

SURVEY

Applications of Machine Learning in Resource Management for RAN-Slicing in 5G and Beyond Networks: A Survey

YASER AZIMI¹, SALEH YOUSEFI¹, HASHEM KALBKHANI²,
AND THOMAS KUNZ³, (Senior Member, IEEE)

¹Department of Computer Engineering, Urmia University, Urmia 57561-51818, Iran

²Faculty of Electrical Engineering, Urmia University of Technology, Urmia 57166-17165, Iran

³Department of Systems and Computer Engineering, Carleton University, Ottawa, ON K1S 5B6, Canada

Corresponding author: Saleh Yousefi (s.yousefi@urmia.ac.ir)

ABSTRACT One of the key foundations of 5th Generation (5G) and beyond 5G (B5G) networks is network slicing, in which the network is partitioned into several separated logical networks, taking into account the requirements of diverse applications. In this context, resource management is of great importance to instantiate and operate network slices and meet their performance and functional requirements. Resource management in Radio Access Networks (RANs) is associated with a range of challenges due to network dynamics and the specific requirements of each application while ensuring performance isolation. In this paper, we present a survey on state-of-the-art works that employ Machine Learning (ML) techniques in RAN slicing. We begin by reviewing the challenges, then we review the existing papers on resource management in a comprehensive manner, and classify the papers based on the used ML algorithm, the addressed challenges, and the type of allocated resources. We evaluate the maturity of current methods and state a number of open challenges and some solutions to address these challenges in RAN resource management.

INDEX TERMS Network slicing, 5G/B5G, machine learning, reinforcement learning, deep learning, RAN slicing.

I. INTRODUCTION

Due to tremendous growth in the number of users, the volume of traffic, various application requirements, heterogeneous devices and protocols, and varying business models, we need a scalable network that provides better performance with larger connectivity density, higher throughput, lower latency, higher mobility range, and ultra-high reliability with regard to security, trust, and privacy compared to the current networks [1]. 5G/B5G technologies are envisioned to address the commercial and technical requirements of infrastructure service providers and the demands of users in 2020 and beyond. 5G/B5G technologies enable a new type of architecture for an end-to-end network that can actualize a fully mobile and connected society. The number of users of this technology is

expected to increase from 200 million in 2019 to 1.02 billion by 2023. Some research has also claimed that 5G/B5G technologies could serve 1.9 billion people by 2024 [2]. According to [3], its economic value will grow to more than \$12.3 trillion by 2035.

To achieve the goals of the 5G/B5G networks, the next-generation wireless networks (NGWNs) are envisioned to form the foundation for various novel applications that support more diversified services to meet a wide range of application requirements. 5G/B5G networks introduce a global standard for the wireless air interface called New Radio to cover the spectrum that is not used in 4G, using a new antenna technology known as massive MIMO (multiple inputs, multiple outputs). MIMO uses multiple transmitters and receivers for achieving a high data transmission rate. But the use of an additional, new radio spectrum is not the only advantage of 5G/B5G technologies. 5G/B5G technologies allow the use of

The associate editor coordinating the review of this manuscript and approving it for publication was Jad Nasreddine¹.

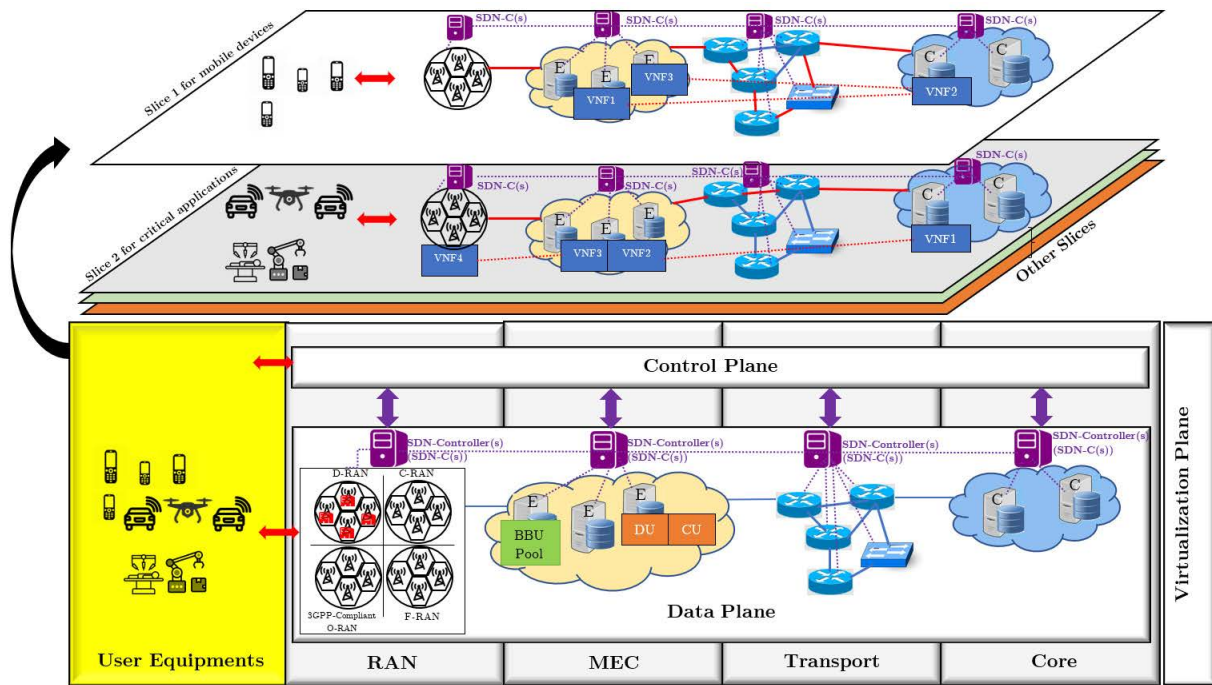


FIGURE 1. Using slicing enablers in network slicing.

a converged, and heterogeneous network with a combination of licensed and unlicensed wireless technologies to increase the available bandwidth to users [1], [4].

5G/B5G networks have introduced a new model of the smart world that supports powerful applications such as self-driving cars, augmented reality, smart industries, smart homes, and smart cities. The applications of 5G network can be generally divided into three types: enhanced mobile broadband connectivity (eMBB), massive machine-type communications (mMTC), and ultra-reliable low latency-critical services (uRLLC) [1], [5]. eMBB enables applications with higher capacity, high data rates, and higher user mobility across a wide coverage area. The mMTC applications are distinguished by a massive number of devices, low volume data transmissions, delay tolerance, and low power consumption, while uRLLC applications have strict higher throughput and low latency requirements for real-time interaction [6]. B5G and the sixth generation (6G) will consider more ambitious services in terms of peak data rate, user-experienced data rate, spectrum efficiency, mobility, latency, connection density, network energy efficiency, and area traffic capacity. In particular, in 6G, eMBB, mMTC, and uRLLC will be expanded to another three services, which are ubiquitous mobile ultra-broadband (uMUB), ultrahigh-speed-with-low-latency communications (uHSLLC), and ultrahigh data density (uHDD). uMUB allows 6G systems to provide any necessary performance throughout the space-aerial-terrestrial-sea region, uHSLLC offers ultrahigh rates and low latency, and uHDD satisfies the data density and high-reliability needs [7].

5G/B5G implementations can be software-defined platforms, in which networking functionality is administered through software rather than hardware. Progress in virtualization, cloud-based technologies, and IT and business process industrialization enables 5G/B5G to be agile and flexible and to provide anytime, anywhere user access. 5G/B5G networks can create software-defined sub-networks that are known as network slices. These slices enable network managers to allocate network functionality based on applications requirements [8]. However, given the very complicated and dynamic environment in 5G/B5G, it is imperative to use an intelligent mechanism for network slicing.

A. IMPORTANCE OF RAN SLICING FOR 5G AND BEYOND

5G/B5G networks can be divided into four parts: radio access network (RAN), mobile edge computing (MEC), transport, and core network (CN) (Fig. 1). A radio access network as a part of a mobile telecommunication system implements a radio access technology. Conceptually, it connects indirectly user equipment (UE) to CN. RANs are important connection points for telecommunications network operators because they perform intensive and complex processing and have significant costs, and now face high demands as more edge and 5G/B5G use cases for customers emerge. A RAN transceiver receives the information via radio waves from UE and sends it to the CN. The resource allocation to UEs is thus a substantial difficulty as a result of the increase in the number of UEs, the disruptive performance needs of users (and slices), and the spatio-temporal changes of the channel quality.

Network slicing is considered as a logical end-to-end network from the RAN to the core that can flexibly provide one or more network services according to the slice requirements, as shown in Fig. 1. In each network slice, the performance requirements are satisfied for users in the slice through implementing certain mechanisms. So far, MEC, transport, and core slicing in 5G/B5G networks have been extensively investigated, but relatively limited efforts have addressed all RAN slicing challenges. Resource management is one key issue in RAN slicing. The limited available resources must be allocated to each user with various Quality of Service (QoS) requirements with regard to traffic changes and network state dynamics. Compared to core network slicing, resource management in RAN slicing is much more challenging, considering radio channel conditions and user mobility.

B. PAPER MOTIVATION AND PAPER CONTRIBUTIONS

Some research efforts have focused on RAN slicing with the aim to minimize network operating costs and maximize resource utilization. In these works (such as [9], [10], [11]), a RAN slicing problem is formulated as an optimization problem that aims to maximize network utilization and ensure the QoS of users in each slice. These complex optimization problems can be solved mainly by classical methods based on iterative optimization algorithms. Therefore, they have high computational and time costs. Also, to model the resource allocation problem in the RAN, we must have accurate prior information about the type of traffic, which is not possible due to the dynamic nature of the network. Hence, to address these limitations and allocate resources efficiently, new RAN slicing methods have used ML techniques. For this reason, in this paper, we focus on RAN slicing and RAN resource management approaches based on ML techniques.

Several surveys introduce network slicing and discuss its challenges and opportunities [12], [13], [14], [15], [16], [17]. However, only [18], [19] directly addressed the application of ML in network slicing for resource management. More concretely, these works rarely discuss the methods based on reinforcement learning and distributed learning (such as transfer learning and federated learning). Furthermore, [18], [19] only focus on some challenges of RAN slicing. Also, [17] just covers the applications of deep reinforcement learning in network slicing. Moreover, a comprehensive survey is not provided to review the proposed works for resource allocation in RAN. This was done in a pretty low-level fashion, which does not meet the needs of those interested in this field. Given these shortcomings and the recent research proposals, a more comprehensive survey is needed that incorporates the recent achievements in the applications of various machine learning methods in different network scenarios of RAN resource management. In this paper, we specifically and in-depth review state-of-the-art applications of various machine learning approaches in RAN resource slicing management. In summary, in contrast to the existing surveys, this paper addresses the following research questions: (i) What are the applications of machine learning techniques to solve the

challenges in RAN slicing? (ii) What are the research directions on ML-based RAN slicing resource allocation?. The main contributions in this paper are summarized as follows:

- This literature review concentrates on ML-based resource management solutions in 5G/B5G networks for RAN slicing.
- We classify the used ML techniques in RAN resource slicing management into supervised learning, unsupervised learning, evolutionary algorithm, reinforcement learning, and distributed learning. Then, we review the state-of-the-art works based on ML in resource management for RAN slicing in 5G/B5G networks.
- This survey classifies the reviewed papers based on the used ML algorithm, the addressed challenges, and the type of allocated resources. We examine the feasibility and applicability of the used ML techniques to solve the existing challenges in RAN slicing.
- We review each paper in detail and categorize them into different sections. At the end of each section, we summarize the lessons learned from the reviewed works. Then, we compare all reviewed methods based on different parameters in RAN slicing, and identify future challenges that need to be considered in RAN slicing. We also state some possible solutions to address these challenges.

C. METHODOLOGY

To address the research questions raised in this paper, we initially define the network slicing enablers, the RAN types and the existing challenges in RAN slicing. Then, we classify the existing ML-based RAN resource management approaches into the used ML techniques, the solved challenges, and the allocated resources. Lastly, we provide research directions in the ML-based RAN slicing resource management area.

To select the works presenting RAN-slicing resource management approaches based on ML, we introduced search keywords on Google Scholar and five electronic databases: Springer, IEEE, ACM, Elsevier, and MDPI. The keywords used are: “Resource management on RAN slicing”, “ML and RAN slicing”, “ML and 5G”, “ML and B5G”, “resource allocation and ML and RAN slicing”, and “resource scheduling and ML and RAN slicing”. The searches resulted in 180 works. Titles and abstracts were reviewed to eliminate works with no relation to the area. Furthermore, recent works from non-recognized conferences and low-impact journals were filtered. As a result, we chose 83 works for a full-text review. We provide in Fig. 2 a percentage breakdown of the ML-based RAN slicing literature based on the publication type. Conference proceedings account for 33% of the papers, while journals account for 59% and preprints account for 8%.

D. OUTLINE OF THE PAPER

The organization of the rest of this paper is as follows. In Section II, we review background concepts of network

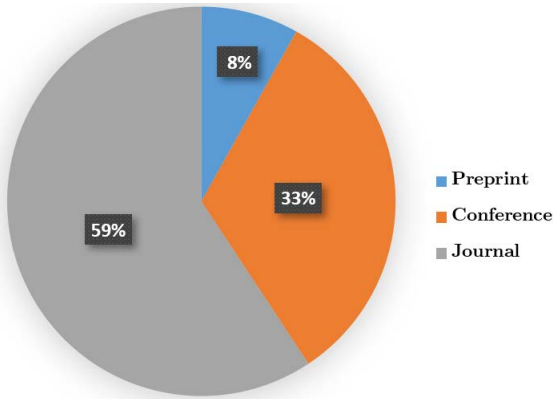


FIGURE 2. Percentage breakdown of the ML-based RAN slicing literature based on the publication type.

slicing in 5G/B5G. Then, we overview different RAN types in Section III. Section IV discusses the key challenges of resource management in RAN slicing. In Section V, we survey the literature of resource management for RAN slicing based on the used ML technique, the addressed challenges, and resource type. We then list the lessons learned from the reviewed papers in each subsection of Section V. We evaluate the reviewed proposals from various aspects in Section VI. In Section VI, we also outline the future challenges and some solutions to address these challenges. Finally, Section VII concludes the paper. Table 1 lists the abbreviations used in this paper.

II. BACKGROUND CONCEPTS OF NETWORK SLICING IN 5G/B5G

One of the important concepts in 5G/B5G networks is network softwarization. Network softwarization allows us to design, implement, build, manage and maintain most equipment, services, and components of the network using software programming. As a result, it gives 5G/B5G networks additional adaptability, programmability, and flexibility. Indeed, end-to-end network slicing can be realized using network softwarization and virtualization employing different techniques such as software-defined network (SDN), network function virtualization (NFV) or a combination of them [15].

Network slicing is introduced as one of the key technologies of 5G/B5G networks that, using enablers such as SDN, NFV, cloud, and MEC, flexibly provides various network services according to users' needs. We can realize all aspects of the 5G/B5G architectures through network slicing. As shown in Fig. 1, network slicing can divide a shared physical network into multiple logical virtual networks, to optimize the allocation of various resources to appropriately support different users of different services. Each slice is able to independently adjust network functions and allocate the corresponding network resources with regard to the needs of commercial and communication scenarios and traffic models, improving flexibility, reliability, and robustness in the whole

network [20]. Although this paper focuses on RAN slicing, some resource management proposals for slicing are end-to-end (from RAN to CN) or a combination of RAN and other network parts (as mentioned in Section I). Therefore, it is beneficial to get acquainted with network slicing enablers before dealing with RAN slicing. In the following, we overview the network slicing enablers including SDN, NFV, cloud computing, and mobile edge computing.

A. SOFTWARE-DEFINED NETWORKING (SDN)

Traditional networking devices were designed to perform a specific set of functionalities, and vendors must update the devices to enable new features. One solution is to build custom hardware products. However, this approach has a high cost. Also, it requires significant effort to provide the latest updated features. Therefore, to tackle these problems, SDN has been introduced. Software-defined networking provides the separation of control functionalities from the data transmission network. The network is divided into two planes, a data plane for data transmission and a control plane for performing control operations. In fact, the control plane of SDN can be logically centralized. Having such a global view of the network, SDN facilitates network management and simplifies the data forwarding hardware [21]. As shown in Fig. 1, one or more SDN controllers can control and manage resource management mechanisms in the whole network from RANs to CN. In SDN, the controller manages packet forwarding through programmable interfaces [22], [23], [24], [25]. Floodlight, OpenDayLight (ODL), Onix, and NOX are the most well-known SDN controllers. Also, OpenFlow as a popular interface is used for controlling the packet flow [8].

The SDN enables logically centralized management of equipment, components, and services across RAN, MEC, transport, and core, enabling end-to-end network slicing throughout the network. In MEC and core, SDN and NFV are used for implementing VNFs and sharing them among different slices. In transport, SDN is beneficial in implementing different virtual circuits for different slices. Especially given the focus of this paper on RAN slicing, the use of programmable radio platforms such as software-defined radio (SDR) can reduce spectrum limitations and improve transceiver power through dynamic power allocation based on the channel information. Moreover, SDN may be beneficial for some VNFs inside RANs (particularly non-centralized RANs) [15].

B. NETWORK FUNCTION VIRTUALIZATION (NFV)

Network function virtualization (NFV) is a technique for transforming the physical network nodes' functions into virtual functions that can be chained together through virtual links to provide different communication services. One or more virtual machines can be used to emulate physical nodes such as switches, servers, routers, load balancers, firewalls, and various hardware resources, and create a virtual network function. In fact, NFV allows running virtual instances of physical nodes on commercial off-the-shelf hardware,

TABLE 1. List of abbreviations.

3GPP	3rd Generation Partnership Project	MBDQN	Multi-branch dueling Q-network
5G	Fifth Generation	MCTS	Monte Carlo Tree Search
5G-PPP	5G Infrastructure Public Private Partnership	MDP	Markov Decision Process
A2C	Advantage Actor-Critic	MEC	Mobile Edge Computing
A3C	Asynchronous Advantage Actor-Critic	MIMO	Multiple inputs, multiple outputs
AAU	Active Antenna Unit	ML	Machine-learning
AC	Actor-Critic	mMTC	Massive Machine-type Communications
ADMM	Alternating direction method of multipliers	MU	Mobile User
AI	Artificial Intelligence	NB	NodeB
AoI	Age of information	NF	Network Function
AP	Access Point	NFV	Network Function Virtualization
ASO	Access side resource allocation	NGWN	Next-generation wireless network
BBU	Base-band Unit	NOMA	Non-orthogonal multiple access
BIP	Binary integer programming	NQL	Nash Q-learning
BS	Base Station	OAI	Open-Air Interface
BSC	BS coverage	OBSAI	Open Base Station Architecture Initiative
CMDP	Constrained MDP	ODL	Open Day Light
CN	Core Network	OFDMA	Orthogonal frequency-division multiplexing
CPRI	Common Public Radio Interface	OPEX	Operational Expenditures
CQI	Channel Quality Indicator	O-RAN	Open RAN
C-RAN	Centralized RAN (cloud C-RAN)	PDCP	Packet Data Convergence Protocol
CSI	Channel State Information	P-GW	Packet gateway
CU	Centralized Unit	PPO	Proximal Policy Optimization
D2D	Device-to-device	PRB	Physical resource block
DDPG	Deep deterministic policy gradient	PSO	Particle swarm optimization
DDQN	Deep Dueling/Double Q-Network	QoE	Quality of Experience
DL	Down-link	QoS	Quality of Service
DNN	Deep Neural Network	RAC-V2V	Relay assisted cellular vehicle-to-vehicle
DPGD	Deterministic policy gradient descent	RAN	Radio Access Network
D-RAN	Distributed RAN	RAT	Radio Access Technology
DRL	Deep Reinforcement Learning	RB	Resource block
DRQN	Deep recurrent Q-Network	RF	radio frequency
DU	Distributed Unit	RL	Reinforcement Learning
EC	Edge controller	RLC	Radio Link Control
eCPRI	Enhanced Common Public Radio Interface	RNN	Recurrent Neural Network
EE	Energy Efficiency	RRC	Radio Resource Control
eMBB	Enhanced Mobile Broadband	RRH	Remote Radio Head
eNB	Evolved Node B	RRU	Remote Radio Unit
F-AP	Fog access points	RSMA	Rate-splitting multiple-access
FCFS	First Come First Service	RSS	Received signal strength
FH	Finite-horizon	RSU	Road-side Unit
FIFO	First In First Out	SARL	Single-agent RL
FL	Federated Learning	SARSA	State-action-reward-state-action
FN	Fog node	SAU	Serve-all-utilities
F-RAN	Fog RAN	SDAP	Service Data Adaptation Protocol
GA	Genetic algorithm	SDN	Software-defined Networking
GAN	Generative adversarial network	SE	Spectral efficiency
GAT	Graph attention	SHU	serve-high-utilities
gNB	Next Generation Node B	SINR	Signal-to-Interference-plus-Noise Ratio
GoS	Grade of Service	SLA	Service Level Agreement
GPU	Graphic processing unit	SMDP	Semi MDP
GRU	Gated recurrent unit	SP	Service Provider
HetNet	Heterogeneous networks	SSR	Service level agreement satisfaction ratio
IH	Infinite-horizon	SVM	Support vector machine
InP	Infrastructure provider	TRL	Transfer Reinforcement Learning
IoT	Internet of Things	UE	User Equipment
IoV	Internet of Vehicles	uHDD	UltraHigh Data Density
IPO	Internal-point Policy Optimization	uHSLLC	UltraHigh-Speed-with-Low-Latency Communications
KNN	K-nearest-neighbor	UL	Up-link
KPI	Key Performance Indicator	uMUB	Ubiquitous Mobile Ultra-Broadband
LFU	Least Frequently Used	uRLLC	Ultra-reliable and Low-latency Communications
LRTQ	Latency Reliability-Throughput Q-learning	V2I	Vehicle-to-Infrastructure
LRU	Least Recently Used	V2V	Vehicle-to-Vehicle
LSTM	Long Short-Term Memory	V2X	Vehicle-to-everything
LTE	Long Term Evolution	vBBUs	Virtual baseband units
MAC	Media Access Control	VNF	Virtual Network Function
MANO	MANagement and Orchestration	VoLTE	Voice over Long-Term Evolution
MARL	Multi-agent RL		

which is therefore general-purpose and cheaper than a dedicated piece of hardware. NFV allows software developers to run their software on generic shared hardware without the need for new dedicated or purpose-built hardware. NFV also increases scalability, efficiency, flexibility, and ease of management of the network [21], [26], [27], [28]. As shown in Fig. 1, one or more VNF can be created in different parts of the network from RANs to CN based on the requirements of various slices.

