU, along with the relocation of Network Function (NF), constitutes a critical aspect of optimizing the performance of specific applications within the O-RAN architecture. This optimization is necessitated by the fact that different layers of the O-RAN architecture are designed to manage distinct operational objectives. For instance, CU, typically housed in data centers, possess superior computing capabilities compared to DU. This distinction renders CU more adept at processing complex network functions, while DUs are more suited to executing algorithms of relative simplicity. Consequently, the strategic allocation of network functions to their appropriate computational units, ensuring that each function is executed within the optimal environment, enhances the overall efficiency and performance of the network. This paper [60] considers the CU-DU placement and user association as a multi-objective optimization problem. a novel DQN-based algorithm is proposed to reduce the cost of the O-RAN deployment and the end-to-end latency of UE by placing the DU-CU network functions in the regional and edge O-Cloud nodes, while jointly link users to RU. The aim of this paper is to enable RU to find the optimal O-Cloud node to run the network functions of DU and CU and the UE can also be assigned the most appropriate RU. In [61], authors add traffic type, delay budget and other requirements in to consideration. In this paper, two actor-critic learning based algorithm are proposed to decide resource allocation intelligently and relocate the resource allocation function dynamically at proper place (CU or DU). The aim of this paper is not only allocating the limited resources to the users but further improve the performance of NF by the proper CU-DU selection. Based on [61] and [62], the authors propose an upgraded version in [63]. In addition to the previous requirements, the agent in this paper needs to consider the propagation delay and energy consumption at the same time before choosing the proper location at DUs.

B. RESOURCE ALLOCATION AND O-RAN SLICING

O-RAN slicing is a significant application that offers more flexibility and precise network services, playing a crucial role in end-to-end network slicing. However, resource management, which involves allocating limited available resources to users while considering their QoS requirements and dynamic network situations, poses a challenging issue in RAN slicing [64]. In [22], an approach that combines a data-driven DRL-based algorithm deployed in as an xApp in the Near-RT RIC with the Colosseum open cellular stack. This integration enables the selection of the best-performing scheduling policy for each RAN slice, considering variations in the number of resource blocks allocated to each slice at different times, thereby ensuring enhanced scheduling accuracy. To address the allocation of communication and computational network resources for URLLC end devices, [65] proposes a two-level O-RAN slicing system. They treat the communication and computation resources as resource blocks from BSs and the CPU usage of MEC servers. In [50], two different xApps are designed for power and resource allocation based on RL, along with a federated learning approach to coordinate the two xApps agents. This enables achieving higher maximum throughput for eMMB slices and lower latency for URLLC slices. Reference [66] propose a policy-gradient based RL to address the allocation of computing resources. The objective is to maximize equitable allocation and optimize the QoS of the users, leading to superior performance compared to conventional methods. Similarly, [67] develops an edge network simulator based on the Q-Learning RL to demonstrate the performance improvement achieved by efficiently allocating resources. Overall, these studies contribute to the advancement of resource management in O-RAN slicing, utilizing RL algorithms and data-driven approaches to ensure more efficient and effective allocation of network resources.

C. TRAFFIC STEERING

In the intricate and multi-faceted ecosystem of an O-RAN, the Traffic Steering (TS) acts as an intelligent manager, responsible for optimizing complex data flows across diverse network elements while balancing the distribution of network resources to prevent overloading. Its primary objective is to enhance end-user experience by facilitating seamless and efficient data transmission. ML-based TS deployed at the near-RT RIC proves more suitable for adapting to evolving network conditions compared to traditional solutions, thanks to its centralized abstraction of the network that learns potential associations between different RAN parameters. In [68], a two-tiered ML algorithm combining Navie Bayes Classifier (NBC) and deep Q-learning is presented for traffic congestion prediction. This method employs SL to predict congestion in each Network Function Virtualization (VFN) and feeds the output data to a Deep Reinforcement Learning (DRL) agent, which dynamically steers traffic to avoid congestion, reduce queuing time, and ensure reliability. Similarly, in [69], an Long Short-Term Memory (LSTM) method and an SCA-based iterative algorithm are introduced for long and short sub-problem traffic prediction, respectively. Handover management, known as Radio Access Technology (RAT) allocation, is another significant use case of O-RAN, particularly crucial as the 5G system supports multiple access technologies, each with different access types. In [70], a Federated Meta-Learning (FML) algorithm is proposed for RAT allocation, significantly improving RL agents' training efficiency and adaptability to dynamic environments. This algorithm achieves a higher caching rate compared to a single RL agent while meeting UE demands. In comparison, [71] propose a comprehensive O-RAN-compliant framework for handover management and dual connectivity in the 3GPP network. To optimize the Traffic Scheduler, Conservative Q-Learning (CQL) and Random Ensemble Mixture (REM) techniques are employed, maximizing throughput and connecting O-RAN to the ns-3 simulation environment to obtain a vast training dataset, thus improving model accuracy.

IX. ENERGY EFFICIENCY OF PREVIOUS RAN TECHNOLOGIES

Research into the energy efficiency of Open RAN is currently in a nascent stage, with only a limited number of studies specifically focusing on this area. In contrast, there is an extensive body of work dedicated to reducing power consumption in legacy RAN architectures. Therefore, this section is going to introduce how the energy efficiency in previous generation and previous RAN.

A. ENERGY EFFICIENCY IN PREVIOUS GENERATIONS

In the era of 3G, energy efficiency wasn't a primary design consideration [72]. 3G networks used Code Division Multiple Access (CDMA) technology, which was more energy-efficient than the Time Division Multiple Access (TDMA) used in 2G networks but it still had significant limitations. Although in [73], the first joint energy-aware Radio Resources Management (RRM) and BS sleeping mechanism is proposed to reduce the energy consumption while ensuring the QoS of subscriber, the power usage in 3G networks was still relatively high due to factors

TABLE 1. O-RAN based ML applications.

Reference	Application	AI/ML approach	Model Deployment (Non/Near-RT RIC)	Advantages
[60]	CU-DU Placement User Assoication	Deep Q-Network(DQN)	Non-RT RIC	Reduce deployment costs and end-to-end latency of UE
[61]	CU-DU Selection Resource Allocation	Actor-Critic(A2C)	Near-RT RIC	Improvement of NF performane
[63]	CU-DU Selection Energy Efficiency	Actor-Critic(A2C)	Near-RT RIC	Improvement of energy efficiency and reducing mean latency
[22]	RAN Slicing Resource Allocation	Deep Q-Network(DQN)	Near-RT RIC	Improved spectrum utilisation smaller buffer size
[65]	RAN slicing Communication & Computational Resources Allocation	Double Deep Q-Network (DDQN)	Near-RT RIC	Improvement of efficiency for meeting QoS requirements
[50]	RAN Slicing Power & Resource Allocation	Federated DRL	Near-RT RIC	Higher throughput and lower latency
[66]	Computing Resources Allocation	Policy-gradient based RL	Near-RT RIC	Improvement of network performance
[67]	Network Resources Allocation	Q-Learning	Near-RT RIC	Improvement of network performance
[68]	Traffic Perdiction	Navie Bayes Classifier (NBC) Deep Q-Learning	Near-RT RIC	Reducing the queuing time and traffic congestion
[69]	Traffic Perdiction	Long Short-term Memory(LSTM) Successive Convex Approximation(SCA)	Non-RT RIC	Highly accurate traffic prediction
[70]	Handover Mangement Dual Connectivity	Federated Meta-learing (FML)	Near-RT RIC	Achieving higher cashing rates
[71]	Handover Mangement Dual Connectivity	Conservative Q-Learning(CQL) Random Ensemble Mixture(REM)	Near-RT RIC	improvement of throughput and spectral efficiency

such as inefficient hardware and a lack of sophisticated energy-saving techniques.

