Aalto University

Ahmad, I.; Shahabuddin, S.; Malik, H.; Harjula, E.; Leppänen, Teemu; Lovén, L.; Anttonen, Antti; Sodhro, A. H.; Alam, M. Mahtab; Juntti, M.; Ylä-Jääski, A.; Sauter, T.; Gurtov, A.; Ylianttila, Mika; Riekki, Jukka

Machine Learning Meets Communication Networks: Current Trends and Future Challenges

Machine Learning Meets Communication Networks: Current Trends and Future Challenges

IJAZ AHMAD[1], (Member, IEEE), SHARIAR SHAHABUDDIN[2],
HASSAN MALIK[3], (Member, IEEE), ERKKI HARJULA[4], (Member, IEEE),
TEEMU LEPPÄNEN[5], (Senior Member, IEEE), LAURI LOVÉN[5], (Senior Member, IEEE),
ANTTI ANTTONEN[1], (Senior Member, IEEE), ALI HASSAN SODHRO[6], (Member, IEEE),
MUHAMMAD MAHTAB ALAM[7], (Senior Member, IEEE), MARKKU JUNTTI[4], (Fellow, IEEE),
ANTTI YLÄ-JÄÄSKI[8], (Member, IEEE), THILO SAUTER[9,10], (Fellow, IEEE),
ANDREI GURTOV[11], (Senior Member, IEEE), MIKA YLIANTTILA[4], (Senior Member, IEEE),
AND JUKKA RIEKKI[5], (Member, IEEE)

[1]VTT Technical Research Centre of Finland, 02044 Espoo, Finland
[2]Nokia, 02610 Espoo, Finland
[3]Computer Science Department, Edge Hill University, Ormskirk L39 4QP, U.K.
[4]Centre for Wireless Communications, University of Oulu, 90570 Oulu, Finland
[5]Center for Ubiquitous Computing, University of Oulu, 90570 Oulu, Finland
[6]Department of Computer and System Science, Mid-Sweden University, Östersund, Sweden
[7]Thomas Johann Seebeck Department of Electronics, Tallinn University of Technology, 12616 Tallinn, Estonia
[8]Department of Computer Science, Aalto University, 02150 Espoo, Finland
[9]Institute of Computer Technology, TU Wien, 1040 Wien, Austria
[10]Department for Integrated Sensor Systems, Danube University Krems, 2700 Wiener Neustadt, Austria
[11]Department of Computer and Information Science, Linköping University, 58183 Linköping, Sweden

Corresponding author: Ijaz Ahmad (ijaz.ahmad@vtt.fi)

ABSTRACT The growing network density and unprecedented increase in network traffic, caused by the massively expanding number of connected devices and online services, require intelligent network operations. Machine Learning (ML) has been applied in this regard in different types of networks and networking technologies to meet the requirements of future communicating devices and services. In this article, we provide a detailed account of current research on the application of ML in communication networks and shed light on future research challenges. Research on the application of ML in communication networks is described in: i) the three layers, i.e., physical, access, and network layers; and ii) novel computing and networking concepts such as Multi-access Edge Computing (MEC), Software Defined Networking (SDN), Network Functions Virtualization (NFV), and a brief overview of ML-based network security. Important future research challenges are identified and presented to help stir further research in key areas in this direction.

INDEX TERMS Communication networks, machine learning, physical layer, MAC layer, network layer, SDN, NFV, MEC, security, artificial intelligence (AI).

I. INTRODUCTION

The security, availability and performance demands of new applications, services and devices are increasing at a pace higher than anticipated. Real-time responsiveness in application areas like e-health, traffic, and industry requires communication networks to make real-time decisions autonomously. Such real-time autonomous decision-making requires that

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

the network must react and learn from the environment, and control itself without human interventions. However, communication networks have until now taken a different path. Traditional networks rely on human involvement to respond manually to changes such as traffic variation, updates in network functions and services, security breaches, and faults. Human-machine interactions have resulted in network downtime [1], have opened the network to security vulnerabilities [2], and lead to many other challenges in current communication networks [3], [4].

The requirement for human interaction or manual configuration constitutes a major hindrance for a network to use its past experiences to adapt to changing requirements. The general idea is to predict the (future) behavior of a service, network segment, user or User Equipment (UE), and tune the network at run-time based on this information. For instance, the movement trajectory of a user can be predicted using straight-forward mechanisms such as measuring the signal strength between consecutive base stations, to minimize the handover latency. Moreover, the requirements of near future services and networks, such as vehicular communication and the Internet of Things (IoT) in Fifth Generation (5G), need the introduction of intelligent network operations [5], [6]. Intelligence in communication networks can be characterized as follows: the network system pursues its goals autonomously, i.e., without human intervention, in the presence of uncertainty that is caused by lack of information [7]. The system reacts and adapts to changes in its environment and learns from experience, based on information collected during its operation.

Machine Learning (ML) with its many disciplines has been on the forefront for automation, embodying intelligence in machines to minimize costs and errors, and to increase efficiency. ML is a sub-field of AI that is concerned with *"the programming of a digital computer to behave in a way which, if done by human beings or animals, would be described as involving the process of learning"* [8]. An ML system improves its performance on future tasks after making observations of its environment [7]. Such observations are represented by *data*, and the sources of the data are referred to as *sensors*. Analysing the data, an ML system creates and potentially updates an internal *model* of its operating environment. In communication systems, ML enables systematic mining and extraction of useful information from traffic data to automatically find the correlations that would otherwise be too complex for human experts [9].

ML systems can be broadly categorized into: supervised learning [7], unsupervised learning [10], [11], semi-supervised learning [12], and reinforcement learning [13], [14]. Given the *input* data ("independent variables", "covariates"), an ML system makes predictions of *output* data ("dependent variable", "response"). Instead of hard-coded rules, the system estimates the mapping between input and output by analysing labeled *training data*, that is, data containing examples of actual inputs and corresponding outputs. The process of estimating the mapping between input and output is referred to as *supervised learning* [7, p. 695]. In unsupervised learning, given input data, the system learns patterns and associations occurring in the data. Unsupervised learning techniques include *clustering*, i.e. partitioning data into separate subsets according to varying criteria [15], [16], and *dimensionality reduction*, where multidimensional data is projected, by any of a multitude of methods, to a lower-dimensional space [17].

In semi-supervised learning, given input data, a small part of which is labelled with the corresponding output data,

the system learns either how to label the unlabelled input data (*transductive learning*), or to predict the output given new input data (*inductive learning*) [12]. As labelling data is often expensive, semi-supervised learning offers in some cases a cost-effective way of training a learning agent. In reinforcement learning, given input data, the system learns to take *actions* that maximise a cumulative *reward* [13], [14]. All these types of learning enrich the systems with intelligence to improve its future performance based on available information.

Communication networks need to utilize the various disciplines, technologies, concepts and methods of ML for many reasons. The major reasons being to mitigate the risks involved with human-control, and empower the networks to self-control, to adapt, and to heal themselves with the changing user, traffic and network conditions, as well as their dynamic requirements [18], [19]. The first use of ML in telecommunication was realized in a network traffic controller in 1990, called NETMAN [20]. NETMAN combined two machine learning techniques, i.e, explanation-based learning and empirical learning, as described in [20]. The main aim was to maximize call completion using ML techniques in circuit-switched networks. Since then, networks and the services using the networks have drastically changed with the emergence of IP-based technologies. The domain of ML, on the other hand, has progressed and widened dramatically with the emergence of fast computing capabilities.

New technological concepts are introduced for communication networks at an increasing pace. These new concepts have an effect from the physical layer to the application layer and beyond the layered architecture. The disciplines of ML can further improve the performance of such technologies. For example, massive Multiple-Input Multiple-Output (MIMO) systems in the physical layer significantly improve the spectral and energy efficiency of wireless networks [21], [22]. ML techniques, in turn, can significantly improve the performance of massive MIMO systems in, for example, avoiding the challenges of pilot signal contamination [23]. The disciplines of ML can be used to improve the performance of cognitive radios in spectrum sharing, and in heterogeneous access networks for sharing different resources [24].

Furthermore, Software Defined Networking (SDN), Network Function Virtualization (NFV), and Multi-access Edge Computing (MEC) are proving to be stepping stones in dynamic and opportunistic network operations. However, these technologies introduce several challenges. For instance, the centralized control platform in SDN and the hypervisor or virtual resource manager in NFV can become potential bottlenecks for entire networks these entities manage. It is important to note that these entities might reside in MEC nodes on the edge of mobile networks. Hence, there is a need to investigate mechanisms that enable the network to learn from the environment, comprehend potential challenges (e.g., reasons or situations leading to bottlenecks), and tune or configure itself at run-time to avoid or mitigate the risks of

these challenges. ML is one such candidate that can enable communication networks with such capabilities [19], [25].

Along with the improvement in communication technologies, several disciplines with distinct algorithms and tools have emerged in ML. These algorithms and tools of ML are actively investigated for numerous use-cases, diverse services, and technologies in communication networks. Therefore, this is the right time to shed light on the convergence of the solutions, algorithms, and tools of ML with the advanced technological concepts in communication networks. In this article, we provide a detailed survey of the solutions, algorithms, and tools of ML in communication networks. Beginning from the physical layer, the use of ML in MAC and network layers, and in technologies such as SDN, NFV, and MEC is described. Future research directions are drawn to help the research community to circumvent the challenges of future services (e.g. for massive IoT) and technologies (e.g. NFV) using ML and grasp attention to bridging the existing gaps. Various surveys on the topic have recently appeared covering some specific applications of communication technologies. However, the major difference between the existing articles and this article is that our work provides an up-to-date overview of the merger of the disciplines of ML, the different communication layers, and emerging technological concepts in communication networks. We also provide a summary of the existing survey articles, and the gaps we identify in this article.

This article is organized as follows: Section II describes the related work. The survey of ML applications in the three layers in communications networks, i.e., physical, MAC, and network layers is presented in Sections III, IV, and V respectively. The application of ML in SDN and NFV is discussed in Section VI. ML for edge computing is discussed in Section VII, and an overview of using ML for network security is provided in Section VIII. Interesting insights into the future of using ML for communication networks are provided in Section IX, and the paper is concluded in Section X. For smooth readability, the most used acronyms are presented in full in Table. 1.

II. STATE-OF-THE-ART AND OUR CONTRIBUTIONS

Due to the visible benefits of intelligent network operations, several studies exist including survey articles, as presented in Table 2 and Table 3. Most of the articles elaborate the concepts and techniques of ML, its applicability in wireless networks, and possible future directions as highlighted in Table 2 and Table 3. Researchers have been looking into different concepts that can be used to embed intelligence into communication networks. For example, bio-inspired networking comprises a class of strategies for scalable and efficient networking in uncertain conditions, profiting from the governing dynamics and fundamental principles of biological systems [56]. A survey on bio-inspired networking is presented in [27]. The article [27] describes fundamental challenges in communication networks and how biological concepts can be used to mitigate those challenges by highlighting the

TABLE 1. List of most common abbreviations.

Acronym	Full term(s)
5G	Fifth Generation
AI	Artificial Intelligence
ANN	Artificial Neural Networks
AR	Autoregression
CIP	Cooperative Interference Prediction
CSI	Channel State Information
CNN	Convolutional Neural Networks
D2D	Device to Device
DBA	Deep Belief Architecture
DL	Deep Learning
DNN	Deep Neural Networks
DNS	Domain Name System
DOA	Direction-of-Arrival
DPD	Digital Pre-Distortion
DPI	Deep Packet Inspection
DT	Decision Trees
DRL	Deep Reinforcement Learning
EC	Edge Computing
ELM	Extreme Learning Machine
FC	Fog Computing
eMMB	enhanced Mobile Broadband
GLM	Generalized Linear Models
GPS	Global Positioning System
HMM	Hidden Markov Models
ICN	Information-Centric Network
IDS	Intrusion Detection System
IoT	Internet of Things
IPS	Intrusion Prevention System
KDN	Knowledge Defined Networking
KNN	K-Nearest Neighbors
KP	Knowledge Plane
KPI	Key Performance Indicator
LTE	Long Term Evolution
LMS	Least Mean Square
LWPAN	Low Power WAN
MAC	Medium Access Control
MANET	Mobile Ad-hoc Network
MEC	Mobile Edge Computing
mIoT	massive IoT
MIMO	Multiple-Input Multiple-Output
ML	Machine Learning
mMTC	massive Machine Type Communication
NFV	Network Function Virtualization
NLP	Natural Language Processing
OFDM	Orthogonal Frequency Division Multiplexing
OSPF	Open Shortest Path First
PA	Power Amplifier
QL	Q-Learning
QoE	Quality of Experience
QoS	Quality of Service
RAN	Radio Access Network
RBF	Radial Basis Function
RF	Radio Frequency
RL	Reinforcement Learning
RNN	Recurrent Neural Networks
SDN	Software Defined Networking
SDR	Software Defined Radio
SNR	Signal to Noise Ratio
SON	Self-Organized Network
SVM	Support Vector Machine
uRLL	ultra-Reliable Low Latency
V2V	Vehicle to Vehicle
VNF	Virtual Network Function
WAN	Wide Area Network
WSN	Wireless Sensor Networks

existing research efforts. The significance of the work is in embodying intelligence in communication networks, much like the biological behavior of living organisms. The work is focused on linking research in bio-inspired approaches and nano-communications. A survey on bio-inspired mechanisms for self-organizing networks (SON) is presented in [29].

TABLE 2. Existing survey and literature review articles.

Publication year	References	Main focus of the survey articles	Scope limitations
2008	[26]	Techniques for Internet traffic classification using ML	Main focus on traffic classification techniques
2010	[27]	Bio-inspired networking and nano-communications	The main focus is on nano-networks, sensors and actuators
2010	[28]	State-of-the-art of ML-based intrusion detection systems	Focuses only on improving security using ML
2014	[29]	Bio-inspired and swarm intelligence-based networking	Limited discussion on novel technologies, e.g., SDN, NFV
2014	[30]	A survey on using ML for WSNs	The article is focused on sensor networks only
2015	[31]	Opportunities and challenges in ML for HetNets	Short article and limited in scope
2016	[32]	A survey on data mining techniques for IDSs	Focuses only IDSs using data mining
2017	[33]	A survey on DL for intelligent network traffic control	Mainly focused on the network layer
2017	[34]	A survey on ML and Big Data mechanisms for IoT	Broadly covering the scope of IoT and its requirements
2018	[35]	A survey on using ML for securing IoT and WSNs	Focuses only on security of IoT and WSNs using ML
2018	[36]	A survey on security of ML techniques	Limited in scope to discussion on security of ML
2018	[37]	A survey on DL for Intelligent Wireless Networks	Lacks recent trends in MEC, SDN, and NFV, etc.
2018	[38]	A survey of ML techniques for optical networks	Limited in scope to optical networks
2018	[39]	An overview of ML techniques for optical networks	Limited in scope for wireless networking technologies
2018	[40]	Big data analytics and ML in future wireless networks	Not comprehensive on networking technologies
2018	[41]	Overview of ML applications in communication systems	Mainly focused on edge, cloud and physical layer
2019	[9]	An extensive survey on DL for wireless networks	Lacks discussion on emerging technologies, e.g., SDN, MEC
2019	[42]	A state-of-the-art study on ML in SDN	Focuses only on SDN
2019	[43]	Application and challenges of RL and DRL in 5G	Focuses only RL and DRL and does not cover all the layers
2019	[44]	Application of DRL in communication networks	Limited in scope on MEC, SDN, NFV, etc.
2019	[45]	ML techniques in different domains of wireless networks	More focused on the basics of ML and few key areas
2019	[46]	A survey on the application of ML in SDN concepts	More focused on ML in general and communication networks
2019	[47]	Workflow of ML in networking	More focused on the basics of ML than its application areas
2019	[48]	Tutorial on using ANNs in wireless networks	Describes the basics of ANNs for wireless networks
2019	[49]	A survey on RL in communication networks	Does not cover any layer in depth
2019	[50]	Study of ML in 5G and beyond	Focused on ML and few use-cases of 5G
2020	[51]	An overview of ML techniques in wireless networks	Very limited discussion on novel technologies and security
2020	[52]	A survey on ML techniques for edge and cloud platforms	Limited to ML in MEC platforms
2020	[53]	Comparison of ANNs, SVM, KNN, and logic regression	Limited to only four techniques and the issue of latency
2020	[54]	Data driven optimization using ML	More focused on big data analytics
2020	[55]	A survey of federated learning in MEC platforms	Limited to discussion and survey on MEC only
2020	[25]	White paper on the role of ML in 6G	Limited discussion on existing technologies

TABLE 3. Existing survey and literature review articles with main focus highlighted and compared to this article.

Publication Year	Reference	Focus	Network layers covered			Beyond layers		
			Network	MAC	Physical	SDN-NFV	MEC	Security
2008	[26]	Internet traffic classification	✓					✓
2010	[28]	ML techniques for improving security						✓
2010	[27]	Bio-inspired networking	✓					
2014	[30]	Techniques of ML for sensor networks		✓				✓
2014	[29]	Swarm intelligence-based networking	✓	✓	✓			
2015	[31]	ML for HetNets, opportunities & challenges	✓		✓		✓	
2016	[32]	Data mining techniques for IDS						✓
2017	[34]	ML and BD techniques for IoT						✓
2017	[33]	Intelligent traffic control using ML	✓					
2018	[39]	ML for optical networks		✓	✓			
2018	[40]	BDA and ML for wireless networks	✓		✓			
2018	[37]	DL for intelligence in wireless networks	✓	✓	✓			✓
2018	[36]	Security of ML techniques and algorithms						✓
2018	[35]	ML for IoT (sensor networks) security						✓
2019	[9]	DL for wireless networks	✓		✓			✓
2019	[42]	ML mechanisms for SDN	✓			✓		✓
2019	[44]	DRL in communication networks	✓	✓	✓			✓
2019	[43]	DRL in 5G and beyond				✓	✓	
2019	[46]	ML in SDN concepts	✓			✓		
2019	[50]	ML in 5G and beyond		✓	✓			
2020	[54]	Proactive optimization in 5G using ML		✓			✓	
2020	[53]	Latency improvement through ML		✓				
2020	[51]	ML history in wireless networks		✓	✓			
2020	[52]	ML in edge and cloud					✓	
2020	[55]	A survey of federated learning in MEC					✓	
2020	[25]	White paper on the role of ML in 6G		✓	✓			✓
Our article		ML for communication networks	✓	✓	✓	✓	✓	✓

Leading to the recent development in ML, an introduction of ML with applications to communication systems is provided in [41]. The article introduces the key concepts of ML, mainly focusing on supervised and unsupervised learning, and discusses using ML for the physical layer at the edge and cloud computing. The state-of-the-art techniques

in addition to the opportunities and challenges of using ML in Heterogeneous Networks (HetNets) are discussed in [31]. The study of [31] describes ML-based techniques for smart HetNet infrastructure and systems paying focus on the challenges of self-configuration, self-healing, and self-optimization. A historical background of ML, spanning 30 years, with an overview of its applications in wireless networks is discussed in [51].

How to apply ML to networking is explained in [47] with a basic workflow defined in several steps. The article [47] sheds light on the recent development in the field and focuses on measurements, prediction, and scheduling for networking leveraging ML. The applications of ML, specifically Artificial Neural Networks (ANN), in wireless networks are described in [48]. Being tutorial in nature, the article [48] provides a thorough description of ANN algorithms, and how ANNs can solve the challenges in wireless communications. A deep overview of using various types of ANNs in Unmanned Aerial Vehicles (UAVs), wireless virtual reality, MEC, spectrum management, and IoT is provided. The main focus in the layered network architecture, though, is on the physical layer.

A survey on the application of machine learning specifically supervised and unsupervised learning, reinforcement learning, Deep Neural Networks (DNNs), and transfer learning, in wireless networks is presented in [45]. Applications are classified into resource management in the MAC layer, mobility management and networking in the network layer, and localization in the application layer. The article [45] also instructs which type of ML technique or algorithms to use in different applications based on suitability conditions. Insights into using ML in novel technologies such as slicing and dealing with big data provide interesting future research directions.