Virtualization in general and NFV simplifies end-to-end slicing since functions are implemented in software and on commodity hardware. It allows network administrators to safely isolate and dynamically allocate the resources independent of the underlying network by creating different service function chains (SFCs) throughout the network. As shown in Fig. 1, virtualization encompasses both control and data layers across all parts of the end-to-end path. Radio resources and computational functions in RAN can be virtualized with the help of SDR. Most of the functions of MEC are easily virtualized using commodity hardware. In the transport part, most of the equipment such as routers and switches can be virtualized. In the core network, different core functions such as mobility management function (AMF) and session management function (SMF), and user plane function (UPF) can be virtualized. NFV, when assisted by SDN, leads to better control and management taking into account a more global view of the network [15].

C. CLOUD COMPUTING AND MOBILE EDGE COMPUTING

Cloud computing provides the delivery of computing services such as computing and storage resources over the Internet and enables users of slices to access the on-demand resources anytime and anywhere. Some functionalities of the core can be implemented in a private/public cloud. It leads to the advantages of scalability, elasticity, resource pooling, and sufficient storage capacity. However, due to the high distance from the UEs, the high latency makes it inappropriate for real-time applications. To tackle this issue and serve users with latency-sensitive applications such as augmented reality and e-health, edge computing is introduced, aiming to push the storage and computing resources to the edge of the network (Fig. 1). In the literature, the three terms cloudlets, fog computing, and mobile edge computing (MEC) are introduced with different scope and natural features for this purpose [21]. In a cloudlet, small-scale data centers are placed at the network edge to run resource-intensive applications. Fog computing uses edge devices, e.g., routers and gateways, by users for storage and computing purpose. Finally, MEC places servers in the BSs [29], [30], [31] and is a common concept discussed in 3GPP documents.

MEC can be utilized to serve the users of delay-sensitive slices and deliver the desired QoS because the processes and data are close to the end users. Additionally, some RAN operational elements like base-band units (BBUs) and distributed unit-centralized units (DU-CUs) can be transferred to MEC, and by virtualizing these elements, we can improve RAN

management and control and boost flexibility. Besides, MEC allows for the allocation of certain resources (such as computing, storage, and caching memory) to users of ultra-low latency slices as virtual functions [15].

III. RAN TYPES

Traditionally, a group of Base Stations (BS) or NodeB (NB) which supply wireless access to mobile users forms a Radio Access Network (RAN). Each BS is composed of an antenna, radio frequency (RF) equipment, digital processors, and base-band units. The base stations are used to create a connection between a user's device and the core network and transmit the traffic (such as voice, data, and video) to the core network.

The base station is formed of two different parts that will operate separately, called the Base-Band Unit (BBU) and the Remote Radio Unit (RRU) or Remote Radio Head (RRH). When the base station receives a signal, the signal or the radio frequency (RF) from the antenna is retrieved by the RRU. Also, the RRU constructs the analog transmitted RF to the baseband unit. Then, the BBU modulates, generates, and processes the digitalized signal. In general, the BBU manages most of the digital processing such as de(coding) and de(modulation). In the following, we outline several RAN architectures based on this generic description.

A. DISTRIBUTED-RAN (D-RAN) RESOURCE MANAGEMENT

In a D-RAN architecture, both units of a base station are located at a cell tower site as shown in Fig. 3. The RRUs are positioned at the top of the tower next to the antennas, and the BBUs are positioned in an equipment room close to the tower itself. The RRUs and BBUs are directly linked via a Common Public Radio Interface (CPRI) or an Open Base Station Architecture Initiative (OBSAI) connection. The data of each RRU will be processed by one BBU [32].

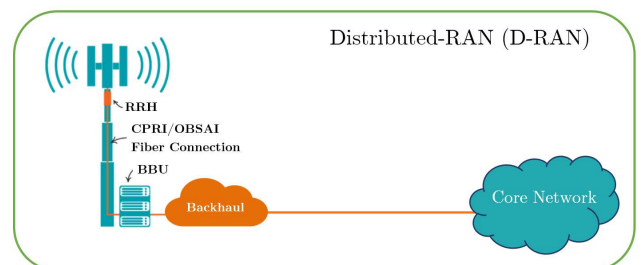


FIGURE 3. Distributed-RAN (D-RAN).

The D-RAN architecture raises several issues for telecoms mainly in terms of managing the space and capacity required at the BSs. The increase in the number of BSs in the region also failed to meet the requirements of users at peak capacity periods. Also, this type of RAN was not designed to meet the needs of short bursts of traffic and real-time applications. All of these issues paved the way for the emergence of new architectures in RAN, e.g., C-RAN.

B. CENTRALIZED-RAN (C-RAN) RESOURCE MANAGEMENT

In a C-RAN architecture, the base station is split up so that the RRUs stay in the place of the cell site, while the BBUs are aggregated into a centralized office called a BBU Pool, with a centralized controller for digital processing as shown in Fig. 4. In this concept, all key functionalities are brought to the central office, in which the computationally intensive baseband computation can be managed and organized by the virtualization of the resources in the BBU pool and minimize the costs of the maintenance of the 5G network. Each BBU can be considered as a virtual node, while the communication with other BBUs is via a virtual link [32], [33].

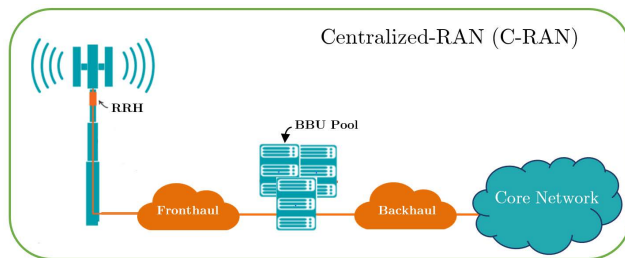


FIGURE 4. Centralized-RAN (C-RAN).

Although the aggregation of BBUs has some advantages (e.g., resource virtualization, better service deployment at the edge instead of the core network, the utilization of advanced technologies for high processing power, data security, and low power consumption), it is faced with challenges in the fronthaul network to meet the requirements of 5G. For this reason, enhanced CPRI (eCPRI) is introduced specifically for the 5G fronthaul transport layer. Also, the C-RAN community has recommended fiber due to its high bandwidth requirements to tackle the issue.

C. 3GPP-COMPLIANT OPEN-RAN (O-RAN) RESOURCE MANAGEMENT

To address the high data rate and low latency requirements in 4G, massive MIMO (multiple-input, multiple-output) is introduced by 5G in which the RRU and the antenna are combined into an Active Antenna Unit (AAU) for reducing the signal loss.

In the 3GPP-compliant O-RAN, the BBU is split up into the Distributed Unit (DU) and the Centralized Unit (CU) as shown in Fig. 5. The DU operates close to the RRU and performs real-time operations in layer 1 (Physical) such as OFDMA and MIMO, and scheduling functions in layer 2 such as RLC (Radio Link Control) and MAC (Media Access Control). The CU sits between the 5G core network and the DUs and controls the eNB/gNB functions based on the functional split option. Also, it is responsible for access control and coordinating high-level protocols such as Radio Resource Control (RRC) in the control plane, Service Data Adaptation Protocol (SDAP), and Packet Data Convergence Protocol (PDCP) in the user plane [34].

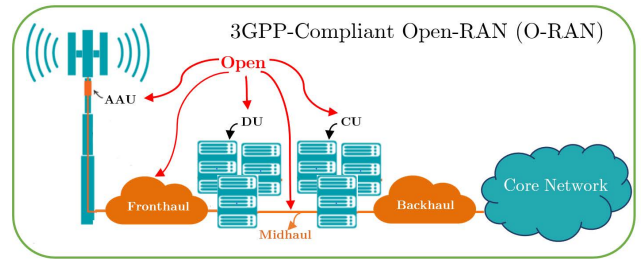


FIGURE 5. 3GPP-Compliant Open-RAN (O-RAN).

C-RAN also at times features a decoupled BBU structure where all parts of the RAN must be from the same vendor. The 3GPP-compliant O-RAN alliance has introduced an open interface specification for AAU-DU and DU-CU. Consequently, a customer can mix and match the different RAN components from various vendors without dependence on a specific vendor, resulting in a 3GPP-compliant open RAN [34].

D. FOG-RAN (F-RAN) RESOURCE MANAGEMENT

The F-RAN was designed to use the advantages of both fog computing and C-RAN to address growing traffic demands and provide higher QoS to the end-users. As shown in Fig. 6, the F-RAN consists of three layers: terminal layer, network access layer, and cloud computing layer. The fog user equipments (F-UEs) in the terminal layer and the fog access points (F-APs) in the network access layer create the mobile fog computing layer. Indeed, the fog computing layer consists of multiple F-APs and F-UEs which are responsible for providing different resources based on the low-latency requirements of IoT applications and other real-time applications. Also, F-UEs in the terminal layer use the high power nodes (HPNs) in the network layer to get the system signaling information. Furthermore, the neighboring F-UEs in the terminal layer can directly communicate with one another by using the device-to-device (D2D) communication mode. The F-APs in the network access layer are responsible for processing and forwarding the received data from F-UEs to the BBU pool in the cloud computing layer [32].

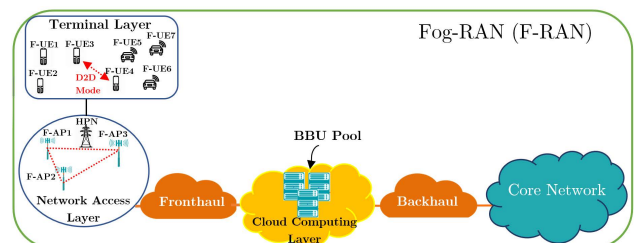


FIGURE 6. Fog-RAN (F-RAN).

IV. KEY CHALLENGES IN RAN SLICING

In this section, we have summarized the challenges identified in the reviewed papers on RAN slicing. Then, in Section V,

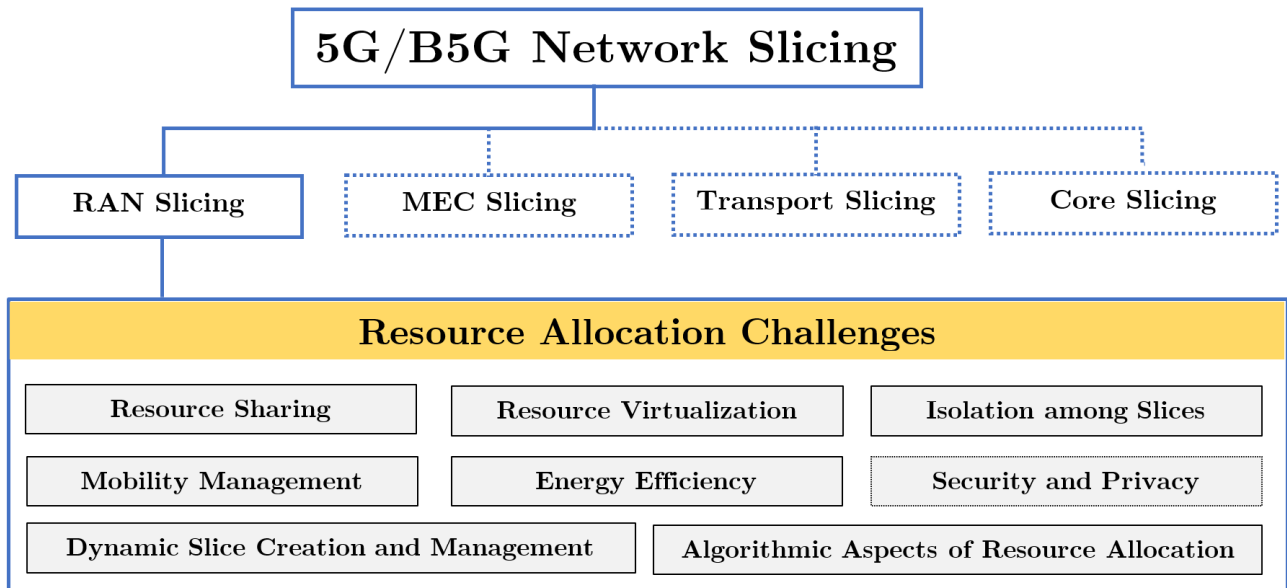


FIGURE 7. Resource allocation challenges in RAN slicing.

we classify the reviewed papers into different categories based on the used ML techniques, the addressed challenges, and the type of allocated resources. In Fig. 7, the important identified challenges for RAN slicing in 5G/B5G networks are summarized, which are resource sharing, resource virtualization, isolation among slices, mobility management, energy efficiency, security and privacy, dynamic slice creation and management, and algorithmic aspects of resource allocation. A proper RAN slicing solution should be able to solve a significant amount of the challenges mentioned in this section. All the reviewed solutions in this paper are compared in terms of addressing these challenges in Tables 3, 5, 7, 9 in Section VI. Although some of these challenges also can exist in other parts of the network such as MEC, transport, and core, they are beyond the scope of this paper.

A. RESOURCE SHARING

Sharing and scheduling of resources among slice tenants is a key issue for the NGMN Alliance. To implement resource sharing, two approaches are considered: static partition sharing or dynamic sharing. In the static method, a specific and fixed amount of resources are allocated to each slice, which potentially wastes resources and reduces the quality of the signal for radio resources. Due to the dynamics of network load, dynamic resource sharing methods can result in higher resource utilization. Although resource sharing yields many benefits to infrastructure providers, it also introduces new challenges such as slice isolation [35].

B. RESOURCE VIRTUALIZATION

Virtualization is seen as a key enabler to achieve network slicing. Virtualization technology with the introduction of

hypervisors, virtual machines, and container-based virtualization has matured over the past years and been utilized mainly in wired networks. A lot of work has concentrated on virtualization problems in the core network, such as VNF placement and management. However, virtualization in wireless communication is more complex, taking into account time-varying channels and interference. So, we cannot simply apply the traditional virtualization methods developed in wired networks for their wireless part. Therefore, designing new virtualization methods for sharing wireless spectrum resources and virtualization of base stations plays a vital role in RAN slicing implementation [36].

C. ISOLATION AMONG SLICES

Isolation is a crucial property of network slicing that ensures performance and security for each tenant, even when various tenants use network slices with contradictory requirements. Although strong isolation among slices can decrease configuration overhead in the network by minimizing the re-configuring operations of resources between slices, it may lead to low resource utilization. At the same time, poor isolation may lead to violations of their performance requirements [37]. Isolation can be addressed (i) by each slice using a different physical resource, (ii) by using virtualization for a shared resource, and (iii) through sharing a resource via appropriate mechanisms (e.g., scheduling) for each tenant.

D. MOBILITY MANAGEMENT

Network slicing is confronted by challenges such as seamless handover and interference in mobility management. For real-time applications, the use of a fast handover ensures its quality of service. In addition, reserving resources based on

users' mobility to ensure E2E communication for accepted services reduces the outage probability due to lack of resource allocation. Consequently, mobility management is necessary for specific slices while considering various mobility requirements. Therefore, to tackle mobility challenges in network slicing requires the design of a slice-oriented mobility management protocol [38].

E. ENERGY EFFICIENCY

Energy consumption management is a key factor in the deployment and evolution of network slicing. Due to the increasing number of connected devices to mobile networks, and the increasing density of users, a high number of base stations and related infrastructure that use high energy are required to serve different users across different slices. Sustainability and green computing issues are new energy management principles that must be considered in this challenge. Furthermore, the service cost for users will increase due to operational expenditure for the management and orchestration (MANO), as increasing the energy consumption results in high energy supply costs for operators. To this end, an efficient energy management system is needed to manage the power consumption in network slicing [39]. The most important approach for maximizing spectrum efficiency and energy efficiency is dynamic power allocation according to the channel conditions, which reduces the energy consumption of UEs and increases the user acceptance rate due to the optimal use of radio resources [40].

F. SECURITY AND PRIVACY

Security attacks or faults occurring in one slice must not impact other slices. Besides, to prevent unauthorized access to read and write in the specification of slices (e.g., configuration/management/accounting information) and privacy-preservation of users, each slice can independently implement its own security functions and log information about the slice [41]. Security and privacy challenges are beyond the scope of this paper.

G. DYNAMIC SLICE CREATION AND MANAGEMENT

One of the concerns for operators of 5G/B5G networks is how to manage and create the network slices, aiming to increase their benefit. For example, life-cycle management of slices with their various requirements in RAN and core network are crucial problems to be solved. In addition, any changes in a slice configuration such as a scale-up/down in resources, removal, creation, activation, deactivation, based on varying service loads should not affect other slices. Also, the management of each slice must be deployed in an automated manner with its own security policy to tackle errors and security issues raised by human operators [12].

H. ALGORITHMIC ASPECTS OF RESOURCE ALLOCATION

A major challenge in network slicing is how to efficiently manage resources. A lot of work based on mathematical methods in operation research or heuristics, meta-heuristics,

or ML are proposed to solve the resource allocation and virtual embedding problems for slices. Due to the dynamic characteristics of a 5G/B5G network and the number of slices and system size, resource allocation algorithms must be able to reconfigure and (re-)allocate slice resources. Also, the proposed resource allocation method must be applicable in terms of time and computational complexity to respond to real-time applications in an acceptable time due to the dynamic changes of the network. ML-based methods have been used extensively in network slicing compared to other methods. Network slicing methods can enhance the quality of service and quality of user experiences to respond to real-time requests through ML, taking into account network dynamics and network changes. Network slicing needs to enlist ML techniques for automated provisioning and proactive management of traffic and services [42]. However, the convergence speed of ML methods is very important because incorrect decisions about resource allocation due to lack of convergence or low convergence speed in the used ML method can cause poor network performance.

V. RESOURCE MANAGEMENT FOR RAN SLICING IN 5G/B5G NETWORKS

The proposed methods for efficient resource management in network slicing can be classified into two classes: admission control in slices and cross-slice resource allocation [18]. The first class is used to control user acceptance in slices because acceptance of all users' requests may decrease the QoS of the accepted users when the system faces resource limitations. The methods of the second class address optimal resource allocation between the slices, taking into account the traffic load changes and network dynamics. Furthermore, to ensure the requirements of slices and maximize the resource utilization, we want to avoid over-allocating resources to slices.

Decisions on admission control in slices and cross-slice resource allocation are made based on two approaches: policy-based approaches (e.g., [43], [44]) and auction-based approaches (e.g., [45], [46]). In the policy-based approaches, the admission control in slices and cross-slice resource allocation are done based on system information and the adopted policy. In the auction-based approaches, users' requests are sent to all tenants, and according to the collected bids of the tenants, the winner of the request is determined based on the user's condition.