In 4G networks, energy efficiency has emerged as a significant consideration. The implementation of advanced techniques such as Orthogonal Frequency-Division Multiple Access (OFDMA), Multiple Input Multiple Output (MIMO) antennas or multi-user MIMO, and improved modulation schemes has contributed to enhanced spectral efficiency. Consequently, the energy efficiency per transmitted bit has improved. Reference [74] provides an extensive survey on energy-saving aspects in 4G wireless networks. The survey covers various aspects, including energy-saving models and energy efficiency metrics, the correlation between energy saving and Qos of user, and the application of energy-saving techniques in different scenarios.

B. ENERGY EFFICIENCY IN DISTRIBUTED/ CLOUD/ VIRTUAL-RAN

The RAN architecture in 5G and 6G networks have witnessed notable advancements in energy conservation with two notable approaches are the centralization and virtualization of network functions through structures like C-RAN and v-RAN (mentioned in Section II). These architectures facilitate efficient energy utilization by consolidating and optimizing network operations. Additionally, AI/ML are deployed for intelligent network management and operation leading to improved energy efficiency by dynamically adjusting resources based on network conditions and user demands. In this section, AI/ML based energy efficiency algorithm deployed on the RAN will be introduced.

1) BS SWITCHING OFF/ON MECHANISMS

The energy consumption of BSs constitutes a significant portion of the total energy consumed by cellular networks. In addition, an adequate number of BSs are deployed to meet the peak traffic demand in a given area [75], [76]. However, during periods of low traffic demand, the continuous operation of numerous BSs results in wasted energy. To address this issue, dynamically switching BSs on and off based on traffic demand has become a widely adopted energy-saving strategy. However, this approach presents an immediate challenge in terms of the potential degradation of QoS for subscribers. In this section, we will explore several ML-based methods that aim to mitigate network energy consumption while simultaneously ensuring the QoS for users. In [77], the authors optimise the overall energy consumption of the network by thinking simultaneously about the conflicting issues of BS sleep control and reducing transmission energy consumption. A deep learning based model is proposed to optimise the activation mode and beamforming weights of the BSs by learning the transmission channel coefficients between the RRH and the user, which reduces the computational complexity and optimises the network energy consumption compared to the traditional numerical approach. In [78], authors address the problem of joint optimization in v-RAN, focusing on BS sleeping and functional split orchestration to reduce energy consumption. Additionally, they take into consideration routing and coverage costs as important factors in the optimization process. When a BS is planned to enter a sleep mode, the traffic within its coverage area needs to be efficiently reassigned to other BS areas to ensure minimal energy consumption while maintaining satisfactory network performance. Another limitation pertaining to the dormancy time of the BS is associated with the period of the synchronisation signal burst, which exhibits variations ranging from 5 to 160 ms, depending on specific requirements. To address this challenge, the authors of [79] propose a classification of the BS's sleep modes into three distinct categories based on the varying signaling burst periods. Subsequently, they employ Distributed Q-learning techniques to enable the BS to dynamically select the optimal sleep mode, taking into account diverse network requirements. The objective is to strike a balance between achieving energy savings and managing the potential increase in system latency resulting from the implementation of energy-saving measures.

2) ML-BASED DATA-DRIVEN ENERGY EFFICIENCY OPTIMIZATION

The previous section highlighted the multifaceted nature of 5G wireless networks, characterized by increased network complexity and stochastic behavior. These networks face challenges such as fluctuating data demands, user mobility, and environmental variables, necessitating adaptive solutions capable of accommodating the inherent dynamism of 5G. However, traditional optimization algorithms often require recomputation when the network environment changes, resulting in high computing overhead [75]. In contrast, ML algorithms offer a promising approach to managing these complexities by leveraging data-driven decision-making and predictive analysis. ML algorithms enable energy optimization under varying conditions while minimizing computing and signaling overhead. Their flexibility allows for efficient adaptation to dynamic network environments. Reference [80] introduces a novel approach for energy-efficient resource allocation in C-RAN using a DDQN. While ML-based intelligent resource allocation algorithms have been previously discussed, the objective of this paper is to maximize energy-efficient rewards while considering constraints on transmission power selection and user rates. By employing DDQN, the paper addresses the overestimation problem encountered in traditional DQN algorithms and achieves superior performance. The proposed method leverages DDQN to optimize the allocation of network resources in C-RAN, thereby improving energy efficiency and overall network performance. In addition to determining user data rates for resource allocation, [81] this paper introduces a more advanced approach that leverages stacked and bidirectional LSTM as a deep learning method, along with A3C as a DRL method. By utilizing these techniques, resource allocation is conducted in a more refined manner, considering both large and small time intervals. The proposed approach significantly enhances energy efficiency, improves the accuracy of resource allocation, and increases the utilization of network slices. By combining deep learning and DRL, [81] contributes to the optimization of resource allocation in terms of energy consumption, precision, and overall network slice utilization.

X. ENERGY EFFICIENCY IN OPEN RAN

Within the context of Open RAN, transformative developments like open interfaces and disaggregation, as discussed in Section III, have contributed to the overall enhancement of energy efficiency across the network. Through these innovations, RICs are empowered to monitor and access information from various nodes within the network via open interfaces. Consequently, decisions pertaining to energy efficiency can be made from a global perspective, transcending local optimization efforts. To achieve this, AI/ML models are deployed within the RICs, leveraging predictive capabilities to anticipate future traffic demand, user mobility patterns, resource utilization, and other relevant factors. By harnessing these AI/ML models, the RICs can make informed optimization decisions, thereby fostering improved energy efficiency within the Open RAN ecosystem. This approach enables a more holistic and proactive approach to energy optimization by taking into account various network dynamics and future trends.This paper [82], authored by prominent European telecoms operators and targeted towards Open RAN vendors, addresses the crucial aspect of energy efficiency in Open RAN systems. The article focuses on four key categories for energy optimization. The first category emphasizes the need for energy-efficient hardware, particularly power amplifiers that can be selectively activated or deactivated based on the traffic volume. The second category highlights the importance of open interfaces within the Open RAN framework, enabling the measurement and monitoring of KPIs related to energy consumption, throughput, traffic load, and more. Furthermore, network hardware components should have the capability to enter sleep mode during periods of low load, adhering to the principle of zero consumption when there is no load. Lastly, intelligent control and optimization of energy efficiency are crucial, achievable through AI/ML models. For instance, this can involve cell switching on or off to directly save energy, or indirectly through efficient resource allocation and traffic steering. The O-RAN Alliance, WG1, presented three different approaches to improve the energy efficiency of O-RAN. These are explained in the following subsections.

A. CARRIER AND O-RU SWITCH OFF/ON

Open Radio Access Network (O-RAN), through its RIC and open interfaces, enables energy consumption assessment across the entire network rather than merely facilitating localized optimization. Typically, multiple carriers are used to cover the same geographical region. Consequently, in scenarios of low traffic/demands, O-RAN is capable of deactivating one or multiple carriers, or even shutting down an entire O-RU, to achieve energy savings. The maximum power saving is obtained by entirely switching off the O-RU, but this comes with drawbacks, such as introducing difficulties for other network units to discover the O-RU. To address this issue, instead of shutting down an entire O-RU, some network functionalities remain operational to maintain discoverability.

Based on the energy efficiency improvements outlined in [83] for switching off/on carriers or cells, within a setup featuring 4-transmit-4-receive (4T4R) antennas, four layers, a bandwidth of 100 MHz, operating at 3.5GHz, and with each antenna transmitting at 30W, shutting down O-RUs during periods of low demand can lead to a reduction in energy consumption by 150-180W for each O-RU. With an expanded deployment of 10,000 O-RUs, and assuming each O-RU is deactivated for merely 3 hours daily, the total annual energy savings could range between 1643 to 1971 MWh. Furthermore, escalating the antenna configuration to 64-transmit-64-receive (64T64R) antennas enhances the potential energy savings per O-RU to 260-340W, culminating in an annual reduction of 2847 to 3723 MWh in energy consumption.