A study on state-of-the-art Deep Learning (DL) for mobile networks is presented in [9]. The authors present the background of DL and explore the application of DL for mobile networks extensively. Insights on tailoring the concepts of DL for mobile networks along with future research perspectives make the article [9] an interesting read. The article [9], however, does not elaborate the potential uses of DL in emerging networking technologies such as SDN and MEC. DL methods for enhancing the performance of wireless networks are studied in [37]. The application of DL methods in different layers of the network, in intrusion prevention and other network functions such as fog computing, is studied.

There are also several articles that focus on the concepts of ML in a particular area of wireless networks (Table 2 and Table 3). For example, [33] describes the state-of-the-art DL for intelligent network traffic control systems. In [42], ML mechanisms for SDN are elaborated with their shortcomings and future research directions. Focusing on the learning problems in cognitive radios, [57] argues for the importance of using ML to achieve real cognitive communication systems and summarizes the state-of-the-art literature on ML for cognitive radios. Cognitive radios use intelligence to efficiently use radio resources, such as the most important and scarce resource of radio frequency.

In [40], the authors discuss the mechanisms of utilizing available data to increase efficiency in next-generation wireless networks leveraging ML. The work describes data sources and drivers for adopting data analytics and ML to enable the network to be self-aware, self-adaptive, proactive and prescriptive. Data-driven coverage and capacity optimization are explained with benefits in load-balancing, mobility, and congestion control, beam-forming, etc. Furthermore, [38] and [39] provides an overview of ML techniques for optical networks respectively. The classification of Internet traffic using ML is discussed in [26]. An overview of ML in Wireless Mesh Networks (WMNs) is presented in [58]. To solve the most important challenges of WMNs such as high bandwidth and coverage, expected QoS and security, and network management, key ML techniques are discussed.

Intelligence is a focal point in IoT due to the increase in the number of deployed devices and the humongous growth projected [34]. In [34], the authors review learning and big data analysis approaches related to IoT. Using ML to cope with the dynamic nature of Wireless Sensor Networks (WSNs), [30] discusses the potential of ML with its applications and algorithms in the area. The article describes the advantages and disadvantages of various algorithms in particular scenarios and presents a guide for WSN designers to select the right ML algorithms. A brief survey on using ML for securing IoT and WSNs is presented in [35]. Various types of attacks on WSNs are presented followed by the overview of different solutions that use ML to secure the networks. An overview of DL approaches for IoT is presented in [59].

A survey on data mining and ML techniques for intrusion detection systems is presented in [32]. The article first describes the methods of data mining and ML and then provides an overview of various articles that use those methods for improving the cyber-security landscape. The authors argue that no single technique or method can be considered generally to be the best approach. Different algorithms can be selected for different scenarios based on the type of attack or security vulnerability. Similarly, the completeness of the data set is of paramount importance; considering both network- and kernel-level data, and if possible the network data must be augmented with the operating system kernel-level data.

An interesting insight into ML is provided in [36]. The authors describe that even though the concepts and techniques of ML have been proposed and used in various fields including cyber-security, the techniques might be vulnerable to security lapses. The article [36] provides a deep overview of the security of various techniques and algorithms of ML and describes defensive measures to secure those. A study of DNS traffic for cyber-security is conducted in [60]. The authors reveal that most of the systems using ML approaches consume longer times than required for real-time security measures. Similarly, [28] studies the state-of-the-art intrusion detection systems (IDS) that use ML. The authors observe the

difficulties and challenges in using ML for IDS compared to other applications of ML.

Reinforcement learning has been used in communication networks to enable network elements to obtain optimal policies for taking, usually a limited number of, decisions or actions. As the number of states or actions grow, Deep Reinforcement Learning (DRL) is used to improve the network performance under uncertain conditions. DRL is a combination of reinforcement learning and deep learning that can meet the challenging dynamics in 5G and beyond [43]. In [43], the applications of DRL to improve energy efficiency, resource utilization (utility maximization), and performance of key technologies in 5G such as network slicing, edge caching, and computation offloading is discussed. Furthermore, various applications of DRL in wireless networks are discussed in [44].

A survey on the use of ML in MEC, mainly to solve the challenges of resource scarcity through these technologies, is presented in [52]. The article describes the main problems of MEC platforms and communication networks-specific ongoing ML research activities. Research on using ML to solve the challenges of task offloading, resource allocation, server deployment, and overhead management are elaborated. The authors conclude that for task offloading ML is mostly used in classification and for resource allocation ML is mostly used in optimization. Similarly, for server deployment ML is mostly used in clustering, and for overhead management ML is mostly used in state transitions. Furthermore, the potential of ML at the edge of wireless networks is also described in [61]. A survey on using federated learning (FL) in MEC is presented in [55].

A comparative study of ML techniques for improving latency in communication networks is carried out in [53]. The authors survey ML techniques in technical depths for bandwidth allocation decisions in converged networks consisting of end-users, machines and robots. The article concludes that ANN has superior uplink latency performance compared to other techniques such as SVM, KNN, and logic regression. A survey of online data-driven proactive 5G network optimization using ML is presented in [54]. The article focuses on the potential of big data analytics along with methods and technologies for proactive network optimization using machine learning in future networks.

A survey on the use of ML in 5G and beyond 5G mobile wireless networks is presented in [50]. The article provides an overview of the fundamentals of ML with its disciplines, mainly the types of learning such as supervised, unsupervised and reinforcement learning, and its applications in wireless networks. Related to 5G, the article discusses the potential use of ML in enhanced mobile broadband, massive machine-type communications, and ultra-reliable low latency communications. The article also discusses the role of ML in beyond 5G or 6G systems. The role of ML in 6G is also discussed in a white paper presented in [25]. The white paper elaborates the potential uses of ML in futuristic technologies that could be used in 6G.

However, none of the existing articles take a deep look into the recent technological developments such as SDN, NFV, and MEC which can leverage ML for a variety of purposes in future communication networks. Table 2 summarizes the existing survey and related articles that outline the concepts of different disciplines of ML for communication networks. The articles are organized with respect to a year, whereas the scope and limitations of each are highlighted. Table 3 presents a comparison of the relevant survey articles, highlighting what is missing from the whole picture, and presenting the differences between this article and the existing ones. The major difference between the existing articles and this article is that our work provides an up-to-date overview of the merger of disciplines of ML with different parts (layers), and novel technological concepts in communication networks. We describe ML in communication networks for the technologies of physical layer, MAC layer, network layer, and the novel concepts and technologies in communication networks such as massive MIMO, Softwarized network functions enabled by SDN and NFV.

MEC can support ML by providing computation for the learning or analysis algorithms in the edge or near the data sources. This article describes how MEC helps the deployment of ML techniques, and how ML can be used to optimize MEC platforms. This paper also discusses how ML can be used to overcome the challenges of security in future networks, as well as provide future directions on improving the use of ML in different domains and technologies of communication networks. The main contributions of this article are summarized as follows:

1) A summary of existing survey articles under the theme of applications of ML in wireless networks is provided.
2) The needs and applications of ML in physical, MAC and network layers are presented to provide the state-of-the-art applications of ML in each layer.
3) The applications of ML in novel technologies such as MEC, SDN, and NFV are presented.
4) An overview of using ML for network security is presented.
5) Interesting insights into the shortcomings of existing techniques to motivate future research are presented to stir further research in this direction.

In a nutshell, this article provides a clear elaboration of the need of intelligence in communication networks, the current trends used for embedding ML-based intelligence in communication networks, and the future directions on how to improve networks to embrace ML, and how to improve the techniques and tools of ML for its efficient use in communication networks. In the following section, we begin from the application of ML in the physical layer.

III. ML FOR LAYER ONE: PHYSICAL LAYER
Physical Layer, also commonly referred to as Layer 1, is the lowest layer of a communication system that deals with the optimal transmission, receiver processing and accurate modeling of channels to ensure reliable data communication over

physical channels. Communication signal processing algorithms for a physical layer are typically designed analytically by applying mathematical optimization, statistics, and information theory. For example, estimators such as maximum likelihood or minimum mean square error have been typically used for signal processing. These estimators are developed using estimation theory, which is a branch of statistics. Similarly, most physical layer algorithms try to solve an optimization problem which deals with maximizing or minimizing a real function. Thus, methods suitable to solve these optimization problems, such as dual ascent, co-ordinate descent, are also widely used for signal processing of physical layer. Optimal layer 1 algorithms are usually derived and realized for relatively simple conditions such as stationary channels and systems, linear processing and Gaussian noise. Therefore, for a practical communication system with non-linearity and imperfections, ML can potentially provide gains over existing physical layer algorithms. In [62], the authors discussed the application of DL for several physical layer applications. The opportunities and challenges of DL for the physical layer are discussed in [63]. In this section, we focus on the application of ML for different parts of the physical layer. We cover Digital Pre-Distortion (DPD) and beamforming, while they were out of the scope of [62] or [63]. In addition, we discuss ML-based end-to-end communication systems, i.e., how an ANN is used to replace the entire receiver. A summary of important ML applications for the physical layer is presented in Table 4.

A. APPLICATION OF NEURAL NETWORKS FOR DIGITAL PRE-DISTORTION

The presence of non-linear distortion can pose severe consequences for digital systems. The primary cause of the non-linear distortion in digital radio systems is the high-power amplifier (HPA). The obvious solution to this problem is to use linear class A amplifiers or to operate the HPA far from the saturation point. However, this solution leads to expensive, bulky and inefficient HPAs. A more efficient solution is to apply a non-linear filter at the transmitter which generates the inverse of an HPA response. This technique to linearize the effects of an HPA by pre-distorting the baseband digital signal is commonly known as DPD. A typical transmitter with a DPD is illustrated in Fig. 1. The baseband input stream $a(KT_0)$ is subjected to a pre-distortion filter which essentially inverts the effects of the HPA. The pre-distorted signal $b(KT_0)$ goes through modulator and demodulator pulse shaping filters with impulse response $g_T(t)$ and $g_R(t)$. The impact of the HPA can be analyzed comparing the output of the demodulator $c(KT_0)$ and pre-distortion filter output $b(KT_0)$. The weight of the pre-distortion filter can be adapted based on the difference of $c(KT_0)$ and $b(KT_0)$. However, the performance of the pre-distortion filter depends on how accurately the HPA has been modeled. As the response of an HPA is a non-linear continuous function, the conventional memory polynomial models are not always accurate. In addition, an HPA has to
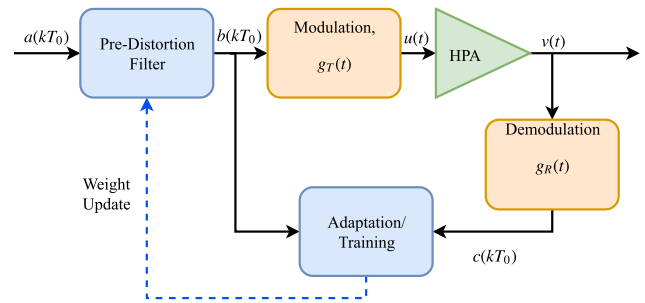

FIGURE 1. A conventional transmitter system with pre-distortion.

support different types of signals which makes the polynomial modeling more difficult. ANNs have been proven as an efficient tool to implement non-linear mappings. It has been shown that feed-forward ANNs with sufficient neurons are universal approximators for an arbitrary continuous function [64]. Therefore, ANN is a natural choice for pre-distortion, which can be trained to generalize a HPA model for different types of carriers.

Several efforts have been made in the last three decades to apply ANN to design sophisticated DPD algorithms. One of the earliest ANN-based DPD design has been proposed in [65]. The pre-distortion filter of Fig. 1 is replaced with an ANN that consists of three inputs, one hidden layer with five neurons and one output. The ANN-based DPD can decrease the Mean-Square Error (MSE) by -29.53 dB. Another early ANN-based DPD has been proposed in [66]. The authors proposed an ANN-based DPD for memoryless non-linear HPA model described by [67]. This work used an ANN, similar to that of [65], which works as a pre-distortion filter. In addition, an ANN is used for training which replaces the adaptation block of Fig. 1. However, the DPD of [66] requires a large number of parameters resulting in a significant computational load. An improved design was proposed in [68] that simplifies the pre-distortion problem by utilizing the knowledge of Radio Frequency (RF) amplifier response properties.

In [69], the author proposed an ANN-based pre-distortion technique for satellite communications. The proposed DPD consists of two separate ANNs. The first ANN is used for Travelling Wave Tube (TWT) amplifier transfer function modeling and the second ANN is used for the inverse of the transfer function. Each ANN contains two-layers and $N = 10$ neurons. Contrary to the use of a known normalized TWT, the authors assumed a non-normalized solid-state HPA with unknown parameters and with intrinsic adaptive behavior in [70]. Each ANN consists of one hidden layer with $N = 10$ neurons and one output layer with a single neuron. In [71], the first experimental ANN-based DPD is demonstrated. The proposed method uses a single Multilayer Perceptron (MLP) for both amplitude and phase correction and can achieve a 25 dB linearity improvement which was evidenced by the measurement results. The experimental setup of the four-layer MLP is trained with ANN toolboxes of MATLAB. A Double Input Double Output (DIDO) two-layer forward ANN combined with a tapped delay line for a HPA with

TABLE 4. ML for Physical Layer.

Application Domain	References	ML Algorithms	Advantage	Disadvantage
Digital Pre-distortion	[65]	ANN	One of the earliest ANN based DPD implementation	Complete performance evaluation is lacking, i.e. total degradation vs back-off
	[66]	Dual ANN	Preliminary results indicates that ANN has potential use in pre-distortion	Requires large number of parameters resulting in significant computational load
	[68]	Dual ANN	Improvements over [66], Simplifies pre-distortion by utilizing HPA response properties	The robustness of the ANN solution for different HPAs and operating regime has not been investigated
	[69]	Dual ANN	Proposed solution is adaptive and can be applied to different memoryless HPAs	Performance for non-linear HPAs with memory is unknown
	[70]	Dual ANN	Compares two new ANN adaptive pre-distortion technique	Comparison with contemporary ANN DPDs from other papers is lacking
	[71]	ANN	First experimental demonstration of ANN DPD, which achieves a 25 dB gain	Performance is dependent on linear gain value
	[72]	DIDO ANN with tapped delay line	Reduces computation load and provides a 15 DB improvement	Spectra spreading is affected by incorrectly corrected phase distortion
	[73]	CasCoR ANN	Distortion caused by memory HPA of OFDM can be greatly compensated	No comparison with contemporary work is provided
	[75]	RBFNN	Requires less training than contemporary implementations	Performance is not satisfactory in smaller bandwidths
	[76]	RVTDNN	FPGA implementation for behavioral modeling	The motivation behind selecting RVTDNN is not clarified
	[77]	RNN	Linearize a GaN class F HPA	Experimental setup is not properly explained
Channel Coding	[79]	ANN	Earliest ANN based channel coding and decoding	Comparison with conventional channel coding methods is not presented
	[81]	ANN	Outperforms conventional decoders in a jamming environment	Performance curve is not simulated with realistic channel model
	[82]	Dual ANN	Solution of Hamming codes	Fails to produce correct error pattern for 3 or more errors
	[83], [85]	ANN and RNN	Performance matches a Viterbi decoder	Performance is not presented for non-stationary channel
	[84]	ANN	Can be used to improve transmission reliability	Complexity analysis does not include training
	[89]	DL	Improves belief propagation algorithm	Performance gain is not significant
	[90]	DL	Results in non-iterative and highly parallel polar decoder	Performance degrades due to partition
	[91]	CNN based DL	CNN is used to remove estimation errors	Loss function is not completely equivalent to measurement results
	[92]	ANN, CNN and RNN	Compares different ANN structures as a decoder	Proposed structures are simple and generic
	[93]	DL	Sloped ReLU based polar code is proposed	Performance is not taken for realistic channel models
Neuro-beamformer	[96], [97]	Hopfield Network	Earliest proposal of neurobeamformer	Fails to match performance of contemporary beamformers
	[99]–[102]	RBF ANN	Perform consistently well with fully operation and degraded phased antenna arrays	For near-field scattering, monopulse algorithm outperforms ANN solutions
	[104]	ANN	Able to accurately calculate phases to synthesize the desired pattern	No comparison is provided
Autoencoders for end-to-end communications	[62]	Autoencoder	Performs better than a BPSK communication system with hard decision decoder	Suffers from synchronization and SFO issues
	[109]	Autoencoder	Alleviates synchronization and SFO issues	Synchornization stage requires a lot of effort for single carrier modulation scheme
	[111]	Autoencoder	Autoencoders for an OFDM system with cyclic prefix, which mitigates SFO issue	Performance for higher order modulation is not provided
Others	[112], [113]	DL	Can achieve optimal MIMO detection performance	Performance in a realistic channel model is lacking
	[114]	DL	Can successfully used for channel estimation [114]	Analytical framework is used for performance evaluation instead of simulations
	[115]	Triple ANN	Modulation classifier for analog and digital modulation	Complexity analysis is lacking
	[116]	KNN, SVM	Classifies CSI and associates with a set of antennas for best performance	Time varying channel model is not used

memory is proposed in [72]. The in-phase and quadrature components of a complex signal served as the two inputs of the DPD. A Cascade-Correlation (CasCoR) ANN-based DPD

for an Orthogonal Frequency Division Multiplexing (OFDM) system is proposed in [73]. The OFDM system is very sensitive to non-linear distortion and thus, a memoryless HPA

model is not suitable for OFDM. The non-linear distortion caused by the memory HPA of the OFDM system can be greatly compensated by the CasCoR ANN-based DPD [73].

In [74], the authors proposed system-level behavioral modeling for HPAs using Real-Valued Time-Delay Neural Network (RVTDNN). The HPA behavioral modeling is useful to analyze the non-linearity of a system without the need for actual HPA hardware. The ANN has been successfully used to model RF and microwave circuits and the authors proposed dynamic modeling of RF HPAs using ANN. A class AB LDMOS HPA is used in this study to generate the input and output data to train and validate the RVTDNN. A two-layer NN is used with two neurons in the input and output and $N = 15$ neurons in the hidden layer. HPA modeling based on Radial-Basis Function Neural Network (RBFNN) has been proposed in [75]. In this work, the envelope of the sampled input and output signals are used rather than in-phase (I) and quadrature-phase (Q) signals. The RBFNN requires less training than traditional I/Q signal based ANNs. The RBFNN could also be successfully used as the inverse model of DPD, i.e. the pre-distortion filter model.

A Field Programmable Gate Array (FPGA) implementation of RVTDNN for HPA behavioral modeling is presented in [76]. The RVTDNN and a Back-Propagation learning algorithm are implemented on a Xilinx Virtex-6 FPGA. The RVTDNN contains two layers with a six neuron hidden layer. The FPGA implementation is compared with a 16-QAM reference signal that is generated with MATLAB. A DPD technique based on a Non-linear Autoregressive Exogenous (NARX) model is proposed in [77]. The NARX network is a class of Recurrent Neural Networks (RNN) which allows efficient modeling of nonlinear systems. The NARX DPD of [77] has been proposed to linearize class F HPAs. The DPD model consists of two identical ANNs where the ANN replacing the pre-distortion filter is static while the ANN replacing the adaptation is dynamic. Each of the ANN consists of three layers and the number of neurons in the hidden layer is varied from $N = 4$ to $N = 10$. The experimental setup linearized a GaN class F HPA that operates on 2 GHz and a Long Term Evolution (LTE) input signal is inserted with a vector signal generator. In [78], the authors compared sigmoid and ReLU activated DNN-based DPDs for a specific number of coefficients. The ACLR of both DPDs was measured using a GaN Doherty power amplifier. This work demonstrated that sigmoid activation outperforms the ReLU for less than 2000 coefficients. If the coefficients are increased more than 2000, then ReLU provides a 3-4 dB gain over sigmoid activation function.