The main challenge of conventional policy-based methods is their high time and computational complexity [18]. Admission control in slices and the cross-slice resource allocation are often modeled as binary programming and integer programming problems, respectively. These problems are non-convex and are known as NP-hard problems. Therefore, these methods cannot be used in real-time applications. Compared to policy-based methods, auction-based methods are effective methods with low time and computational complexity. But these methods require a strong auction mechanism and strict rules. In addition, there are challenges, such as

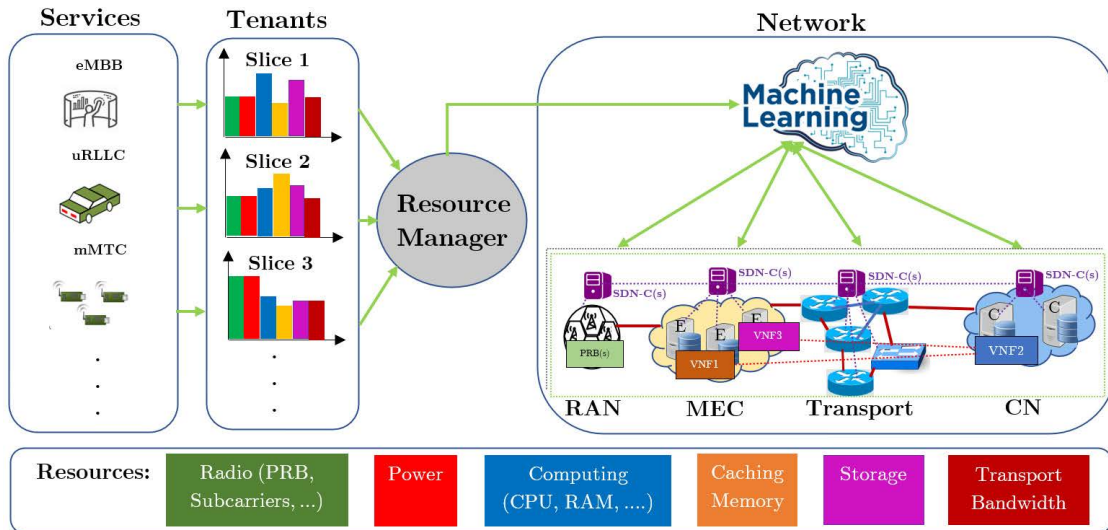


FIGURE 8. Using ML techniques in network slicing.

multi-round auction overhead, biased bidding, and cheating in these methods.

Recently, ML has been considered as an approach for RAN slicing in 5G/B5G networks. It can take advantage of both policy-based methods and auction-based methods while reducing the complexity and increasing the speed. In Fig. 8, one or multiple ML techniques use the received information from different environments for various applications, or interact online and dynamically with these environments and achieve knowledge over time for managing and allocating resources such as radio, transmission power, virtual computational resources, caching memory, storage, and transport bandwidth to different slices from RANs to CN. In this section, we mainly concentrate on ML-based RAN resource management algorithms in 5G/B5G networks using supervised learning, unsupervised learning, evolutionary learning, reinforcement learning, and distributed learning, as shown in Fig. 9.

In Fig. 10, a taxonomy of the applications of ML for addressing the challenges of resource allocation in RAN slicing (Fig. 7) to allocate various resources (i.e., radio, transmission power, computational, caching memory, and transport bandwidth) in RAN, MEC, transport, and core is shown. Although this paper focuses on RAN slicing, some resource management proposals for slicing are end-to-end (from RAN to CN) or a combination of RAN and other network parts, i.e., MEC, transport, and core. So, we also review these proposals that are a combination of RAN and other parts of the network. We have classified the proposals based on the used ML algorithm, the addressed challenges, and the type of allocated resources. The main reason for this classification in this survey is the effect of these parameters on the applicability of ML algorithms and solving the challenges in RAN slicing. In the following, if the RAN type is not mentioned explicitly, the RAN type is equal to D-RAN.

In each subsection, we discuss the advantages and disadvantages of the resource management proposals based on different parameters for RAN slicing in Tables 2-9. Comparison criteria for the reviewed proposals are the allocated resource types, the type of ML technique (Section V), isolation (Section IV-C), dynamic power allocation (Section IV-E), computational and time complexity (Section IV-H), convergence speed (Section IV-H), energy efficiency (Section IV-E), RAN type (Section V), the use of SDN/NFV (Section II-A, II-B, and IV-B), mobility management (Section IV-D), dynamic slice creation and management (Section IV-G), and the simulation Tools.

A. SUPERVISED AND UNSUPERVISED LEARNING

Supervised learning algorithms consist of input data (which is called training data) and a target/label variable that is to be predicted from a given set of predictors. We can generate a function for mapping inputs to desired outputs by using these data. The training process ends when the model reaches the desired level of accuracy on the training data. In unsupervised learning algorithms, input data does not have a label. Rather, the learning task aims to deduce structures present in the input data through a mathematical process or similarity analysis. This allows us to group similar data [47], [48], [49], [50].

1) RESOURCE SHARING

In the following, we review the proposals that address resource sharing. In these proposals, power and radio resources are considered for resource allocation. In [51], a supervised DNN method for radio resource allocation in the RAN is presented. In this method, the resource allocation for each application is done based on the tags of the packets in the P-GW (packet gateway). The simulation result shows that the proposed method outperforms greedy and fair-sharing (equal division of bandwidth between applications) methods

Type of Machine Learning Techniques		
Techniques	Applications	Methods
Supervised Learning	Classification/Regression/Estimation	Neural Networks/ Support Vector Machine/ Bayesian Networks/ ...
Unsupervised Learning	Clustering/Dimensionality Reduction/Prediction	K-means/ Gaussian Mixture Model/ Dirichlet Process/ Mixture Model/ Feature Selection/ Linear Discriminant Analysis (LDA)/ ...
Evolutionary Learning	Biological Processes	Genetic Algorithm/ Genetic Programming/ Evolutionary Programming/ Gene Expression Programming/ Evolution Strategy/ ...
Reinforcement Learning	Online Learning/ ...	Model-Free
		Policy Optimization • Policy Gradient/ A2C/ A3C/ PPO/ TRPO/...
		Q-Learning • DQN/ C51/ Double DQN/ Dueling-DQN/...
		Policy Optimization+Q-Learning • DDPG/ TD3/ ...
		Model-Based
		Learn the Model • World Models/ I2A/ MBMF/ MBVE/
	Given the Model • Alpha-Zero/ ...	
Distributed Learning	Classification/Regression/Estimation/Clustering/ Dimensionality Reduction/Prediction/Biological Processes/Online Learning/ ...	Distributed or Federated AC/ A2C/ A3C/ DQN/ LSTM/ NN/ ...

FIGURE 9. Classification of used machine learning techniques in RAN Slicing.

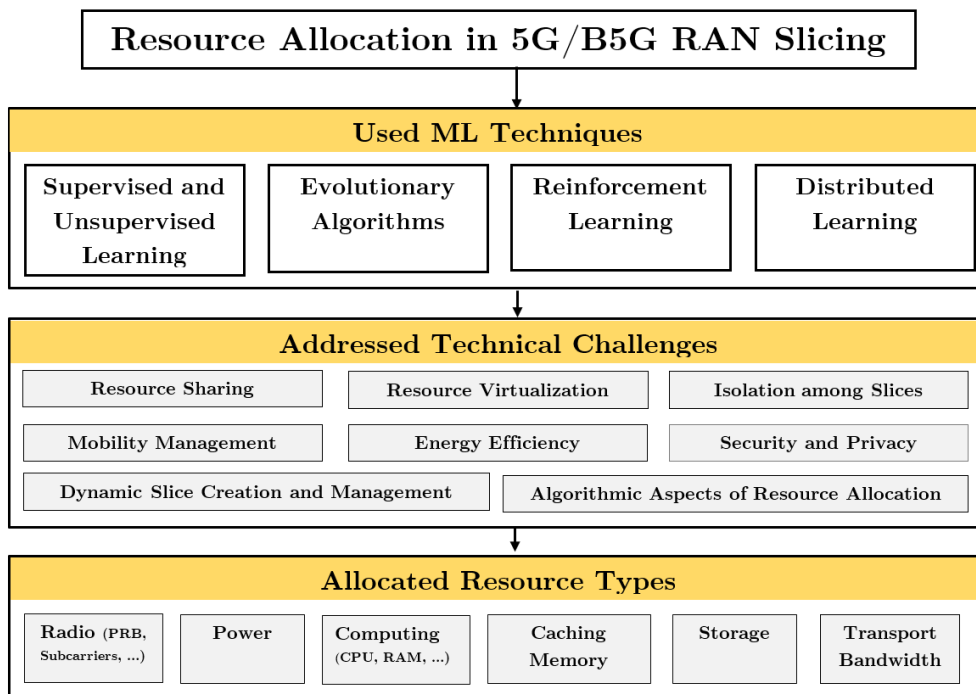


FIGURE 10. A taxonomy of the applications of ML for RAN slicing to allocate various resources.

in terms of throughput. The authors in [52] have proposed an approach using a supervised convolutional neural network and a supervised random forest called DeepSlice for E2E

resource allocation to eMBB, uRLLC, and mMTC slices. DeepSlice uses the Key Performance Indicator (KPI) as a dataset from both the network and the devices, such as the

type of device used to connect, UE category, QoS Class Identifier, packet delay budget, maximum packet loss, time and day of the week. Supervised random forest is used for the identification of the type of application of the device to map to the desired slice using collected information, and the supervised convolutional neural network is used to predict the required resources in RAN and CN. The simulation result shows that DeepSlice assigns the users to the appropriate slices with high accuracy and has a high throughput and a high device acceptance ratio. In [53], a deep learning algorithm is presented to predict if a service provider can fulfill a new request in the slice, taking into account the conditions of the channel and the allocated resources. Due to the transmission data sequence, a Long Short Term Memory as a supervised learning method is used to predict the channel conditions in the near future. The simulation result shows that the proposed method has higher accuracy than a recurrent neural network for predicting channel conditions. In [54], a prediction-based method using a supervised DNN is proposed for spectrum allocation, aiming to minimize spectrum allocation costs for the tenants, maximize radio resource utilization, and guarantee desired service level agreements (SLAs) of the tenants. The simulation result shows that the proposed method predicts the traffic with high accuracy. In [55], a method is proposed for allocating resources in the RAN that aims to reduce the Channel Quality Indicator (CQI) transmission load while a channel is stable. In this method, queuing theory is used to allocate resources, and four algorithms, including supervised support vector machine (SVM), supervised neural network, supervised LSTM, and the proposed optimal method are used to predict channel stability. The simulation results for different mobility models show that LSTM and the proposed optimal method can estimate the channel conditions with higher accuracy.

2) RESOURCE VIRTUALIZATION

Compared to the proposals in Section V-A1, [56], [57] address resource virtualization besides resource sharing. In [56], radio, power, and computational resources are allocated to the slices. Compared to [56], storage in addition to radio and power resources is allocated to the slices in [57]. In [56], a supervised DNN method is proposed using the KPI as a dataset that includes traffic load per slice, downlink (DL) physical resource blocks (PRBs) at a transmission/reception point (TRP), radio resource connected (RRC) users' licenses at the virtual base-band units (vBBUs), CPU load, and back-haul capacity for provisioning resources in each slice, taking into account the constraints on transmission rate. The simulation result shows that the proposed method fulfills the desired objectives based on different simulation parameters for eMBB, social media, and browsing slices. The authors in [57] have proposed an E2E slicing approach using supervised DNN and slicing enablers such as SDN and NFV for the identification of the application type to map to the proper slice based on traffic information collected in an MVNO (Mobile Virtual Network Operator). In this approach,

traffic from UEs to VNFs is classified by supervised DNN in the data plane, and the control plane is responsible for signaling among the different elements. The simulation result shows that the proposed approach classifies various applications into desired slices with high accuracy.

3) ISOLATION

[58] and [59] consider the isolation among slices compared to the proposals in Section V-A2. In [58], an approach is proposed utilizing supervised DNN, SDN, and NFV that uses the Key Performance Indicator (KPI) as a dataset to allocate radio, power, and computational resources to the slices. Virtual SDN controllers are used for signaling between internal elements of the RAN. In this approach, a low-complexity predictor based on a soft gated recurrent unit (GRU) is used for the estimation of the traffic of the slices. Then, to estimate the required resources based on the traffic per slice in each virtual function, a joint multi-slice deep neural network is presented by considering violation rate-based SLA and resource bounds-based SLA. The simulation result shows that the proposed approach fulfills the desired objectives based on different simulation parameters for eMBB, social media, and browsing slices. In [59], a self-sustained RAN slicing framework is proposed to allocate radio, power, computational, and caching memory resources in RAN by transfer neural network, SDN, and NFV. The proposed RAN slicing framework allocates resources into three levels, i.e., network-level slicing, next-generation NodeB-level (gNodeB) slicing, and packet scheduling level slicing. At the network level, the resources are allocated to each gNodeB on a large timescale. Then, the configuration of each slice in the cell is adjusted by each gNodeB in the gNodeB-level on the large timescale. Finally, the radio resources are allocated to users in each network slice on a small timescale at the packet scheduling level. The SDN controller manages the set of gNodeBs, and the computational and caching memory resources are allocated as VNFs close to the users or in the CN based on the various QoS requirements for different slices. Also, a transfer neural network is used to predict the required resources for each gNodeB at the network level, which can use the prior knowledge of the other RANs. The simulation results show that the proposed RAN slicing framework enhances dramatically the QoS for users.

4) LESSONS LEARNED

In Tables 2 and 3, we summarize the reviewed proposals that use supervised and unsupervised learning for resource allocation in RAN slicing. The allocated resource types in each proposal are shown in Table 2. Also, we compare the proposals based on different features in Table 3 as mentioned previously at the beginning of this section. The allocated resource types in each proposal are shown in Table 2. Also, we compare the proposals based on different features in Table 3 as mentioned earlier in Section 4. As the proposals in groups {1-6} show, the use of supervised and unsupervised methods for resource allocation can be a suitable alternative

TABLE 2. Summary of related work for supervised and unsupervised learning methods based on allocated resource types.

Ref.	Allocated Resource Types					
	Radio	Power	Computational	Storage	Caching Memory	Transport Bandwidth
[51], [52],[53], [54], [55]	✓	✓	-	-	-	-
[56], [58]	✓	✓	✓	-	-	-
[57]	✓	✓	✓	✓	-	-
[59]	✓	✓	✓	-	✓	-

TABLE 3. Summary of related work for resource allocation in RAN slicing based on supervised and unsupervised learning.

Group ID →	Ref.					
	1	2	3	4	5	6
Features ↓	[51]	[54], [56]	[58], [59]	[57]	[52], [53]	[55]
Isolation	-	-	✓	-	-	-
Dynamic Power Allocation	-	-	-	-	-	-
Computational Complexity	High	High	High	High	High	High
Convergence Speed	Low	Low	Low	Low	Low	Low
Energy Efficiency	-	-	-	-	-	-
Time Complexity	High	High	High	High	High	High
SDN	-	-	✓	✓	-	-
NFV	-	-	✓	✓	-	-
RAN Type	D-RAN	D-RAN	D-RAN	D-RAN	D-RAN	D-RAN
Mobility Management	-	-	-	-	-	-
Dynamic Slice Creation and Management	-	-	-	-	-	-
Simulation Tools	OAI/TensorFlow	-	-	OAI/TensorFlow	Keras/TensorFlow	NS3/MATLAB

to traditional methods with high computational complexity solved by optimization algorithms. Although these proposals need to gather an appropriate database to learn the optimal policy for resource allocation, using SDN can be a great aid in data gathering and system management (e.g., groups {3-4}). In groups {3-4}, the use of NFV increases the resource utilization and isolation between the slices.

B. EVOLUTIONARY ALGORITHMS

ML techniques are used for prediction, optimization, classification, etc. One of the machine learning methods used for optimization are evolutionary algorithms. Although some researchers do not accept these algorithms as part of ML techniques, the evolutionary algorithms in [60] and [61] have been introduced as a subset of ML methods. Evolutionary algorithms use mechanisms inspired by nature and solve problems through processes that imitate the behaviors of living organisms. In other words, biological evolution can be used as a learning process. First, a population of possible solutions is created and scored based on a fitness function to determine how good each generation is. Generations evolve over time until the solution will improve. One of the most popular evolutionary algorithms is the genetic algorithm (GA) [62], [63], [64].

1) RESOURCE SHARING

In the following, we review the proposals that address resource sharing. In these proposals, power and radio resources are allocated to the slices. Furthermore, the computational resources and transport bandwidth are also considered in [70] and [72], respectively. In [65], a hierarchical method for slicing in telemedicine services is provided

for physical resource block allocation. In this method, telemedicine services are divided into four categories and requests are classified in each category based on the delay and transmission rate required using the supervised radial basis function (RBF) neural network. Then, a genetic algorithm is used to allocate and adapt resources in each slice. The simulation result shows that the proposed method performs better than methods without hierarchical specification in terms of delay, bit rate, and utilization. In [66], an evolutionary method for PRB allocation is presented based on social relationships between different users connected to several network slices in a dynamic and evolutionary manner, aiming to maximize transmission bit rate and resource utilization. In this method, users are classified into different groups based on the similarities between them, and it is assumed that all users in a group need a similar service. The simulation result shows that the proposed method has a higher transmission rate, resource utilization, and request acceptance rate than [73] for data and video slices. In [67], a GA-based method is presented for allocating radio resources and power to users of the different tenants in a macro cell (consisting of several small cells) to improve energy efficiency and energy consumption and users’ acceptance rates. This method tries to assign the users to the appropriate BS under its tenant management to meet their QoS needs. The simulation results show that this method meets the set goals compared to the greedy method. In [70], an evolutionary algorithm based on the genetic algorithm is presented to allocate PRB and computational resources. In this method, the slicing strategy is encoded as a binary sequence for request-and-decision mechanism. Each gene represents the request for each slice, 0 means reject the request and 1 means accept the request.

TABLE 4. Summary of related work for evolutionary algorithms based on allocated resource types.

Ref.	Allocated Resource Types					
	Radio	Power	Computational	Storage	Caching Memory	Transport Bandwidth
[65], [66], [67], [68], [69]	✓	✓	-	-	-	-
[70], [71]	✓	✓	✓	-	-	-
[72]	✓	✓	-	-	-	✓

TABLE 5. Summary of related work for resource allocation in RAN slicing based on evolutionary algorithms.

Group ID →	Ref.				
	1	2	3	4	5
Features ↓	[65], [66]	[67], [72]	[70]	[68], [69]	[71]
Isolation	-	-	-	✓	-
Dynamic Power Allocation	-	✓	-	✓	-
Computational Complexity	High	High	High	High	High
Convergence Speed	Low	Low	Low	Low	Low
Energy Efficiency	-	-	-	-	-
Time Complexity	High	High	High	High	High
SDN	-	-	-	-	✓
NFV	-	-	-	-	✓
RAN Type	D-RAN	D-RAN	D-RAN	D-RAN	D-RAN
Mobility Management	-	-	-	-	✓
Dynamic Slice Creation and Management	-	-	-	-	-
Simulation Tools	NS3/MATLAB	-	-	MATLAB	MiniNet

The goal of the fitness function is to maximize utility. The simulation result for 2 slices shows that the proposed method has a higher resource utilization than greedy (accepting all requests), conservative (rejecting all requests of slice 1), and Opportunistic (rejecting all requests of slice 2) methods. The authors in [72] have proposed a method for network slicing in both RAN and core networks to allocate radio resources and power in the RAN, and transmission capacity in the CN based on a genetic algorithm. In this method, several BSs are considered, and each user can connect to one BS using one slice. Also, the access mobility management function (AMF) module is deployed in the CN and is responsible for the deployment and management of network slices. This scenario is considered as a matrix, the GA tries to find the optimal answer for this matrix by considering transmission bit rate and delay constraints. The simulation results show that the proposed method performs better than RSS-based (received signal strength) methods and greedy methods in terms of satisfaction degree, data rate, utilization, throughput, and accepted users.

2) ISOLATION

In addition to resource sharing, the isolation among slices is also considered in [68] and [69]. In [68] and [69], an evolutionary algorithm based on GA is proposed to allocate radio and power resources in RAN, aiming to maximize the bit rate for users in each slice. These methods isolate the radio resource of slices in neighboring cells from inter-cell interference for ensuring QoS for each slice. The simulation result shows that the proposed methods fulfill the desired objectives based on different simulation parameters.

3) RESOURCE VIRTUALIZATION AND MOBILITY MANAGEMENT

Compared to the proposals in Section V-B1, [71] has considered the virtualization and mobility challenges in addition to resource sharing for RAN slicing. In [71], a user-centric service slicing approach based on a genetic algorithm, SDN, and NFV is presented to allocate radio and power resources in RAN and computational resources in the core network by considering delay and transmission bit-rate. To address the delay in the user-centric scenarios, the separation of the control and data planes are used for signaling and managing the different flows. Also, different components of RAN and resources in the CN have been considered as the VNF. The simulation results show that the proposed method fulfills the requirement of video streaming and voice over IP slices.