B. RF CHANNEL RECONFIGURATION OFF/ON

To boost the throughput and capacity of O-RUs, beamforming stands as a prevalent strategy in mobile networks with massive MIMO (mMIMO) antennas. Energy efficiency can be optimized during low-traffic periods (such as nighttime) by turning off specific segments of the Tx/Rx arrays. Subsequent to the partial deactivation of O-RU Tx/Rx arrays, adjustments must be made to the configurations related to these arrays, including modifications to the transmit power of the O-RU antennas and the quantity of Synchronization Signal Block (SSB) beams, among other parameters.

In O-RAN, through RF channel reconfiguration, reducing the antenna configuration from 4T4R to 2T2R within the same O-RU setup as discussed in the previous section can achieve up to 80W of energy savings per O-RU. Similarly, decreasing the antenna configuration from 64T64R to 32T32R can result in a maximum energy saving of 230W per O-RU [82].

C. ADVANCED SLEEP MODE

Advanced Sleep Modes (ASMs) are a feature in cellular networks that involve the gradual deactivation of different components of a O-RU based on the time needed for each component to deactivate and reactivate, known as the transition time [84], [85]. ASMs allow for the definition of different levels with varying characteristics such as duration and power consumption. These modes are highly efficient in terms of energy conservation, capable of reducing energy consumption by up to 90% [86] at low loads. However, the trade-off is an increase in latency due to the waiting time for a user requesting a service while the O-RU is in sleep mode. To optimize this trade-off between energy conservation and delay. [87] proposes a framework dynamically adjusts ASMs by implementing the Delay Conservative Advanced Sleep Modes (DCASM) approach for BS operation. This approach involves deriving proper ASM parameter settings based on traffic predictions to meet specific delay constraints while optimizing energy savings. By using a closed-form expression, the framework ensures that ASMs parameters are tailored to the needs of Mobile Network Operators (MNOs) and vertical industries, allowing for estimation of power consumption based on traffic prediction and real-time adjustments. The DCASM approach guarantees the desired average reactivation delay, such as 1 ms, under any predetermined arrival rate, ensuring compliance with delay requirements.

In the O-RAN framework, it is imperative for O-RUs to share their ASM settings with O-DUs. This encompasses the durations of various sleep phases, transition intervals between these modes, and the requisite activation times. Subsequently, the O-DU communicates this data to both the RIC and SMO. Unlike in conventional networks operated by a single vendor, where internal data flows seamlessly enabling straightforward sleep mode management, O-RAN's disaggregated architecture implies a lack of direct awareness about O-RU specifics by the E2 node [82]. Therefore, the deployment of ASMs in O-RAN demands that details regarding the O-RU's sleep management be disseminated across the network's components via open interfaces. With this information at hand, the RIC proceeds to train models aimed at optimizing the trade-off between sleep mode durations and the latency associated with reactivation.

XI. POWER CONSUMPTION MODEL IN O-RAN

A. POWER CONSUMPTION MODELS OF RU

1) EARTH MODEL OF RU

The study in [88] defines power consumption model known as EARTH[1] model for a single BS that can be widely used. This is a typical BS consists of multiple transceivers, each of which serves one transmit antenna element. Therefore, power consumption of a BS can be regarded as the power consumption of all transceivers, each one including a Power Amplifier (PA), a RF module and a BBU, a DC-DC power supply, an active cooling system, and mains supply for connection to the electrical power grid. This BS power consumption is given by

$$P_{BS} = N_{TRX} * \frac{\frac{P_{out}}{\eta_{PA}(1-\sigma_{feed})} + P_{BB} + P_{RF}}{(1-\sigma_{co})(1-\sigma_{DC})(1-\sigma_{MS})} \tag{3}$$

where N_{TRX} denotes the total number of RF transceivers in a BS. P_{out} is the transmission power, scaling with traffic load, η_{PA} is the PA efficiency, σ_{feed} is the power losses by antenna feeder since the location of the macro BS is often different from its antenna, but this problem was addressed by having the RRH in the same site. P_{BB} and P_{RF} are the power consumption of BBU and RF module respectively. The power losses in active cooling system, DC-DC power supply and main power supply are denoting σ_{co}, σ_{DC} and σ_{MS}.

The simulation in [88] shown that only PA scales with BS load, other components hardly scale with the load. So the relation between BS power consumption P_{in} and RF output power P_{out} are nearly linear and the equation can be simplified into

$$P = \begin{cases} N_{TRX} * P_0 + \xi_P P_{out} & 0 < P_{out} < P_{max} \\ N_{TRX} * P_{sleep} & \mathrm{P}_{out} = 0 \end{cases} \tag{4}$$

where P_0 and ξ_P indicate the fixed power consumption and the slope of the load-dependent power consumption in different cell type, respectively. Reference [88] argues that putting BS into sleep mode when its has no service to offer is critical to energy efficiency. Therefore, the P_{sleep} $\left(P_{sleep} < P_{out}\right)$ indicates the power consumption when BS is entering sleep mode when there is no traffic to transmit.

In the context of O-RAN, RU is a physical node [6] and a major power consumption of RU is related to RF

[1] Energy Aware Radio and neTwork tecHnologies (EARTH) was an EU FP7 multi national project (2010-2012) which investigated the energy consumption of different network components.

functionalities and power amplification [8]. Therefore, the power consumption of RU can be formulated as follow base on the EARTH model:

$$P_{RU} = \begin{cases} P_{RF} + \dfrac{P_{out}}{\eta} & 0 < P_{out} < P_{max} \\ P_{sleep} & \mathrm{P}_{out} = 0 \end{cases} \tag{5}$$

where η is the power amplifier drain efficiency. P_{RF} is refer to fixed power consumption of RU such as RF circuits power consumption. P_{sleep} is the power consumption when RU is sleep.

2) CARRIER AGGREGATION POWER CONSUMPTION MODEL OF RU

Carrier Aggregation (CA), introduced in the 3GPP Long Term Evolution Advanced (LTE-A) standard, serves as a pivotal feature to enhance cell throughput [89]. Fundamentally, CA permits the aggregation of multiple Carrier Components (CCs), thereby expanding bandwidth and augmenting data rates for mobile devices. This mechanism optimizes the use of the available spectrum by amalgamating carriers from either identical or disparate frequency bands. In its early stages, CA supported the aggregation of a maximum of 5 CCs with a bandwidth of up to 20 MHz. However, with the advent of 5G NR, its capabilities have been broadened to accommodate up to 16 CCs with a bandwidth reaching 1 GHz. This versatile technology has been instrumental in various areas, encompassing capacity augmentation, coverage enhancement, and facilitation of advanced functionalities such as Licensed Assisted Access (LAA) and dual connectivity [75].

Given this context, when constructing a power consumption model for RU utilizing CA, it becomes imperative to formulate a function that captures the power consumption's correlation with the number of active CCs, denoted as N_{cc}. Integrating insights from both the O-RAN framework and the study by [90], energy model for the RU is illustrated as follow:

$$P_{RU} = \sum_{j=1}^{N_{cc}} (P_{TX_j} + B_j P^{CA}_{CP_j}) + P^{CAi}_{CP}, \tag{6}$$

where $P_{TX_j} = \frac{P_{out_j}}{\eta}$ denotes the effective transmit power used by CC j. B_j and $P^{CA}_{CP_j}$ are represented as the bandwidth of CC j and the variable circuit power consumption, which scales linearly with both the number of active CCs, and their bandwidth, B_j. However, P^{CAi}_{CP} is the static circuit power consumption of CA system.