B. LEARNING TO DECODE

Channel coding is a technique to control errors of data communication over noisy channels. The application of ANN for channel coding and decoding was first introduced by Bruck and Blaum in 1989 [79]. They have shown that given an error-correction code, an ANN can be constructed in which every local maximum is a codeword and vice-versa. In [80],

the authors showed an application of ANN to decode the error-correction codes. In [81], the authors described the use of ANN for channel decoders. The proposed ANN-based decoders outperform the conventional decoders when Additive White Gaussian Noise (AWGN) or Binary Symmetric Channel (BSC) assumptions are violated, i.e., in a jamming environment. Instead of using each output node for single bits, the authors proposed an ANN for Hamming codes that use every output node for one codeword. They examined the possible solution of Hamming codes with two different ANNs, namely counter-propagation neural network and back-propagation. The use of only the syndrome as the input of ANN to solve Hamming codes is proposed in [82]. The application of ANN to decode convolutional codes is shown in [83]. The authors showed that the ANN network performance matches the performance of an ideal Viterbi decoder. In [84], an ANN is used to predict the presence of errors in turbo coded data. The ANN can be used to improve the reliability of communication by triggering re-transmission requests during the decoding process. An RNN is proposed for decoding convolutional codes for 3G systems in [85]. The authors claimed that the ANN decoder performs close to Viterbi decoding and implementable for certain constraint length. In [86], the authors proposed a random ANN-based soft-decision decoder for block codes. The advantage of the decoder over traditional algebraic decoder is its ability to decoder non-binary codes.

A huge drawback of the application of ANN for decoding error-correction codes was mentioned in [83]. The decoding problem has far more possibilities than a conventional pattern recognition problem. Thus, the application of ANN was limited for short codes during the 90s. Besides, the standard training methods with a large number of layers and neurons also made the decoding unsuitable for long codewords. Thus, the interest of using ANNs for decoding dwindled albeit a few minor improvements [87]. However, the introduction of layer-by-layer unsupervised learning followed by the Gradient Descent fine tuning in 2016 led to the renaissance of the application of ANN for applications like channel coding [88].

Several ANN-based channel coding methods can be found in the literature during the last few years. The use of DL to decode linear codes can be found in [89]. The DL method improves the belief propagation algorithm and different LDPC are used to demonstrate the improvements. In [90], the polar decoder is enhanced by applying ANN for decoding sub-blocks. The authors partition the encoding graph and train them individually and thus, reach near-optimal performance per sub-block. The resulting decoding algorithm is non-iterative and highly parallel. Nevertheless, the codeword length is limited to short codes as the partitioning limits the overall performance. An iterative belief propagation with a Convolutional Neural Network (CNN) architecture is proposed in [91]. A conventional belief propagation decoder is used to estimate the coded bits, which is followed by a CNN to remove the estimation errors. In [92], the performance of MLP, CNN and RNN are compared for channel decoding.

They found that RNN has the best decoding performance with the highest complexity. They also found that the length of the codeword influences the fitting of the ANN. The term saturation length for each ANN is coined in this work which is caused by the restricted learning abilities.

In [93], the activation functions of DL are explored for channel decoding problems of a polar code. The paper considers the Rectified Linear Unit (ReLU) and its variants as the activation functions. The authors also proposed a novel variant, called sloped ReLU for the positive domain range. The error-rate comparison shows that conventional ReLU variants do not provide much performance improvement. However, the sloped ReLU, which is derived from the analogy of the likelihood function in coding theory, provides performance improvements. The idea of the sloped ReLU can be utilized for other decoding algorithms. As mentioned earlier, the partitioned ANN decoder utilizes multiple ANN decoders which are connected with belief propagation decoding. The belief propagation decoders affect the decoding performance detrimentally. A neural successive cancellation is proposed in [94] where multiple ANN decoders are connected through successive cancellation decoding. The decoder achieves the same performance of the partitioned ANN decoders while reducing the decoding latency by 42.5%. In [95], a practical deep learning aided polar decoder is presented for any code length. The computational complexity of the proposed decoder is close to the original belief propagation algorithm. The authors also proposed a hardware architecture of the deep learning model. The proposed decoder outperforms the belief propagation algorithm in error-rate simulations.

C. NEUROBEAMFORMER

Beamforming is a signal processing technique used for directional signal transmission and reception. The basic idea is to set the phase angles of an antenna array in such a way that signals at a certain angle experience constructive interference and thus, focus the signal in the desired direction. The signal processing related to phase angle calculation are typically done in the digital domain and thus, can be considered as a part of the physical layer. Contrary to pre-equalization or preoding for MIMO systems, the beamformers corresponds to steering a beem towards a particular direction. The history of ANN-based beamformers can be traced back to the 80's. A typical setup of an ANN-based beamformer, also known as neurobeamformer, is shown in Fig. 2. The neurobeamformer uses an ANN to set the angles of the phased arrays so that transmitted signals from different antennas can be focused on the direction of the target user.

The earliest literature on beamforming using ANN was proposed by Speidel in 1987–89 in [96], [97]. The straightforward implementation of adaptive beamformer could not match the interference-cancellation performance of beamformers, which include sidelobe cancellers. The author proposed a neuroprocessor which incorporates the beamformer as a component and provides cancellation of sidelobes, enhances source discrimination, and angle estimation
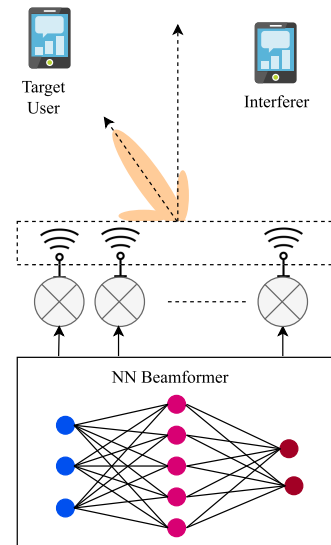

FIGURE 2. A typical setup of a Neurobeamformer where an ANN is used to predict the beamforming weights.

through the interaction of beams. Speidel coined the term neurobeamformer for this setup which, in theory, is implementable in analog circuitry without the need for any control code. The idea is to use a Hopfield network, a type of RNN, which can be employed to solve optimization problems using analog crossbar network. The Hopfield network is used to establish a direct relationship between the beamforming error and the energy function of the circuit. The network aims to minimize the energy function and in the process also minimize the average squared error at the output. The data used for training the ANN were obtained from a practical measurement setup, which includes a phase array with nine summed rows. The phase array was immersed 30 meters deep and was traveling at 15 knots. The Hopfield beamformer outperforms the classical Least Mean-Square (LMS) algorithm in terms of convergence rate [98].

During the 90s, Radial Basis Function (RBF) based ANNs for adaptive beamforming were proposed in [99]–[102]. Conventional beamforming algorithms, such as Monopulse and Multiple Signal Classification (Music) requires highly calibrated and nearly identical antenna elements for accurate results. These antenna beamforming algorithms are not designed for hardware imperfections. According to the authors, ANN excel in such problems where nonlinear or unknown antenna element behaviors need to approximate with a certain degree of accuracy. The authors apply an ANN to determine whether the network could learn the desired beamforming function and adapt the function to nonlinear element failures and degradation. The first neurobeamformer is designed to approximate the relationship between received antenna radiation and the location of the target emitting the radiation. The neurobeamformer performs consistently well with the full operation and degraded phased antenna arrays. The ANN typically detects 100% of the targets at 13 dB signal-to-noise ratio (SNR). The training method used for

the adaptive beamforming is based on back-propagation. The authors capture antenna measurements with an eight-array X-band antenna located at different positions. After each data run, 121 information samples are collected and a subset of them is used for training.

The idea of RBF ANN is extended in [103]. The authors train the adaptive RBF ANN with a Gradient Descent algorithm and a linear algebra-based network that trains using an LMS error solution. The architecture of three-layer RBF consists of an input layer for pre-processing the antenna measurements, a hidden layer with Gaussian RBFs and an output layer with summation nodes. The authors use an eight-element phase array to gather training data. For every degree of the azimuth angles, data is captured in the far field of the array. The adaptive RBF finds a minimum in the error-weight surface with only a few iterations. The ANN beamformers perform very well except for data with severe near-field scattering conditions. In [104], a novel ANN structure is proposed to implement an antenna array beamforming. The ANN consists of two hidden layers. The first layer is divided into sublayers which are equal to the number of inputs. The sublayers are fully connected to the second hidden layer. The ANN is trained using data from minimum variance distortionless response (MVDR) beamforming. The proposed ANN structure outperforms conventional ANN structures for beamforming. We invite interested readers to browse through the reviews of [105] and [106] to know more about neurobeamformers.

D. AUTOENCODERS FOR END-TO-END COMMUNICATION SYSTEMS

The communication systems are typically designed with smaller and independent signal processing blocks that individually execute their functions. The modular design process of transceivers results in controllable and efficient designs. However, the process of individually optimizing each sub-block can lead to sub-optimal designs [62]. For example, separating source and channel coding is sub-optimal according to to [107]. On the other hand, the joint optimization of these parts of the transceiver can be very complex by analytical methods. DL methods, which do not require a mathematically tractable model, can be utilized for such a problem. A communication system can be viewed as a type of autoencoder from the DL perspective.

An example of such an autoencoder is presented in [62]. A communication system typically consists of a transmitter, a channel and a receiver in its simplest form. The transmitter of [62] aims to send a message *s* out of *M* possible messages, which is encoded as a one-hot vector. The transmitter consists of a feedforward ANN with multiple dense layers. A normalization layer is added at the end of the transmitter ANN that ensures that the energy or total power constraint is satisfied which results in a transmitted signal $\mathbf{x} = f(s)$. The communication channel is represented by an additive noise layer with a fixed variance. The receiver is also implemented as a feedforward ANN. The final layer of the receiver ANN
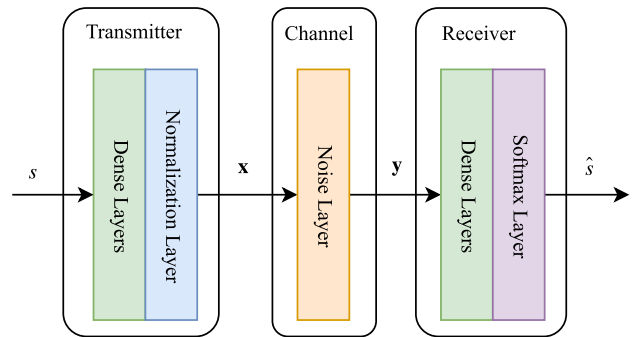

FIGURE 3. A simple autoencoder for end-to-end communication.

is a softmax activation layer. The output of the activation layer is a probability vector in which the sum of elements is equal to 1. The index of the largest element with the highest probability determines which of the *M* possible message is the decoded $\hat{s}$. A block diagram of an autoencoder, which is similar to that of [62], is presented in Fig. 3. The autoencoder of [62] is trained with gradient descent at a fixed SNR. The error-rate simulation of the autoencoder is compared with a communication system employing Binary Phase-Shift Keying (BPSK) and a Hamming (7,4) code with either hard-decision decoder or Maximum Likelihood Decoding (MLD). The simulation results show that the autoencoder performs better than uncoded BPSK and Hamming (7,4) with hard decision decoding.

The performance of the autoencoders can be adjusted for specific communication scenarios or to accelerate the training phase by changing the DL architecture. In [62], certain parametric transformations are shown to correspond to the effect of the communication channel. The inverse transform of this effect can compensate for the negative impact of the channel. The authors proposed an extension to the simple autoencoder by adding dense layers with a linear activation function and a deterministic transformation layer. A parameter vector ω is learned from the received vector $\mathbf{y}$ by the linear activation block. The transformation block results in $\tilde{\mathbf{y}}$ from $\mathbf{y}$ and ω. The simulation results show that this extended architecture outperforms the plain autoencoder. The autoencoders, extended multi-user scenarios, are also explored in [62]. Two autoencoder based transmitter and receiver pairs are considered that attempt to communicate simultaneously over the same interfering channel. Here, each transmitter-receiver pair tries to optimize the system to propagate its own message accurately. The autoencoder based communication is extended for MIMO channels in [108]. The autoencoder of [62] cannot be trained for a large number of messages due to the complexity of training. The autoencoder must transmit in smaller blocks of messages which requires synchronization at the receiver side. Therefore, the autoencoder becomes vulnerable to sampling frequency offset (SFO). In addition, the inter-symbol interference (ISI) over multiple message blocks also need to be included in the training mechanism. To alleviate the ISI problem, a sequence decoder is introduced

in [109] that decodes several message blocks in parallel with the aid of multiple ANNs working in a parallel fashion. The authors proposed another ANN for phase estimation to tackle SFO problem. The idea of autoencoders for end-to-end communications systems has been extended to OFDM systems in [110]. This work shows that OFDM with a cyclic prefix can mitigate the SFO issue of autoencoders and simplifies equalization over multipath channels.

E. OTHERS

MIMO technology was introduced to boost the capacity of wireless communication. Due to the large number of antennas introduced in a massive MIMO system, complexity of MIMO detection has been a key challenge. In addition, approximate inversion based detection mechanisms do not work well when the ratio between number of antennas and users are relatively low. Therefore, ML based methods can be an attractive alternative which can provide optimal performance with similar complexity to exact inversion based detection methods. In [112], the authors proposed a MIMO detector based on DL. The authors introduced DetNET, a DL network, for binary MIMO detection. DetNET achieves optimal detection performance and can be implemented in real-time. In [113], a model-driven deep learning framework has been presented for MIMO detection. The network trains some adjustable parameters of orthogonal approximate message passing (OAMP) detection. The OAMP with deep learning significantly outperforms the original OAMP detection in terms of error-rates. Similarly, ML can be a viable alternative of existing MIMO pre-equalization and other precoding methods. In addition, ML will also be used as a complementary technique to improve performance of the existing detection or precoding methods. Many of convex optimization based algorithms are currently tuned manually to optimize their performance. ML can be used to tune detection or precoding algorithms automatically. For example, in [117], ANN has been used to optimize biConvex 1-bit PrecOding (C2PO) algorithm which achieves same error-rate performance of the original C2PO algorithm with $2\times$ lower complexity.

ML methods can be also be an alternative for channel estimation methods. Conventional pilot based training of channel estimation can be challenging in terms of performance and complexity. According to [118] and [119], ML methods are effective for single-input single-output (SISO) channel estimation. In [118], the efficacy of DL for SISO channel estimation and signal detection is shown for an OFDM system. The proposed DL approach, which is trained off-line, estimates the Channel State Information (CSI) implicitly and detect the transmitted symbols. The DL method can address the channel distortion and recover the transmitted symbol with a performance comparable to the MMSE channel estimator. In addition, the proposed approach is more robust with fewer training pilots. In [119], the authors present a fully complex extreme learning machine (C-ELM) based SISO channel estimation and equalization method. Unlike [118], C-ELM performs the training step also online at the receiver

side. ML methods are also shown to be effective for MIMO channel estimation schemes. For example, channel estimation is very complex when the number of RF chains is limited in the millimeter-wave MIMO receiver. In [114], a Learned Denoising-based Approximate Message Passing (LDAMP) ANN is proposed to address this problem. The results show the potential of DL for mmWave channel estimation. In [120], a DL method is proposed for Direction-of-Arrival (DOA) and channel estimation. The deep ANN is applied for offline and online learning to characterize the channel statistics in the angle domain. ML can also be used as a complementary technique rather than a replacement of existing channel estimation methods. For example, in [121], a DL based pilot allocation scheme is proposed. This technique improves the overall performance of the system by alleviating pilot contamination through learning the relationship between pilot assignment and user distribution. ML has been also used for other applications of MIMO systems.

In [116], ML-based antenna selection technique for wireless communication has been proposed. The author's applied multiclass classification algorithms, i.e. multiclass k-nearest neighbors (KNN) and a Support Vector Machine (SVM) to classify the CSI and associate it with a set of antennas that provide the best communication performance.

The ML algorithms excel for classification tasks and CNN is commonly used in image classification. The same principle can be applied in the physical layer for the modulation recognition task. In [115], an ANN architecture is proposed as a modulation classifier for analog and digital modulations. The architecture is comprised of three main blocks: pre-processing the key features of the signal, the training and learning phase, and the test phase to decide the modulation. The authors carried out extensive simulations for twelve analog and six digital modulation signals and the success rate of the ANN was over 96% at the SNR of 15 dB. A CNN based modulation classifier is presented in [62]. The classifier consisted of a series of convolutional layers, followed by dense layers and terminated with a dense softmax layer. The classifier is trained by 1.2M sequences for IQ samples covering 10 different digital and analog modulation schemes. The system takes into consideration the multipath fading effects, sample rate offset and center frequency offset.

F. LESSONS LEARNED

Despite the potential of ML schemes for the physical layer, major challenges remain due to the nature of the physical layer algorithms and the complexity of ANNs. Many physical layer algorithms have closed-form mathematical expressions and it is difficult to justify the use of an ML scheme for such a scenario. However, as we mentioned earlier, the closed-form expressions are often based on the assumption of oversimplified and unrealistic system models. The pattern recognition problems, such as modulation recognition, can be replaced with the information provided in the 5G or LTE transmit frame header. A simple flag can be used in the receiver which indicates what modulation scheme is used. Therefore, it is

essential to justify the use of ML for a particular pattern recognition application. The main challenge related to the use of ML for MIMO equalization or detection is the quality of channel estimates. Most of the detection problems assume non-varying CSI at the receiver. The ML schemes must cope with the changes in channel parameters. It is possible that re-training an ANN for varying channel conditions might make a detection algorithm prohibitively complex. It should be noted that the traditional pilot based channel estimation algorithms are also not simple when the antenna dimensions go higher [122]. In the case of channel decoding, the use of ML methods is still complex for long codewords. Despite the introduction of DL, most of the DL decoders are proposed for short codewords. The applicability of DL for decoding long codewords still requires tremendous research efforts. In a nutshell, the application ML is perfectly justified for physical layer algorithms that are highly non-linear in nature and where the mathematical model is far from perfection. Therefore, ML will continue to excel for applications like pre-distortion. On the other hand, the sub-optimal algorithms can provide very good performance with feasible complexity for many baseband applications. Therefore, more research is necessary to make the ML solutions competitive against those applications.

IV. ML FOR LAYER TWO: MEDIUM ACCESS CONTROL

In wireless networks, the spectrum is a scarce resource, and therefore, robust MAC providing channel access to multiple users is required to maximize the spectrum utilization and guarantee the Quality of Service (QoS). With the increasing number of connected devices that expand the optimization domain with an ever-growing number of parameters and tighter QoS requirements, MAC is expected to reach unprecedented complexity. MAC protocol depends on the combination of network architecture, communication model and duplex mechanisms (i.e., enabling bi-direction communication via time division duplexing (TDD), frequency division duplexing (FDD), full-duplexing), and therefore, the constant evolution in a wireless network with new technological components (e.g. flexible duplexing, adaptive frame numerologies, the convergence of heterogeneous wireless networks with multi-interface radio devices) increase the complexity of MAC tasks even further. Thus, MAC can be regarded as a large-scale control problem with diverse QoS constraints and optimization of such a problem with traditional rule-based algorithms is not the optimal choice. Until today, most of the algorithms for MAC radio resource allocation are based on an optimization approach that requires assumptions to relax the non-convex problem and provides a sub-optimal solution as complexity prevents solving the full non-convex problem [123], [124].

Moreover, wireless networks are rich in data, where data is continuously gathered from a massive amount of user devices and network entities in the form of radio and system measurements [125], [126]. However, the current MAC protocols derive little insight from such data as it considered

to be short-lived and localized commodity due to aging and user mobility [127]. The evolution in the field of ML and wireless networks provides an opportunity to exploit such data in multiple dimensions and create data-driven wireless networks that are more robust and autonomous in changing environments. This section presents a holistic overview of the potential research directions and the associated challenges brought forward by the use of ML algorithms in the MAC protocol design for future wireless networks.