4) LESSONS LEARNED

We summarize the reviewed proposals that use evolutionary learning for resource allocation in RAN slicing in Tables 4 and 5. The allocated resource types in each proposal are shown in Table 4. Also, we compare the proposals based on different features in Table 5. It can be learned from proposals in groups {1-5} that the use of GA methods for resource allocation can reduce the dependency of resource allocation methods on datasets and increase interaction with dynamic environments. Also, dynamic power allocation in groups {2, 4} decreases the power consumption and improves the channel conditions. In [71], the use of SDN and NFV facilitates RAN management and resource allocation to the slices. However, due to the high cost of computational and time complexity in these algorithms, the use of methods of sections V-C and V-D has been considered in recent years.

C. REINFORCEMENT LEARNING

Reinforcement Learning (RL) is a type of ML method that enables an agent to learn in an interactive environment using trial and error and using feedback from its actions and experiences. In reinforcement learning, when an agent performs an action in a particular situation, it receives a reward. In this type of machine learning, the goal of the agent will be to maximize the received reward in the long run. Although both supervised learning and reinforcement learning use mapping between inputs and outputs, unlike supervised learning, rewards and penalties are used as signals to improve the final performance of the system. The main difference between RL and other methods of ML is that, in reinforcement learning, the agent does not use prior knowledge for decision-making. The agent chooses an action among multiple actions in each state and, based on feedback from the system, it learns how good or bad an action is. In this way, complex decision-making problems can often be solved by providing the least amount of necessary information to solve the problem [74], [75], [76], [77].

1) RESOURCE SHARING

In the following, we review the proposals that address resource sharing.

- Addressing Radio and Power Allocation

In [78], an online network slicing solution called ONETS based on the multi-armed bandit (MAB) model was presented to maximize network slicing efficiency by accepting requests above the available capacity of the network. The simulation result shows that the proposed method has higher achievable multiplexing gains and outperforms FCFS, random, and $\epsilon - greedy$ scenarios.

In [79], a method for allocating radio resources and power in V2I communications and relay assisted cellular vehicle-to-vehicle (RAC-V2V) in UL and DL is presented to increase throughput, load balancing, and minimize energy consumption. Two layers are considered to allocate resources in several macro BS and small BS: Layer-1) Soft AC (SAC) for allocating resources to different BSs. Layer-2) An iterative algorithm called alternative slicing ratio search (ASRS) for allocating resources to RAC-V2V with considering QoS of users and channel conditions. The simulation results show that this two-layer resource allocation method, compared to different methods that used alternative algorithms instead of SAC or ASRS, has high resource utilization, transmission rate, and low energy consumption of vehicles, and fast convergence speed. In [80], an AC-based method called RAWs is proposed for Internet of Vehicles (IoV) by considering the load distribution between neighboring base stations. In this method, all BSs are connected to a controller and the controller is responsible for collecting environment information and RAN slicing. Furthermore, QoS measurement criteria are provided for users of delay-sensitive and delay-tolerant

slices. The simulation results show that RAWs with load distribution between neighboring BS reduces the computational cost and traffic load in a BS compared to Twin Delayed DDPG (TD3), RAWs methods without load distribution, and random allocation methods. Also, the proposed method has a higher transmission rate and higher resource utilization than the compared methods. In [81], a method called constrained discrete-continuous soft actor-critic (CDC-SAC) is presented, taking into account the queue length of packets and energy costs. In this method, a discrete action space is used for PRB allocation, while a continuous action space for power allocation is considered. The purpose is to select two suitable actions for these resources at the same time. Furthermore, the Lagrange coefficient method has been used to determine the power of UEs. Also, a replay memory has been used to increase the decision stability in CDC-SAC. The simulation results show that CDC-SAC has a better performance compared to DDPG, TD3, random (random action selection), and SAC algorithms in meeting the considered constraints. In [82], a method called DNAF is presented that uses deterministic policy gradient descent (DPGD) to prevent unnecessary calculations of Q values by considering the spectral efficiency (SE) of the channel and the QoE of users. Besides, DPGD only performs in continuous action space, so the proposed method uses k-nearest-neighbor (KNN) to find the nearest action in continuous space to prevent the recalculation of Q values. The simulation results for VoLTE, video, and uRLLC slices show that DNAF has a higher SE and QoE and converges faster than equal allocation and Q-learning methods. In [83], a ML-based method is proposed to allocate PRB to different slices by considering resource utilization and maximizing network throughput. To manage the underlying RAN nodes, a software defined RAN controller is used. The proposed method consists of four parts: 1) Using ML techniques (e.g., regression tree) to classify the demand of different users 2) Using a Decision Tree to predict the demand rate of each slice 3) Presenting an admission control model using the knapsack optimization problem 4) Tuning the allocated resources using DDPG. The simulation results show that the method has better performance in terms of resource utilization and network throughput compared to static slicing and uninformed random slicing.

In [84], a resource allocation method based on RL is proposed for allocating RBs to elastic and real-time applications in the smart grid for transmitting power data aiming to increase spectral efficiency and resource utilization. To solve the problem, a DQN method based on the $\epsilon - greedy$ is developed. Also, to address the overestimation problem and improve decision-making in DQN, the authors proposed Double DQN. The simulation results show that the DDQN method performs better than DQN in terms of computational cost and resource utilization.

In [85], first the resource allocation problem is modeled based on constrained MDP (CMDP) for allocating radio and power resources in the up-link mode of the RAN, aiming to reduce the packet latency, increase the battery life of UEs and increase the transmission rate of UEs for the three slices eMBB, mMTC, and uRLLC. Due to the dimensions of the CMDP problem, a DQN-based method has been proposed to reduce the dimensions of the problem. The simulation results show that this method outperforms the QSI-based (queue state information) method, the CSI-based (channel state information) method, and the random allocation method in terms of convergence speed, delay, transmission rate, and battery life of UE. In [86], a DQN-based method called RS-DRL (resource slicing DRL) is proposed to allocate radio resources and power in the RAN taking into account energy consumption and bandwidth for mobile, video, and vehicle slices. First, the resource allocation problem is modeled as a semi-MDP problem and then solved using DQN. The simulation results demonstrate that RS-DRL has a higher resource utilization and higher transmission rate than random and Q-learning methods. Also, the proposed method learns the optimal allocation policy better than the compared methods. In [87], network slicing is defined in three layers: core cloud, edge cloud (edge layer), and access units (local layer). The resources in the edge cloud and the access units are allocated to the users of slices. A mixed-integer programming method is proposed to allocate RBs and power resources in the local layer by considering QoE including latency, transmission rate, and interference. Also, a DQN-based method is proposed to estimate the users' requests and adjust the resources of slices in the edge layer. The simulation results show that the proposed method performs better in terms of the mentioned QoE factors compared to the fixed or dynamic (considering the priority and traffic of each slice) methods. In [88], the issue of radio and power allocation is modeled as a binary non-convex problem in the mixed-numerology interference environment taking into account channel state information, aiming to increase the service capacity for different slices. Then, a DQN-based method is used to overcome the computational complexity of the problem. The simulation results for different slices show that this method has a higher transmission rate and SIR close to the optimal method compared to static and optimal (selection of suitable sub-channel with exhaustive search) methods. In [89], a slice-based virtual resource scheduling scheme for eMBB and uRLLC based on PPO is proposed for allocating sub-carriers to uRLLC and eMBB slices, with the aim to minimize the latency for uRLLC packages. To maximize the rate and real-time response to uRLLC slices, the eMBB packets are punctured and the uRLLC packets are embedded in the eMBB packets. The simulation results show that the proposed method compared to the random

method (selection of sub-carriers to send uRLLC packages with equal probability and uniform distribution) and the aggressive method (random selection of sub-carriers) and the threshold proportional (puncturing of each eMBB package with a certain ratio) has less latency for uRLLC packages and a lower outage probability for eMBB packets. In [90], a DRL method based on DQN is proposed to assign the physical resource blocks (RBs) in the 5G/B5G systems with aiming to guarantee the quality of service (QoS) requirements for eMBB, uRLLC, and mMTC slices. To reduce the state-action sizes in DQN, the proposed method uses an elimination function to remove undesirable actions which cannot guarantee the QoS requirements for the desired slices. The simulation results show that the proposed method achieves a higher throughput performance and better resource allocation policies compared to the equal allocation, slicing based on regression tree, and vanilla DQN methods. In [91], a multi-agent method based on DQN called graph attention DQN (GAT-DQN) is proposed for allocating radio resources and power, taking into account spectrum efficiency and the requirements of different slices, the SLA satisfaction ratio (SSR) in several BS. In this method, each BS is managed by an agent and GAT is used for coordination between BSs. The distribution of BSs is considered as a graph and GAT is used to track temporal and spatial fluctuations of users' requests. The simulation results for eMBB, uRLLC, and VoLTE slices show that GAT-DQN has higher SE and SSR compared to hard slicing and DQN methods and has more resource utilization than the compared methods. Also, in [92], the authors evaluated the impact of using value-based methods such as DQN and value-based and policy-based methods such as A2C as the agent on GAT, and their results show that the use of GAT with different types of RL algorithms has a significant impact on improving resource utilization, SE, and SSR. In [93], a method using blockchain, Stackelberg game, RL, and SDN controller is presented for allocating the resources in the RAN to users to maximize the QoS of users, resource utilization, and InP profit by increasing user acceptance. Several BSs within a RAN communicate with each other using SDN controllers and exchange CSI with each other through the controllers. Also, the Stackelberg game is used to model how to buy and sell InP resources, and a blockchain-based broker is used to create a smart contract between buyers and sellers. Furthermore, Dueling DQN is used to learn the optimal policy of pricing and demands from the Stackelberg game problem. The simulation results for eMBB, uRLLC, and mMTC show that the proposed method, compared to methods that use DQN, GA, and Q-learning instead of DDQN, can meet all the objectives set for the intended slices.

The authors in [94] concentrate on network slicing based on RBs allocation in O-RAN. The authors first discuss

the challenges, features, and limitations of O-RAN, then, it is shown that using DRL can add data-driven, autonomous, and self-optimizing properties to the cellular network. The simulation results demonstrate that the Spectrum Efficiency in DRL is higher than the proportionally fair, water-filling, and round-robin allocation methods for eMBB, uRLLC, and mMTC slices. The authors of [95] have proposed an elastic method based on distributed game theory and RL to allocate resources in a macro-cell BS within O-RAN to meet the needs of three different classes of Industrial IoT. In this approach, the authors aimed to balance the age of information (AoI) and energy efficiency while maximizing service rates using synchronization between multiple devices. In this method, game theory is used to prioritize services based on devices and channel conditions. Also, AC is used along with a replay buffer (to increase the stability of AC decision-making) to decide on resource allocation. The simulation results show that this method has a better performance in balancing the mentioned parameters compared to policy gradient and PPO.

- Addressing Computational Allocation

Some proposals consider computational resources besides radio and power for resource allocation in resource management of RAN slicing. In [96], a reinforcement deep learning algorithm based on the DQN method is presented for allocating and managing resources in both RAN and core network, taking into account users' QoE and channel performance. Resource allocation in the proposed method is based on the priority between the slices and a minimum bandwidth guarantee for users in slices. In this method, two neural networks are used to accurately estimate Q-values and select the best action between slices. The simulation results show that this method is better than estimation-based methods in terms of accuracy and speed of decision-making for resource allocation, computational cost, and interaction with the dynamic environment for the VoIP, video, and uRLLC slices. In [97], a DQN-based method is proposed to allocate RBs, power, and computational resources in a gNodeB for down-link communications by considering prioritizing, throughput, CPU usage, and latency of each user's request per slice. The authors first model the resource allocation problem as an optimization problem called myopic, and then solve it using the DQN method. The simulation results show that the DQN algorithm has optimal resource utilization and lower latency compared to myopic, random allocation, and FCFS allocation methods. In [98], a DQN-based method is proposed for allocating RBs, transmission power, and computing resources to users' devices in a gNodeB for eMBB, uRLLC, and mMTC slices, taking into account prioritizing slices and increasing acceptance of user

requests and reducing packet latency and increasing CPU utilization. The simulation results show that the proposed method fulfills the set goals compared to other allocation methods such as myopic, greedy, random, and FCFS. In [99], first, an auction mechanism based on game theory is used to allocate radio resources in the BS and computational resources in the MEC between several SPs. An SDN-orchestrator is responsible for performing auctions between different SPs and its aim is to increase the expected long-term payoff. In addition to considering the constraints of radio resources and channel allocation to different slices, the energy constraint of each UE and AoI (Age of Information) received data packets from mobile services are also considered for resource allocation. To address the lack of basic knowledge about the environment and the very large state space, a DQN-based method has been proposed. The simulation results show that the number of transmitted packets is higher in the DQN method compared to Channel-aware (channel allocation at the beginning of each time-slot), Queue-aware (calculation of queue time or channel usage) and Random (random assignment of each MU to one channel) methods. In [100], the system model is considered as a cluster with DU as the central server and several RUs as the cluster subset. Radio resources in the RU and computational resources in the DU (which perform radio signaling operations) are considered to allocate for slices by considering energy consumption in the DU. Because the cost of creating the required resources in DU is high due to high energy consumption and its delay, the required resources for each slice in RU must be provided. Therefore, the resource allocation problem in O-RAN is expressed as a 2D bin packing problem. The 2D bin packing problem is a method in which the resource allocation problem is converted to a Bin problem that aims to minimize one dimension of this Bin. Since this is an NP-hard problem, a DQN-based method has been proposed to solve it. The simulation results show that the proposed DQN method has lower latency, lower computational costs, lower energy consumption, and higher resource utilization than three methods including the heuristic virtual resource allocation algorithm (HVRAA) [101], the lego heuristics for bin packing [102], and the vanilla MCTS using Monte-Carlo. In [103], a resource allocation method is presented for network slicing using a cluster of F-UEs in a F-RAN. These F-UEs synchronize with each other through an edge coordinator (EC). EC decides to allocate resources/tasks for each service in each F-UE or refers them to the cloud. In this proposal, two slices, called cloud and edge slices, with radio resources and resources needed for IoV and smart city applications are considered that can be assigned to users. First, the resource allocation and task assignment problems are modeled as an MDP problem. The problem

is solved by a DQN method called DQN-EC. To measure the quality of service in DQN-EC, the authors use a measure called Key Performance Indicator, which is a combination of grade of service (GoS) and utilization. GoS is the ratio of the number of high-utility accepted requests to the total number of high-utility requests, and Utilization defines the ratio of the amount of used resources to total resources in a specified period of time. The simulation results show that DQN-EC, compared to the serve-all-utilities (SAU) methods that serve all requests if resources are available - serve-high-utilities (SHU), which serves high-utility requests when resources are available, and Q-learning, has higher user GoS and resource utilization and learns resource allocation policies faster than the other methods. The authors in [104] have proposed a DRL-based method for allocating radio and power resources in F-RAN for both orthogonal and multiplexed sub-channels. Also, the computational resource is considered as an NF in cloud or fog to allocate to users of slices. In this proposal, F-RAN is divided into three layers: cloud computing, network access, and terminal, which includes the BBU pool in the cloud layer, distributed RRHs in the network layer, and UE antennas in the terminal layer. The resource allocation problem and mode selection (choosing between cloud and fog for resource allocation) in F-RAN are modeled as a mix-integer-programming problem, then both problems are solved using DQN. The simulation results with different parameters show that the proposed algorithm has a lower latency and lower energy consumption compared to particle swarm optimization (PSO) and Q-learning methods and converges faster.

- Addressing Storage Allocation

Furthermore, some proposals allocate storage along with radio, power, and computational resources to different slices with regard to their needs. In [105], a resource allocation method based on deep dueling Q-learning was introduced to allocate spectrum, computing, and storage resources to various users. Deep dueling Q-learning does not update the Q-value for unnecessary actions, which achieves the optimal policy faster than the conventional Q-learning algorithm. The simulation results for 3 classes of slices including utilities (class-1), automotive (class-2), and manufacturing (class-3) show that dueling DQN has higher revenue and converges faster than Q-learning, double DQN, and DQN methods. In [106], an DDPG-based method is proposed that is responsible for resource allocation to different tenants in the RAN. In the scenario, several BSs form a cluster in the RAN, and MEC are considered for allocating computing and storage resources to delay-sensitive users. An SDN controller manages the RAN and is responsible for allocating radio and power resources in the cluster and computing and storage resources in

the MEC. In this method, tenant type and the desired slice, and the required resources for users are determined by Internet Protocol (IP) addresses. The simulation results show that the proposed method, in comparison with fixed allocation, heuristic strategy and optimum (frequent reconfiguration of slices) allocation, has better performance in terms of bandwidth utilization and QoS characteristics. In [107], a method is presented to allocate radio, power, processing, and storage resources to IoT users in uRLLC slices with delay-sensitive or processing priority in F-RAN. Two scenarios, infinite-horizon (IH) MDP and finite-horizon (FH) MDP are considered for resource allocation. In the IH scenario, there is no time limit for resource allocation, and increasing the delay will increase the penalty. While in the FH scenario, there is a time limit for allocating resources to users with priority and delay constraints. At first, the resource allocation problem is modeled as an MDP problem and then is solved by different RL methods such as Q-learning (QL), SARSA, Expected SARSA (ESARSA), and Monte Carlo (MC). The simulation results for different RL methods show that the RL methods learn the optimal policy faster and are more efficient than a method that uses a time threshold for allocating resources to users in F-RAN. In [108], a two-stage Q-learning method called TSQL is presented, which aims to allocate resources in C-RAN. C-RAN is divided into two layers, the upper layer is CU/DU, and the lower layer is RRU. NFVs are mapped at the upper layer. Also, radio and power resources and the appropriate RRU are determined at the lower layer. Two Q-learning algorithms are used to solve the resource allocation problem in both layers. The simulation results show that the TSQL method outperforms in terms of the number of accepted users, resource utilization, and transmission rate compared to the methods such as minimum cost function chain deployment, maximum SINR radio resource allocation, function chain deployment based on Q-Learning.

2) MOBILITY MANAGEMENT

Compared to the proposals in Section V-C1, some proposals address mobility management besides resource sharing and allocate power and radio resources to the considered slices. In [109], a method called LSTM-DDPG is provided for allocating the resources in two small and large time scales in the VANET. LSTM has been used to estimate the required resources on a large-time scale using the history of resource allocation data. AC has been used as a well-known DDPG algorithm to allocate resources on a small-time scale based on changes in mobility and channel conditions. The simulation results for the three considered slices showed that the LSTM-DDPG method is more stable in providing services to users than the DQN, A2C, DDPG, and demand-based

algorithm methods. In [110], a method based on RL and Deep Learning for allocating radio and power resources to vehicles is presented in C.V2X mode 4 standards. In this method, each vehicle in each slice automatically and independently of other vehicles selects the appropriate PRB for V2V communication and sends its observations of the environment and channel conditions to the eNB. The eNB adjusts the network slice configuration using AC based on the current observations sent by vehicles and the predicted results through current and previous observations using LSTM. The simulation results show that the method presented in this paper has less latency for the transmission packets compared to deep recurrent Q-Network (DRQN).

3) ENERGY EFFICIENCY

In addition to resource sharing, some proposals have focused on energy efficiency. The proposals allocate the radio and power resources to the slices. In [111], a method called CNDDQN based on DDQN (Double Deep Q Network) is proposed for resource allocation in cognitive RAN slicing. In this method, radio and power resources are allocated to eMBB and uRLLC slices by considering SE and QoE of users (which is defined as the rate of sent packets to the total number of packets for each user). There is a minimum transmission rate limit for eMBB users and a minimum transmission delay limit for uRLLC users. The simulation results show that CNDDQN converges faster than Q-learning and DQN and solves the overestimation problem in DQN. The authors have developed a method using RL and SDN for allocating radio resources on the smart grid in [112]. In this method, the SDN controller is responsible for managing the access network to allocate radio resources to different slices and applies the RL algorithm, aiming to increase QoE of users and channel efficiency by considering the priority of each slice and the achievable data rate. In this method, transmission power is allocated based on the priority level of each slice. The simulation results for three slices with different generation rates show that both DQN and Dueling DQN algorithms perform better in terms of resource utilization compared to fair allocation methods. In [113], a DRL-based method is proposed for the dynamic allocation of power resources and bandwidth between the users and the RRHs taking into account the user's different unicast and multicast requirements in TV broadcast in a C-RAN. At first, the problem is molded as a convex optimization problem, then, it is solved by DQN. To predict the traffic demand for each slice in RRHs, LSTM is used. The simulation result shows that the proposed method outperforms the fair allocation method in terms of energy efficiency, number of accepted users, and energy consumption.