B. POWER CONSUMPTION MODELS OF DU AND CU

Within the paper [91], two distinct power consumption models are delineated. The first centers on processing and is built upon the foundations of the EARTH model. This model seeks to establish a function representing the average CPU load, denoted as $\bar{l}$, for each Edge Processing Module (EPM) in a time slot t over a duration T. The modeling of this equation is presented as follows:

$$E^{p}_{epm,j}(t) = \left(\mathbb{I}_{(\bar{l}^t_{epm,j}>0)} P_{epm} + P'_{epm} \cdot \frac{\bar{l}^t_{epm,j}}{C_{epm}}\right)T, \tag{7}$$

where P_{epm} is the power consumption of EPM where DUs are implemented, which represents the fixed costs of running the server, such as cooling, power amplification, and network switches. P'_{epm} is dynamic or load-dependent power consumption increases linearly with the EPM's load(i.e., average CPU load). $\mathbb{I}_{(\bar{l}^t_{epm,j}>0)}$ is an indicator variable that takes the value 1 when EPM j is busy and 0 when it is idle.

Similarly, the energy consumption of a Central Processing Module (CPM) k where CUs are implemented is given as:

$$E^{p}_{cpm,k}(t) = \left(\mathbb{I}_{(\bar{l}^k_{cpm,j}>0)} P_{cpm} + P'_{cpm} \cdot \frac{\bar{l}^t_{cpm,k}}{C_{cpm}}\right)T, \tag{8}$$

where P_{cpm} and P'_{cpm} are static and dynamic power consumption respectively.

While the EARTH model offers insights into power consumption, it is not directly applicable for estimating the energy usage of the DU/CU. This limitation arises primarily for two reasons [92], [93]. Firstly, considering that multiple DU/CUs operate within a cloud infrastructure, the energy footprint of an individual DU/CU is expected to be reduced. Secondly, due to the dynamic resource allocation inherent in O-RAN systems, applications residing on the DU/CU aren't continuously active.

Consequently, in the context of the O-RAN framework where the DU/CU essentially constitutes a segment of the virtualized BS, the primary determinant of its power consumption is attributed to the utilization of CPU cores [93]. Thus, the power consumption model of a DU/CU for a given time slot t can be articulated as follows:

$$P^t_{DU/CU} = N_c(P_{DU/CU,min} + \Delta_{P_{DU/CU}} \delta_c s^{\beta}), \tag{9}$$

where $\Delta_{P_{DU/CU}}$ is denoted as follow:

$$\Delta_{P_{DU/CU}} = \left(P_{DU/CU,max} - P_{DU/CU,min}\right) / s_0^{\beta} \tag{10}$$

In this model, the power consumption of DU is linear with the number of active CPU cores and CPU load as well. Where, N_c represents the number of active CPU cores in DU, $P_{m,min}$ and $P_{m,max}$ are denoted as the minimum and maximum power consumption of each CPU core.δ_c is the percentage of CPU load on active cores, s is the CPU speed and s_0 is the reference CPU speed. β is the exponential coefficient of CPU speed. Δ_{P_m} indicates the slope of the load-dependent power consumption of DU.

The percentage of CPU load on active cores, denoted as δ_c, can be ascertained through the subsequent function: [93]:

$$\delta_c = \frac{Q(r)}{N_c s} = \frac{c_0 + kr}{N_c s} \tag{11}$$

where $Q(r)$ is the actual instructions per unit time and $N_c s$ represents the maximum instructions available per unit time. c_0 is constant coefficient of instruction speed. k is rate varying coefficient of instruction. r is transmission rate. Base on eq (9) and eq (11), the power consumption model can be modeled as:

$$P^t_{DU/CU} = N_c P_{DU/CU,min} + \Delta_{P_{DU/CU}} c_0 s^{\beta-1} + \Delta_{P_{DU/CU}} krs^{\beta-1} \quad (12)$$

In [94], the authors proposed an energy consumption model of activating servers and instantiating O-RAN applications, representing as follows:

$$E_s(x_s, y_s) = x_s * E_s^{base} + \sum_{a\in\mathcal{A}} \sum_{r\in\mathcal{R}} y_{r,a,s} e_{a,s}, \quad (13)$$

where x_s indicates the server activation profile and E_s^{base} represents the fixed energy consumption when the server s is active. $y_s = (y_{r,a,s})_{(r,a)\in\mathcal{R}\times\mathcal{A}}$ which is an allocation variable indicates the load of server s, that is how many instances of app a with request r have been deployed on that server. $e_{a,s}$ shows the energy consumption of an application a which scales linearly with the server load.

Energy efficiency studies based on O-RAN are still in their nascent stages. Currently, only a few articles have considered and analyzed the Energy efficiency of O-RAN. Current analyses of O-RAN energy consumption models primarily rely on models developed for C-RAN and v-RAN. Although these network structures share similarities with O-RAN, they do not fully meet all the requirements of O-RAN. Therefore, further research is needed to delve into energy efficiency optimization that aligns with O-RAN's unique characteristics, aiming to directly reduce the overall energy consumption of O-RAN. For example, developing a distinct power consumption model unique to O-RAN, which includes the energy used during transmission and operation by various hardware and software components, rather than merely adopting or slightly modifying the existing EARTH model.

TABLE 2. Simulation parameters [92].

Parameter	Value
Channel bandwidth,W	20MHz
RF circuit power,P_{RF}	12.9W
Maximum RU output power,P_{out}	20W
Power Amplifier efficiency (PA),η	31.1%
Noise figure by UE, F	9dB
Noise spectral density, N_0	-174dBm/Hz
Maximum power per CPU core, $P_{DU/CU,max}$	20W
Minimum power per CPU core, $P_{DU/CU,min}$	5W
CPU speed,s	2GHz
CPU reference speed,s_0	2GHz
Constant coefficient of instruction speed, C_0	$7 * 10^8$
Exponential Coefficient of CPU speed, β	2
Rate varying coefficient of instruction speed, K	35

XII. CASE STUDY

In this section, we illustrate the numerical results in Figure 6, highlighting the difference in power consumption between a traditional integrated BS such as eNodeB and an O-RAN configuration under various transmission power scenarios. The power consumption of O-RAN is estimated using (5) and (7). (5), derived from the EARTH's model, calculates the energy utilization of RUs, while (7) assesses the energy demands of DUs and CUs based on CPU core usage. For simplicity, we assume they have equal CPU utilisation, hence equal power consumption. The specific parameters applied in the O-RAN energy simulation are detailed in TABLE 2. For the conventional BS, we employ the EARTH model as illustrated at (4), selecting a macro BS configuration that incorporates RRHs located at the BS sites to eliminate feeder losses. The relevant parameters are provided in TABLE 3. However, it is important to note that we have opted for a single RF transceiver in a BS for this model, with $N_{TRX} = 1$.

TABLE 3. Power model parameters for different BS types [88].