A. INTERFERENCE PREDICTION

Air is a shared medium and transmission of signals by multiple devices in the same frequency disturbs the reception of these signals. This phenomenon is coined as "interference". Despite the advances in modeling interference dynamics in wireless networks [128], [129] using approaches such as Interference Alignment, researchers have not been able to devise algorithms that can harness the full potential of interference knowledge to improve the QoS of communication. The basic idea of Interference Alignment is to coordinate multiple transmitters to align the mutual interference at the receiver. However, there are ongoing efforts to use machine learning like an autoregressive (AR) model by which the next state is predicted with some past states to improve the performance of interference alignment algorithms [130]. One of the main challenges in using the interference information is its randomness and short-time validity due to mobility. Furthermore, the identification of the exact source of interference among multiple sources and the randomness of an interferer in each time instance make the problem even more challenging.

As the next generation of communication systems are expected to support new communication paradigms such as IoT and Machine-to-Machine (M2M) services besides the traditional voice and data services, the expected impact of co-channel and inter-cell interference is even more rigorous. Particularly, with the newly introduced technologies for licensed (i.e., NB-IoT, LTE-M) and unlicensed (i.e., LoRa, Sigfox) spectrum, specifically designed to support massive IoT systems, interference in such networks raises serious concerns on the potential of these technologies. In the case of licensed spectrum technologies, a number of scenarios can be expected where transmission overlaps with other transmissions in a network in frequency and time domain. This is due to the provision of operation within the existing LTE band and the allocation of frequency resources to a device with 1-subcarrier granularity. On the other hand, in the unlicensed spectrum, the large number of devices competing for the free spectrum causes collisions. Interference limits the potential of these technologies and degrades the overall network performance. However, in areas without the network coverage of licensed spectrum, unlicensed technologies are the only possible options.

To cope with the wireless interference problem, a device must estimate interference dynamics and predict the interference well before the transmission. The main benefit of

predicting the interference is to reduce the control signaling overhead, particularly in system with strict latency requirements such as URLLC. Moreover, interference prediction will help to manage resources or beam in a proactive manner and improve efficiency. Moreover, prior knowledge of interference can be used to either cancel the unwanted component from the received signal or otherwise manage the interference to reduce the impact and improve the spectral efficiency of the system.

No doubt, interference prediction is challenging as a wireless signal can be corrupted by a variety of ambient wireless signals such as Bluetooth, Wi-Fi, and LTE and the signal can be further attenuated by walls and other obstacles. However, the advancement in ML provides prediction algorithms to address this challenge and to improve the performance of wireless networks as illustrated in Table 5. The algorithm using K-means and stacked autoencoders clustering (AEC) to extract key features of wireless signals to predict the interference is presented in [131]. A comparison with traditional adaptive filtering with the LMS approach shows that up to 18 dB gain can be achieved in SNR. Similarly, in [132], the interference prediction algorithm inspired by semi-blind channel estimation is proposed based on the learning traffic pattern and the transmission behavior of the interference nodes. The proposed scheme underestimates the interference as it does not consider the newly originated transmission. This problem can be tackled by predicting inference with learning techniques but at the cost of increased complexity. Two interference prediction algorithms based on Hidden Markov Models (HMM) and Concordance Algorithm (CA) for unlicensed band ZigBee transmission to predict the white spaces, i.e., space where no transmission occurs, particularly for Wi-Fi signal, are presented in [133]. The HMM algorithm relies on training data to predict the next occurrence of white space, whereas CA relies on past data and does not require training. The CA is shown to perform better than HMM.

For the licensed spectrum, Cooperative Interference Prediction (CIP) is proposed [134]–[136] to predict the expected inter-cell interference from neighboring cell devices. The main idea of CIP is to exchange information about the expected scheduling of devices between the wireless access points based on predicted channel conditions, thus each access point knows the potential interferer and will be able to estimate the interference and perform channel allocation accordingly to mitigate the impact of interference. Particularly, the technique is beneficial for licensed spectrum technologies where access point allocates the channel resources. Moreover, in the case of licensed IoT technologies like NB-IoT and LTE-M, CIP is expected to perform better as devices with fixed locations cause less channel variation. However, as the algorithm is based on channel prediction between devices and access points, ML can play a vital role in improving the performance of CIP to improve the channel estimation and also by predicting the expected user scheduling based on the traffic pattern. Furthermore, in [137], a Kalman filter-based interference prediction algorithm is presented. The algorithm observes the co-channel temporal correlation to predict interference during the contiguous data transmission. The desired performance is achieved by determining the required transmission power based on the predicted interference. An interference prediction algorithm based on General-order Linear Continuous-time (GLC) mobility model for wireless Ad-hoc networks is presented in [138]. The algorithm uses GLC to derive Mean and Moment Generating Function (MGF) of interference prediction. The presented closed-form expressions only exist in special cases, though. Closed-form approximations can be derived for these statistics using a cumulant-based approach as in [139].

No doubt, interference is the major factor limiting the performance of wireless networks and the impact of interference depends on a number of factors such as network topology, traffic pattern, duration of traffic, and transmission power. Due to such a large number of parameters and their corresponding randomness, an ML-based algorithm to predict the expected interference can help to improve the performance of MAC tasks such as resource allocation, asymmetric traffic accommodation, and transmission power control which in turn improve the performance of the overall wireless network [140], [141].

B. USER MOBILITY AND ASSOCIATION
User mobility and localization techniques are finding their way as an integral part of 5G due to the significant benefits in terms of enabling SON (i.e. enabling proactive handover and resource management) [142], [143] and location-aware services such as factory and process automation, intelligent transportation systems, unmanned aerial vehicles, etc [144]. The expected positioning accuracy target for 5G is less than 1 m in urban (and indoor) scenarios and less than 2 m in suburban scenarios where the vehicle speeds are up to 100 Km/h [145].

To support such variety of applications and performance requirements, 5G is characterized with several disruptive features, which have direct implications to positioning and mobility [146]. These features include network densification, mm-Wave, and massive MIMO, as well as device-to-device communication. The use of mmWave brings a two-fold advantage: large available bandwidth and the possibility to pack a large number of antenna elements even in small spaces (e.g., in a smartphone). Wideband signals offer better time resolution and robustness to multipath thus improving the performance of Observed/Uplink Time Difference Of Arrival OTDOA/UTDOA schemes, as well as paving the way to new positioning methods such as multipath-assisted localization exploiting specular multipath components to obtain additional position information from radio signals. Despite the improvement in the precision of angle-of-arrival based localization systems, particular attention has to be paid to hardware complexity and cost complexity considerations; whereas power consumption and computational burden turn out to be key challenges for IoT-based localization. No doubt, the larger bandwidths in 5G systems allow for a higher degree

of delay resolution. On the other, higher carrier frequencies result in fewer propagation paths and the possibility to pack more antennas into a given area. All the above, clearly, leads to a high degree of resolvability of multipath signals and, in turn, enhanced positioning accuracy. Network densification is also beneficial in that it maximizes the probability of having LOS condition with, possibly, multiple base stations. Along with that, the availability of device-to-device links also provides an additional source of positioning information. The narrow bandwidth and large coverage of long-range IoT solutions (LoRa, Sigfox, NB-IoT, etc.), on the contrary, limit to a large extent their achievable positioning performance.

Traditionally, the solutions proposed for user association and localization are based on heuristic approaches and automation of processes are only limited to low complexity solution like triggering [147]. The solutions so far do not really capitalize on the potential of information that can be retrieved from the wireless networks. There have been some efforts to realize the potential of advance ML techniques in enabling such networks [148]–[156] as presented in Table 5. In [148], the author presents an algorithm for user association problems for infrastructure-based networks using online policy gradient RL by modeling the problem as a Markov Decision Process. The proposed model considers traffic dynamics which make it possible to optimize performance from the measurements directly perceived from the users. Another algorithm based on RNN for user association to a base station and mobility prediction for seamless handover is presented in [149]. It is shown that the proposed algorithm significantly improves mobility prediction and facilities virtual cell formation for user mobility management. Similarly, in [150]–[153], RL based admission control algorithms for infrastructure-based networks are presented for seamless handover and load balancing among base stations to ensure improved QoS.

For Mobile Ad-Hoc Networks (MANET), in [154], the author presents a mobility prediction approach using Extreme Learning Machine (ELM) based on single feed-forward architecture. ELM does not require tuning of parameters and the initial weights also have no impact on the performance of the algorithm. Furthermore, an algorithm based on an RNN for mobility prediction is presented in [155]. However, the algorithm is based on node location which is based on the Global Positioning System (GPS) which limits the accuracy and reliability of the proposed solution. Besides the mobility prediction, ML techniques can also help to address node movement prediction in MANETs as in [156]. The idea is to allow nodes to control their position to optimize connectivity.

However, the accuracy of mobility and user association algorithms presented are highly based on accurate prediction of positioning or localization. Many studies develop localization algorithms based on Wi-Fi, cellular networks and GPS signals [157]–[160]. However, in terms of cellular networks, the positioning algorithms which are even part of the 3GPP standard such as OTDOA and UTDOA are clearly not able

to meet the requirement of future 5G networks as they are designed for a target accuracy of 50 m [161].

In this regard, many researcher studies ML techniques for localization to improve the accuracy of these algorithms by training the algorithms with phase information, estimate the angle of arrival with an ML using phase fingerprinting [162]–[165]. In [162], an indoor positioning algorithm based on KNN is proposed using historical data for estimation of mobile user position. Similarly, in [163], an algorithm for positioning estimates based on DNN is presented. The proposed algorithm employs stacked denoising autoencoder (SDA) and Hidden-Markov model (HMM) to minimize the feature set to smooth the original location without degrading information content. In [164], the author combines the SVM and ANN to estimate the position of the user based on the Received Signal Strength Indicator (RSSI). However, the focus of work is on the boundary-level localization rather than the actual position of the user. In [165], the authors present a CNN based fingerprinting technique for indoor localization based on channel frequency response (CFR). Similarly, in [166], the author presents the deep long short-term memory (LF-DLSTM) approach for indoor localization based on RSSI. The proposed algorithm reduces the noise effects to improve positioning accuracy. However, the fundamental issue in most of the proposed positioning algorithms based on ML techniques and even in the conventional algorithms (i.e., OTDOA, UTDOA, RSSI, Cell-ID) is the reliance on single value estimates (SVEs) which can be either phase information, angle of arrival or RSSI. Therefore, localization estimate depends heavily on the quality of such SVEs, which degrade or have high random in the wireless environment due to multipath and Non-Line-of-Sight (NLOS). To fully exploit the potential of learning techniques in terms of user mobility and localization, the possible direction is to explore the cooperative positioning techniques through data fusion of 5G with other sources such as camera, Device-to-Device (D2D) links, GPS, etc. Moreover, jointly optimize the communication and positioning targets which often overlap.

C. RADIO RESOURCE SCHEDULING

Radio resource scheduling is one of the major challenges for future wireless networks due to the large dimensionality of parameters and conditions in a network coupled with a number of configuration parameters. Particularly for the 5G system, the cardinality of scheduling decisions is dependent on a massive number of devices, the range of operating frequency bands, flexible frame duration, sub-carrier spacing, etc. Moreover, due to the diverse landscape of applications with different requirements and limitations, MAC scheduler is expected to be one of the most challenging tasks.

This is the case with the new Low-Power Wide Area Network (LPWAN) technologies like NB-IoT and LTE-M, where the scheduling can be performed on the sub-carrier level or as per 3rd Generation Partnership Project (3GPP) recommendation on different resource unit configurations [167].

TABLE 5. ML for MAC Layer.

Application Domain	References	ML Algorithms	Key features	Remarks
Interference Prediction	[131]	KNN, AEC	Exploit cyclostationary repeating patterns to predict interference	No results with clusters of different sizes or density, large-scale evaluation is missing.
	[132]	Semi-blind Estimation	Predict interference on traffic pattern and transmission behaviors	Under-estimate interference by ignoring newly originated transmissions.
	[133]	HHM, CA	Predict white spaces in Wi-Fi	HMM and CA rely on training data.
	[134]–[136]	Supervised learning	Learn from channel estimation to predict future scheduling	Requires cooperation which introduce overhead and large backhaul delays.
	[137]	Kalman filter	Learn the co-channel temporal correlation to predict interference	Kalman filtering assumes linear dynamics and Gaussian PDF. The model has to be perfectly known, i.e., no uncertainty.
User Mobility Association	[148]	Policy Gradient RL	Self-optimized association algorithm	High variance in estimating the gradient (i.e., Monte Carlo), Bad gradient estimate could adversely impact the performance.
	[149]	RNN	Mobility prediction for seamless handover among base stations	Prediction of optimal triggering conditions and load on expected base station is not considered.
	[150]–[153]	RL	Handover and load balancing algorithm	Can only be used in controlled environment.
	[154]	ELM	Predict node mobility pattern	Increased time consumption and complexity.
	[155]	RNN	Mobility prediction	Requires GPS which limits the accuracy and reliability of proposed scheme.
	[156]	RL	Node movement to control position and optimize connectivity	However, it is impossible to control node movement in ad-hoc networks.
	[162]	KNN	Historical data of RSSI is exploit for positioning estimate	Does not provide high positioning accuracy as RSSI is highly time dependent.
	[163]	SDA and HMM	SDA is used for dimensions reduction and HMM is applied for localization improvement	Scalability issue is not addressed and system performances is evaluated at different layers making it hard to see the real potential of the proposal.
	[164]	SVM and ANN	RSSI based positioning estimate	The work focus on defining the boundary region rather than accurate positioning of user.
	[165]	CNN	Learn the amplitude of CFR and depend on number of radio link and subcarriers of OFDM	Prior tuning of parameters is required to provide the best localization performance.
	[166]	LF-DLSTM	Reduces the noise effects to obtain more abstract information from RSSI	The scheme depends on the temporal dependencies among signal, however, for accurate position spatial dependencies need to be addressed.
Radio Resource Scheduling	[169]	Policy gradient RL	User scheduling based on traffic variation	Bad gradient estimate could adversely impact the optimization.
	[170]	Q-Learning	Explore the potential spectrum hole for femtocell in HetNet	Selection of RAT in terms of speed, exploration times, and convergence is not considered.
	[171]	Bagging tree, boosted tree, KNN, SVM, and Kohonen networks	Evaluation of prediction techniques for resource allocation in LTE	small network deployment (i.e, maximum 4 small cell).
	[172]	RL	Interference-aware resource allocation	Decentralized prediction approach can lead to frequent changes in frequency assignments, results inaccurate interference prediction.
	[173]	Random forest Algorithm	Exploit mobility prediction for resource scheduling	LoS assumption and accurate location measurement limits the performance.
	[174]	RL	Resource allocation for D2D users operate as an underlay with cellular network	Cooperative approach that lead to significant overhead.
	[175]	Generic algorithm, FLS, and ANN	Evaluation of learning techniques for intelligent task scheduling	FLS outperform, however, more calculation and research is required on the buffer weights.
	[179]	LSTM	Exploiting spatial-temporal correlation for traffic prediction	Focus on network-wide traffic volume prediction.
	[180]	RL	Dynamic resource management based on traffic variation	Based on traffic dataset and ignore external factors such as base station information and user distribution.
	[181]	LSTM	Exploits the temporal and long distance spatial dependencies	Ignore the external factors and focus on network-wide traffic prediction.
	[182]	LSTM-RNN	Long-term traffic prediction	Ignore the external factors and focus on network-wide traffic prediction.
Power Management	[111]	Learn-α	Power saving model (i.e., sleep/wake up interval prediction)	Model assume fixed size packets during the sleep interval, unrealistic assumption.
	[183]	RL	Predictive transmission power control	Complex infrastructure.
	[184]	RL	Downlink power control	Assume stationary network, which is unrealistic.
	[185]	RL	Transmission power and modulation prediction	Introduce long delays.

The scheduling of massive devices on such granularity further increases the complexity considering that each device has its own set of requirements. On the other hand, the stringent requirement of Ultra-Reliable Low Latency Communication (URLLC) reduces the execution time of the scheduling process to less than $100\mu s$, challenging the task even further. Similarly, in MANETs, D2D communication is one of the promising solutions for multiple use case scenarios such as Vehicle-to-Vehicle (V2V) communications. D2D communication may operate as an underlay to the cellular communication and therefore requires a robust scheduling algorithm to improve the spectrum efficiency under extreme application as well as overall network constraint.

To deal with such complex gravity of the problem, scheduling optimization is crucial to boost the system performance. However, most of the scheduling optimization in wireless networks is a combinatorial optimization problem and provides the globally optimal solution to such complex problems is practically infeasible in real networks due to the demand of high computational complexity and time particularly for large-scale networks (i.e., massive devices) [168]. The traditional optimization algorithms make a trade-off between complexity and the quality of the solution, therefore knowing the dilemma of optimization techniques, it is necessary to look for a new solution approach for enabling near to real-time decision making for MAC scheduling.

In this context, ML techniques are promising to provide an alternative to the design approach for complex and dynamic systems. Scheduling in cellular networks is addressed in [169] using the DRL approach considering traffic variation by formulating the problem as Markov Decision Process (MDP). With the proposed scheme mobile network can transmit 14.7% more data. In [170], a multi-objective strategy using RL is proposed to address the problem of resource allocation and interference coordination in Heterogeneous Networks (HetNet) network of femtocell under macro cell coverage. The proposed algorithm enables the femtocells to identify the available spectrum resource for opportunistic use. Furthermore, in [171], a comprehensive comparison of different machine learning approaches i.e., bagging tree, boosted tree, KNN, SVM, and Kohonen networks is presented by formulating a dynamic resource allocation scheme for the Quality of Experience (QoE) provision in cellular networks. It is concluded that bagging tree outperforms other techniques in their case. An interference aware resource allocation scheme based on reinforcement for a cellular network is presented in [172]. The approach is based on decentralized frequency allocation in the presence of incomplete information from neighboring cells. However, such a decentralized approach can lead to a significant increase in interference among cells and eventually results in performance degradation. In [173], random forest algorithm based resource allocation for cellular networks is presented. The proposed scheme exploits the location of mobile users to improve the scheduling decision considering a line of sight communication. The assumption of LOS communication and

the requirement for accurate location measurements limit the performance of the proposed scheme. Furthermore, a cooperative reinforcement algorithm for resource allocation in D2D communication operating as an underlay in a cellular network is presented in [174]. No doubt, the proposed algorithm shows significant improvement in throughput under a single-tier network, whereas, the expected future wireless network will be multi-tier HetNet. In [175], three machine learning techniques i.e., generic algorithm, fuzzy Lyapunov synthesis (FLS), and ANN are studied for scheduling sensor networks. The generic algorithm is shown to outperform the others.

Beside this, ML algorithms can be used to provide novel solutions for the asymmetric traffic accommodation in wireless networks by enhancing the flexible duplexing capabilities of 5G networks which is one of the promising technique to optimize resources based on traffic demand in either down-link and uplink [176]. The aim of flexible duplexing is to perform adaptive resource allocation considering the asymmetric uplink and downlink traffic and jointly optimize in the time-frequency domain, such that the distinction between TDD and FDD is blurred or even completely removed. In [177], the author presents radio resource allocation based on flexible duplexing to avoid inter-cell interference. Similarly, in [178], the author presents the potential benefits and the corresponding challenges for flexible duplexing in wireless networks.