4) ISOLATION

Some proposals have addressed resource sharing and isolation among slices. Compared to [114], [115], [116], and [117]

that allocate radio and power resource, the caching memory is also allocated in [118]. In [114], a two-tier method is provided for allocating radio resources and power in a BS to users of eMBB and uRLLC slices, considering a large time-scale QoS and a small time-scale SE. To prevent reconfiguration of slices and isolation between the slices, double DQN is used to provide QoS of users on a large time-scale in the slices in the top layer. At the lower layer, DDPG is used to allocate radio resources and power to different UEs. To reduce the state and action space in this algorithm, a mixed-integer programming method is proposed, which is solved using Lagrange coefficients. The simulation results show that this method has a better QoS of users for the slices compared to [119]. The authors in [115] have developed a model based on an advantage actor-critic (A2C) algorithm and LSTM to improve system performance for mobile users. In this method, RBs are assigned to users of each slice based on channel conditions and SLA satisfaction ratio (SSR). LSTM is used to detect the temporal relationship between user requests in each slice during the time to make optimal resource utilization. The simulation results for the VoLTE, eMBB, and uRLLC slices show that this method has a higher resource utilization and spectral efficiency than A2C, DQN, GAN-DQN, and Hard slicing methods (allocation of one-third bandwidth per slice), and the SSR for each slice is less violated in this method. In [116], a DQN method is presented for radio resource allocation for rate-based and delay-based slices taking into account the isolation of each slice. In this method, the transmission power is considered fixed in each slice. An innovative algorithm is used to reserve radio resources for a specific period of time. In addition, an optimization algorithm is proposed to determine the required resources for rate-based users that is solved using the ADMM method based on Lagrange coefficients. The simulation results show that the DQN method has a higher resource utilization and better SLA satisfaction than methods in [120] and [121]. In [117], a DQN-based method using ϵ -greedy is presented to allocate radio resources and power to the C-RAN taking into account constraints of isolation and transmission rate. In this model, the C-RAN includes several RRHs, and each link between RRH and users is considered as an agent. The simulation results show that this method fulfills all requirements of the considered slice basis on different simulation parameters. In [118], a resource allocation method is proposed to users of both hotspot and V2I (vehicle to infrastructure) slices in F-RAN. In this method, several radio resource units (RRUs) (to form F-APs) have been considered in the F-RAN to allocate communication and caching memory resources, and a cloud server decides on mode selection for caching information in F-RAN or the cloud server and allocates radio and power resources. The problem is stated as an integer optimization problem. Due to the computational and time constraints, a DQN-based method is presented that considers constraints on the transmission rate, caching, isolation of each slice, and the number of connected UEs to each AP.

In this method, the power allocation problem is expressed as a sub-problem of the Perron Frobenius theorem and proximal theory, which is solved using an iterative method based on Lagrange coefficients. The simulation results show that the DRL method, compared to the three methods that use Least Recently Used (LRU), Least Frequently Used (LFU), and First In First Out (FIFO) for caching data, has better performance in terms of resource utilization and cache hit rate.

5) RESOURCE VIRTUALIZATION

Compared to the proposals in Section V-C1, some proposals address resource sharing and resource virtualization. [122] and [123] allocate radio, power, and computational resources to the slices. The authors in [122] introduce a framework for E2E resource allocation for IoT slices in the smart city. Then, they use a combination algorithm using DQN and LSTM for traffic prediction and resource allocation to different slices. The simulation results for eMBB and uRLLC slices show that this method achieves a higher resource utilization compared to methods in [124] and [116]. In [123], a method called dynamic CU-DU selection technique A2C (DSCD-A2C) is provided to allocate radio resources and network functions (NFs) taking into account latency, slicing priority, type of traffic, and processing power in O-RAN for video, augmented reality, and V2X slices. In this method, two A2C algorithms are used, one is used to allocate radio resources to UEs in the RU and another is used to allocate NFs in the DU or CU according to the users' SLA and the QoS requirements. The simulation results show that DSCD-A2C, compared to NF-DU (in which the DU always handles the NF) and NF-CU (in which the CU always handles the NF), can significantly improve the latency, packet delivery ratio, and throughput.

Some proposals consider the radio, power, storage, and computational resources for resource allocation. In [125], a method using DQN and NFV is proposed for spectrum/power allocation, computational and storage allocation in RAN. At first, the resource allocation problem is stated as CMDP. Then, using an adaptive constrained reinforcement learning algorithm based on Internal-point Policy Optimization (IPO), is solved. The simulation results for the Video, VoLTE, and uRLLC slices show that this method has high throughput and less delay compared to the one-third equal allocation, user-number-based allocation, packet-number-based allocation, and traffic-demand-based allocation methods. In [126], a deep intent-based network slicing system using SDN and VNF based on GAN deep learning was designed to slice and manage the core network and RAN resources. To implement the SDN-based RAN controller, FlexRAN is used to manage and control the slices in RAN, and OpenStack is used to implement the virtual network in core network. The simulation result shows that the proposed method allows network operators to automate the configuration, slice creation, and management process. In [127], a method called CLARA, based on the

constraint MDP (CMDP) problem, is provided to allocate radio resources in a BS for video, VoLTE, uRLLC slices with considering instantaneous constraints (e.g., delay and interference), and long-term constraints (e.g., throughput and outage probability). In this method, the NFV infrastructure is used to convert the physical resources to virtual resources by a virtualization layer. In the simulation results, the CMDP problem is solved with three RL algorithms including IPO, PPO, and PPO+softlayer. The results show that CLARA meets the QoS requirements for each slice compared to the one-third equal, user-number-based, packet-number-based, and traffic-demand-based resource allocation methods. The authors have emphasized the Artificial Intelligence (AI) capability for use in network slicing in [35], and present an RL method based on actor-critic (AC) for allocating resources in RAN and core network to demonstrate the advantages of AI. In this method, Docker is used for allocating storage resources, CPU, etc., in the core network. Also, RBs are considered for allocation in the RAN. The aim of the proposed method is to strike a balance between latency, decoding errors, and CPU usage to minimize the operating costs of CPU reservations and to maximize system performance by reducing decryption error rates and latency. The simulation results show that this method has higher CPU efficiency and lower latency than a static method. In [128], a method called prioritized twin delayed distributional deep deterministic policy gradient (D-TD3) based on AC is presented, aiming to allocate computing and storage resources as VNFs and CPUs in distributed APs in one RAN and a central server. The central server is responsible for allocating resources in the APs by considering the accepted user rate, CPU efficiency, and transmission data rate and transmission delay and power consumption, and the cost of creating VNFs in APs. In D-TD3, a prioritized buffer and memory have been used to reduce network instability in the learning process and better decision-making in the AC method. The simulation results show that this method converges faster than the double DQN and AC methods and outperforms in terms of the mentioned parameters. Also, in [129], a method called twin-delayed double-Q soft Actor-Critic (TDSAC) is provided for allocating CPU resources and controlling power consumption in DU-CU based on the C-RAN architecture. To increase the stability and improve the decision quality in TDSAC, double DQN is used to tune the AC parameters. The simulation results show that TDSAC manages energy consumption better than DDPG, TD3, and SAC methods and increases CPU utilization optimally.

Also, the radio, power, transport bandwidth, and computational resources are allocated in [130] and [131]. In [130], a DRL-based method is presented for resource allocation in three layers: access, transport bandwidth, and core. OpenAirInterface (OAI) is used to implement radio resources in the access layer, SDN switches are considered as bandwidth resources in the transport layer, and GPUs are used for computing resources in the core layer. The simulation results show that this method has a higher resource utilization

TABLE 6. Summary of related work for reinforcement learning methods based on allocated resource types.

Ref.	Allocated Resource Types					
	Radio	Power	Computational	Storage	Caching Memory	Transport Bandwidth
[78], [79], [80], [81], [82], [84], [85], [86], [87], [88], [89], [90], [91], [92], [83], [93], [94], [95], [109], [110], [111], [112], [113], [114], [115], [116], [117], [135]	✓	✓	-	-	-	-
[96], [97], [98], [99], [100], [103], [104], [122], [123], [134]	✓	✓	✓	-	-	-
[130], [131]	✓	✓	✓	-	-	✓
[105], [106], [107], [108], [125], [126], [127], [35], [128], [129]	✓	✓	✓	✓	-	-
[118]	✓	✓	-	-	✓	-

compared to the proportional method. In [131], an end-to-end method called constraint-aware deep reinforcement learning (CaDRL) is presented to allocate resources in a BS within RAN, MEC, transport layer, and server. In this paper, DDPG is used to solve the CMDP problem, and a joint learning method including online and offline learning is used to increase the efficiency of the DDPG algorithm. This method has been implemented using OpenAirInterface LTE for RAN, OpenDayLight-based SDN for transport layer, and CUDA GPU for computational resources. The simulation results show that CaDRL has less latency, higher resource utilization, and faster convergence speed compared to methods in [132] and [133].

6) RESOURCE VIRTUALIZATION AND ISOLATION

In [134], a resource allocation method using RL, NFV, and SDN is proposed to address resource sharing, resource virtualization, and isolation among slices. The authors allocate radio, power, and computational resources as NFs in BSs based on traffic and network dynamics by the proposed heuristic algorithm. A heuristic method based on 0-1 multiple knapsacks is presented for resource allocation to users of shape-based slices while guaranteeing the required transmission rate of these users. Then, to increase resource utilization and ensure QoS and isolation of each slice in each BS, they use Dueling DQN to adapt the assigned resources according to the users' requirements of each slice. An SDN controller is used to communicate between different agents in different BSs. The simulation results show that DDQN has a higher convergence speed, higher resource utilization, and higher isolation than Q-learning, [120], [121] methods.

7) ALGORITHMIC ASPECTS OF RESOURCE ALLOCATION

In addition to resource sharing, [135] addresses algorithmic aspects of the resource allocation method. In [135], a multi-agent RL (MARL) method based on DQN is proposed for allocating radio and power resources to different tenants in multi-cell environments with aiming to guarantee users' SLAs and maximize resource utilization. In this method, each RL agent is assigned to a tenant, and each tenant centrally decides from observation of the environment. The simulation results show that the MARL method converges faster than

the single-agent method and guarantees the required rate for different slices of various tenants.

8) LESSONS LEARNED

We summarize the reviewed proposals that use reinforcement learning for resource allocation in RAN slicing in Tables 6 and 7. The allocated resource types in each proposal are shown in Table 6. Also, we compare the proposals based on different features in Table 7. In RL methods, GAN-DQN ([126]) and GAT-DQN ([91], [92]) methods have been used to improve decision making in DQN methods. Also, Double DQN ([84], [111], [114]) and Dueling DQN ([93], [105], [112], [134]) have also been used to improve decision-making and overestimation in DQN-based methods. Due to the use of AC methods in [79], [80], and [81], decisions may fluctuate, so A2C and DDPG are used to address this problem in [109], [82], [114], [115], [83], [106], and [131]. To increase the convergence speed and decrease the computational and time complexity in resource allocation methods in RAN slicing, the methods in V-D are presented. The methods in Sections V-C5 and V-C6 also use VNF for the virtualization of resources to allocate to users of different slices in the RAN. Some reviewed proposals in Section V-C use SDN to manage resource allocation or implement the transport layer.

It can be learned from methods in groups {19-21, 32-34} that reserving the resources for each slice can guarantee isolation among slices over a specified period of time. Also, dynamic power allocation according to channel conditions in groups {22-34} can reduce the energy consumption of users' devices, reduce interference, increase energy efficiency, and increase user acceptance rate due to better resource utilization. Due to the centralized nature of D-RAN and the concentration of all resources in the RAN, most of the reviewed proposals in Section V-C will be applicable in D-RANs.

It can be learned from [129] that by tuning the AC parameters at specified times, the decision fluctuation in [35] and [128] can be reduced. In the Q-learning-based method ([108]), if the state and action spaces are very large, then the size of the Q-Table will be very large. To overcome this problem, other RL methods use a neural network to estimate the Q-Value. Compared to the reviewed methods in D-RAN, if the slices need more resources, the resources in

TABLE 7. Summary of related work for resource allocation in RAN slicing based on reinforcement learning.

GID	Ref.	Features											
		Isolation	DPA	CC	CS	EE	TC	SDN	NFV	RT	MM	DSCM	ST
1	[35]	-	-	High	Low	-	High	-	-	C-RAN	-	-	Docker/SDR
2	[82], [84], [89], [91], [92], [96], [105]	-	-	High	Low	-	High	-	-	D-RAN	-	-	-
3	[78]	-	-	High	Low	-	High	-	-	D-RAN	-	-	NS3/MATLAB
4	[103], [107]	-	-	High	Low	-	High	-	-	F-RAN	-	-	-
5	[94], [100]	-	-	High	Low	-	High	-	-	O-RAN	-	-	Real Expr.
6	[128], [129]	-	-	High	Low	-	High	-	✓	C-RAN	-	-	Python
7	[83], [106]	-	-	High	Low	-	High	✓	-	D-RAN	-	-	MATLAB
8	[99]	-	-	High	Low	-	High	✓	-	D-RAN	-	-	-
9	[130]	-	-	High	Low	-	High	✓	-	D-RAN	-	-	OpenDayLight/ OAI
10	[125], [127], [122]	-	-	High	Low	-	High	-	✓	D-RAN	-	-	-
11	[79], [90], [97]	-	-	High	Low	-	High	-	-	D-RAN	-	-	Python
12	[126]	-	-	High	Low	-	High	✓	✓	D-RAN	-	-	OpenStack/ FlexRAN/OAI
13	[109]	-	-	High	Low	-	High	-	-	D-RAN	✓	-	-
14	[110]	-	-	High	Low	-	High	-	-	D-RAN	✓	-	MATLAB/ Keras/Python
15	[95]	-	-	High	Low	-	High	-	-	O-RAN	-	-	Python
16	[131]	-	-	High	Low	-	High	✓	✓	O-RAN	-	-	CUDA/OAI /FlexRAN
17	[123]	-	-	High	Low	-	High	-	✓	O-RAN	-	-	Pytorch
18	[93]	-	-	High	Low	-	High	✓	-	D-RAN	-	-	Python
19	[115]	✓	-	High	Low	-	High	-	-	D-RAN	-	-	-
20	[116]	✓	-	High	Low	-	High	-	-	D-RAN	-	-	MATLAB
21	[134]	✓	-	High	Low	-	High	✓	✓	D-RAN	-	-	Python
22	[112]	-	✓	High	Low	✓	High	✓	-	D-RAN	-	-	Python
23	[111]	-	✓	High	Low	✓	High	-	-	D-RAN	-	-	-
24	[113]	-	✓	High	Low	✓	High	-	-	C-RAN	-	-	-
25	[85], [87]	-	✓	High	Low	-	High	-	-	D-RAN	-	-	-
26	[108]	-	✓	High	Low	-	High	-	✓	C-RAN	-	-	-
27	[80]	-	✓	High	Low	-	High	✓	-	D-RAN	-	-	MATLAB
28	[86], [88]	-	✓	High	Low	-	High	-	-	D-RAN	-	-	MATLAB
29	[104]	-	✓	High	Low	-	High	-	-	F-RAN	-	-	-
30	[81], [98]	-	✓	High	Low	-	High	-	-	D-RAN	-	-	-
31	[135]	-	✓	Low	Low	-	Low	-	-	D-RAN	-	-	-
32	[118]	✓	✓	High	Low	-	High	-	-	F-RAN	-	-	-
33	[114]	✓	✓	High	Low	-	High	-	-	D-RAN	-	-	MATLAB/ Python/Keras
34	[117]	✓	✓	Low	Low	-	Low	-	-	C-RAN	-	-	OpenAI/Gym

DPA: Dynamic Power Allocation, CC: Computational Complexity, CS: Convergence Speed, EE: Energy Efficiency, TC: Time Complexity, GID: Group ID, RT: RAN-Type, MM: Mobility Management, DSCM: Dynamic Slice Creation and Management, ST: Simulation Tools, SDR: Software Defined Radio

the DU or CU can be used in the form of NF to allocate to users of slices in C-RAN and O-RAN based on the requirements of slices such as latency and priority such as [100]. In the reviewed methods for O-RAN (e.g., [94], [100], [123]), NN have been used to overcome the size of Q-Table in Q-learning. To increase decision stability in AC [95], we can use a memory buffer to eliminate inefficient decisions or use A2C like [123].

It can be learned from the reviewed proposals methods for F-RAN (e.g., [107]) that the service provider can allocate resources to delay-sensitive slices with a minimum delay using the F-RAN architecture. In F-RAN, we can keep the content close to users by caching the needed content in F-UEs, and minimize the access delay for users. Also, we can provide the required resources for the slices from the F-UEs, and minimize the processing delay for users. In F-RAN, there are several entities, so the use of a coordinator between these entities will lead to better control and management of F-RAN [103]. In [118], the constraint of slice isolation can prevent re-configuring the resources of slices during a specified time. In [118] and [104], one of the goals of the proposed method is to maximize energy efficiency, which is achieved by dynamic power allocation.

D. DISTRIBUTED LEARNING

In recent years, the use of distributed methods (such as transfer learning and federated learning) in ML has been

considered due to the common challenges in centralized methods such as centralization, security, privacy, and time and computational complexity. In distributed methods, a ML model runs distributed on different devices, or the combination of several agents in different devices is used to implement the desired model [136], [137], [138], [139].

1) RESOURCE SHARING AND ALGORITHMIC ASPECTS OF RESOURCE ALLOCATION

Some proposals address resource sharing and consider the algorithmic aspects of the resource allocation method. In [140] - [141], radio and power resources are allocated in a RAN. In [140], the convergence speed of DRL algorithms for the allocation of radio resources in different BSs has been investigated. To increase the convergence speed of agents in the system, a transfer learning-based method for DRL algorithms is proposed, in which expert agents in a BS transfer their knowledge to new BSs (which enter the system). The simulation results for different TRL-based algorithms compared to allocation methods based on user-number-based, traffic-demand-based, and hard slicing (allocation of resources equally between slices) show that DRL+TL methods have high convergence speed. In [142], a DQN-based method is proposed to assign RBs to network slices. In order to increase the scalability, the proposed method uses Ape-X to create parallel agents when increasing the number of slices. To manage and coordinate between slices,

a network slice controller is used. The simulation results show that the proposed method converges faster than the methods without slicing, hard slicing (allocation of RBs between slices equally), demand-based (allocation of RBs based on received packet rate in each slice), and DQN without Ape-X. Furthermore, the proposed method has a higher resource utilization than the other compared methods. In [143], a federated learning model called FL-CA was introduced that incorporates actor-critic (AC) to radio and power allocation. Federated learning trains a shared network model across many participating edge devices while keeping all the training data locally. In this method, the actor network is used for sharing weights and gradients between the shared network model and edge devices. The simulation result shows that FL-CA performs better than AC and greedy power allocation methods in terms of spectrum efficiency. In [144] and [145], two methods based on multi-branch dueling Q-network (MBDQN) are proposed for allocating radio resources to solve inter-numerology interference and maximize network throughput. First, the resource allocation problem is modeled as a non-convex integer problem and then solved by MBDQN. In this method, each sub-channel is managed by an agent and the sub-actions of the agents form a general action. The simulation results show that these methods increase the utilization of radio resources and network throughput compared to other allocation methods such as the optimal method (exhaustive search of all sub-channels to find the appropriate sub-channel), single branch allocation and Round Robin. In [146], a Distributed Deep Q-Network (DDQN) based on the generative adversarial network (GAN) using SDN is proposed to improve the SSR (SLA Satisfaction Ratio) of the users and spectral efficiency (SE) of the channel in each slice. SSR represents the rate of exchanged packets between the user and BS taking into account the delay constraint and the minimum transmission rate constraint in both down-link and up-link modes. In this method, the SDN controller is used for managing and coordinating between slices and resource allocation in each BS. The simulation results for VoLTE, video, and uRLLC show that the two GAN-DDQN and Dueling GAN-DDQN algorithms improve SE and SSR quality and resource efficiency compared to the DQN algorithm. In [141], a multi-agent method based on Q-learning called SARA is used to solve the multi-radio access technology problem, to maximize network throughput while ensuring the QoS of users in various slices. The resource allocation problem is modeled as a semi MDP and then is solved by SARA. In this method, each access is managed by an agent which is connected to a controller to assign a UE to feasible access. The simulation results show that SARA performs better in a time-varying network environment than conventional approaches like Q-Learning, Monte Carlo Tree Search (MCTS), and [147] in terms of QoS parameters such as throughput and bandwidth.

In addition to radio and power resources, computational resources are also allocated in [148]. The authors of [148] have proposed a multi-agent method called knowledge

transfer-based resource allocation (KTRA) to allocate PRB and computational resources for two slices, uRLLC and eMBB, to minimize packet latency and task processing time. In this method, each BS acts as a DQN agent and exchanges its knowledge with other agents to increase the convergence speed and fast discovery of the environment. To prevent sending tasks to the cloud, the MEC has been used to allocate computing resources and reduce task processing time. The simulation results show that KTRA has lower packet delay and higher convergence speed compared to the QLRA method (which is similar to the KTRA method without knowledge exchange between agents).