BS Type	N_{TRX}	$P_{max}(W)$	$P_0(W)$	ξ_p	$P_{sleep}(W)$
Marco	6	20	130.0	4.7	75
RRH	6	20	84	2.8	56
Micro	2	6.3	56.0	2.6	39
Pico	2	0.13	6.8	4.0	4.3
Femto	2	0.05	4.8	8.0	2.9

Figure 6 distinctly illustrates that the initial energy consumption of a conventional BS significantly exceeds that of the O-RAN. Both of them have same radio head and same RF circuit power consumption. The disparity is primarily due to the BBU in the conventional setup, which consume approximately 29.6 watts [88] in their active state, even during periods of idle time. Contrastingly, O-RAN's energy expenditure in idle states is significantly reduced due to its architectural design that inherently incorporates virtualization of network functions. This design not only allows for more efficient energy use when network components are inactive but also provides a more dynamic and responsive energy utilisation model that adapts to varying network demands [11]. Another observation is the curvilinear nature of O-RAN's energy consumption. This characteristic shape arises because O-RAN's energy usage is bifurcated into two distinct segments (RUs and DUs,CUs). The segment to the energy utilized by the DUs and CUs, correlating with the CPU core load relative to the transmission rate, as detailed in (12). Given that the transmission rate is computed using a logarithmic function (log_2), the relationship between the transmission rate and energy consumption doesn't follow a linear trajectory but exhibits a curved pattern instead. The crossover in energy consumption between O-RAN and conventional BSs occurs at a transmission power threshold of roughly 14 W. Beyond this point, O-RAN begins to consume more energy. This phenomenon is attributed to the inherent energy consumption characteristics of the two systems. In conventional BSs, energy usage is a fixed value that scales with the number of RF transceivers, with the exception of power amplifiers (PAs) that vary based on transmission power. In contrast, O-RAN experiences a dual variability: not only does the energy consumption of the PAs fluctuate with transmission power, but the CPU load also scales in response to these power adjustments.

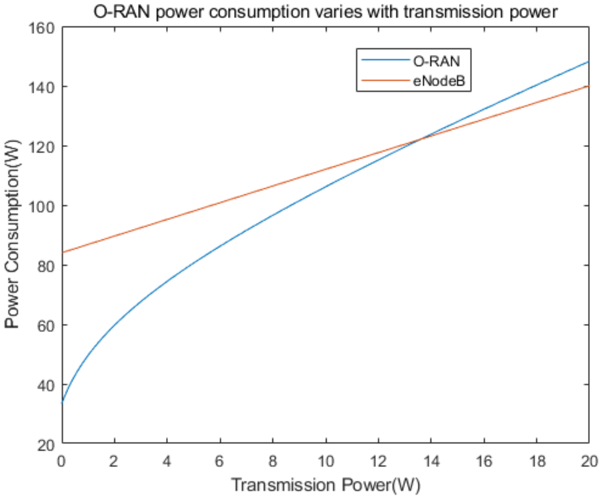


FIGURE 6. Comparison of power consumption of O-RAN and eNodeB with transmission power variation.

Consequently, as transmission power escalates, O-RAN's energy consumption trajectory steepens more drastically compared to that of its traditional counterparts, owing to its compounded sensitivity to changes in transmission power. In conclusion, while virtualization is a common factor in both O-RAN and non-ORAN systems, O-RAN's default incorporation of this technology, coupled with its open architecture, offers a more integrated and efficient approach to energy management. This inherent capability of O-RAN to dynamically adjust network functions and manage resources smartly positions it as a more energy-efficient solution in the evolving landscape of network technologies.'

XIII. CONCLUSION

This paper thoroughly investigated Open RAN architectures, tracing the development of RAN technologies from D-RAN to v-RAN. It highlighted the new features that Open RAN brings to the table. Detailed explanations were provided about the roles and connections of the Near-RT RIC and Non-RT RIC within the SMO framework, as well as the essential AI/ML processes they support. The investigation then advanced into an exploration of ML technologies, spotlighting their role in infusing intelligence and automation into O-RAN operations, and highlighted the energy implications of deploying ML models. It offered a detailed look at different ML applications and research related to O-RAN. Additionally, it reviewed how ML methods developed over time to improve energy efficiency in various network designs and how these methods fit into O-RAN's framework. A pivotal aspect of this study was the in-depth case study examining the energy consumption profiles of RU and DU/CU, employing the EARTH model alongside a CPU core power model respectively. This case study facilitated a comparative analysis with traditional BSs, elucidating the energy efficiency advantages inherent in O-RAN architectures. This paper not only offered a detailed exploration of Open RAN and its underpinning technologies but also underscored the critical role of ML in optimizing network operations and energy consumption.

REFERENCES

[1] A. Fonseca, R. Kazman, and P. Lago, "A manifesto for energy-aware software," *IEEE Softw.*, vol. 36, no. 6, pp. 79–82, Nov. 2019.

[2] P. Lähdekorpi, M. Hronec, P. Jolma, and J. Moilanen, "Energy efficiency of 5G mobile networks with base station sleep modes," in *Proc. IEEE Conf. Standards Commun. Netw. (CSCN)*, Sep. 2017, pp. 163–168.

[3] L. Kundu, X. Lin, and R. Gadiyar, "Towards energy efficient RAN: From industry standards to trending practice," 2024, *arXiv:2402.11993*.

[4] H. Ahmadi, A. Nag, Z. Khar, K. Sayrafian, and S. Rahardja, "Networked twins and twins of networks: An overview on the relationship between digital twins and 6G," *IEEE Commun. Standards Mag.*, vol. 5, no. 4, pp. 154–160, Dec. 2021.

[5] M. Polese, L. Bonati, S. D'Oro, S. Basagni, and T. Melodia, "Understanding O-RAN: Architecture, interfaces, algorithms, security, and research challenges," *IEEE Commun. Surveys Tuts.*, vol. 25, no. 2, pp. 1376–1411, 1st Quart., 2023.

[6] O-RAN Alliance WG1. (2022). *O-RAN Architecture Description*. Accessed: Jan. 2, 2023. [Online]. Available: https://orandownloadsweb.azurewebsites.net/specifications

[7] O-RAN Alliance WG2. (2022). *A1 Interface: General Aspects and Principles 3.0*. Accessed: Dec. 12, 2022. [Online]. Available: https://orandownloadsweb.azurewebsites.net/specifications

[8] A. Ibrahim Abubakar, O. Onireti, Y. Sambo, L. Zhang, G. K. Ragesh, and M. Ali Imran, "Energy efficiency of open radio access network: A survey," in *Proc. IEEE 97th Veh. Technol. Conf. (VTC-Spring)*, Jun. 2023, pp. 1–7.

[9] O-RAN Alliance WG3. (2021). *Near-Real-Time RAN Intelligent Controller Near-RT RIC Architecture*. Accessed: Dec. 15, 2022. [Online]. Available: https://orandownloadsweb.azurewebsites.net/specifications

[10] O-RAN Alliance WG2. (2022). *O-RAN Non-RT RIC Architecture 2.01*. Accessed: Dec. 5, 2022. [Online]. Available: https://orandownloadsweb.azurewebsites.net/specifications

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

[12] S. Niknam, A. Roy, H. S. Dhillon, S. Singh, R. Banerji, J. H. Reed, N. Saxena, and S. Yoon, "Intelligent O-RAN for beyond 5G and 6G wireless networks," in *Proc. IEEE Globecom Workshops*, 2022, pp. 215–220.

[13] D. Sesto-Castilla, E. Garcia-Villegas, G. Lyberopoulos, and E. Theodoropoulou, "Use of machine learning for energy efficiency in present and future mobile networks," in *Proc. IEEE Wireless Commun. Netw. Conf.*, Apr. 2019, pp. 1–6.

[14] D. Sabella, P. Rost, Y. Sheng, E. Pateromichelakis, U. Salim, P. Guitton-Ouhamou, M. Di Girolamo, and G. Giuliani, "RAN as a service: Challenges of designing a flexible RAN architecture in a cloud-based heterogeneous mobile network," *Future Network Mobile Summit*, pp. 1–8, 2013.

[15] K. Chen, C. Cui, Y. Huang, B. Huang, J. Wu, S. Rangan, and H. Zhang, "C-RAN: A green RAN framework," in *Green Communications: Theoretical Fundamentals, Algorithms and Applications*. CRC Press, 2013, pp. 279–304.

[16] I. A. Alimi, A. L. Teixeira, and P. P. Monteiro, "Toward an efficient C-RAN optical fronthaul for the future networks: A tutorial on technologies, requirements, challenges, and solutions," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 1, pp. 708–769, 1st Quart., 2018.