However, to fully exploit the potential of flexible duplexing and to adaptively optimize the resource, it is of utmost importance to predict or learn the traffic behavior of the network. Many researchers have conducted work to predict traffic flow based on linear models such as autoregressive (AR), autoregressive moving average (ARMA), and moving average (MA) [186], [187]. These methods achieve good results for small-scale sparse networks, however, considering the growing nature of the wireless network, many nonlinear factors like multi-dependency and abruptness play a significant role in the performance. Therefore, recently many machine learning-based models have been proposed based on ANN, SVM, least-square support vector machines(LSSVM), LSTM, etc [179]–[182]. These models learn from the historical traffic data of the network to learn the randomness and to improve the prediction of traffic flow. In [179], a 2D-LSTM based short-term traffic prediction model is presented which consider the spatial-temporal correlation. However, the study only focuses on traffic volume prediction. Similarly, in [180], the traffic scheduling problem of IoT networks is addressed by adapting to traffic variation dynamically and improving the network utility function using reinforcement learning. The proposed scheme shows significant improvement over traditional strategies, however, the focus of the work is on predicting the aggregate network traffic. In [181], the author presents the framework base done LSTM for long-term traffic prediction by exploiting the temporal and long-distance spatial dependencies. Another deep learning-based traffic prediction model based on LSTM-RNN is presented in [182]. In the case of traffic prediction models, most of the works

focus on the traffic datasets only and ignores the external factors such as base station information and user distribution. However, it is well known that these factors directly impact the generation of traffic [188], [189]. Moreover, recent works focus on network-wide traffic prediction and fail to identify the pattern diversity or similarity among different services.

D. POWER MANAGEMENT

Energy efficiency has become one of the crucial parameters in the design of wireless communication systems due to environmental concerns and the requirement that IoT devices need to operate from few months to few years without replacing or recharging the battery. Energy conservation in a wireless network can be performed at different layers of the system, but conservation at the MAC layer is more effective as radio can be controlled directly. MAC layer can improve energy efficiency by maximizing sleep duration, minimizing idle listening, eliminating the hidden terminal problem and collision of packets, controlling transmit power, scheduling node with a less interfering resource which in turn result in less re-transmission, etc. Therefore, protocols like discontinuous reception (DRX) [190], power-saving mode (PSM) [191], and optimized power control have been introduced. DRX is a part of existing LTE introduced which enables devices to connect to the network on a need basis.

Devices can remain inactive or in sleep mode for hours. The network and devices can negotiate the period of time for which the device can be in sleep mode. During this period, the device switches off its receiver and is not listening to the paging or downlink control channel. After the expiry of the set period, the device wakes up and starts listening. Moreover, in 3GPP Release 13 extended DRX is introduced for IoT devices with an increased hour of sleep time [192]. On the other hand, the PSM feature was introduced in Release 12 of LTE and is also used in Wi-Fi. PSM is similar to power-off, however, the device still remains registered with the network and when the device wakes up, it does not have to perform the registration process with the network. This reduces the signaling overhead and optimizes the device power consumption. However, these solutions are driven by static rules and provide a posteriori response to traffic and context changes. A traffic driven power saving mechanism is one of the promising ways to improve the energy performance of networks.

Moreover, in traditional wireless networks, the reliability of the system is improved by increasing transmit power, specifically in high-interference scenarios. However, it has a detrimental impact on the global performance of the system in terms of energy. As the wireless channel and the interference are variable, the transmission power is also variable and can be controlled at the MAC level. For this reason, predicting the appropriate transmission power based on the actual network conditions results in improved energy and spectrum efficiency of the overall system. [183] presents such RL based predictive transmission power control. The proposed solution uses a database to manage the machine learning algorithms

which results in a single point of failure and more complex infrastructure. [184] presents an RL approach for downlink power control in cellular networks. The approach assumes the network activity to have Markov property and to be stationary as well, which is an unrealistic assumption as node mobility is not considered. [111] proposes a learning-based power saving model to determine the sleep and wake up the interval of the device. The algorithm is based on Learn-α machine learning approach which has the advantage of not assuming a statistical distribution of network activities. However, the model assumes the packets arrive uniformly during the sleep interval which is not the case in real networks. Furthermore, in [185], a reinforcement based algorithm to predict transmission power and modulation in a wireless sensor network is presented. Prediction is based on learning from previous channel gains and packet queue sizes. The primary limitation of the proposed solution is that the decision of transmission is made at discrete intervals, therefore, the node must wait for a while in its current state before moving to the next one, which limits the protocol application particularly in time-critical systems.

E. LESSONS LEARNED

In future wireless networks, users will be able to access services through different access technologies, such as WLAN, LTE, 5G and DVB, from the same or different network operators, and switch seamlessly between different networks with active communications. For example, in the case of wireless access in vehicular environments (WAVE), especially for road safety, continuous and seamless connectivity is a significant factor in providing services. Therefore, heterogeneous wireless access is needed to obtain continuity through licensed (LTE (LTE direct), 5G (NB-IoT, URLLC)) and unlicensed (IEEE 802.11p, Wifi-direct, LoRA) technologies. The availability of multiple channels and multiple radio interfaces can lead to substantial improvements in the performance of wireless access networks.

However, the selection of the most suitable radio interface for a specific connection depends on various factors such as the availability of the interface in specific devices, the required communication bandwidth, the cost of maintaining an active interface (in terms of energy use), and available neighbors. Therefore, an intelligent interface selection scheme is required to address heterogeneous networks interoperability, adaptive spectrum re-utilization, and network self-optimization along with intelligent response for fault-tolerance. In this regard, ML techniques are able to provide such a self-X feature required by HetNets.

Furthermore, most of the expected future wireless systems such as NB-IoT, LTE-M are based on the symmetric allocation of frequency between downlink and uplink due to half-duplex frequency division duplex (HD-FDD). However, such symmetric allocation would result in under-utilization of spectrum resources due to the nature of traffic which is mostly asymmetric. As in the case of IoT networks, data traffic is expected to be more in the uplink as compared to

downlink which potentially leads to the resource wastage in the downlink. Similarly, in broadband communication, downlink traffic is more and thus results in under-utilization of uplink resources. ML techniques can help to provide an intelligent solution to such problems for the seamless integration of all the environmental parameters and enable proactive MAC functionalists rather than traditional reactive ones.

V. ML FOR LAYER THREE: NETWORK LAYER

The network layer ensures end-to-end packet delivery through a calculation of routes throughout the network [193]. With the increasing complexity and diversity of communicating entities, the network must be capable to configure the routing nodes with least human intervention. Configure here means adjusting the behavior of network devices (routers and switches) for end-to-end delivery of traffic from different applications. This results in several challenges. For instance, traditional network traffic control systems heavily rely on pre-defined policies, which makes the network stagnant and difficult to adapt to changing environments. Whereas, wireless networks are prone to sudden changes in traffic patterns and volumes. Therefore, new mechanisms must be sought out to ensure end-to-end packet delivery in the most resilient and optimal way. ML has proven to be one of the most effective mechanisms to solve such challenges. In this section, we describe how ML can be used to improve the end-to-end packet delivery through efficient network traffic control and routing mechanisms in communication networks. The frameworks along with the ML mechanisms and offered services are summarized in Table 6, and described below.

A. NETWORK TRAFFIC CONTROL

Efficient network traffic control mechanisms have been fundamental to the overall performance of a network, whether fixed networks [194], [195], wireless networks [196], [197], or virtual networks [198], [199]. Network traffic engineering provides mechanisms to reduce network congestion and improve utilization by balancing the load among multiple paths [200], [201]. Efficient traffic engineering empowers communication networks to deliver data optimally between two points even when some links or routers fail. However, today's networks typically perform failure recovery and traffic engineering independently which results in complicated and compromised network traffic control systems [202]. Furthermore, the dynamic nature of novel services and devices necessitate automated network traffic control systems. Hence, ML-based approaches are applied for efficient traffic control.

DL for intelligent traffic control has been studied in [33]. The authors provide an overview of the state-of-the-art DL architectures and algorithms relevant to network traffic control systems. The fundamental function of a network traffic control system is to efficiently route packets between two entities. Conventional routing protocols are non-intelligent, thus, they are not capable to learn from past events to take

autonomous decisions [203]. Thus, traditional network protocols that rely on algorithmic insights from human experts must be replaced by data-driven, predictive or in other words, ML-based approaches. Traffic steering is another approach that can use ML techniques to avoid congested paths of the network or even maximize QoS among existing alternative paths, or heterogeneous networks, as demonstrated in [204] through Q-learning. A simple network traffic control system using ML is presented in Fig. 4, where traffic patterns are used to train the system to provide the best routing paths. In the following sub-sections, we describe ML aided routing (protocols) for different network architectures and networking technologies.

B. ROUTING

Routing, generally, means how packets are delivered from a source to a destination with optimal use of network resources, as well as, maintaining the expected QoS and QoE. Conventionally, route-optimization deals with the uncertainty of future traffic conditions by either i) optimizing routes with respect to previous traffic conditions, hoping that the routing configurations will deal efficiently with future traffic, or ii) optimizing routes with a possible range of traffic scenarios [205]–[207]. In the first case, routing configurations optimized for specific traffic conditions can fail in achieving good performance in slightly different traffic conditions, as described in [208]. In the second case, optimization for a broad range of traffic conditions makes it hard to achieve optimum results for the actual traffic conditions [208]. Therefore, ML has been adopted to meet such challenges.

Leveraging ML for routing has been discussed in [208]. The main theme of the work is to evaluate whether ML can be used to automatically generate good routing configurations. Considering intra-domain traffic engineering [209] as a case study, ML-guided routing optimization within a single, self-administered network is examined with a focus on i) formulating routing as an ML problem, and ii) finding suitable representations for input and output in the learning domains. The authors conclude that data-driven routing, in other words, ML-based routing, can highly improve the network performance, but can also pose significant challenges when not properly applied. The challenges are in terms of network overhead and delay if proper techniques and learning parameters are not selected. For instance, supervised learning with very large sets of parameters can be highly ineffective if the traffic conditions have high irregularity. On the other hand, carefully selected few parameters yet not loosing too much information for expressiveness for RL yields much better results.

DL for HetNet traffic control with challenges and future perspectives have been discussed in [210]. The authors propose methods for input and output characterizations and a DNN system for traffic in HetNets. To minimize the computational overhead and scalability limitations, the proposed DL system in [210] comprises multiple hidden layers in which each layer computes a non-linear transforma-

TABLE 6. ML approaches for Network Layer.

Application domain	References	ML algorithms or mechanisms	Key features	Remarks
Network Traffic Control	[33]	DL mechanisms	Generic overview of DL mechanisms for traffic control	The work outlines the benefits of data-driven traffic control approaches.
	[46]	Many ML techniques	A survey on ML applications in networking and network traffic control	More focused on ML in general.
	[210]	DL mechanisms	Input/output characterization	Chances of cascading of errors from one layer to another.
	[235]	DL	Channel assignment in SDN-IoT	Main focus is on traffic prediction in IoT.
	[236]	Bayesian nonlinear regression	Optimal path performance prediction in optical networks	Other approaches are also evaluated in terms of bit error rate.
Routing	[208]	Generic ML approaches for routing	Generating automatic routing configurations	Challenges, e.g., high costs in supervised learning with large sets of parameters, are highlighted.
	[213]	Swarm intelligence	Routing efficiency through sharing intelligence	Frequent sharing required, resulting in increased network overhead.
	[218]	ML-based route estimation	QoT Estimation in Optical Networks	Main focus is routing and spectrum assignment.
	[224]	ANN and decision trees	Improved packets delivery with minimum latency and overhead	Increases the system complexity.
	[229]	QL algorithm	An approach that couples automatic learning into routing	Usually used in controlled environments.
	[237]	Supervised Deep Belief Architectures	Increased routing efficiency in terms of signaling overhead and throughput in SDN	The article provides interesting results in hindsight, i.e., DL, may not be suitable in every condition.
	[238]	Supervised learning	Application Aware Multipath Flow Routing	Scalability challenges in classification of higher number of applications.
	[239]	Semi-supervised ML (Laplacian SVM)	QoS through routing efficiency	With increasing the number of features, the accuracy increases but the cost of network overhead also increases.
	[240]	Generative DNNs	Flexible management and optimization in cognitive networks	Limited experiments and evaluation, more context information means more network overhead.
	[241]	Generic ML techniques and SDN	Increased network automation	Important challenges of ML in the context of networking are presented.
Routing in Dynamic (ad-hoc) Networks	[227]	Q-routing with RL	Minimal latency in routing in changing network environments	Need switching nodes to maintain RL modules.
	[231]	Adaptive Q-routing Full Echo algorithm	Route exploration in adaptive networks	Scalability and computation overhead challenges.
	[219]	RL using ant-based routing	Multi-path routing in dynamic networks	Rate of change in the network does not effect the routing overhead.
	[234]	Collaborative RL	Routing efficiency in MANETs	Signaling overhead due to feedback and sharing of route information among non-stationary or moving nodes.
	[228]	Gradient-descent algorithm for RL	Decentralized routing system	Applicable to homogeneous networks only.
	[233]	Improved Q-learning	Improved learning behavior and settling time for adaptive routing	Reduced instability compared to other Q-learning approaches.
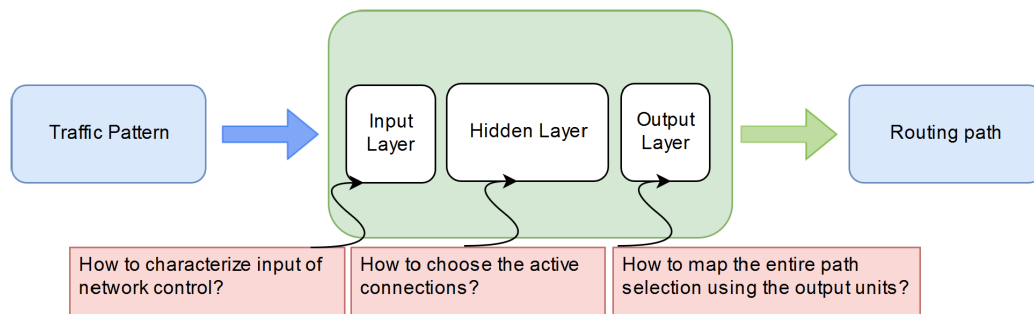
FIGURE 4. General learning approaches for network traffic control.

tion of the previous layer. Similarly, a greedy layer-wise training method is used to initialize the learning system and the back-propagation algorithm is used to fine-tune the DL training. To demonstrate the proposed system's efficiency, a proof-of-concept is developed for comparison with conventional routing, i.e., Open Shortest Path First

(OSPF) [211], [212], that shows significant improvement in terms of signaling overhead, throughput and delay.

Using the concepts of swarm intelligence for routing is discussed in [213]. Swarm intelligence [214] refers to solving problems through collaborative interactions among multiple simple processing units. Swarms can be small processing units and their coupling can have a wide variety of characteristics, however, there must be interaction among them [214]. The interaction is based on primitive actions to complete complex tasks with no supervision [215]. The article [213] overviews several algorithms that leverage swarm intelligence for routing. Swarm intelligence for routing in MANETs is proposed in [216], [217], providing significantly better packet delivery ratio and minimizing end-to-end delay with comparable control overhead, in juxtaposition to conventional routing protocols for MANETs.

Leveraging ML for improving the quality of transmission in optical networks, an integrated approach for routing and spectrum assignment is proposed in [218]. The proposed approach, using probabilistic outputs of ML-based quality of transmission, saves up to 30 percent of the spectrum and reduces the number of risky lightpaths. Risky lightpaths are those that exceed a certain threshold of bit error rate. For traffic prediction, supervised learning mechanisms using historical data and user profiles can predict future traffic requirements and consequent resource needs [39].

C. ROUTING IN DYNAMIC (AD-HOC) NETWORKS

Dynamic networks are subject to frequent and mostly unpredictable changes in topology and link costs. Routing in such networks places a high load on the routers to guarantee an effective performance that limits scalability and increases various costs (e.g., the chances of congestion). A case study on routing in dynamic networks using RL for routing is presented in [219]. The study investigates and improves on the ant-based single path routing algorithm [220], and proposes a multi-path routing algorithm. The main advantage of the proposed algorithm in [219] is that the routing traffic does not increase with the rate of change in the network, making it suitable for dynamic networks.

Opportunistic networks are dynamic and sporadic networks in which routes between communicating peers are established at run-time without prior knowledge of a route between the communicating peers [4]. Being ad-hoc in nature, opportunistic networks are delay tolerant with applications such as text messages and emails. However, costs such as in-network caching introduce challenges when nodes join and leave intermittently. Routing or finding routes towards the desired destination in such disconnected environments are one of the most compelling challenges faced by ad-hoc or opportunistic networks [4]. Most of the nodes in such networks try to learn from other nodes whether a destination can be reached [222], and such knowledge can highly increase the routing performance [4], [223].

ML-based protocol for efficient routing in opportunistic networks has been proposed in [224]. The routing protocol improves the performance of PROPHET+ protocol [225] with ML techniques to train itself based on several factors such as hop count, buffer capacity, node energy, moving speed of a node, popularity of a node, and a number of successful deliveries. The ML algorithm uses the past routing data to compute the capability of a node to deliver messages to intended destinations. Two ML models, i.e., ANN and DTs are used to calculate the probability of correct delivery of messages. The proposed protocol improves the delivery probability, and minimizes the average latency, overhead and buffer size requirements. Similarly, the wavelet neural network prediction model was used in [226] for multi-path routing in wireless mesh networks and was reported to have better adaptability and robustness to link failures and congestion.

Using RL for dynamic and adaptive routing has been described in [227], [228] respectively. The article [227] describes Q-routing algorithm [229] for packet routing using the RL module embedded into nodes of a switching network. The nodes keep statistics of local communication to derive routing decisions that lead to minimal latency in packet delivery in a changing network environment. In [228], gradient descent algorithm for RL [230] in routing has been suggested and evaluated. Learning the optimal route by trial and error repetitions, the algorithm avoids centralized control and global sharing of information regarding the structure of the network. However, the network considered for evaluation of the algorithm is a homogeneous one, which could be true in the 1990s, but not anymore.

Adaptive learning rates in RL for routing has been proposed in [231]. The proposed routing algorithm uses Adaptive Q-routing Full Echo algorithm [227] as well as adaptive learning rates to improve the exploration behavior. Q-routing [229] itself is an adaptive routing algorithm that uses QL to tell agents what optimal action to take in a controlled environment [232]. The routing approach in [233] further extends it with routing memory to reduce instability of routing under high load conditions and improve the performance in terms of settling the learning time.

Collaborative RL for optimizing routing in MANETs using negative and positive feedback mechanisms is presented in [234]. Collaborative RL enables groups of RL agents to solve system optimization problems in a dynamic, adaptive and decentralized manner. The routing agents use feedback about link selection to adapt and optimize routing according to the changing network conditions and properties. By exploiting stable routes for traffic flows, the protocol avoids interference and congestion, thus, optimizing throughput in ad-hoc networks. However, continuous feedback and sharing of route information among non-stationary or rapidly moving nodes incur extra signaling and computation costs for the agents.

One possibility to handle a heterogeneous network in terms of bandwidth, node capacities, and QoS requirements, as suggested in [228], is the centralized control framework. In Section VII, we describe routing through a centralized control framework, called SDN.
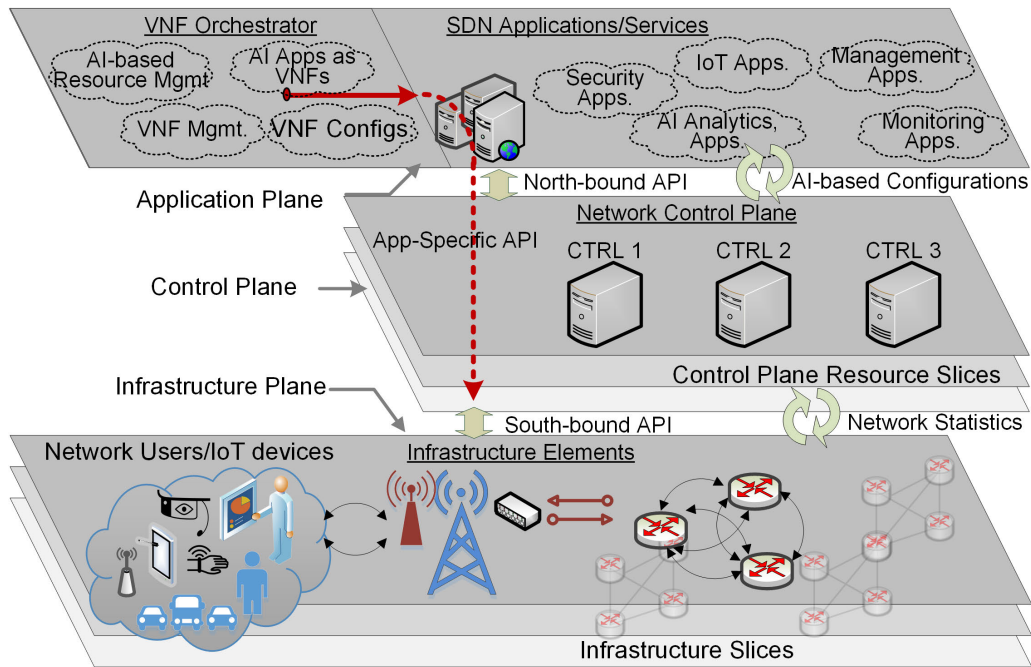
FIGURE 5. Generic SDN and NFV architecture.