Compared to [148], radio, power, and cache memory are allocated in [149]. In [149], a hierarchical and multi-agent method based on transfer RL (TRL) is presented for allocating the resources in multiple BSs of a RAN. In this method, two methods of Q-value transfer RL (QTRL) and action selection transfer RL (ASTRL) have been used to decide on resource allocation to improve the quality of current decisions using the acquired knowledge. Also, each BS works as an independent agent to allocate resources and shares its prior knowledge with other BSs. The proposed method consists of two layers: 1) global resource manager (GRM), which is responsible for executing the RL algorithm in the agents, and 2) slice resource managers (SRMs), which distribute radio resources between devices and manage user content in cache. The simulation results show that this method has less packet latency and higher throughput and faster convergence speed for uRLLC and eMBB slices than the model-free Q-learning, model-based priority proportional fairness and time-to-live (PPF-TTL) methods.

2) DYNAMIC SLICE CREATION AND MANAGEMENT

In addition to addressing the listed challenges in Section V-D1, the methods in this section address the dynamic creation and management of slices. In [150], a multi-agent method based on Q-learning called COQRA for inter-slice PRB and power allocation is proposed for uRLLC and eMBB slices to minimize packet latency. In this method, each slice acts as an agent and exchanges Q values with other agents through the SDN controller to coordinate the decisions. The simulation results show that COQRA has lower latency and packet drop rate compared to Nash Q-learning (NQL) and Latency Reliability-Throughput Q-learning (LRTQ) techniques.

In [151], a framework is proposed for resource allocation according to 3GPP and O-RAN specifications. Also, a multi-agent DQN method has been used to allocate radio, power, and computational resources in O-RAN to each slice using the geographical features of each cell. Each slice is considered as an agent and the service management and orchestration (SMO) layer manages and coordinates the agents. The simulation results show that the use of DQN increases the user acceptance and quality of SLA in each mobile network operator compared to static allocation methods.

3) MOBILITY MANAGEMENT

The main goal of [152] is to address resource sharing and the algorithmic aspects of the resource allocation method by considering the mobility of nodes for improving the resource utilization. In [152], a multi-agent method based on DQN is proposed for radio and power allocation in V2V and V2I communication links, aiming to increase the utilization of the link. In this method, each link manages with an agent, and the agent interacts with an unknown environment and tries to assign the resources in a competitive environment with other agents. The simulation results for MARL (multi-agent RL) and SARL (single-agent RL) methods show that MARL efficiently increases resource utilization.

4) ISOLATION

[119] focuses on addressing resource sharing, the algorithmic aspects of the resource allocation method, and isolation among slices. In [119], an intelligent resource planning scheduling (iRSS) for RAN slicing was proposed by combining deep learning for decision making on a large time-scale and reinforcement learning for decision making on a small time-scale to guarantee isolation of the slices. In this paper, long short-term memory (LSTM) and A3C are used to predict the required RBs on a large time-scale in RAN slices and allocate the PRB on a small time-scale dynamically, respectively. It is assumed that the allocated power to users is constant and the channel condition is not considered in the resource allocation. Using distributed and parallel A3C in the method has reduced the time and computational complexity.

5) ENERGY EFFICIENCY AND DYNAMIC SLICE CREATION AND MANAGEMENT

Compared to the method in Section V-D4, [40] addresses other challenges such as energy efficiency and dynamic slice creation and management as well. To address dynamic power allocation and energy efficiency problems in [119], a method called EE-DRL-RA is proposed in [40]. EE-DRL-RA uses a collaborative learning framework that includes A3C for decision-making on resource allocation on a small time-scale, and SBiLSTM for decision-making on resource allocation on a large time-scale. Also, RBs and power allocation for rate-based users are formulated as a non-convex optimization problem and solved by an efficient iterative algorithm called EE-PA. In EE-DRL-RA, each slice has been considered as an agent and decides independently on resource allocation to its users. The simulation results show that EE-DRL-RA outperforms compared to [119], [119]+EE-PA methods in terms of convergence speed, the accuracy of LSTM, energy efficiency, the number of accepted users, and isolation degree.

6) RESOURCE VIRTUALIZATION AND DYNAMIC SLICE CREATION AND MANAGEMENT

The authors in [153] address resource virtualization and dynamic slice creation and management, along with the

listed challenges in Section V-D4. Radio, power, transport bandwidth, and computational resources are allocated in this method to the slices. In [153], a method called OnSlicing is provided for end-to-end resource allocation in four domains including radio access, transport, core, and edge for UL/DL communications. This method has two parts, the first part includes an orchestrator and the second part is a manager. In this approach, each slice is managed by a PPO agent and the orchestrator is responsible for coordinating the agents. Also, the manager includes four domain managers called RDM, TDM, CDM, and EDM, and its task is to collect information for agents and implement configuration policies of slices in the system. In this method, to ensure isolation between the slices in the RAN domain, PRB resources are exclusively allocated to the slices and in other domains, the isolation of the slices is ensured by using virtualization. The simulation environment is implemented using OpenAirInterface for RAN, OpenDayLight SDN Platform for transport layer, docker for VNFs in MEC, and OpenAir-CN for Core Network. The simulation results for the three slices show that OnSlicing retains the SLA with a high percentage and guarantees resource utilization and isolation of the slices.

7) RESOURCE VIRTUALIZATION AND ENERGY EFFICIENCY

In addition to the listed challenges in Section V-D1, the methods in this section also address resource virtualization and energy efficiency. In [154], a Deep Federated Q-Learning (DFQL) method using NFV and SDN is proposed to allocate radio and power resources for virtual slices in industrial IoT applications. At first, a distributed multi-agent method based on DQL called MAQL is proposed to determine the radio resources and the power in the slices, taking into account the QoS requirements of each slice. Then, using the DFL method, each agent sends partial observations from its local model to the global model in the SDN controller for choosing the appropriate actions in each slice and applying the optimal policy to ensure QoS of users in each slice and increasing the overall reward. The simulation results for DFQL and [155] (the centralized approach based on the ML tools) show that the proposed method is better than the compared method in terms of energy consumption, energy efficiency, and delay and transmission rate.

In [132], a distributed and multi-agent method based on decentralized actor-critic called EdgeSlice is provided for allocating radio, power, transport bandwidth, and computational resources in three layers: radio, transmission, and computing. EdgeSlice creates an orchestrator between network slices to ensure the performance of each slice based on its SLA. Each slice is considered as an agent that interacts with the orchestrator. In this method, the OpenAirInterface is used to implement radio resources, OpenDayLight switches are used as transmission resources and CUDA GPU is used as computing resources. The simulation results show that the performance of each slice satisfies based on its SLA for both computational and video slices.

TABLE 8. Summary of related work for distributed learning methods based on allocated resource types.

Ref.	Allocated Resource Types					
	Radio	Power	Computational	Storage	Caching Memory	Transport Bandwidth
[150], [140], [142], [143], [144], [145], [146], [141], [152], [119], [40], [154], [156]	✓	✓	-	-	-	-
[151], [148]	✓	✓	✓	-	-	-
[149]	✓	✓	-	-	✓	-
[153], [132]	✓	✓	✓	-	-	✓

TABLE 9. Summary of related work for resource allocation in RAN slicing based on distributed learning.

GID	Ref.	Features											
		Isolation	DPA	CC	CS	EE	TC	SDN	NFV	RT	MM	DSCM	ST
1	[150]	-	-	Low	Low	-	Low	✓	-	D-RAN	-	✓	MATLAB
2	[132]	-	-	Low	Low	-	Low	✓	✓	D-RAN	-	-	CuDA/OAI /FlexRAN
3	[140]	-	-	Low	High	-	Low	-	-	D-RAN	-	-	Python
4	[146], [141]	-	-	Low	Low	-	Low	✓	-	D-RAN	-	-	-
5	[142]	-	-	Low	Low	-	Low	-	-	D-RAN	-	-	Ape-X
6	[151]	-	-	Low	Low	-	Low	-	-	O-RAN	-	✓	Python
7	[148]	-	-	Low	High	-	Low	-	-	D-RAN	-	-	-
8	[144], [145]	-	-	Low	Low	-	Low	-	-	D-RAN	-	-	MATLAB
9	[149]	-	-	Low	High	-	Low	-	-	D-RAN	-	-	MATLAB
10	[119]	✓	-	Low	High	-	Low	-	-	D-RAN	-	✓	Python
11	[153]	✓	-	Low	Low	-	Low	✓	✓	D-RAN	-	✓	OAI&CN/ Docker/ODL
12	[143]	-	✓	Low	High	-	Low	-	-	D-RAN	-	-	-
13	[152]	-	✓	Low	Low	-	Low	-	-	D-RAN	✓	-	-
14	[154]	-	✓	Low	High	✓	Low	✓	✓	D-RAN	-	-	Python
15	[156]	-	✓	Low	Low	-	Low	-	-	F-RAN	✓	✓	-
16	[40]	✓	✓	Low	High	✓	Low	-	-	D-RAN	-	✓	Python

DPA: Dynamic Power Allocation, CC: Computational Complexity, CS: Convergence Speed, EE: Energy Efficiency, TC: Time Complexity, GID: Group ID, RT: RAN-Type, MM: Mobility Management, DSCM: Dynamic Slice Creation and Management, ST: Simulation Tools

8) MOBILITY MANAGEMENT AND DYNAMIC SLICE CREATION AND MANAGEMENT

The authors in [156] address mobility management and dynamic slice creation and management, along with the listed challenges in Section V-D1. In [156], a distributed method called UC-PA-RA is provided to allocate power and radio blocks to user devices within clusters in multiple F-APs into a single F-RAN. In this method, users’ devices in a specific cluster of a F-AP can receive their information through similar RBs. Rate-splitting multiple-access (RSMA) is used to cluster user devices, which is a way to manage interference in multi-user communications. This problem is divided into two sub-problems of clustering and resource allocation. To solve the clustering sub-problem, a multi-agent RL based on Stochastic Learning Automata (SLA) is used to cluster users’ devices in which each user device is considered as an agent. Fractional programming is used to solve the second sub-problem. The simulation results show that the UC-PA-RA method achieves a higher transmission rate compared to classical methods that use clustering algorithms such as k-means method.

9) LESSONS LEARNED

We summarize the reviewed proposals that use distributed learning for resource allocation in RAN slicing in Tables 8 and 9. The allocated resource types in each proposal are shown in Table 8. Also, we compare the proposals based on different features in Table 9 as mentioned previously at the beginning of this section. To increase the convergence

speed and decrease the computational and time complexity in resource allocation methods in RAN slicing, the methods in V-D are presented. The exchange of acquired knowledge between different entities in the system in transfer learning increases the speed of convergence of resource allocation algorithms in RAN [140], [148], [149]. Also, the use of federated learning distributes the resource allocation algorithm between different entities in the RAN and reduces the computational and time complexity of the algorithm [143], [154]. In [40], [119], [142], [152], [144], [145], [141], and [132], the speed of resource allocation to users of slices in RAN increases by distributing the resource allocation algorithm between different agents, also, the running of each agent on separate CPU thread can facilitate to create and manage the slices. In [40] and [119] the use of parallel blocks in A3C increases the speed of environment exploration and convergence of the algorithm. Some reviewed proposals in Section V-C use SDN to manage resource allocation or implement the transport layer, but using SDN in distributed methods is a vital part because it coordinates different agents and exchanges the information between different entities [132], [141], [146], [153], [154]. Also, dynamic power allocation with regard to the channel condition in groups {12-16} increases the energy efficiency and users’ acceptance rate.

VI. DISCUSSION AND FUTURE RESEARCH DIRECTIONS

In Table 10, we categorize the reviewed proposals based on the ML technique and the type and number of used algorithms. According to our analysis of the literature review on

TABLE 10. Distribution of ML methods for resource management in RAN.

ML Approaches	Ref.	Type	Num. of Papers	
Supervised and Unsupervised Learning	[51], [52], [54], [58], [56], [57]	DNN	6	
	[53], [55]	LSTM	2	
	[59]	T-NN	1	
Evolutionary Algorithms	[65], [66], [68], [69], [67], [70], [72], [71]	GA	8	
	[35], [79], [80], [81], [128], [129], [95]	AC	7	
Reinforcement Learning	[78]	MAB	1	
	[110], [85], [116], [86], [87], [88], [135], [90], [96], [97], [98], [150], [125], [99], [122], [130], [113], [117], [94], [100], [103], [118], [104]	DQN	24	
	[108], [107]	Q-Learning	2	
	[114], [111], [84]	Double DQN	3	
	[115], [123]	A2C	2	
	[109], [82], [114], [83], [106], [131]	DDPG	6	
	[91], [92]	GAT-DQN	2	
	[89], [127]	PPO	2	
	[105], [112], [93], [134]	Ducl. DQN	4	
	[126]	GAN-DQN	1	
	Distributed Learning	[146]	GAN-DQN	1
		[142], [152], [151]	DQN	3
		[40], [119]	A3C	2
[143]		F-AC	1	
[154]		F-DQN	1	
[140], [148], [149]		T-DQN	3	
[141]		Q-Learning	1	
[132]		AC	1	
[153]		PPO	1	
[156]		SLA-RL	1	
[144], [145]		MBDQN	2	

resource management in RAN slicing, as shown in Table 10, most of the recent proposals are based on RL or distributed learning. These ML methods, as mentioned earlier, are dynamic and online techniques that can interact with dynamic network changes and choose the best decision for each condition. Furthermore, due to the distributed nature of these algorithms, such as transfer learning, federated learning and distributed DQNs that run on different devices (e.g. BSs, APs, and UEs) or can run on different CPU threads such as A3C, they are also applicable for real-time applications in terms of computational and time complexity. In addition, transfer learning has been used in the new proposals of resource allocation in RAN slicing, in which the use of prior knowledge of other entities by the new entities can increase the convergence speed of the proposed method. Due to the simple structure of DQN-based methods, most authors have used it for resource management in RAN slicing (see Table 10).

A summary evaluation of the reviewed proposals on resource management in RAN slicing is shown in Tables 2-10. We have categorized the reviewed proposals into multiple groups based on their advantages and disadvantages. The most important advantage of ML approaches is solving the resource management and resource allocation tasks, which are often non-convex and NP-Hard. These methods can learn the proper resource allocation policy over time and enforce them in the system.

The use of SDN and NFV has addressed some challenges in RAN slicing. For example, in the reviewed proposals, the SDN controllers are used as coordinators between different agents in distributed machine learning algorithms or different entities in the distributed methods. In addition, NFV has been used to virtualize resources such as computational, storage, and caching memory, in which hardware resources can be

optimally utilized and the slice isolation of each slice can be ensured.

Most of the reviewed proposals are in accordance with D-RAN architecture and use a centralized resource allocation algorithm. The implementation of these proposals in new architectures faces serious challenges, due to a lack of computational resources. Because a significant part of the computational operations in the new architectures of RAN is done in BBU or DU/CU. Also, most of the reviewed proposals based on supervised and unsupervised learning are used to map the user to the appropriate slice and predict the required traffic for each slice. These methods are impractical due to the need for prior knowledge and a proper dataset concerning a rapidly changing dynamic network. In evolutionary algorithms such as GA, the selection of the proper “fitness” function has special importance. This is sometimes considered as a drawback of GAs because the selection of a “fitness” function can be difficult in some applications. Recently, distributed and RL methods have been widely used for RAN slicing in 5G/B5G networks, which are a good options for interacting with dynamic environments due to their online nature. Choosing the proper state, action, and reward function in RL algorithms and the type of algorithm in the distributed methods is a major challenge. The key difference between GA and RL methods is that the agent of GA does not know its own fitness, and so does not learn from a fitness signal in the same way that an RL agent learns from reward signals. As shown in Tables 2-10, a comprehensive method has not been proposed to address all challenges outlined in Section IV.

Significant challenges remain for the realization of an efficient RAN slicing solution while ensuring the quality of service and the desired SLA of each slice. In the following, we have mentioned future research directions in RAN slicing

based on Tables 2-10. We also state possible solutions for each challenge.

- **Isolation:** The issue of isolation of radio resources in RAN is another challenge that should be properly considered because not ensuring the isolation of each slice increases the reconfiguration overhead and wastes resources. Although some proposals such as [40], [58], [59], [68], [69], [114], [115], [119], [116], [134], [153], [117], [118] have addressed this issue, creating a trade-off between isolation and optimal resource utilization is an open issue in RAN slicing.

Possible Solution: The isolation of each slice means that the allocated resources to each slice do not change over a specific period of time. To ensure the quality of service, a degree of isolation must be enforced to each slice so that the traffic load variation in a slice does not affect the other slices. To this end, we must reserve resources for future users of each slice over a specific period of time to avoid frequent reconfiguration of the resources of each slice. We can use LSTM [157] to predict the required resources for the future users using the prior information of the allocated resources (e.g., [40], [119]).

- **Dynamic Power Allocation:** In most proposals, the allocated transmission power to the UEs is fixed and not determined dynamically. Indeed, a low transmission power value may increase the outage probability. The simple solution is to consider the maximum available transmission power, but this increases the power consumption, which in turn increases the interference level and operational costs of network providers. Channel conditions and energy efficiency are not considered as important factors in resource allocation in RAN slicing. In addition, the problem of frequency interference in inter-cells, intra-cells, inter-RANs, or intra-RANs should be considered. Although some proposals such as [40], [67], [68], [69], [72], [80], [81], [111], [114], [85], [86], [87], [88], [98], [112], [135], [104], [108], [113], [117], [118], [143], [152], [154], [156] have addressed the dynamic power allocation issue, presenting an efficient dynamic power allocation method with a low computational and time complexity is a vital requirement.

Possible Solution: Dynamic power allocation in existing proposals has been done in two manners: 1) power allocation with an iterative method using Lagrangian coefficients taking into account channel conditions and interference (e.g., [81], [116], [118]) 2) considering power as an action in online algorithms and selecting the desired power by the agent for each user, evaluating the selected power in accordance with the channel conditions and interference such as [117].

- **Mobility Management:** The proposed frameworks and algorithms for resource management in network slicing should take into account user mobility and allocate or reserve resources to users of each slice according to the

users' mobility. Also, user handovers should be considered because resource allocation to users of a slice when the target slice is overloaded is a serious challenge. Few proposals have considered mobility on resource management in RAN slicing (e.g., [71], [109], [110], [135], [152]), but examining the impact of different mobility models on resource management algorithms is an open issue.

Possible Solution: User mobility prediction through methods such as LSTM by the resource manager or the mobile devices can help resource allocation methods to reserve the required resources for mobile users in future steps. Although the use of mobility prediction methods in resource management will also prevent the outage probability of users due to lack of resources, and improve the QoS of users, the privacy-preserving of users is an important challenge in mobility prediction methods that must be considered.

- **Convergence Speed:** The convergence speed of the reviewed proposals is proportional to the speed of environment discovery. Therefore, the convergence speed is directly related to the network size and the number of slices. Most RL proposals have only one agent to explore the environment, which seriously challenges the convergence speed in these methods.

Possible Solution: The use of parallel agents like A3C (e.g., [40], [119]) and the prior and acquired knowledge of other agents such as TRL (e.g., [140], [148], [149]) and federated learning (e.g., [143], [154]) can increase the speed of convergence of ML algorithms and the discovery of the environment.

- **Dynamic Slice Creation and Management:** The reviewed proposals have not considered dynamic creation and management of slices in resource management. Re-configuring the network when adding or removing a slice incurs the non-negligible effects on other slices. Also, most proposals evaluate their scenarios using two or three slices, so the scaling of these proposals would raise issues in practical scenarios. Currently, the dynamic slice creation and management is an open issue in the RAN slicing.

Possible Solution: The implementation of each slice independently of other slices (each slice runs on one thread) and the lack of effect of other slices on the resource allocation decisions of each slice can make it easy to manage the slices (e.g., [40]). In this condition, we can manage slices (add, remove, reconfigure of the slices) without affecting other slices.

- **Computational and Time Complexity:** Computation in most of the reviewed proposals is performed in a centralized manner. So, these methods are not practically feasible in real-time applications with regard to the new architectures of RAN. In some proposals such as [40], [119], [132], [135], [140], [141], [142], [143], [144], [145], [146], [148], [149], [150], [152], [154], [117], [151], [153], [156] computational and time complexity

have been considered as an important issue, but in other reviewed proposals this issue has not been accounted for RAN slicing. Therefore, given the changes in the network and responding to users' requests at an acceptable time, this is a vital challenge that must be considered.

Possible Solution: The implementation of each slice independently of other slices (each slice runs on one thread) and the use of distributed methods can reduce the computational and time complexity for the proposed methods (e.g., [40]).

- **Priority Between Slices:** Due to the existence of some critical slices for real-time applications, it is necessary to create a priority between the slices in order to allocate resources to them. Some proposals such as [158], [159] have considered the impact of this issue on the profit of the InPs. But, the impact of the priority between the slices on the QoS of users in all slices is an important challenge that is less addressed.

Possible Solution: An efficient priority queue can guarantee the QoS of users in all slices by choosing a proper resource scheduling strategy for each queue such as [96], [123], [112].

- **Adaptation to New RAN Architectures:** Due to the emergence of new RAN architectures, it is necessary to propose applicable resource management methods in accordance with the new architectures. Most of the centralized proposals are not applicable in new RAN architectures due to the lack of sufficient computational resources on the RRH side, so providing a resource management method in accordance with new RAN architectures is an important issue that is less addressed.