[17] M. Antonio Marotta, H. Ahmadi, J. Rochol, L. DaSilva, and C. Bonato Both, "Characterizing the relation between processing power and distance between BBU and RRH in a cloud RAN," *IEEE Wireless Commun. Lett.*, vol. 7, no. 3, pp. 472–475, Jun. 2018.

[18] J. van de Belt, H. Ahmadi, and L. E. Doyle, "Defining and surveying wireless link virtualization and wireless network virtualization," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1603–1627, 3rd Quart., 2017.

[19] L. Gavrilovska, V. Rakovic, and D. Denkovski, "From cloud RAN to open RAN," *Wireless Pers. Commun.*, vol. 113, no. 3, pp. 1523–1539, Aug. 2020.

[20] B. Brik, K. Boutiba, and A. Ksentini, "Deep learning for B5G open radio access network: Evolution, survey, case studies, and challenges," *IEEE Open J. Commun. Soc.*, vol. 3, pp. 228–250, 2022.

[21] A. S. Abdalla, P. S. Upadhyaya, V. K. Shah, and V. Marojevic, "Toward next generation open radio access networks: What O-RAN can and cannot do!" *IEEE Netw.*, vol. 36, no. 6, pp. 206–213, Nov. 2022.

[22] L. Bonati, S. D'Oro, M. Polese, S. Basagni, and T. Melodia, "Intelligence and learning in O-RAN for data-driven NextG cellular networks," *IEEE Commun. Mag.*, vol. 59, no. 10, pp. 21–27, Oct. 2021.

[23] O-RAN Alliance WG6. (2023). *Cloud Architecture and Deployment Scenarios for O-RAN Virtualized RAN*. Accessed: Oct. 12, 2023. [Online]. Available: https://orandownloadsweb.azurewebsites.net/specifications

[24] D. Sabella, A. de Domenico, E. Katranaras, M. A. Imran, M. D. Girolamo, U. Salim, M. Lalam, K. Samdanis, and A. Maeder, "Energy efficiency benefits of RAN-as-a-service concept for a cloud-based 5G mobile network infrastructure," *IEEE Access*, vol. 2, pp. 1586–1597, 2014.

[25] T. Pamuklu, M. Erol-Kantarci, and C. Ersoy, "Reinforcement learning based dynamic function splitting in disaggregated green open RANs," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2021, pp. 1–6.

[26] 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.

[27] O-RAN Alliance WG6. (2021). *O-RAN Cloud Architecture and Deployment Scenarios for O-RAN Virtualized RAN 5.0*. Accessed: Dec. 15, 2022. [Online]. Available: https://orandownloadsweb.azurewebsites.net/specifications

[28] O-RAN Alliance WG3. (2021). *Near-Real-Time RAN Intelligent Controller Architecture*. Accessed: Dec. 12, 2022. [Online]. Available: https://orandownloadsweb.azurewebsites.net/specifications

[29] O-RAN Alliance WG11. (2023). *O-RAN Security Requirements Specification 7.0*. Accessed: Aug. 15, 2023. [Online]. Available: https://orandownloadsweb.azurewebsites.net/specifications

[30] O-RAN Alliance WG3. (2022). *Near-Real-Time RAN Intelligent Controller Architecture E2 General Aspects and Principles*. Accessed: Jan. 13, 2023. [Online]. Available: https://orandownloadsweb.azurewebsites.net/specifications

[31] O-RAN Alliance WG2. (Jun. 2021). *Non-RT RIC: Functional Architecture*. Accessed: Jan. 13, 2023. [Online]. Available: https://orandownloadsweb.azurewebsites.net/specifications

[32] (2022). *A1 Interface: General Aspects and Principles*. Accessed: Jan. 13, 2023. [Online]. Available: https://orandownloadsweb.azurewebsites.net/specifications

[33] (2022). *A1 Interface: Application Protocol*. Accessed: Jan. 13, 2023. [Online]. Available: https://orandownloadsweb.azurewebsites.net/specifications

[34] (2021). *AI/ML Workflow Description and Requirements*. Accessed: Jan. 15, 2023. [Online]. Available: https://orandownloadsweb.azurewebsites.net/specifications

[35] H. Fourati, R. Maaloul, and L. Chaari, "A survey of 5G network systems: Challenges and machine learning approaches," *Int. J. Mach. Learn. Cybern.*, vol. 12, no. 2, pp. 385–431, Feb. 2021.

[36] E. Alpaydin, "Supervised learning," in *Introduction to Machine Learning*. Cham, Switzerland: Springer, 2014, pp. 21–47.

[37] V. Nasteski, "An overview of the supervised machine learning methods," *Horizons.B*, vol. 4, pp. 51–62, Dec. 2017.

[38] M. Somvanshi, P. Chavan, S. Tambade, and S. V. Shinde, "A review of machine learning techniques using decision tree and support vector machine," in *Proc. Int. Conf. Comput. Commun. Control Autom. (ICCUBEA)*, Aug. 2016, pp. 1–7.

[39] K. Makantasis, K. Karantzalos, A. Doulamis, and N. Doulamis, "Deep supervised learning for hyperspectral data classification through convolutional neural networks," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2015, pp. 4959–4962.

[40] R. Caruana and A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms," in *Proc. 23rd Int. Conf. Mach. Learn.*, 2006, pp. 161–168.

[41] H. B. Barlow, "Unsupervised learning," *Neural Comput.*, vol. 1, no. 3, pp. 295–311, Sep. 1989.

[42] M. E. Celebi and K. Aydin, *Unsupervised Learning Algorithms*, vol. 9. Cham, Switzerland: Springer, 2016.

[43] K. P. Sinaga and M.-S. Yang, "Unsupervised K-means clustering algorithm," *IEEE Access*, vol. 8, pp. 80716–80727, 2020.

[44] S. J. Wetzel, "Unsupervised learning of phase transitions: From principal component analysis to variational autoencoders," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 96, no. 2, Aug. 2017, Art. no. 022140.

[45] N. Morozs, "Accelerating reinforcement learning for dynamic spectrum access in cognitive wireless networks," Ph.D. dissertation, Univ. York, 2015.

[46] J. Moysen and L. Giupponi, "From 4G to 5G: Self-organized network management meets machine learning," *Comput. Commun.*, vol. 129, pp. 248–268, Sep. 2018.

[47] E. Akleman, "Deep learning," *Computer*, vol. 53, no. 9, pp. 1–17, Sep. 2020.

[48] V. Sze, Y.-H. Chen, T.-J. Yang, and J. S. Emer, "Efficient processing of deep neural networks: A tutorial and survey," *Proc. IEEE*, vol. 105, no. 12, pp. 2295–2329, Dec. 2017.

[49] H. Zhou, M. Erol-Kantarci, and H. V. Poor, "Knowledge transfer and reuse: A case study of AI-enabled resource management in RAN slicing," *IEEE Wireless Commun.*, vol. 30, no. 5, pp. 160–169, Dec. 2022.

[50] H. Zhang, H. Zhou, and M. Erol-Kantarci, "Federated deep reinforcement learning for resource allocation in O-RAN slicing," in *Proc. IEEE Global Commun. Conf.*, Dec. 2022, pp. 958–963.

[51] E. García-Martín, C. F. Rodrigues, G. Riley, and H. Grahn, "Estimation of energy consumption in machine learning," *J. Parallel Distrib. Comput.*, vol. 134, pp. 75–88, Dec. 2019.

[52] J. L. Hennessy and D. A. Patterson, *Computer Architecture: A Quantitative Approach*. Amsterdam, The Netherlands: Elsevier, 2011.