D. LESSONS LEARNED

Different requirements of the network layer must be considered when designing ML-based routing and traffic control systems. For example, when designing an end-to-end routing protocol using ML, the latency costs due to ML processing (gathering the data, training the model, and distributing the results) must not increase beyond the acceptable threshold. Similarly, the network overhead in terms of link budget, storage or caching, and processing resources required for ML in the routing or switching nodes need to be counted when evaluating the resulting benefits of ML in the network layer. As stated in [242], in large networks where the data is gathered from multiple diverse nodes and heterogeneous links, the costs are much higher. In the case of dynamic routing, transient disruptions during routing protocol convergence cause huge network overhead due to topology information dissemination [202].

Furthermore, conventional ML techniques such as ANNs have evident shortcomings in terms of scalability and computation efficiency when considered for routing, as elaborated in [210]. Hence, adding more resource constraints through ML in routing on the data plane or routing devices would exacerbate the existing congestion and thus, the resulting latency challenges. Therefore, it is necessary to first evaluate the cost vs benefits before indulging or using ML-based techniques for routing. For example, DL based routing in backhaul and core networks is compared to the traditional OSPF routing mechanism in [237]. The results reveal that when the signaling interval between routers is more than a certain threshold (in milliseconds), both OSPF and DL yield the same result in terms of throughput and average delay.

Integrating DL-based mechanisms, nonetheless, require more computational and storage resources. Therefore, latency and scalability of the overall system must be evaluated before using ML in the network layer.

VI. ML FOR SDN AND NFV

Conventional communication networks relied on vendor-specific devices with closed proprietary solutions, requiring low-level vendor specific configurations resulting in many challenges including network down-time due to human errors, security, and management complexity, etc. To mitigate those challenges, a clean slate approach [243] that discards the existing ossified architecture and focuses on a completely new architecture, SDN [244], [245] with its de-facto implementation, i.e., OpenFlow [246], has gained a lot of momentum. SDN centralizes the network control logic by separating it from the data forwarding planes. The control logic, called the SDN controller, decides how to forward traffic on the behest of the data plane.

Intelligence gathering from the data plane through the south-bound API, as shown in Fig. 5, is already one of the basic functions of the controller. However, embedding ML into the controller can further boost the controller intelligence to avoid several risks involved with centralized control. For instance, in [247] the authors demonstrate how ML algorithms such as DT [248], naive Bayes [249], and SVN [250] increase the tolerance of the controller under security attacks. The proposed mechanisms enable the controller to deploy network-wide drop rule for malicious traffic, as opposed to local rule additions. Therefore, the coupling of SDN and

ML can have significant improvement in terms of network efficiency and security.

Network virtualization separates the network software logic from the hardware logic to enable sharing the network resources among multiple users (tenants) [251], [252]. By decoupling the network functions from hardware, NFV enables mapping multiple network functions (software instances) to single network element, or map a single network function to multiple network elements, thus increasing flexibility and scalability. Network virtualization is identified to solve many challenges that exist in communication networks, and is, therefore, considered from the physical layer up to the application layer [251], [253], [254], and beyond the layers.

In the following sub-sections, we discuss how ML-based approaches can improve the efficiency of SDN and NFV.

A. ML FOR SDN

SDN enables applications to manipulate the behavior of the network through generating flow forwarding rules [255], [256]. Application-aware traffic control opens new opportunities to prioritize and de-prioritize network traffic based on charging, QoS, and QoE policies. Utilizing network resource visibility and granular control over traffic flows, SDN paves the way for application-aware network traffic control. In [257], the authors proposed ML-based traffic classification through flow feature extraction. The main aim of this work is to enable application-aware policy enforcement with the help of ML and the OpenFlow protocol.

Leveraging the granularity provided by SDN, flow-routing using RL has been discussed in [258]. Flow-routing, unlike traditional source-destination IP-based routing, provides an opportunity to dynamically route packets of a flow through different network segments or even intermediary networks. The Q-learning approach used in [258] enables flow-preserving multipath routing. The proposed approach have benefits in terms of minimizing latency for moderate to high loads and accommodating changes. However, it brings forth scalability challenges in terms of convergence, specifically in large networks. The work, thus, stirs further research to benefit from RL for flow-routing in future networks.

Traffic prediction and optimal path performance prediction in SDN using ML have been discussed in [236]. Gaussian process regression (GPR), a Bayesian nonlinear regression model, is used for future traffic prediction that reduces the consumption of optical network resources (e.g. bandwidth) by 9 percent. Furthermore, various ML algorithms such as penalized linear regressions, nonlinear regressions, and ensembles of regression trees are compared in terms of bit error rate to predict optimal path performance. Finding efficient routes or paths for traffic flows in SDN based on different priorities or requirements of applications is presented in [238]. The multi-path routing framework [238] uses ML techniques to evaluate the characteristics of the flow and possible paths to optimize delay and efficiently use the available bandwidth.

A survey on the application of ML in SDN is presented in [46]. The article outlines how ML can play a major role in network resource management, fault identification and recovery, network security, traffic scheduling, and route planning. Various applications and algorithms of ML suitable for SDN are discussed, followed by future development of both fields, i.e., ML and SDN, together. Interesting future research includes how ML can help SDN extract knowledge from network logs, besides security, resource management, and routing. Moreover, in [240] the authors elaborate on how SDN coupled with ML techniques, such as DNN, enable flexible actuation and management of complex networked systems. The proposed methods exploit flow information to provide QoE-aware resource management as a step towards the intelligent and self-organized cognitive network.

Knowledge Defined Networking (KDN), coupling the ideas of ML and SDN, is presented in [241]. KDN extends the concepts of the Knowledge Plane (KP) for the Internet [259] through practical implementation using SDN. KP builds on Artificial Intelligence (AI) and cognitive systems [260] to enable the network to assemble itself given high-level instructions, adapt to changing requirements, automatically discover malfunctions and fix them or report why it cannot fix them [259]. However, the idea of KP has not been used or deployed yet due to the inherent limitations in the underlying network architectures such as distributed control systems ripe with complexity. By logically centralizing the network control plane, SDN mitigates such complexity, and with the global network state visibility, intelligence gathering and sharing are further facilitated [261]. The article [241] advocates that SDN, through logically centralized network control and global visibility of the network state, facilitates the deployment of the concept of KP which can leverage DL techniques towards realizing KDN.

Network traffic classification is highly important in SDN, mainly to ensure QoS for different traffic flows. Compared to traditional traffic classification for applications, QoS aware traffic classification has several benefits in SDN. For example, scalability is a major challenge in SDN due to the centralized control plane. Applications-based traffic classification will exacerbate the challenges of the availability of the control plane. QoS-aware traffic classification will yield the benefit of defining a few QoS classes, and then prioritizing the routing decisions based on those classes, rather than a many-fold higher number of applications. A QoS-aware traffic classification framework for SDN is proposed in [239] that defines suitable routing paths for a set of traffic flows. The path selection is based on i) local traffic identification component in the switches and ii) a global traffic classifier in the controller. The traffic classifier in the controller performs a mapping function based on selected features of flow such as packet arrival time, Hurst parameter, and port number, to provide a QoS class for a flow. Thus, the global classifier in the controller learns, builds, and refines the mapping function on historical traffic information to classify traffic in various

QoS categories. Similarly, ML-based QoE improvement is proposed and evaluated in [262].

By introducing programmability and centralizing the network control, SDN along with all benefits also opens networks to security challenges, as detailed in [261]. The centralized control frameworks, such as SDN controllers, oversee and control the entire network from a central vintage point, making it a favorite choice for resource exhaustion and Denial of Service (DoS) attacks [254]. Overwhelming the controller with any kind of traffic can cause a jamming effect, resource exhaustion, or a DoS attack as described in [263], [264]. Traditional mechanisms are mostly reactive, i.e., a challenge occurs and then a solution is deployed, which introduces delay or interruption in services. ML, in this vain, can provide proactive and predictive mechanisms to protect the network from possible threats [42], [265]. Therefore, there are a number of research efforts in using ML for increasing the security of SDNs, mainly the SDN controllers. Different ML techniques are evaluated in [266] for improving SDN security against Denial of Service (DoS) attacks. Authors in [247] used ML to identify DoS attacks and improve network resilience. Similarly, SVM is used in [267] for DoS detection in SDN, and various supervised-machine learning techniques are used in [268] for intrusion detection in SDN controllers. DL-based channel assignment in SDN-IoT is investigated in [146], mainly for traffic prediction to avoid network congestion.

B. ML FOR NFV

The efficient assignment of physical resources based on a history of services or service usage peak-times is an interesting area where the disciplines of ML can yield a significant improvement in NFV [18]. The authors outline that accurate resource allocation through service demand prediction using ML can be attained in NFV. The prediction mechanisms of ML along with NFV can also be used to save energy in large data centers and networks [269]. Similarly, ML can be used to detect or anticipate the sources of performance degradation in virtualized environments, and apply corrective measures well in advance [270]. NFV provides a greater level of elasticity. Hence, ML can be used to detect topology changes, and facilitate virtual machine migration to further improve the dynamicity of NFV [18].

In [271], ML-based NFV resource allocation is evaluated. Markov Decision Processes (MDP) [272], [273] is used to dynamically allocate NFV components to cloud resources, whereas ML is used to predict resource reliability in the allocation process. MDP is generally used for dynamic resource allocation problems that center on policy establishment considering long-term effects, cost factors, and optimal strategies. However, MDP has high overhead costs. Therefore, Bayesian learning methods are combined with MDP in [271], to improve the performance of MDP in dynamically predicting the reliability of cloud services based on usage history. Thus, using ML in predicting resource reliability for NFV components can significantly improve the allocation

and performance of the overall environment as demonstrated in [271].

ML-based classification algorithms are evaluated in [274]. The work studies the impact of flow features on machine classifiers and their effectiveness for different protocols. It is demonstrated that selecting the appropriate flow features increases accuracy and decreases classification delay. Furthermore, ML classification in a VNF environment where an NFV controller dynamically selects the classifiers and flow features to categorize network flows results in a flexible and scalable solution. The main significance of this work is that leveraging the techniques of NFV and VNFS, classification in ML can yield better results specifically due to the flexibility provided by these techniques of virtualization. This also constitutes an interesting research area to further explore the concepts of VNFs for improving the performance of various disciplines of ML in communication networks.

Due to the increasing number of VNFs, service function chains are highly important to put in order and connect different functions in specific ways to meet user demands [275], [276]. To utilize the underlying resources most efficiently, automated mapping and linking of different functions under strict latency can be achieved with ML. For example, authors of [277] use the restricted Boltzmann machine [278], a classic machine learning network structure, with a heuristic closed-loop feedback algorithm for chaining multiple different functions. Low-latency paths are searched to chain and connect functions to users in order to minimize resource utilization ratio and meet latency constraints.

Considering a specific use-case, virtualization has many benefits in data centers. For instance, non-virtualized data centers exhibit many challenges in security, QoS, and manageability mainly due to the dynamicity of applications and services [279]. However, virtualization alone may not suffice. Intelligent and dynamic sharing of resources among applications and services need the disciplines of ML to play an important role. For example, learning the use of a particular resource and proactively sharing resources among various services is one key area where the ML can play a major role. Mapping resources to services in a cost-effective manner, specifically when the number of services is increasing constantly, is a major challenge in data center virtualization [280].

C. SELF-DRIVING SOFTWARIZED NETWORKS

By enabling programmability of the underlying network infrastructure, SDN paves the way for efficient Virtual Network Functions (VNFs) placement [281], [282]. SDN also facilitates NFV to efficiently share network resources or deploy network functions in different network perimeters at run-time [281], [282] as shown in Fig. 5. SDN, thus, facilitates deploying ML services as applications or VNFs on one hand, and virtualize network resources on the other hand, as depicted in Fig. 5. SDN can also support the design of VNFs and increase the efficiency of NFV [283]. Similarly, NFV strengthens the concepts of SDN by providing

mechanisms to virtualize the SDN control or data planes [284]. Therefore, the two technologies, i.e., SDN and NFV, are highly complementary to each other [282]. The resulting network, combing the concepts and technologies of NFV and SDN, can be termed as a softwarized network, as discussed in [285].

By adding ML-based data-driven capabilities, a softwarized network can be extended with adaptation as elaborated in [285]. The authors discuss that the resulting softwarized network can react to the environment using ML-based data-driven observations, decision-making and control. Furthermore, by employing the concepts of empowerment, a self-driving network can be attained. Self-driving networks can measure, analyze and control themselves in an automated manner and can react to changes in the environment [286]. Since self-driving networks exploit existing flexibilities offered by the network to optimize themselves, in softwarized networks the programmability yields even more promising opportunities in this direction. Therefore, there are a number of research efforts in this direction. For example, in [287] the authors use SDN software switches to deploy data-driven approaches to improve network configurations.

D. LESSONS LEARNED

Even though SDN and NFV coupled with ML can further improve the performance of communication networks, the research in this direction is very limited and ignores a very important aspect of both technologies. In the case of SDN, most of the research is related to traffic flow handling and controller security. ML-based traffic flow handling yields a number of benefits such as minimizing flow setup latency, and improving QoS and QoE based on application-specific flow attributes. One of the main challenges of SDN is the security of the control plane, and ML has been proved to enhance the security of SDN controllers mainly from DoS attacks. However, very limited work has been carried out on improving the scalability of the SDN control plane with ML. Similarly, the involvement of the controller in ML can further increase the scalability challenges, and this has been not considered in most of the state-of-the-art research work.

For virtualization or NFV, the hypervisor is a key entity that sits in between the physical resources and the services that use those resources. Hypervisors map the physical resources to diverse services, as well as overlook and control both. Being central to the overall network like the SDN controller, hypervisor also has nearly similar scalability and security challenges. Therefore, improving the scalability and security of hypervisor through ML is an important research area that is not given full consideration in the state-of-the-art. ML can be used to predict traffic and service demands and, thus, proactively mitigate the challenges of the hypervisor being a bottleneck. Similarly, understanding the behavior of different instances of virtual functions, associated users and network slices can help avoid the challenges of over-provisioning of physical resources.

VII. ML FOR EDGE COMPUTING

The current de-facto architecture of mobile and IoT mobile services includes local user interfacing, sensor data collection and actuation, and data center-based service and application logic at the middle (edge) layer between the cloud and the devices. Such a three-tier architecture has proven benefits, including resource-efficiency at the local level and global availability of services. However, the edge paradigm is challenged by two major trends of today: the fast growth of sensor and user-generated data, and real-time requirements by modern applications related to e.g. augmented and virtual reality [288]. The current centralized cloud architecture has been shown to fail to meet the performance and scalability requirements of such applications and services [61], [288]. This development has intrigued strong interests towards realizing an *intelligent edge* between end-user/IoT devices and the cloud architecture.

Edge Computing (EC) [289]–[291] is a concept which pushes various computing and data analysis capabilities from centralized locations towards the edges of networks. It enables services to exploit the proximity of devices by e.g. providing ultra-low latency and high data rate communication, and provides tools to control and limit the propagation of private data of users. Fog Computing (FC) [292], [293], is a closely related concept to EC, covering caching, data processing and analytics near the source of the data to improve the performance, reduce the burden on data centers and core networks and improve the resilience against networking problems [289], [294]. Although EC and FC are frequently mixed and various overlapping definitions can be found in literature, the main distinction is that whereas EC mainly refers to the computational Edge infrastructure services (IaaS), FC focuses more on providing a platform for enterprise services (PaaS/SaaS) above Edge infrastructure.

Multi-Access Edge Computing (MEC) is a standard technology by European Telecommunications Standards Institute (ETSI) [295] for running applications and providing data storage and computational resources for mobile and IoT devices at the edges of new generation Radio Access Networks (RAN, 5G onwards). Current implementations following ETSI MEC include e.g. Nokia's vMEC [296], Huawei's MEC@CloudEdge [297], as well as OpenStack [298] -based Saguna MEC [299]. The main benefits provided by MEC include ultra-low latency communications towards mobile and IoT devices allowing application offloading, extra data storage and computational capacity for data processing and reducing the burden on data centers. Cloudlet [300] is another technology for realizing EC and FC. The main difference to MEC is that Cloudlet is considered a lightweight virtualization platform, thus it can be deployed on less resourceful edge nodes and used as a standalone implementation, that is configured and managed locally.

Following the vision of collaborative ML and EC, presented in [61], we recognize a two-fold role for ML in the context of edge/fog computing: 1) ML provides new capabilities such as predictive or decentralized control and

system orchestration, and predictive maintenance of network infrastructure and 2) distributed edge platforms provide the ML models with massive amounts of local data in a timely fashion to improve through online training. These roles of ML are visualized in Fig. 6.

In the first role, ML can be either seen as a fog level function for optimizing the placement of different functionality of applications, or edge level functionality for e.g. predicting changes in workload. Examples for fog level optimization include decision-making on how to partition computational tasks across the local, edge and cloud/data center level. Examples for edge level optimization include orchestration and initiation of EC resources, and decision-making on application offloading, with regard to the workload on edge nodes or user mobility across the edge platform. Fig. 6 visualizes this two-fold role of ML in optimizing edge/fog system operation and providing services for mobile applications.

In the second role, ML takes input from the underlying edge/fog platform to optimize its operation. Training ML at the edge over wireless networks while taking into account latency and reliability opens up novel research directions [61]. The benefits of ML optimization based on the behavioral and workload (balancing) information from edge/fog platform includes e.g. the ability to predict changes in workload, with respect to spatial and temporal constraints and user mobility, in order to orchestrate edge resources and task offloading in run-time.

Since EC and FC are relatively fresh research topics, despite the hype around them, increasing the research body is currently appearing that concerns cooperation of edge with ML in these two roles. In the following subsections, we summarize the most significant studies published so far.

A. ML-BASED OPTIMIZATION OF CLOUD/FOG SERVICES

Authors in [301] outline an intelligent approach for IoT analytics facilitating automated transitions between edge and cloud depending on the dynamic conditions of the IoT infrastructure and applications' requirements. The authors identify two central challenges: 1) multiple applications compete for the same resources, and 2) user application contexts change rapidly and unpredictably. As a potential solution, the authors propose an architecture which, based on the system context and user objectives, divides the analytic task into segments that can be offloaded to surrogate edge nodes that then return the results to the user application. This adaptive orchestration requires flexible platform architectures and system support services and enhances design-time deployment configurations by provisioning of cloud resources on the edge.

An intelligent resource allocation algorithm for optimal distribution of edge and cloud resources based on genetic algorithms is presented in [302]. With an iterative process, the proposed algorithm provides an optimal solution matching cloud provider resources with the applications. Based on evaluations, the authors conclude that the algorithm is effective in finding a Pareto-optimal match of cloud provider resources and can be tuned to specific application needs.

The authors also state that provider resource ranking and an application user prediction are the most interesting future directions for their work. The most significant challenges concern the optimized use of resources with multiple objectives of application providers and the federation operator. The facts that every application has unique optimization objectives and resources are very heterogeneous, make the process of optimal resource allocation for multiple applications complicated.