Possible Solution: This issue can be addressed by using distributed methods or methods with low computational complexity in base stations (e.g., [151], [156]) to adapt with various RAN architectures.

VII. CONCLUSION

We have presented a survey of the state-of-the-art in RAN slicing of 5G/B5G networks. To this end, we have discussed the important challenges in resource management such as resource sharing, virtualization in the RAN, isolation, mobility, etc. We investigated some RAN architectures and classified existing work based on the used ML algorithm, the addressed challenges, and the allocated resource type. Then, we review lessons learned from the reviewed papers in each section to compare the methods. Furthermore, we summarized the reviewed papers in a comprehensive manner in Tables 2-10. We assessed the papers and compared their advantages and disadvantages in Tables 2-10. Finally, future challenges and some solutions to address the challenges are introduced for research direction in Section VI.

REFERENCES

- [1] X. Lin and N. Lee, *5G and Beyond*. Cham, Switzerland: Springer, 2021.
- [2] W. Xiang, K. Zheng, and X. S. Shen, *5G Mobile Communications*. Cham, Switzerland: Springer, 2016.

- [3] *Is it Worth Investing in 5G Yet?*. Accessed: Feb. 2, 2021. [Online]. Available: <https://internationalbanker.com/brokerage/is-it-worth-investing-in-5g-yet/>
- [4] X. Foukas, G. Patounas, A. Elmokashfi, and M. K. Marina, "Network slicing in 5G: Survey and challenges," *IEEE Commun. Mag.*, vol. 55, no. 5, pp. 94–100, May 2017.
- [5] *Network Sharing; Architecture and Functional Description (Release 15)*, document TR 21.915 V0.5.0, 3GPP, 2018.
- [6] *5G Testing | MMTc use Cases | Rohde & Schwarz*. Accessed: Jan. 11, 2020. [Online]. Available: https://www.rohde-schwarz.com/us/solutions/test-and-measurement/wireless-communication/wireless-5g-and-cellular/5g-test-and-measurement/mmtc/mmtc_233752.html
- [7] S. Alraih, I. Shayea, M. Behjati, R. Nordin, N. F. Abdullah, A. Abu-Samah, and D. Nandi, "Revolution or evolution? Technical requirements and considerations towards 6G mobile communications," *Sensors*, vol. 22, no. 3, p. 762, Jan. 2022.
- [8] S. M. A. Kazmi, L. U. Khan, N. H. Tran, and C. S. Hong, *Network Slicing for 5G and Beyond Networks*. Cham, Switzerland: Springer, 2019.
- [9] P. L. Vo, M. N. H. Nguyen, T. A. Le, and N. H. Tran, "Slicing the edge: Resource allocation for RAN network slicing," *IEEE Wireless Commun. Lett.*, vol. 7, no. 6, pp. 970–973, Dec. 2018.
- [10] F. Y.-S. Lin, C.-H. Hsiao, Y.-F. Wen, and Y.-S. Wu, "Optimization-based resource management strategies for 5G C-RAN slicing capabilities," in *Proc. 10th Int. Conf. Ubiquitous Future Netw. (ICUFN)*, Jul. 2018, pp. 346–351.
- [11] D. Ginthor, R. Guillaume, M. Schungel, and H. D. Schotten, "5G RAN slicing for deterministic traffic," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Mar. 2021, pp. 1–6.
- [12] X. Li, M. Samaka, H. A. Chan, D. Bhamare, L. Gupta, C. Guo, and R. Jain, "Network slicing for 5G: Challenges and opportunities," *IEEE Internet Comput.*, vol. 21, no. 5, pp. 20–27, Sep. 2017.
- [13] J. Ordonez-Lucena, P. Ameigeiras, D. Lopez, J. J. Ramos-Munoz, J. Lorca, and J. Folgueira, "Network slicing for 5G with SDN/NFV: Concepts, architectures, and challenges," *IEEE Commun. Mag.*, vol. 55, no. 5, pp. 80–87, May 2017.
- [14] I. Afolabi, T. Taleb, K. Samdanis, A. Ksentini, and H. Flinck, "Network slicing and softwarization: A survey on principles, enabling technologies, and solutions," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 3, pp. 2429–2453, 3rd Quart., 2018.
- [15] A. A. Barakabitze, A. Ahmad, R. Mijumbi, and A. Hines, "5G network slicing using SDN and NFV: A survey of taxonomy, architectures and future challenges," *Comput. Netw.*, vol. 167, Feb. 2020, Art. no. 106984.
- [16] A. Kaloxylou, C. Mannweiler, G. Zimmermann, M. Di Girolamo, P. Marsch, J. Belschner, A. Tzanakaki, R. Trivisonno, Ö. Bulakçı, P. Spapis, P. Rost, P. Arnold, and N. Nikaen, "Network slicing," in *5G System Design: Architectural and Functional Considerations and Long Term Research*. Hoboken, NJ, USA: Wiley, 2018, pp. 181–205.
- [17] J. A. Hurtado Sánchez, K. Casilimas, and O. M. C. Rendon, "Deep reinforcement learning for resource management on network slicing: A survey," *Sensors*, vol. 22, no. 8, p. 3031, Apr. 2022.
- [18] B. Han and H. D. Schotten, "Machine learning for network slicing resource management: A comprehensive survey," 2020, *arXiv:2001.07974*.
- [19] X. Shen, J. Gao, W. Wu, K. Lyu, M. Li, W. Zhuang, X. Li, and J. Rao, "AI-assisted network-slicing based next-generation wireless networks," *IEEE Open J. Veh. Technol.*, vol. 1, pp. 45–66, 2020.
- [20] M. Richart, J. Baliosian, J. Serrat, and J.-L. Gorricho, "Resource slicing in virtual wireless networks: A survey," *IEEE Trans. Netw. Serv. Manage.*, vol. 13, no. 3, pp. 462–476, Sep. 2016.
- [21] P. P. Ray and N. Kumar, "SDN/NFV architectures for edge-cloud oriented IoT: A systematic review," *Comput. Commun.*, vol. 169, pp. 129–153, Mar. 2021.
- [22] L. Yang, B. Ng, W. K. G. Seah, L. Groves, and D. Singh, "A survey on network forwarding in software-defined networking," *J. Netw. Comput. Appl.*, vol. 176, Feb. 2021, Art. no. 102947.
- [23] Z. I. Rasool, R. S. A. Ali, and M. M. Abdulzahra, "Network management in software-defined network: A survey," in *Proc. IOP Conf. Mater. Sci. Eng.*, vol. 1094, no. 1, Feb. 2021, Art. no. 012055.
- [24] B. Rudra et al., "Architecture and deployment models-SDN protocols, APIs, and layers, applications and implementations," in *Software Defined Internet of Everything*. Cham, Switzerland: Springer, 2022, pp. 63–77.
- [25] S. Saraswat, V. Agarwal, H. P. Gupta, R. Mishra, A. Gupta, and T. Dutta, "Challenges and solutions in software defined networking: A survey," *J. Netw. Comput. Appl.*, vol. 141, pp. 23–58, Sep. 2019.

- [26] F. Schardong, I. Nunes, and A. Schaeffer-Filho, "NFV resource allocation: A systematic review and taxonomy of VNF forwarding graph embedding," *Comput. Netw.*, vol. 185, Feb. 2021, Art. no. 107726.
- [27] S. Mostafavi, V. Hakami, and M. Sanaci, "Quality of service provisioning in network function virtualization: A survey," *Computing*, vol. 103, no. 5, pp. 917–991, May 2021.
- [28] M. Golkarifard, C. F. Chiasserini, F. Malandrino, and A. Movaghar, "Dynamic VNF placement, resource allocation and traffic routing in 5G," *Comput. Netw.*, vol. 188, Apr. 2021, Art. no. 107830.
- [29] M. Laroui, B. Nour, H. Moungra, M. A. Cherif, H. Afifi, and M. Guizani, "Edge and fog computing for IoT: A survey on current research activities future directions," *Comput. Commun.*, vol. 180, pp. 210–231, Dec. 2021.
- [30] A. Shakarami, H. Shakarami, M. Ghobaei-Arani, E. Nikougofar, and M. Faraji-Mehmandar, "Resource provisioning in edge/fog computing: A comprehensive and systematic review," *J. Syst. Archit.*, vol. 122, Jan. 2022, Art. no. 102362.
- [31] Z. M. Nayeri, T. Ghafarian, and B. Javadi, "Application placement in fog computing with AI approach: Taxonomy and a state of the art survey," *J. Netw. Comput. Appl.*, vol. 185, Jul. 2021, Art. no. 103078.
- [32] M. A. Habibi, M. Nasimi, B. Han, and H. D. Schotten, "A comprehensive survey of RAN architectures toward 5G mobile communication system," *IEEE Access*, vol. 7, pp. 70371–70421, 2019.
- [33] W. Ejaz, S. K. Sharma, S. Saadat, M. Naeem, A. Anpalagan, and N. A. Chughtai, "A comprehensive survey on resource allocation for CRAN in 5G and beyond networks," *J. Netw. Comput. Appl.*, vol. 160, Jun. 2020, Art. no. 102638.
- [34] M. Polese, L. Bonati, S. D'Oro, S. Basagni, and T. Melodia, "Understanding O-RAN: Architecture, interfaces, algorithms, security, and research challenges," 2022, *arXiv:2202.01032*.
- [35] D. Bega, M. Gramaglia, A. Garcia-Saavedra, M. Fiore, A. Banchs, and X. Costa-Perez, "Network slicing meets artificial intelligence: An AI-based framework for slice management," *IEEE Commun. Mag.*, vol. 58, no. 6, pp. 32–38, Jun. 2020.
- [36] L. U. Khan, I. Yaqoob, N. H. Tran, Z. Han, and C. S. Hong, "Network slicing: Recent advances, taxonomy, requirements, and open research challenges," *IEEE Access*, vol. 8, pp. 36009–36028, 2020.
- [37] A. J. Gonzalez, J. Ordonez-Lucena, B. E. Helvik, G. Nencioni, M. Xie, D. R. Lopez, and P. Grønsund, "The isolation concept in the 5G network slicing," in *Proc. Eur. Conf. New. Commun. (EuCNC)*, Jun. 2020, pp. 12–16.
- [38] M. M. Sajjad, C. J. Bernardos, D. Jayalath, and Y.-C. Tian, "Inter-slice mobility management in 5G: Motivations, standard principles, challenges and research directions," 2020, *arXiv:2003.11343*.
- [39] C. Tipantuña and X. Hesselbach, "Adaptive energy management in 5G network slicing: Requirements, architecture, and strategies," *Energies*, vol. 13, no. 15, p. 3984, Aug. 2020.
- [40] 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. 2021.
- [41] A. Thantharate, R. Paropkari, V. Walunj, C. Beard, and P. Kankariya, "Secure5G: A deep learning framework towards a secure network slicing in 5G and beyond," in *Proc. 10th Annu. Comput. Commun. Workshop Conf. (CCWC)*, Jan. 2020, pp. 0852–0857.
- [42] F. Debbabi, R. Jmal, L. C. Fourati, and A. Ksentini, "Algorithmics and modeling aspects of network slicing in 5G and beyond network: Survey," *IEEE Access*, vol. 8, pp. 162748–162762, 2020.
- [43] Y. Sun, L. Zhang, G. Feng, and M. A. Imran, "Access control and handoff policy design for RAN slicing," in *Radio Access Network Slicing and Virtualization for 5G Vertical Industries*. Hoboken, NJ, USA: Wiley, 2021, pp. 167–187.
- [44] R. Shrivastava, K. Samdanis, and A. Bakry, "On policy based RAN slicing for emerging 5G TDD networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2018, pp. 1–6.
- [45] L. Liang, Y. Wu, and G. Feng, "Online auction-based resource allocation for service-oriented network slicing," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 8063–8074, Jun. 2019.
- [46] T. Ma, Y. Zhang, S. Yuan, and Z. Cheng, "Cognitive RAN slicing resource allocation based on Stackelberg game," *China Commun.*, vol. 19, no. 5, pp. 12–23, May 2021.
- [47] M. W. Berry, A. Mohamed, and B. W. Yap, *Supervised and Unsupervised Learning for Data Science*. Cham, Switzerland: Springer, 2019.
- [48] S. C. H. Hoi, D. Sahoo, J. Lu, and P. Zhao, "Online learning: A comprehensive survey," *Neurocomputing*, vol. 459, pp. 249–289, Oct. 2021.
- [49] H. Nozari, M. E. Sadeghi, and J. G. Nahr, "Artificial intelligence and machine learning for real-world problems (a survey)," *Int. J. Innov. Eng.*, vol. 1, no. 3, pp. 38–47, 2021.
- [50] K. Bochie, M. S. Gilbert, L. Gantert, M. S. M. Barbosa, D. S. V. Medeiros, and M. E. M. Campista, "A survey on deep learning for challenged networks: Applications and trends," *J. Netw. Comput. Appl.*, vol. 194, Nov. 2021, Art. no. 103213.
- [51] P. Du and A. Nakao, "Deep learning-based application specific RAN slicing for mobile networks," in *Proc. IEEE 7th Int. Conf. Cloud Netw. (CloudNet)*, Oct. 2018, pp. 1–3.
- [52] A. Thantharate, R. Paropkari, V. Walunj, and C. Beard, "DeepSlice: A deep learning approach towards an efficient and reliable network slicing in 5G networks," in *Proc. IEEE 10th Annu. Ubiquitous Comput., Electron. Mobile Commun. Conf. (UEMCON)*, Oct. 2019, pp. 0762–0767.
- [53] M. Toscano, F. Grunwald, M. Richart, J. Baliosian, E. Grampín, and A. Castro, "Machine learning aided network slicing," in *Proc. 21st Int. Conf. Transparent Opt. Netw. (ICTON)*, Jul. 2019, pp. 1–4.
- [54] R. Abozariba, M. K. Naeem, M. Asaduzzaman, and M. Patwary, "Uncertainty-aware RAN slicing via machine learning predictions in next-generation networks," in *Proc. IEEE 92nd Veh. Technol. Conf. (VTC-Fall)*, Nov. 2020, pp. 1–6.
- [55] S. Bakri, P. A. Frangoudis, A. Ksentini, and M. Bouaziz, "Data-driven RAN slicing mechanisms for 5G and beyond," *IEEE Trans. Netw. Service Manage.*, vol. 18, no. 4, pp. 4654–4668, Dec. 2021.
- [56] H. Chergui and C. Verikoukis, "OPEX-limited 5G RAN slicing: An over-dataset constrained deep learning approach," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2020, pp. 1–6.
- [57] A. Nakao and P. Du, "Toward in-network deep machine learning for identifying mobile applications and enabling application specific network slicing," *IEICE Trans. Commun.*, vol. 101, pp. 1536–1543, Jul. 2018.
- [58] H. Chergui and C. Verikoukis, "Offline SLA-constrained deep learning for 5G networks reliable and dynamic End-to-End slicing," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 2, pp. 350–360, Feb. 2019.
- [59] J. Mei, X. Wang, and K. Zheng, "An intelligent self-sustained RAN slicing framework for diverse service provisioning in 5G-beyond and 6G networks," *Intell. Converged Netw.*, vol. 1, no. 3, pp. 281–294, Dec. 2020.
- [60] Y. Sun, M. Peng, Y. Zhou, Y. Huang, and S. Mao, "Application of machine learning in wireless networks: Key techniques and open issues," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3072–3108, 4th Quart., 2019.
- [61] G. Marcus and E. Davis, *Rebooting AI: Building Artificial Intelligence we can Trust*. Vintage, 2019.
- [62] X. Yu and M. Gen, *Introduction to Evolutionary Algorithms*. Cham, Switzerland: Springer, 2010.
- [63] A. Telikani, A. Tahmassebi, W. Banzhaf, and A. H. Gandomi, "Evolutionary machine learning: A survey," *ACM Comput. Surv.*, vol. 54, no. 8, pp. 1–35, Nov. 2021.
- [64] Y. Tian, L. Si, X. Zhang, R. Cheng, C. He, K. C. Tan, and Y. Jin, "Evolutionary large-scale multi-objective optimization: A survey," *ACM Comput. Surv.*, vol. 54, no. 8, pp. 1–34, Nov. 2021.
- [65] Y. Liu, L. Wang, X. Wen, Z. Lu, and L. Liu, "A network slicing strategy for telemedicine based on classification," in *Proc. IEEE/CIC Int. Conf. Commun. China (ICCC Workshops)*, Aug. 2020, pp. 191–196.
- [66] D. Wu, Z. Zhang, S. Wu, J. Yang, and R. Wang, "Biologically inspired resource allocation for network slices in 5G-enabled Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 9266–9279, Dec. 2018.
- [67] S. S. Jayanthi, Y. L. Lee, and Y. C. Chang, "User association for multi-tenant heterogeneous network slicing using genetic algorithm," in *Proc. 8th Int. Conf. Comput. Commun. Eng. (ICCCCE)*, Jun. 2021, pp. 326–330.
- [68] X. Yang, Y. Liu, I. C. Wong, Y. Wang, and L. Cuthbert, "Genetic algorithm for inter-slice resource management in 5G network with isolation," in *Proc. Int. Conf. Softw., Telecommun. Comput. Netw. (SoftCOM)*, Sep. 2020, pp. 1–6.
- [69] X. Yang, Y. Wang, I. C. Wong, Y. Liu, and L. Cuthbert, "Genetic algorithm in resource allocation of RAN slicing with QoS isolation and fairness," in *Proc. IEEE Latin-American Conf. Commun. (LATINCOM)*, Nov. 2020, pp. 1–6.
- [70] B. Han, J. Lianghai, and H. D. Schotten, "Slice as an evolutionary service: Genetic optimization for inter-slice resource management in 5G networks," *IEEE Access*, vol. 6, pp. 33137–33147, 2018.
- [71] X. Xu, H. Zhang, X. Dai, Y. Hou, X. Tao, and P. Zhang, "SDN based next generation mobile network with service slicing and trials," *China Commun.*, vol. 11, no. 2, pp. 65–77, Feb. 2014.