[53] R. Bertran, M. Gonzalez, X. Martorell, N. Navarro, and E. Ayguade, "Decomposable and responsive power models for multicore processors using performance counters," in *Proc. 24th ACM Int. Conf. Super Comput.*, Jun. 2010, pp. 147–158.

[54] S. Han, J. Pool, J. Tran, and W. Dally, "Learning both weights and connections for efficient neural network," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 1–9.

[55] T.-J. Yang, Y.-H. Chen, and V. Sze, "Designing energy-efficient convolutional neural networks using energy-aware pruning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6071–6079.

[56] T.-J. Yang, Y.-H. Chen, J. Emer, and V. Sze, "A method to estimate the energy consumption of deep neural networks," in *Proc. 51st Asilomar Conf. Signals, Syst., Comput.*, Oct. 2017, pp. 1916–1920.

[57] C. F. Rodrigues, G. Riley, and M. Luján, "SyNERGY: An energy measurement and prediction framework for convolutional neural networks on Jetson TX1," in *Proc. Int. Conf. Parallel Distrib. Process. Techn. Appl. (PDPTA)*, 2018, pp. 375–382.

[58] E. Cai, D.-C. Juan, D. Stamoulis, and D. Marculescu, "NeuralPower: Predict and deploy energy-efficient convolutional neural networks," in *Proc. Asian Conf. Mach. Learn.*, 2017, pp. 622–637.

[59] M. Hodak, M. Gorkovenko, and A. Dholakia, "Towards power efficiency in deep learning on data center hardware," in *Proc. IEEE Int. Conf. Big Data*, Dec. 2019, pp. 1814–1820.

[60] R. Joda, T. Pamuklu, P. E. Iturria-Rivera, and M. Erol-Kantarci, "Deep reinforcement learning-based joint user association and CU–DU placement in O-RAN," *IEEE Trans. Netw. Service Manage.*, vol. 19, no. 4, pp. 4097–4110, Dec. 2022.

[61] S. Mollahasani, M. Erol-Kantarci, and R. Wilson, "Dynamic CU–DU selection for resource allocation in O-RAN using actor-critic learning," in *Proc. IEEE Global Commun. Conf.*, Dec. 2021, pp. 1–6.

[62] T. Pamuklu, S. Mollahasani, and M. Erol-Kantarci, "Energy-efficient and delay-guaranteed joint resource allocation and DU selection in O-RAN," in *Proc. IEEE 4th 5G World Forum*, Oct. 2021, pp. 99–104.

[63] S. Mollahasani, T. Pamuklu, R. Wilson, and M. Erol-Kantarci, "Energy-aware dynamic DU selection and NF relocation in O-RAN using actor–critic learning," *Sensors*, vol. 22, no. 13, p. 5029, Jul. 2022.

[64] Y. Azimi, S. Yousefi, H. Kalbkhani, and T. Kunz, "Applications of machine learning in resource management for RAN-slicing in 5G and beyond networks: A survey," *IEEE Access*, vol. 10, pp. 106581–106612, 2022.

[65] A. Filali, B. Nour, S. Cherkaoui, and A. Kobbane, "Communication and computation O-RAN resource slicing for URLLC services using deep reinforcement learning," *IEEE Commun. Standards Mag.*, vol. 7, no. 1, pp. 66–73, Mar. 2023.

[66] M. Sharara, T. Pamuklu, S. Hoteit, V. Vèque, and M. Erol-Kantarci, "Policy-gradient-based reinforcement learning for computing resources allocation in O-RAN," in *Proc. IEEE 11th Int. Conf. Cloud Netw.*, Nov. 2022, pp. 229–236.

[67] N. F. Cheng, T. Pamuklu, and M. Erol-Kantarci, "Reinforcement learning based resource allocation for network slices in O-RAN midhaul," in *Proc. IEEE 20th Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2023, pp. 140–145.

[68] I. Tamim, S. Aleyadeh, and A. Shami, “Intelligent O-RAN traffic steering for URLLC through deep reinforcement learning,” 2023, *arXiv:2303.01960*.

[69] F. Kavehmadavani, V.-D. Nguyen, T. X. Vu, and S. Chatzinotas, “Intelligent traffic steering in beyond 5G open RAN based on LSTM traffic prediction,” *IEEE Trans. Wireless Commun.*, vol. 22, no. 11, pp. 7727–7742, Mar. 2023.

[70] H. Erdol, X. Wang, P. Li, J. D. Thomas, R. Piechocki, G. Oikonomou, R. Inacio, A. Ahmad, K. Briggs, and S. Kapoor, “Federated meta-learning for traffic steering in O-RAN,” in *Proc. IEEE 96th Veh. Technol. Conf.*, Sep. 2022, pp. 1–7.

[71] A. Lacava, M. Polese, R. Sivaraj, R. Soundrarajan, B. S. Bhati, T. Singh, T. Zugno, F. Cuomo, and T. Melodia, “Programmable and customized intelligence for traffic steering in 5G networks using open RAN architectures,” *IEEE Trans. Mobile Comput.*, Apr. 2023, doi: 10.1109/TMC.2023.3266642.

[72] A. Mughees, M. Tahir, M. A. Sheikh, and A. Ahad, “Towards energy efficient 5G networks using machine learning: Taxonomy, research challenges, and future research directions,” *IEEE Access*, vol. 8, pp. 187498–187522, 2020.

[73] L. Saker, S. E. Elayoubi, and H. O. Scheck, “System selection and sleep mode for energy saving in cooperative 2G/3G networks,” in *Proc. IEEE 70th Veh. Technol. Conf. Fall*, Sep. 2009, pp. 1–5.

[74] D. Feng, C. Jiang, G. Lim, L. J. Cimini, G. Feng, and G. Y. Li, “A survey of energy-efficient wireless communications,” *IEEE Commun. Surveys Tuts.*, vol. 15, no. 1, pp. 167–178, 1st Quart., 2013.

[75] D López-Pérez, A. De Domenico, N. Piovesan, G. Xinli, H. Bao, S. Qitao, and M. Debbah, “A survey on 5G radio access network energy efficiency: Massive MIMO, lean carrier design, sleep modes, and machine learning,” *IEEE Commun. Surveys Tuts.*, vol. 24, no. 1, pp. 653–697, 1st Quart., 2022.

[76] X. Liang, A. Al-Tahmeesschi, Q. Wang, S. Chetty, C. Sun, and H. Ahmadi, “Enhancing energy efficiency in O-RAN through intelligent xApps deployment,” 2024, *arXiv:2405.10116*.

[77] G. Du, L. Wang, Q. Liao, and H. Hu, “Deep neural network based cell sleeping control and beamforming optimization in cloud-RAN,” in *Proc. IEEE 90th Veh. Technol. Conf.*, Sep. 2019, pp. 1–5.

[78] Z. Zhu, H. Li, Y. Chen, X. Wen, Z. Lu, and L. Wang, “Joint base station sleeping and functional split orchestration in crosshaul-based V-RAN,” in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Mar. 2023, pp. 1–6.

[79] A. El-Amine, M. Iturralde, H. A. Haj Hassan, and L. Nuaymi, “A distributed Q-learning approach for adaptive sleep modes in 5G networks,” in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2019, pp. 1–6.

[80] A. Iqbal, M.-L. Tham, and Y. C. Chang, “Double deep Q-network-based energy-efficient resource allocation in cloud radio access network,” *IEEE Access*, vol. 9, pp. 20440–20449, 2021.

[81] Y. Azimi, S. Yousefi, H. Kalbkhani, and T. Kunz, “Energy-efficient deep reinforcement learning assisted resource allocation for 5G-RAN slicing,” *IEEE Trans. Veh. Technol.*, vol. 71, no. 1, pp. 856–871, Jan. 2022.