An architecture for composing analytics functionality on the cloud and deploying them for use at the edge is presented in [303]. The architecture allows the processing of data streams to be distributed in a controlled manner, with intensive but well-structured processing taking place at the edge. The authors see effectiveness in reducing latency, minimizing data movement and preserving local data privacy as the main benefits of EC. They investigate the functionality split between observer and controllers components and identify the supporting system tools required for efficiently composing distributed collaborative analysis applications for streaming data. The aim is to enable support for collaborative autonomous systems, principally by applying generative policy models.

B. ML-BASED OPTIMIZATION OF THE EDGE INFRASTRUCTURE

The joint offloading and autoscaling problem in energy harvesting MEC systems has been studied in [304]. They found that foresightedness and adaptivity are keys to the reliable and efficient operation of renewable-powered MEC. To enable fast learning in the presence of a priori unknown system parameters, a PDS based RL algorithm was developed to learn the optimal offloading and autoscaling policy by exploiting the special structure of the considered problem. The proposed model uses both online and offline RL algorithms to achieve improvements in learning rate and runtime performance when compared to standard RL algorithms such as QL. The simulations showed that the proposed scheme can significantly improve the EC performance, even being powered by intermittent and unpredictable renewable energy.

Authors in [305] present an architecture for energy-efficient D2D EC and discuss examples of different links and access patterns. With respect to offloading, the architecture incorporates both D2D communication and the cooperative relaying technique. Traffic offloading and balancing technologies exhibit further improvement of the total energy efficiency and the QoS to the edge users, and alleviates both inter and intra-cell interference and congestion. As future work, the authors consider utilizing ML and big data techniques to obtain satisfactory solutions to offloading, balancing, and allocation problems and wireless resource virtualization to simplify the process of the entire network.

There are several proposals for tasks and resource optimization in MEC platforms and MEC-based networks. A multi-stack reinforcement learning algorithm is proposed in [306] for MEC to optimize tasks or resource allocations
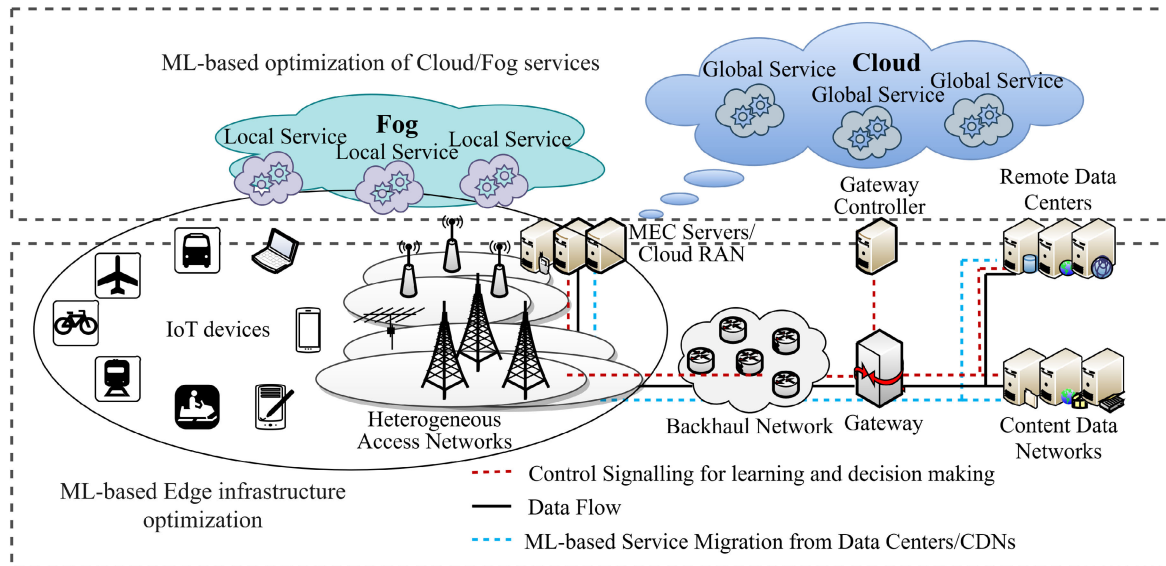
FIGURE 6. ML and Edge for service and infrastructure optimization.

for users. The proposed algorithm improves the convergence speed and learning efficiency. How to efficiently utilize MEC resources in the wake of increasing number of complex devices and services with growing number of configuration parameters with the help of ML is discussed in [52]. The work in [52] also highlights which computation and communication challenges related to MEC can be solved by different ML solutions. Collaboration between edge and centralized cloud resources using ML is proposed and evaluated in [307]. A distributed DL-based task offloading algorithm is proposed that generates offloading decisions among the end-user device, the edge cloud, and the central cloud servers.

A number of proposals and approaches for improving particular technology using MEC through ML exist. For example, [308] uses DRL to improve online computing offloading for non-orthogonal multiple access, [309] and [310] use multi-agent RL for cooperative caching and task offloading in MEC, respectively. Authors in [311] propose DRL-based energy-efficient task offloading for machine type communication in the edge. Minimizing the per-bit energy consumption in 5G MECs using DNNs is discussed in [312]. The work improves the energy efficiency of users through developing a digital twin of the real system and training the DL system in it. Similarly, stochastic online learning for MEC proposed in [313] minimizes the time-averaged operational costs of MECs. Service function chain placement and scaling in MEC using ML is presented in [314]. In [315], the authors describe the need of AI or ML based offloading in MEC and discuss some state-of-the-art research in this direction.

Also, the industry has shown notable interest in *Edge AI*. MediaTek has prepared an SW/HW solution for optimizing Edge AI performance [316]. According to the authors, AI on the edge devices takes the advantages of rapid response with low latency, high privacy, better robustness, and more efficient use of network bandwidth. The paper introduces

challenges and technology trends of Edge AI including ML, neural network acceleration and reduction, and heterogeneous run-time mechanism. The main contribution of the paper is the design of a dedicated AI processing unit to provide better power efficiency, eliminating 95 % of energy consumption. MediaTek sees power and computation efficiency as centric requirements for Edge AI. The proposed solution supports current AI frameworks, including Caffe, Tensorflow, MXNet, and NNabla. The toolchains including model translators in NeuroPilot allow programmers to enable AI applications on devices. Therefore, MediaTek's Edge AI solution can be realized in a wide range of applications.

C. EDGE-BASED ML OPTIMIZATION
The new breed of intelligent devices and high-stake applications such as drones, augmented and virtual reality, and autonomous systems, require a novel paradigm change calling for distributed, low-latency and reliable ML at the wireless network edge (referred to as *Edge ML*) in [61]. The authors studied the requirements and key building blocks for enabling Edge ML, different ANN architectures and their tradeoffs, as well as theoretical and technical enablers stemming from a wide range of mathematical disciplines. In Edge ML, training data is unevenly distributed over a large number of edge nodes each having access to only a tiny fraction of the data, and training and inference are carried out collectively over wireless links, where edge devices communicate and exchange their learned models. Several case studies pertaining to various high-stake applications are presented demonstrating the effectiveness of Edge ML in unlocking the full potential of 5G and beyond.

An interesting approach for edge ML training architecture is federated learning (FL) [317], [318]. FL enables computing nodes to collaboratively learn a shared prediction model while keeping the training data on the device [317]. Authors

in [319] present a method for FL of deep networks based on iterative model averaging. They conduct an empirical evaluation demonstrating the robustness of the approach to unbalanced and non-independent and identically distributed (Non-IID) data distributions. Communication costs are seen as the main constraint. The experiments show that federated learning can be made feasible in constrained environments since their model can train high-quality models using relatively few rounds of communication, as compared to synchronized stochastic gradient descent. According to the authors, FL also offers many practical privacy benefits, such as providing stronger guarantees via differential privacy, secure multi-party computation, and their combination. They see this research direction as a promising avenue for future work.

A framework for the implementation of FL algorithms in wireless networks is presented in [317]. The framework enables FL to be optimized and corroborated between cellular base stations and wireless users to overcome the challenges associated with wireless nature of the network. Optimization of the convergence time of FL has been researched in [320], [321]. The authors have elaborated how to minimize the convergence time while optimizing the FL performance in a resource constrained cellular wireless environment. Authors of [322] have studied FL for optimizing ML prediction models using local training data. They show that existing algorithms are not suitable for this setting, and propose a new algorithm for sparse convex problems. They demonstrate that it is possible to design efficient algorithms utilizing local training data that does not leave the source node at any phase. The work also sets a path for future research needed in the context of federated optimization. Examples of future work include developing a fully asynchronous version of the proposed algorithm where the updates are applied as soon as they arrive, and gaining a better theoretical understanding of the developed algorithm. Furthermore, a survey on using FL in MEC platforms is presented in [55]. The article discusses the background of FL in MEC, implementation challenges and solutions, and future research directions.

There are also several proposals and techniques that help in particular way to improve the performance of ML using resources in the edge. For example, authors in [323] propose distribution of ML tasks for IoT in an MEC setup and [324] discusses how to improve privacy of AI or ML solutions used in the edge. One of the main advantage for using ML in the MEC is that MEC efficiently mitigates most of the challenges related to real-time responsiveness to systems using ML [52], [288]. Coupled with ML techniques such as FL, MEC also improves the privacy of user data compared to centralized cloud systems as discussed in [325]. Therefore, MEC brings its own benefits for ML optimization ranging from minimizing latency to increasing privacy.

D. LESSONS LEARNED

ML introduces high demands in terms of energy, memory and computing resources and therefore limits their adoption for resource-constrained edge devices, as already stated in [61].

The problem is emphasized in the envisioned IoT-edge scenarios, where edge devices are typically highly constrained in terms of energy, memory and computing resources. New types of ubiquitous computing scenarios where digital services follow users through smart surroundings wherever they move, applications such as augmented/virtual reality, as well as devices such as drones and applications require novel paradigm change calling for distributed, low-latency and reliable edge and fog ML technologies.

In front of the emerging challenges, new types of virtualization technologies enabling dynamically deployable Edge ML functions suitable for constrained devices are needed. Benefits are then seen in increased scalability. Container technologies, such as Docker [326] and intelligent orchestration using technologies such as Kubernetes [327] are promising tools for realizing lightweight virtualization with such resource-constrained edge nodes. Therefore, also ML solutions need to be further studied in this direction to enable dynamic placement of lightweight ML functionality in edge and fog nodes.

VIII. AN OVERVIEW OF ML FOR NETWORK SECURITY

Analysis of large amounts of data or monitoring the network traffic-in-transit for security requires a paradigm of proactive, self-aware and self-adaptive intelligent systems. Such systems employ novel algorithms and technologies of ML, and as a result, cyber-security may become one of the best application areas for ML. Conventionally, a security attack or a security lapse happens and then patching starts once the attack or the lapse has been recognized. This needs to change since critical infrastructures such as electricity smart grids, transportation, and health-care systems are moving towards online connectivity through network infrastructures, and reactive security measures will not suffice [328]. The change must be towards proactive security measures due to the criticality of the security of these infrastructures. Proactive security measures require continuous intelligence gathering and using that intelligence to mitigate the possibility of security risks and lapses. ML, with its promising algorithms and full solutions gaining appreciation in other fields, is also used in the realm of network security.

Research on using ML for security can be traced back to the Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHA) that brought the two domains (security and ML) at an intersection [329]. CAPTCHA is widely used in commercial contexts where a human is differentiated from a bot through recognizing distorted characters or a sequence of characters [330]. As pattern recognition gets more sophisticated, the drive to maintain security will also drive more sophisticated use of ML. Consequently, the advances in decision procedures, model checking, and recently, Boolean satisfiability have grasped the attention of cyber-security researchers [329]. ML was for the first time used for intrusion detection through flow classification in 1994 [331]. This also led to the work on using ML for traffic classification over the Internet [26]. Firewalls using

Deep Packet Inspection (DPI) can be considered as another instantiating technology of ML in security [332]. However, more efforts are needed to develop specific ML-based security approaches and solutions for future networks [332]. Below, we present an overview of the most pertinent and relevant ML-based security approaches for future communication networks.

A. ML FOR SECURITY OF COMMUNICATION NETWORKS

Due to the diversity of services and devices in next-generation networks, autonomous decision making in terms of security policy verification, policy conversion to configurations and subsequent deployment require leveraging ML for the purpose. As described in [40], ML has the potential to help network operators in situations where there are no prior data or experiences, or the data is too complicated to understand with traditional approaches. With the conglomeration of diverse IoT devices, UAVs, V2X, wearables and smart home appliances into communication networks, differentiating a security attack from legitimate traffic will be practically impossible or unmanageable without using the concepts of ML [328], [333]. One of the stringent requirements of IoT, or for instance UAVs and V2X communication, will be latency. Security services such as authentication and access control need to be proactively carried out within the time constraints in order to meet the main service requirements such as service migration from one edge node to another. In doing so, ML will play a critical role to timely identify the terminal actions and requirements to avoid service interruptions [333].

The most prominent use of ML in network security would be to recognize malicious traffic. Hence, traffic classification mechanisms helping in detecting patterns lay the foundation of identifying anomalies over the network [26], [334]. Traditional IP traffic classification mechanisms directly inspect the content of packets, targeting port numbers or payloads in the packets [335]. However, using encryption mechanisms to obfuscate packet contents including TCP or UDP port numbers, and changing packet payload structures make such mechanisms either not accurate or too costly. Therefore, ML-based approaches have been around that take into consideration several other factors such as inter-arrival times of packets, flow-duration, payload size, and bandwidth [26], [334]. Such factors used in combination can apparently provide better results, leading to interesting research on using ML for anomaly and Intrusion Detection Systems (IDSs) as described in the next sub-section.

Using the tools of ML such as ANN and DTs, [336] demonstrates revealing hidden communication by mobile malware or malicious software. Similarly, authors in [337] use fuzzy logic, ANN and trend analysis in IDSs. Moreover, various ML algorithms and solutions such as Bayesian methods [338] to deal with uncertainty in terms of finding malicious activity, as in DPI, opens new horizons for strengthening the security of networks of a massive number of connected devices (e.g. IoT), applications, and diversified services. A communication-efficient and failure-robust

secure protocol for aggregation of high-dimensional data are demonstrated in [339]. One of the main objectives of the protocol [339] is the privacy preservation of sensitive data using ML in multi-party computation systems.

A number of use-cases have been evaluated where the disciplines of ML are used to secure the communication network from cyber-attacks. For example, in [333] the authors demonstrate using ML techniques to secure vehicular communication and effectively counter Denial-of-Service (DoS) attacks. Stealth attacks, that cannot be detected with traditional detection mechanisms using state estimation, are rising in smart grid systems. Two approaches using the disciplines of ML (supervised, unsupervised learning) are evaluated in [340] and compared to traditional detection schemes. The ML approaches are capable to detect false data injection and protect the system effectively [340].

A cloud monitoring model based on cooperative intelligent agents is proposed in [341]. The agents, located in different units of the cloud, learn about the environment using specific monitoring methods, communicate with each other, and make decisions based on ML. These agents detect abnormalities, malfunctions and security threats within the systems. There are also interesting opensource tools used for intrusion detection leveraging ML. For instance, TensorFlow [342] is a collection of open-source ML libraries developed by google that utilizes complex data structures through ANN. The usefulness of TensorFlow-based malicious traffic detection is demonstrated in [343].

B. ML FOR NETWORK INTRUSION DETECTION

IDSs prevent information systems from unauthorized use and safeguard online resources from malignant activities [344]. IDS monitors network traffic and checks the traffic for doubtful or malicious activities. Therefore, traffic classification is highly important in IDSs. Since using ML for intrusion detection is a highly researched topic, e.g., used ML for IDS in 1990s [345], in this sub-section the existing survey article on the topic of ML-based intrusion systems is highlighted. A survey of different techniques to classify Internet traffic using ML is presented in [26]. The target of the classification in [26] is two-folded, i.e., for QoS assurance and lawful interception. A comparative study of ML, statistical-based and knowledge-based IDS is conducted in [346]. Albeit being capable of robust global searches, and reaching accurate thresholds through learning and training, ML-based systems, however, have higher costs in terms of computations and overhead [346]. Therefore, distributed traffic analysis engines using agent-oriented systems (as described in Section IX) are proposed to tackle the challenges of computation and communication overhead [347].

A thorough survey on using the concepts of data mining and ML for intrusion detection is presented in [32]. Since data is highly important for ML, data mining techniques along with algorithms used in both ML and data mining are described. Three types of cyber analytics for IDS are defined. i) Misuse-based that comprise techniques designed to detect

known attacks through signatures, ii) anomaly-based that find abnormalities or deviations from normal behavior, and iii) hybrid using a combination of both. The article [32] then describes various methods of ML such as ANNs, Bayesian Networks, clustering, etc. and discusses the methods used in the three types of cyber analytics. The article, thus, provides a detailed overview of different approaches in different scenarios related to using ML for IDSs.

C. LESSONS LEARNED

In communication networks, one of the most pertinent uses of ML is improving network security, and that is why there are a lot of research efforts in this direction. ML-based traffic classification and inspection mechanisms, specifically in the age of big data, have highly improved the security of communication systems. Furthermore, ML has been used to improve the security of different technologies used in communication networks such as cloud and edge computing platforms, IoT devices and services, as well as interfaces between different services and systems. However, there are certain very important issues that have received less research attention. For example, one important aspect of security that has received less research attention is the security of ML systems themselves, as discussed in [348]. A detailed study on vulnerabilities in ML systems is carried out in [348], [349] presenting a variety of attacks against ML systems. For instance, if the learning is affected, surely the output will be affected, and an adversarial opponent can produce the desired output through compromising the learning systems by only feeding wrong information. Similarly, communication systems will serve highly time-sensitive services and systems. ML systems, on the other hand, first need to gather enough data to train ML models correctly and then deploy intelligence-based security procedures. Such systems, thus, will incur higher delays that may not be appropriate for time-sensitive services and systems such as V2X communication.

Furthermore, conventionally the use of ML for improving network security is highly researched, however, on the contrary, e.g., using ML for devising and deploying malicious attacks, must also be investigated. Security attacks leveraging ML can be more challenging to detect or stop. For example, it is demonstrated in [350] that ANNs can learn how to perform encryption or decryption to an extent, and maybe quite effective in making sense of metadata and in traffic analysis. Thus, it can be deduced that the disciplines and techniques of ML, can be used in the future for cyber attacks, to counterfeit or compromise the confidentiality and integrity of the communication systems. The solution to such challenges lie in ensuring the correct operation of the learning systems, the security of the raw data and gathered information and compliance and monitoring of the intelligent actionable output.

IX. DISCUSSION AND FUTURE RESEARCH DIRECTIONS

ML is highly important in communication networks, as discussed throughout this article. In this study, we identify many

areas that still need further investigation, though many areas like the physical, MAC, and network layers have strong state-of-the-art and thus require more detailed studies. The future of ML for intelligent decision making in communication applications looks promising. In many cases, it is even a necessity, as the communication performance requirements keep aggressively increasing whereas available resources and technological development trends remain limited. For example, the general DL paradigm, which acts as a data-driven universal function approximation tool, is an obvious direction to go in many of the areas covered in this survey. However, it is not obvious which particular learning algorithm setup works best for a given application. Moreover, a thorough understanding of how these nonlinear and iterative learning algorithms actually produce their outputs as well as their inherent learning delay with respect to emerging ultra-low latency applications possess important challenges yet to be resolved.

Communication networks need their own definitions, disciplines, algorithms, and tools of ML. Having said that, fine-tuning of the definitions, disciplines, algorithms and tools of ML for specific use-cases of communication networks are still in infancy. Most of the concepts of ML are used *as is* in communication networks, optimizing one objective while overlooking other constraints such as latency, link, storage and processing overhead. A more concerning point is the modest interest towards a cross-layered approach, in which ML in one layer could also benefit or help in optimization in another layer. Moreover, the negative effects on the performance of other layers such as increased latency in finding the optimal route on scheduling in the MAC and physical layer are not properly investigated. Therefore, in this section, we describe what remains to be done to effectively apply the techniques of ML in communication networks.