- [72] G. Zhao, S. Qin, G. Feng, and Y. Sun, "Network slice selection in softwarization-based mobile networks," *Trans. Emerg. Telecommun. Technol.*, vol. 31, no. 1, p. e3617, 2020.
- [73] R. Kokku, R. Mahindra, H. Zhang, and S. Rangarajan, "CellSlice: Cellular wireless resource slicing for active RAN sharing," in *Proc. 5th Int. Conf. Commun. Syst. Netw. (COMSNETS)*, Jan. 2013, pp. 1–10.
- [74] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 2018.
- [75] S. Gupta, S. Pal, K. Kumar, and K. Ghosh, "Development of conceptual framework for reinforcement learning based optimal control," in *Innovation in Electrical Power Engineering, Communication, and Computing Technology*. Cham, Switzerland: Springer, 2022, pp. 339–348.
- [76] Z. Yang, F. Wu, and J. Zhao, "A survey: Limited data problem and strategy of reinforcement learning," in *Proc. Chin. Intell. Automat. Conf.* Cham, Switzerland: Springer, 2022, pp. 471–481.
- [77] J. Parker-Holder, R. Rajan, X. Song, A. Biedenkapp, Y. Miao, T. Eimer, B. Zhang, V. Nguyen, R. Calandra, A. Faust, F. Hutter, and M. Lindauer, "Automated reinforcement learning (AutoRL): A survey and open problems," 2022, *arXiv:2201.03916*.
- [78] V. Sciancalepore, L. Zanzi, X. Costa-Perez, and A. Capone, "ONETS: Online network slice broker from theory to practice," *IEEE Trans. Wireless Commun.*, vol. 21, no. 1, pp. 121–134, Jan. 2021.
- [79] K. Yu, H. Zhou, Z. Tang, X. Shen, and F. Hou, "Deep reinforcement learning-based RAN slicing for UL/DL decoupled cellular V2X," *IEEE Trans. Wireless Commun.*, vol. 21, no. 5, pp. 3523–3535, May 2021.
- [80] W. Wu, N. Chen, C. Zhou, M. Li, X. Shen, W. Zhuang, and X. Li, "Dynamic RAN slicing for service-oriented vehicular networks via constrained learning," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 7, pp. 2076–2089, Jul. 2020.
- [81] Y. Xu, Z. Zhao, P. Cheng, Z. Chen, M. Ding, B. Vucetic, and Y. Li, "Constrained reinforcement learning for resource allocation in network slicing," *IEEE Commun. Lett.*, vol. 25, no. 5, pp. 1554–1558, May 2021.
- [82] C. Qi, Y. Hua, R. Li, Z. Zhao, and H. Zhang, "Deep reinforcement learning with discrete normalized advantage functions for resource management in network slicing," *IEEE Commun. Lett.*, vol. 23, no. 8, pp. 1337–1341, Aug. 2019.
- [83] N. Salhab, R. Langar, and R. Rahim, "5G network slices resource orchestration using machine learning techniques," *Comput. Netw.*, vol. 188, Apr. 2021, Art. no. 107829.
- [84] S. Meng, Z. Wang, H. Ding, S. Wu, X. Li, P. Zhao, C. Zhu, and X. Wang, "RAN slice strategy based on deep reinforcement learning for smart grid," in *Proc. Comput., Commun. IoT Appl. (ComComAp)*, Oct. 2019, pp. 6–11.
- [85] F. Song, J. Li, C. Ma, Y. Zhang, L. Shi, and D. N. K. Jayakody, "Dynamic virtual resource allocation for 5G and beyond network slicing," *IEEE Open J. Veh. Technol.*, vol. 1, pp. 215–226, 2020.
- [86] R. Xi, X. Chen, Y. Chen, and Z. Li, "Real-time resource slicing for 5G RAN via deep reinforcement learning," in *Proc. IEEE 25th Int. Conf. Parallel Distrib. Syst. (ICPADS)*, Dec. 2019, pp. 625–632.
- [87] G. Yang, Q. Liu, X. Zhou, Y. Qian, and W. Wu, "Two-tier resource allocation in dynamic network slicing paradigm with deep reinforcement learning," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2019, pp. 1–6.
- [88] M. Zambianco and G. Verticale, "Spectrum allocation for network slices with inter-numerology interference using deep reinforcement learning," in *Proc. IEEE 31st Annu. Int. Symp. Pers., Indoor Mobile Radio Commun.*, Aug. 2020, pp. 1–7.
- [89] F. Saggese, L. Pasqualini, M. Moretti, and A. Abrardo, "Deep reinforcement learning for URLLC data management on top of scheduled eMBB traffic," 2021, *arXiv:2103.01801*.
- [90] K. Suh, S. Kim, Y. Ahn, S. Kim, H. Ju, and B. Shim, "Deep reinforcement learning-based network slicing for beyond 5g," *IEEE Access*, vol. 10, pp. 7384–7395, 2022.
- [91] Y. Shao, R. Li, Z. Zhao, and H. Zhang, "Graph attention network-based DRL for network slicing management in dense cellular networks," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Mar. 2021, pp. 1–6.
- [92] Y. Shao, R. Li, B. Hu, Y. Wu, Z. Zhao, and H. Zhang, "Graph attention network-based multi-agent reinforcement learning for slicing resource management in dense cellular network," *IEEE Trans. Veh. Technol.*, vol. 70, no. 10, pp. 10792–10803, Oct. 2021.
- [93] G. O. Boateng, D. Ayepah-Mensah, D. M. Doe, A. Mohammed, G. Sun, and G. Liu, "Blockchain-enabled resource trading and deep reinforcement learning-based autonomous RAN slicing in 5G," *IEEE Trans. Netw. Service Manage.*, vol. 19, no. 1, pp. 216–227, Mar. 2021.
- [94] 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.
- [95] S. F. Abedin, A. Mahmood, N. H. Tran, Z. Han, and M. Gidlund, "Elastic O-RAN slicing for industrial monitoring and control: A distributed matching game and deep reinforcement learning approach," *IEEE Trans. Veh. Technol.*, pp. 1–15, 2021.
- [96] R. Li, Z. Zhao, Q. Sun, I. Chih-Lin, C. Yang, X. Chen, M. Zhao, and H. Zhang, "Deep reinforcement learning for resource management in network slicing," *IEEE Access*, vol. 6, pp. 74429–74441, 2018.
- [97] Y. Shi, Y. E. Sagduyu, and T. Erpek, "Reinforcement learning for dynamic resource optimization in 5G radio access network slicing," in *Proc. IEEE 25th Int. Workshop Comput. Aided Model. Design Commun. Links Netw. (CAMAD)*, Sep. 2020, pp. 1–6.
- [98] Y. Shi, P. Rahimzadeh, M. Costa, T. Erpek, and Y. E. Sagduyu, "Deep reinforcement learning for 5G radio access network slicing with spectrum coexistence," *TechRxiv*, 2021. Accessed: Feb. 2, 2022, doi: 10.36227/techrxiv.16632526.v1.
- [99] X. Chen, C. Wu, T. Chen, N. Wu, H. Zhang, and Y. Ji, "Age of information-aware multi-tenant resource orchestration in network slicing," in *Proc. IEEE Int. Conf. Dependable, Autonomic Secure Comput., Int. Conf. Pervasive Intell. Comput., Int. Conf. Cloud Big Data Comput., Int. Conf. Cyber Sci. Technol. Congr. (DASC/PiCom/CBDCom/CyberSciTech)*, Aug. 2019, pp. 1001–1007.
- [100] X. Wang, J. D. Thomas, R. J. Piechocki, S. Kapoor, R. Santos-Rodríguez, and A. Parekh, "Self-play learning strategies for resource assignment in open-RAN networks," *Comput. Netw.*, vol. 206, Apr. 2022, Art. no. 108682.
- [101] W. Zhu, Y. Zhuang, and L. Zhang, "A three-dimensional virtual resource scheduling method for energy saving in cloud computing," *Future Gener. Comput. Syst.*, vol. 69, pp. 66–74, Apr. 2017.
- [102] L. Duan, H. Hu, Y. Qian, Y. Gong, X. Zhang, Y. Xu, and J. Wei, "A multi-task selected learning approach for solving 3D flexible bin packing problem," 2018, *arXiv:1804.06896*.
- [103] A. Nassar and Y. Yilmaz, "Deep reinforcement learning for adaptive network slicing in 5G for intelligent vehicular systems and smart cities," *IEEE Internet Things J.*, vol. 9, no. 1, pp. 222–235, Jan. 2021.
- [104] H. Xiang, M. Peng, Y. Sun, and S. Yan, "Mode selection and resource allocation in sliced fog radio access networks: A reinforcement learning approach," *IEEE Trans. Veh. Technol.*, vol. 69, no. 4, pp. 4271–4284, Apr. 2020.
- [105] N. Van Huynh, D. T. Hoang, D. N. Nguyen, and E. Dutkiewicz, "Real-time network slicing with uncertain demand: A deep learning approach," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2019, pp. 1–6.
- [106] S. Martiradonna, A. Abrardo, M. Moretti, G. Piro, and G. Boggia, "Deep reinforcement learning-aided RAN slicing enforcement supporting latency sensitive services in B5G networks," *Internet Technol. Lett.*, vol. 4, no. 6, p. e328, Nov. 2021.
- [107] A. T. Nassar and Y. Yilmaz, "Reinforcement learning-based resource allocation in fog RAN for IoT with heterogeneous latency requirements," 2018, *arXiv:1806.04582*.
- [108] X. Wang and T. Zhang, "Reinforcement learning based resource allocation for network slicing in 5G C-RAN," in *Proc. Comput., Commun. IoT Appl. (ComComAp)*, Oct. 2019, pp. 106–111.
- [109] Y. Cui, X. Huang, P. He, D. Wu, and R. Wang, "A two-timescale resource allocation scheme in vehicular network slicing," in *Proc. IEEE 93rd Veh. Technol. Conf. (VTC-Spring)*, Apr. 2021, pp. 1–5.
- [110] J. Mei, X. Wang, and K. Zheng, "Semi-decentralized network slicing for reliable V2V service provisioning: A model-free deep reinforcement learning approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 12108–12120, Aug. 2021.
- [111] S. Yuan, Y. Zhang, W. Qie, T. Ma, and S. Li, "Deep reinforcement learning for resource allocation with network slicing in cognitive radio network," *Comput. Sci. Inf. Syst.*, vol. 18, no. 3, p. 55, 2020.
- [112] M. Liu, Y. Wang, S. Meng, X. Zhao, and S. Geng, "Deep reinforcement learning-based resource allocation for smart grid in RAN network slice," in *Advances in Wireless Communications and Applications*. Cham, Switzerland: Springer, 2021, pp. 199–215.
- [113] P. Yu, F. Zhou, X. Zhang, X. Qiu, M. Kadoch, and M. Cheriet, "Deep learning-based resource allocation for 5G broadband TV service," *IEEE Trans. Broadcast.*, vol. 66, no. 4, pp. 800–813, Dec. 2020.
- [114] J. Mei, X. Wang, K. Zheng, G. Boudreau, A. B. Sediq, and H. Abou-Zeid, "Intelligent radio access network slicing for service provisioning in 6G: A hierarchical deep reinforcement learning approach," *IEEE Trans. Commun.*, vol. 69, no. 9, pp. 6063–6078, Sep. 2021.

- [115] R. Li, C. Wang, Z. Zhao, R. Guo, and H. Zhang, "The LSTM-based advantage actor-critic learning for resource management in network slicing with user mobility," *IEEE Commun. Lett.*, vol. 24, no. 9, pp. 2005–2009, Sep. 2020.
- [116] G. Sun, Z. T. Gebrekidan, G. O. Boateng, D. Ayepah-Mensah, and W. Jiang, "Dynamic reservation and deep reinforcement learning based autonomous resource slicing for virtualized radio access networks," *IEEE Access*, vol. 7, pp. 45758–45772, 2019.
- [117] M. Mohsenivatani, M. Darabi, S. Parsaeefard, M. Ardebilipour, and B. Maham, "Throughput maximization in C-RAN enabled virtualized wireless networks via multi-agent deep reinforcement learning," in *Proc. IEEE 31st Annu. Int. Symp. Pers., Indoor Mobile Radio Commun.*, Aug. 2020, pp. 1–6.
- [118] H. Xiang, S. Yan, and M. Peng, "A realization of fog-RAN slicing via deep reinforcement learning," *IEEE Trans. Wireless Commun.*, vol. 19, no. 4, pp. 2515–2527, Apr. 2020.
- [119] M. Yan, G. Feng, J. Zhou, Y. Sun, and Y. Liang, "Intelligent resource scheduling for 5G radio access network slicing," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 7691–7703, Aug. 2019.
- [120] R. Kokku, R. Mahindra, H. Zhang, and S. Rangarajan, "NVS: A substrate for virtualizing wireless resources in cellular networks," *IEEE/ACM Trans. Netw.*, vol. 20, no. 5, pp. 1333–1346, Oct. 2011.
- [121] R. Mahindra, M. A. Khojastepour, H. Zhang, and S. Rangarajan, "Radio access network sharing in cellular networks," in *Proc. 21st IEEE Int. Conf. Netw. Protocols (ICNP)*, Oct. 2013, pp. 1–10.
- [122] F. Zhou, P. Yu, L. Feng, X. Qiu, Z. Wang, L. Meng, M. Kadoch, L. Gong, and X. Yao, "Automatic network slicing for IoT in smart city," *IEEE Wireless Commun.*, vol. 27, no. 6, pp. 108–115, Dec. 2020.
- [123] S. Mollahasani, M. Erol-Kantarci, and R. Wilson, "Dynamic CU-DU selection for resource allocation in O-RAN using actor-critic learning," 2021, *arXiv:2110.00492*.
- [124] A. Baumgartner, T. Bauschert, A. M. C. A. Koster, and V. S. Reddy, "Optimisation models for robust and survivable network slice design: A comparative analysis," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2017, pp. 1–7.
- [125] Y. Liu, J. Ding, and X. Liu, "A constrained reinforcement learning based approach for network slicing," in *Proc. IEEE 28th Int. Conf. Netw. Protocols (ICNP)*, Oct. 2020, pp. 1–6.
- [126] K. Abbas, M. Afaq, T. A. Khan, A. Rafiq, and W.-C. Song, "Slicing the core network and radio access network domains through intent-based networking for 5G networks," *Electronics*, vol. 9, no. 10, p. 1710, Oct. 2020.
- [127] Y. Liu, J. Ding, Z.-L. Zhang, and X. Liu, "CLARA: A constrained reinforcement learning based resource allocation framework for network slicing," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2021, pp. 1427–1437.
- [128] F. Rezazadeh, H. Chergui, and C. Verikoukis, "Zero-touch continuous network slicing control via scalable actor-critic learning," 2021, *arXiv:2101.06654*.
- [129] F. Rezazadeh, H. Chergui, L. Christofi, and C. Verikoukis, "Actor-critic-based learning for zero-touch joint resource and energy control in network slicing," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2021, pp. 1–6.
- [130] Q. Liu and T. Han, "When network slicing meets deep reinforcement learning," in *Proc. 15th Int. Conf. Emerg. Netw. Experiments Technol.*, Dec. 2019, pp. 29–30.
- [131] Q. Liu, N. Choi, and T. Han, "Constraint-aware deep reinforcement learning for end-to-end resource orchestration in mobile networks," in *Proc. IEEE 29th Int. Conf. Netw. Protocols (ICNP)*, Nov. 2021, pp. 1–11.
- [132] Q. Liu, T. Han, and E. Moges, "EdgeSlice: Slicing wireless edge computing network with decentralized deep reinforcement learning," in *Proc. IEEE 40th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Nov. 2020, pp. 234–244.
- [133] J. Achiam, D. Held, A. Tamar, and P. Abbeel, "Constrained policy optimization," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 22–31.
- [134] G. Sun, K. Xiong, G. O. Boateng, G. Liu, and W. Jiang, "Resource slicing and customization in RAN with dueling deep Q-network," *J. Netw. Comput. Appl.*, vol. 157, May 2020, Art. no. 102573.
- [135] I. Vila, J. Perez-Romero, O. Sallent, and A. Umbert, "A multi-agent reinforcement learning approach for capacity sharing in multi-tenant scenarios," *IEEE Trans. Veh. Technol.*, vol. 70, no. 9, pp. 9450–9465, Sep. 2021.
- [136] D. Peteiro-Barral and B. Gujarro-Berdiñas, "A survey of methods for distributed machine learning," *Prog. Artif. Intell.*, vol. 2, no. 1, pp. 1–11, Mar. 2013.
- [137] D. Janu, K. Singh, and S. Kumar, "Machine learning for cooperative spectrum sensing and sharing: A survey," *Trans. Emerg. Telecommun. Technol.*, vol. 33, no. 1, p. e4352, Jan. 2022.
- [138] T. Zhang and S. Mao, "An introduction to the federated learning standard," *GetMobile, Mobile Comput. Commun.*, vol. 25, no. 3, pp. 18–22, Jan. 2022.
- [139] C. Zhang, Y. Xie, H. Bai, B. Yu, W. Li, and Y. Gao, "A survey on federated learning," *Knowl.-Based Syst.*, vol. 216, Mar. 2021, Art. no. 106775.
- [140] A. M. Nagib, H. Abou-Zeid, and H. S. Hassanein, "Transfer learning-based accelerated deep reinforcement learning for 5G RAN slicing," in *Proc. IEEE 46th Conf. Local Comput. Netw. (LCN)*, Oct. 2021, pp. 249–256.
- [141] M. Yan, G. Feng, J. Zhou, and S. Qin, "Smart multi-RAT access based on multiagent reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4539–4551, May 2018.
- [142] Y. Abiko, T. Saito, D. Ikeda, K. Ohta, T. Mizuno, and H. Mineno, "Flexible resource block allocation to multiple slices for radio access network slicing using deep reinforcement learning," *IEEE Access*, vol. 8, pp. 68183–68198, 2020.
- [143] M. Yan, B. Chen, G. Feng, and S. Qin, "Federated cooperation and augmentation for power allocation in decentralized wireless networks," *IEEE Access*, vol. 8, pp. 48088–48100, 2020.
- [144] M. Zambianco and G. Verticale, "Intelligent multi-branch allocation of spectrum slices for inter-numerology interference minimization," *Comput. Netw.*, vol. 196, Sep. 2021, Art. no. 108254.
- [145] M. Zambianco and G. Verticale, "Mixed-numerology interference-aware spectrum allocation for eMBB and URLLC network slices," in *Proc. 19th Medit. Commun. Comput. Netw. Conf. (MedComNet)*, Jun. 2021, pp. 1–8.
- [146] Y. Hua, R. Li, Z. Zhao, X. Chen, and H. Zhang, "GAN-powered deep distributional reinforcement learning for resource management in network slicing," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 2, pp. 334–349, Feb. 2019.
- [147] H. Kim, K. Kim, Y. Han, and S. Yun, "A proportional fair scheduling for multicarrier transmission systems," in *Proc. IEEE 60th Veh. Technol. Conf.*, vol. 1, Sep. 2004, pp. 409–413.
- [148] H. Zhou and M. Erol-Kantarci, "Knowledge transfer based radio and computation resource allocation for 5G RAN slicing," in *Proc. IEEE 19th Annu. Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2022, pp. 617–623.
- [149] H. Zhou, M. Erol-Kantarci, and V. Poor, "Learning from peers: Deep transfer reinforcement learning for joint radio and cache resource allocation in 5G RAN slicing," 2021, *arXiv:2109.07999*.
- [150] H. Zhou, M. Elsayed, and M. Erol-Kantarci, "RAN resource slicing in 5G using multi-agent correlated Q-learning," in *Proc. IEEE 32nd Annu. Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2021, pp. 1179–1184.
- [151] R. Ferrús, J. Pérez-Romero, O. Sallent, I. Vilà, and R. Agustí, "Machine learning-assisted cross-slice radio resource optimization: Implementation framework and algorithmic solution," *ITU J. Future Evolving Technol.*, vol. 1, no. 1, pp. 1–18, 2020.
- [152] L. Liang, H. Ye, and G. Y. Li, "Spectrum sharing in vehicular networks based on multi-agent reinforcement learning," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 10, pp. 2282–2292, Oct. 2019.
- [153] Q. Liu, N. Choi, and T. Han, "OnSlicing: Online end-to-end network slicing with reinforcement learning," in *Proc. 17th Int. Conf. Emerg. Netw. Experiments Technol.*, Dec. 2021, pp. 141–153.
- [154] S. Messaoud, A. Bradai, O. B. Ahmed, P. T. A. Quang, M. Atri, and M. S. Hossain, "Deep federated Q-learning-based network slicing for industrial IoT," *IEEE Trans. Ind. Informat.*, vol. 17, no. 8, pp. 5572–5582, Aug. 2020.
- [155] S. Messaoud, A. Bradai, and E. Moulay, "Online GMM clustering and mini-batch gradient descent based optimization for industrial IoT 4.0," *IEEE Trans. Ind. Informat.*, vol. 16, no. 2, pp. 1427–1435, Feb. 2019.
- [156] M. Z. Hassan, M. J. Hossain, J. Cheng, and V. C. M. Leung, "Joint throughput-power optimization of fog-RAN using rate-splitting multiple access and reinforcement-learning based user clustering," *IEEE Trans. Veh. Technol.*, vol. 70, no. 8, pp. 8019–8036, Aug. 2021.
- [157] Y. Yu, X. Si, C. Hu, and Z. Jianxun, "A review of recurrent neural networks: LSTM cells and network architectures," *Neural Comput.*, vol. 31, no. 7, pp. 1235–1270, Jul. 2019.
- [158] M. Vincenzi, E. Lopez-Aguilera, and E. Garcia-Villegas, "Maximizing infrastructure providers' revenue through network slicing in 5G," *IEEE Access*, vol. 7, pp. 128283–128297, 2019.
- [159] M. Vincenzi, E. Lopez-Aguilera, and E. Garcia-Villegas, "Timely admission control for network slicing in 5G with machine learning," *IEEE Access*, vol. 9, pp. 127595–127610, 2021.



YASER AZIMI received the B.Sc. and M.Sc. degrees in software and network engineering from Urmia University, Iran, where he is currently pursuing the Ph.D. degree. His research interests include AI-enabled wireless networking, next-generation cellular networks, data analytics, and network security.



HASHEM KALBKHANI received the B.Sc., M.Sc., and Ph.D. degrees in electrical engineering from Urmia University, Iran. He is currently an Assistant Professor at the Faculty of Electrical Engineering, Urmia University of Technology, Urmia, Iran. His research interests include wireless networks, machine learning, and signal processing.



SALEH YOUSEFI received the B.S. and M.S. degrees in computer engineering (in the hardware engineering field) and the Ph.D. degree in computer engineering (in the networking field) from the Iran University of Science and Technology, Tehran, Iran, in 1999, 2002, and 2008, respectively. He is currently an Associate Professor with the Computer Engineering Department, Urmia University, Urmia, Iran. His research interests include mobile and wireless networks, machine learning and optimization techniques for computer/telecommunication networks, and vehicular networks.



THOMAS KUNZ (Senior Member, IEEE) received the dual degree (Hons.) in computer science and business administration and the Dr. Ing. degree in computer science from the Technical University of Darmstadt, Germany, in 1990 and 1994, respectively. He is currently a Professor with the Department of Systems and Computer Engineering, Carleton University, Canada. He heads the Mobile Computing Group, researching wireless network architectures, network protocols, and middleware layers for innovative wireless applications. He is the author or coauthor of more than 70 journals and 190 conference papers. He is a Senior Member of the Association for Computing Machinery (ACM). He has received a number of awards and best paper prizes.

...