[82] (2022). *Open RAN Technical Priorities Focus on Energy Efficiency*. Accessed: May 15, 2023. [Online]. Available: https://orandownloadsweb.azurewebsites.net/specifications

[83] O-RAN Alliance WG1. (2023). *Network Energy Saving use Cases Technical Report*. Accessed: Aug. 15, 2023. [Online]. Available: https://orandownloadsweb.azurewebsites.net/specifications

[84] F. E. Salem, A. Gati, Z. Altman, and T. Chahed, “Advanced sleep modes and their impact on flow-level performance of 5G networks,” in *Proc. IEEE 86th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2017, pp. 1–7.

[85] F. E. Salem, Z. Altman, A. Gati, T. Chahed, and E. Altman, “Reinforcement learning approach for advanced sleep modes management in 5G networks,” in *Proc. IEEE 88th Veh. Technol. Conf. (VTC-Fall)*, Aug. 2018, pp. 1–5.

[86] F. E. Salem, T. Chahed, E. Altman, A. Gati, and Z. Altman, “Optimal policies of advanced sleep modes for energy-efficient 5G networks,” in *Proc. IEEE 18th Int. Symp. Netw. Comput. Appl. (NCA)*, Sep. 2019, pp. 1–7.

[87] D. Renga, Z. Umar, and M. Meo, “Trading off delay and energy saving through advanced sleep modes in 5G RANs,” *IEEE Trans. Wireless Commun.*, pp. 1–12, 2023.

[88] G. Auer, V. Giannini, C. Desset, I. Godor, P. Skillermark, M. Olsson, M. A. Imran, D. Sabella, M. J. Gonzalez, O. Blume, and A. Fehske, “How much energy is needed to run a wireless network?” *IEEE Wireless Commun.*, vol. 18, no. 5, pp. 40–49, Oct. 2011.

[89] Z. Khan, H. Ahmadi, E. Hossain, M. Coupechoux, L. A. Dasilva, and J. J. Lehtomäki, “Carrier aggregation/channel bonding in next generation cellular networks: Methods and challenges,” *IEEE Netw.*, vol. 28, no. 6, pp. 34–40, Nov. 2014.

[90] G. Yu, Q. Chen, R. Yin, H. Zhang, and G. Y. Li, “Joint downlink and uplink resource allocation for energy-efficient carrier aggregation,” *IEEE Trans. Wireless Commun.*, vol. 14, no. 6, pp. 3207–3218, Jun. 2015.

[91] R. Singh, C. Hasan, X. Foukas, M. Fiore, M. K. Marina, and Y. Wang, “Energy-efficient orchestration of metro-scale 5G radio access networks,” in *Proc. IEEE Conf. Comput. Commun.*, May 2021, pp. 1–10.

[92] H. B. Nafea, M. M. Sallam, and F. W. Zaki, “Study of DRX sleep mode performance on virtual base station energy saving in 5G networks,” *Wireless Pers. Commun.*, vol. 118, no. 4, pp. 3251–3270, Jun. 2021.

[93] T. Zhao, J. Wu, S. Zhou, and Z. Niu, “Energy-delay tradeoffs of virtual base stations with a computational-resource-aware energy consumption model,” in *Proc. IEEE Int. Conf. Commun. Syst.*, Aug. 2014, pp. 26–30.

[94] S. Maxenti, S. D’Oro, L. Bonati, M. Polese, A. Capone, and T. Melodia, “ScalO-RAN: Energy-aware network intelligence scaling in open RAN,” 2023, *arXiv:2312.05096*.

XUANYU LIANG (Student Member, IEEE) received the B.E. degree in communication engineering from Hangzhou City University, Zhejiang, China, in 2021, and the M.S. degree in communication engineering from the University of York, U.K., in 2023, where he is currently pursuing the Ph.D. degree, focusing on network energy efficiency, machine learning, and open radio access networks.

QIAO WANG (Member, IEEE) received the master’s degree from Tampere University of Technology (currently Tampere University), in 2014, and the Ph.D. degree from the University of York, in 2022, with the subject being “Mobility-Prediction Based Proactive Caching in Vehicular Networks.” Since 2015, he has been an LTE System Developer with Ericsson. He is currently a Research Associate with the Yorkshire Open RAN (YORAN) Project funded by the Department for Science, Innovation and Technology (DSIT), U.K. His current research interests include intelligent open RAN, machine learning, and federated learning in O-RAN.

AHMED AL-TAHMEESSCHI (Member, IEEE) received the Ph.D. degree in electrical engineering and electronics from the University of Liverpool, U.K., in 2018. He was a Postdoctoral Researcher with Tokyo University of Agriculture and Technology, from 2019 to 2021, contributing to 5G-Enhance a Horizon 2020 joint project. From 2021 to 2023, he was a Research Associate with the University of Helsinki, and a member of the Finnish Centre of Artificial Intelligence. Currently, he holds a position as a Research Fellow with the University of York, contributing to the YO-RAN Project. His research interests include cognitive radio networks, dynamic spectrum access techniques, algorithm design, localization, open radio access networks, and integration of machine learning into wireless networks.

SWARNA B. CHETTY (Member, IEEE) received the B.E. degree in electronics and communication engineering from Sathyabama University, Chennai, India, in 2014, the M.S. degree in mobile communication systems from the University of Surrey, U.K., in 2016, and the Ph.D. degree from University College Dublin, Ireland, in 2023. Prior to her doctoral studies, she gained professional expertise as a Software Developer. She is currently a Research Associate with the University of York, actively contributing to the YO-RAN Project. Her research interests include network virtualization, resource allocations, microservices, machine learning (especially reinforcement and deep learning), open radio network access, 5G, and beyond communications.

DAVID GRACE (Senior Member, IEEE) received the Ph.D. degree from the University of York, in 1999. He is currently a Professor (Research) and leads the Challenging Environments Research Theme with the School of Physics, Engineering and Technology, a Pillar Lead for Advanced Communications in the University's Institute for Safe Autonomy, and the Director of the Centre for High Altitude Platform Applications. He is a work package lead for the York-led £7.8M U.K. Government YO-RAN Project developing 5G open radio access networks, and recently a lead investigator on U.K. Government funded MANY, dealing with 5G trials in rural areas. He is the author of over 280 papers and the author/an editor of two books. His current research interests include 5G/6G O-RAN systems, application of artificial intelligence to wireless communications, dynamic spectrum access, and interference management. He is a member of U.K. Telecom Infrastructure Network's Expert Working Group on Non-Terrestrial Networks, which aims at influencing government policy and bring together disparate strands of expertise. He is the former Chair of IEEE Technical Committee on Cognitive Networks, from 2013 to 2014; and a Founding Member of the IEEE Technical Committee on Green Communications and Computing. From 2014 to 2018, he was the non-executive Director of Stratospheric Platforms Ltd., developing high altitude platform based wireless systems.

HAMED AHMADI (Senior Member, IEEE) received the Ph.D. degree from the National University of Singapore, in 2012. He is a Reader in digital engineering with the School of Physics, Engineering and Technology, University of York, U.K. He is also an Adjunct Academic with the School of Electrical and Electronic Engineering, University College Dublin, Ireland. He was a SINGA Ph.D. Scholar with the Institute for Infocomm Research, A-STAR. Since then, he has been with different academic and industrial positions in the Republic of Ireland and U.K. He has published more than 90 peer reviewed book chapters, journals, and conference papers. His current research interests include design, analysis, and optimization of wireless communications networks; the application of machine learning in wireless networks; open radio access and networking; green networks; airborne networks; digital twins of networks; and the Internet of Things. He is a fellow of U.K. Higher Education Academy. He has been the Networks Working Group Chair of COST Actions CA15104 (IRACON) and CA20120 (INTERACT). He is a member of editorial board of *IEEE Communication Standards Magazine*, IEEE Systems, and *Wireless Networks* (Springer).