A. ML FOR LAYER ONE: PHYSICAL LAYER

The physical layer can highly benefit from the techniques of ML. There are some fundamental challenges, however. For instance, common benchmarks and data sets are crucial for comparing the performance of various ML algorithms. Yet no comparable benchmarks or common data sets of physical layer data, such as channel observations, are available [62]. Furthermore, the varying nature of channel conditions such as SNR values is deemed to complicate the overall scenario. What is more, most wireless signal processing algorithms are designed with complex valued signals, whereas the most widely used NN hardware use real arithmetic. Finally, it is challenging to scale the learning algorithms for complex end-to-end communication systems and systems that require feedback mechanisms such as those used for adaptive communication.

Using the disciplines of ML such as DL on the physical layer needs further investigation from multiple points of view. The proven DL-based modulation recognition methods [351], [352] illustrate their usefulness in feature extraction and recognition in the domain. This implies that DL methods

can be also used in recognition of system parameters for source coding, extracting CSI, and others, from wireless signals that can enable knowledgeable or intelligent physical layer based communication systems [63]. Moreover, combining the intelligence on the physical layer with the upper layers can yield a flexible and intelligent overall network. Cross-layer intelligence sharing using the techniques of ML lacks research and constitutes an interesting future research topic.

B. ML FOR LAYER TWO: MEDIUM ACCESS CONTROL

MAC in wireless networks can be regarded as a large-scale control problem, where multiple devices must either be controlled via a central entity (i.e., infrastructure-based networks) or the devices autonomously manage themselves (i.e., ad-hoc networks) to get access to the shared resource. Applications set requirements for access, in terms of bandwidth, latency, power, etc. However, the large amount and high variety of devices increase the challenges of MAC control. Traditional MAC algorithms behave reactively, collecting information and making decisions using mostly sub-optimal solutions due to the complexity of the problem. Therefore, to meet the requirement of future wireless networks, to guarantee QoS, and to provide access to the massive number of devices simultaneously, MAC protocols need to behave proactively.

Some progress has already been made in proactive MAC algorithms with advanced learning techniques. However, there is still a need to investigate and devise more robust and rigorous solutions for challenges such as interference prediction, resource allocation, power optimization, and mobility prediction. Besides these, the accommodation of asymmetric traffic between uplink and downlink is one of the major issues in future wireless networks, as most of the IoT based cellular network is based on HD-FDD and would result in under-utilization of spectrum resources. Other open research directions, where ML techniques can provide significant improvements from the MAC perspective, are the convergence of the licensed and the unlicensed spectra, and the exploitation of spectrum availability in both of the frequency bands to improve the efficiency of wireless networks as a whole.

Further, another open research direction is related to emerging LPWAN technologies (i.e., NB-IoT, LTE M). IoT devices typically have low computation power. Therefore, a delay needs to be scheduled between a control channel message and the associated data transmission to ensure the IoT devices have enough time to decode the control information. Currently, the scheduling delay is fixed for all devices, which results in reduced QoS and inefficient spectrum utilization [353]. ML methods could potentially be used to autonomously select the scheduling delay, based on the capabilities of the connected node, thus improving performance.

Finally, in current wireless networks, control channels are fixed, which incurs large overheads on spectrum resources.

Some efforts have been made to optimize the control plane to allow small data transmissions along with control information [354]. However, with the help of machine learning techniques, it may be possible to design an adaptive control channel, based on the traffic and the requirements of the connected devices. In particular, ML methods can potentially help MAC algorithms predict the expected needs of the devices and respond accordingly.

C. ML FOR LAYER THREE: NETWORK LAYER

The architecture of the network layer has recently been a central theme of research, predicted even to undergo a paradigm shift. Two major issues, however, withhold the major changes required. First, the network control plane is tightly coupled with the network infrastructure or forwarding plane. Second, the network control part comprises proprietary solutions forming a closed and distributed network control architecture. These two features lead to a network that requires manual configurations of high-level policies in command line interface environments, and a lack of global visibility of the network state. The results are obvious, i) difficulties in updating network-level policies in response to changing network conditions, ii) manual configurations resulting in errors, and iii) the network is closed to innovations. Accordingly, the network architecture has been termed as ossified [246]. To break these barriers, loosing the control from the infrastructure and programmability of the network have been proposed in different forms from active networking [355] to the 4D approach [356] and Ethane [357]. The most recent one that is still a high research focus is SDN, with its de-facto implementation, called the OpenFlow architecture [246]. However, the work on using ML for SDN is very limited, as discussed in the subsequent subsection.

Apart from separating the network control from the data forwarding plane, there are important considerations in terms of using ML. For instance, a lot of research has been carried out on using ML in routing protocols for MANETs. MANETs, in principle, are made randomly and spuriously. Learning, on the other hand, requires information gathering, analyzing the information and making some conclusions that could be converted into decisions. The outcome of the learning may not be useful when new devices join, or existing devices leave the network. Security of such learning algorithms is another challenge. In the case of decentralized systems, misleading or wrong information spread by sporadic devices might worsen the behavior of the entire network or routing protocols. Therefore, latency or time constraints-based use of ML in MANETs and security of the network are two interesting research challenges. The network overhead created with retrieving information for learning, the latency in learning-based decisions, and security of the learning environment are challenges for routing protocols in nearly all types of networks. Therefore, these are and will remain interesting research challenges that need further investigation.

D. ML FOR SDN AND NFV

The most prominent feature of SDN is simplifying the data forwarding mechanisms in communication networks through the control-data planes' separation [358]. However, this separation brings new challenges that will circumvent the efficiency of forwarding. First, SDN applications are capable to manipulate the network behavior. Thus, authorizing applications to modify forwarding rules, and limiting applications not to change flow rules defined by other applications is one challenge [359]. The second, and more concerning challenge in SDN, is the scalability or ensuring the availability of the centralized, though logically centralized, control plane. Therefore, the most pertinent use of ML would be to i) learn the requirements of applications and solicit their behavior to utilize the underlying (in many cases shared) infrastructure in an efficient way, and ii) proactively estimate the burden of controllers and protect the controller from being the bottleneck for the whole network. The SDN controller oversees or controls the underlying forwarding devices, and gathers information (flow statistics) for various purposes, e.g., load balancing. Adding software modules to the controller to gather more intelligence for learning and processing will simply increase the controller scalability challenges. Therefore, further research is needed to avoid the controller scalability challenges, yet improve the controller intelligence for both application and data forwarding planes. Furthermore, one of the strengths of SDN is network abstraction. Mapping network resources with services using ML to avoid network congestion using ML needs further research.

Research on improving network or network function virtualization in future networks using ML is very limited. There exist a lot of research on using ML for IDS and IPS. ML-based IDS/IPS can efficiently secure the hypervisors, yet no significant research is visible on this topic at the time of this writing. Security of slices, isolated services, and verticals, and virtualization management entities using ML have not been investigated yet. The other way around, using virtualization techniques to improve the performance of ML algorithms is also very limited. ML algorithms deployed in VNFs that can be moved around in network premises or different network perimeters can yield a significant improvement in terms of minimizing delay and network overhead. Traditionally, ML algorithms run in a centralized location and gather knowledge from different network locations and links. Such systems clearly incur high link overhead and delay. A VNF running ML that can be deployed at run-time in any network perimeter will thus minimize link overhead and delay. This area also needs further investigation. Therefore, the use of ML in virtualization technologies and vice versa makes interesting future research topics not properly researched yet.

E. ML FOR EDGE COMPUTING

Given the proliferation of smart surroundings and fast growth of sensor data gathered by mobile user devices and IoT

devices embedded in our surroundings, the current centralized cloud architecture will fail to meet the performance and scalability requirements. This development has intrigued strong interests towards realizing an intelligent edge between end-user/IoT devices and the cloud architecture. In edge and fog computing, ML has a centric role in optimizing the system functionality, performance and resource-efficiency. ML can be seen as a method for optimizing the placement of different functions of applications and for optimizing the underlying edge architecture. Furthermore, ML can also take input from underlying edge and fog architecture to optimize its operation. Since edge and fog computing are quite new concepts and not yet widely taken into commercial use, the need for further research is obvious.

The avenues for future research include at least development of methods for using ML as a central component in deciding which parts of application and service data and processing should be managed locally, which part should be managed at the edge and which part to be sent to be handled by data centers (ML for fog computing), considering, e.g., performance, efficiency and privacy. Similarly, an interesting avenue for future research includes the use of ML as a central component for managing the edge infrastructure, i.e., coordinating which nodes serve users/end devices at certain logical/physical area or in certain application types, or how/where to migrate processing when users/end devices move across the networks (ML for edge computing).

Another interesting viewpoint to be considered in the future research is related to the use of large amounts of data from diverse sources. For example, using large amounts of data available from different sources such as edge infrastructure, end-user, and different sensor devices will need proper contextual information of the sources. This will further increase the complexity for the learning systems. Furthermore, current ML technologies are relatively heavy and generally not optimal in situations where computing needs to be handled at mobile, IoT and other constrained devices. Hence, there is a clear need for lightweight intelligent Edge ML mechanisms, where ML computing is optimally placed on the edge-based three-tier architecture to provide the needed functionality, performance and resource efficiency. Furthermore, privacy is an omnipresent challenge related to all of the above-mentioned research directions. Privacy preserving techniques that can protect the privacy of data while being in use must be incorporated to the usage of data for purposes other than the intended ones are highly important. Such the leaking of learning data, and maintain learning data to leak outside the system or used of purposes other than the intended ones.

F. ML FOR NETWORK SECURITY

An interesting study focusing on the limited deployment of ML-based techniques for intrusion detection is presented in [28]. The authors outline that the premise of considering ML-based anomaly detection to be mainly suitable for finding novel attacks does not hold grounds for two

reasons. First, the authors argue that the strength of ML is finding correlations between concurring events, taking into consideration previously observed activities. Second, the cost of errors can be very high compared to the use of ML in other applications. Therefore, in the case of a fresh attack for which previous data is unavailable, accuracy in intrusion detection will be questionable. However, a lesson can be learned from these observations; ML-based security systems might provide better results in intrusion prevention systems. It is also important to note that ML-based systems usually take longer times to respond to security lapses. For example, Torabi et. al [60] conducted a study of security systems for Domain Name System (DNS) security. The study reveals that most of the systems that employ ML require hours or even days to detect threats. Therefore, new faster learning and convergence systems must be designed for ML-based security systems. Similarly, light-weight ML-based security systems must be designed for future networks. The computation necessary to perform security analysis might not be available for all systems that can generate large amounts of data, for instance, IoT networks. Hence, further research is needed to develop ML-based security systems that are capable of running on low computation devices. One direction is the collaborative or federated learning approaches [360], [361] in which the computation and communication overhead is divided among the nodes.

X. CONCLUSION

This article provides a detailed overview of the state-of-the-art of disciplines, techniques, and tools of ML that are used in communication networks. First, the applications of ML in wireless networks are described in three layers, i.e., physical, MAC, and network layers. Then the applications of ML are discussed in novel technological concepts such as SDN, NFV, and MEC. An overview of ML in network security is provided, mainly to highlight the need for ML-based security operations in wireless networks. This paper highlights the merger of major research efforts between the disciplines, techniques, and tools of ML and the technologies of communication networks. Since ML has yet to be deployed on a practical basis in wireless networks, there are many interesting open research questions that need further investigation. Therefore, we presented interesting future research directions from the physical layer up to the network layer, and in the newly emerging networking technologies and services.

REFERENCES

[1] R. Mahajan, D. Wetherall, and T. Anderson, "Understanding BGP misconfiguration," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 32, no. 4, pp. 3–16, 2002.

[2] ONF. (Oct. Oct. 2013). *SDN Security Considerations in the Data Center.* Open Networking Foundation. [Online]. Available: https://www.opennetworking.org/sdn-resources/sdn-library

[3] A. Wool, "A quantitative study of firewall configuration errors," *Computer*, vol. 37, no. 6, pp. 62–67, Jun. 2004.

[4] H. Hamed and E. Al-Shaer, "Taxonomy of conflicts in network security policies," *IEEE Commun. Mag.*, vol. 44, no. 3, pp. 134–141, Mar. 2006.

[5] M. E. Morocho Cayamcela and W. Lim, "Artificial intelligence in 5G technology: A survey," in *Proc. Int. Conf. Inf. Commun. Technol. Converg. (ICTC)*, Oct. 2018, pp. 860–865.

[6] P. Demestichas, A. Georgakopoulos, D. Karvounas, K. Tsagkaris, V. Stavroulaki, J. Lu, C. Xiong, and J. Yao, "5G on the horizon: Key challenges for the radio-access network," *IEEE Veh. Technol. Mag.*, vol. 8, no. 3, pp. 47–53, Sep. 2013.

[7] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2009.

[8] A. L. Samuel, "Some studies in machine learning using the game of checkers," *IBM J. Res. Develop.*, vol. 3, no. 3, pp. 210–229, Jul. 1959.

[9] C. Zhang, P. Patras, and H. Haddadi, "Deep learning in mobile and wireless networking: A survey," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2224–2287, 3rd Quart., 2019.

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

[11] Z. Ghahramani, "Unsupervised learning," in *Summer School on Machine Learning*. Berlin, Germany: Springer, 2003, pp. 72–112.

[12] X. Zhu, "Semi-supervised learning literature survey," Dept. Comput. Sci., Univ. Wisconsin-Madison, Madison, WI, USA, Tech. Rep. 1530, 2005.

[13] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 1998.

[14] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *J. Artif. Intell. Res.*, vol. 4, no. 1, pp. 237–285, Jan. 1996.

[15] R. C. Tryon, *Cluster Analysis: Correlation Profile and Orthometric (Factor) Analysis for the Isolation of Unities in Mind and Personality*. Lillington, NC, USA: Edwards brother, Inc., 1939.

[16] V. Estivill-Castro, "Why so many clustering algorithms: A position paper," *ACM SIGKDD Explor. Newslett.*, vol. 4, no. 1, pp. 65–75, 2002.

[17] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2327, 2000.

[18] T. S. Buda, H. Assem, L. Xu, D. Raz, U. Margolin, E. Rosensweig, D. R. Lopez, M.-I. Corici, M. Smirnov, R. Mullins, O. Uryupina, A. Mozo, B. Ordozgoiti, A. Martin, A. Alloush, P. O'Sullivan, and I. G. Ben Yahia, "Can machine learning aid in delivering new use cases and scenarios in 5G?" in *Proc. IEEE/IFIP Netw. Oper. Manage. Symp. (NOMS)*, Apr. 2016, pp. 1279–1284.

[19] I. Ahmad, S. Shahabuddin, T. Kumar, E. Harjula, M. Meisel, M. Juntti, T. Sauter, and M. Ylianttila, "Challenges of AI in wireless networks for IoT," *IEEE Ind. Electron. Mag.*, vol. 15, no. 1, pp. 1–16, Dec. 2020.

[20] B. Silver, "Netman: A learning network traffic controller," in *Proc. 3rd Int. Conf. Ind. Eng. Appl. Artif. Intell. Expert Syst. (IEA/AIE)*, vol. 2. New York, NY, USA, 1990, pp. 923–931, doi: 10.1145/98894.99101.

[21] L. Lu, G. Y. Li, A. L. Swindlehurst, A. Ashikhmin, and R. Zhang, "An overview of massive MIMO: Benefits and challenges," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 742–758, Oct. 2014.

[22] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 186–195, Feb. 2014.

[23] C.-K. Wen, S. Jin, K.-K. Wong, J.-C. Chen, and P. Ting, "Channel estimation for massive MIMO using Gaussian-mixture Bayesian learning," *IEEE Trans. Wireless Commun.*, vol. 14, no. 3, pp. 1356–1368, Mar. 2015.

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

[25] S. Ali, "6G white paper on machine learning in wireless communication networks," 2020, *arXiv:2004.13875*. [Online]. Available: http://arxiv.org/abs/2004.13875

[26] T. T. T. Nguyen and G. Armitage, "A survey of techniques for Internet traffic classification using machine learning," *IEEE Commun. Surveys Tuts.*, vol. 10, no. 4, pp. 56–76, 4th Quart., 2008.

[27] F. Dressler and O. B. Akan, "A survey on bio-inspired networking," *Comput. Netw.*, vol. 54, no. 6, pp. 881–900, Apr. 2010.

[28] R. Sommer and V. Paxson, "Outside the closed world: On using machine learning for network intrusion detection," in *Proc. IEEE Symp. Secur. Privacy*, May 2010, pp. 305–316.

[29] Z. Zhang, K. Long, J. Wang, and F. Dressler, "On swarm intelligence inspired self-organized networking: Its bionic mechanisms, designing principles and optimization approaches," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 513–537, 1st Quart., 2014.

[30] M. A. Alsheikh, S. Lin, D. Niyato, and H.-P. Tan, "Machine learning in wireless sensor networks: Algorithms, strategies, and applications," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 4, pp. 1996–2018, 4th Quart., 2014.

[31] X. Wang, X. Li, and V. C. M. Leung, "Artificial intelligence-based techniques for emerging heterogeneous network: State of the arts, opportunities, and challenges," *IEEE Access*, vol. 3, pp. 1379–1391, 2015.

[32] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1153–1176, 2nd Quart., 2016.

[33] Z. M. Fadlullah, F. Tang, B. Mao, N. Kato, O. Akashi, T. Inoue, and K. Mizutani, "State-of-the-art deep learning: Evolving machine intelligence toward tomorrow's intelligent network traffic control systems," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2432–2455, 4th Quart., 2017.

[34] O. B. Sezer, E. Dogdu, and A. M. Ozbayoglu, "Context-aware computing, learning, and big data in Internet of Things: A survey," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 1–27, Feb. 2018.

[35] M. Mamdouh, M. A. I. Elrukhsi, and A. Khattab, "Securing the Internet of things and wireless sensor networks via machine learning: A survey," in *Proc. Int. Conf. Comput. Appl. (ICCA)*, Aug. 2018, pp. 215–218.

[36] Q. Liu, P. Li, W. Zhao, W. Cai, S. Yu, and V. C. M. Leung, "A survey on security threats and defensive techniques of machine learning: A data driven view," *IEEE Access*, vol. 6, pp. 12103–12117, 2018.

[37] Q. Mao, F. Hu, and Q. Hao, "Deep learning for intelligent wireless networks: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 2595–2621, 4th Quart., 2018.

[38] J. Mata, I. de Miguel, R. J. Durán, N. Merayo, S. K. Singh, A. Jukan, and M. Chamania, "Artificial intelligence (AI) methods in optical networks: A comprehensive survey," *Opt. Switching Netw.*, vol. 28, pp. 43–57, Apr. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S157342771730231X

[39] F. Musumeci, C. Rottondi, A. Nag, I. Macaluso, D. Zibar, M. Ruffini, and M. Tornatore, "An overview on application of machine learning techniques in optical networks," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 2, pp. 1383–1408, 2nd Quart., 2019.

[40] M. G. Kibria, K. Nguyen, G. P. Villardi, O. Zhao, K. Ishizu, and F. Kojima, "Big data analytics, machine learning, and artificial intelligence in next-generation wireless networks," *IEEE Access*, vol. 6, pp. 32328–32338.

[41] O. Simeone, "A very brief introduction to machine learning with applications to communication systems," *IEEE Trans. Cognit. Commun. Netw.*, vol. 4, no. 4, pp. 648–664, Dec. 2018.

[42] J. Xie, F. R. Yu, T. Huang, R. Xie, J. Liu, C. Wang, and Y. Liu, "A survey of machine learning techniques applied to software defined networking (SDN): Research issues and challenges," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 393–430, 1st Quart., 2019.

[43] Z. Xiong, Y. Zhang, D. Niyato, R. Deng, P. Wang, and L. Wang, "Deep reinforcement learning for mobile 5G and beyond: Fundamentals, applications, and challenges," *IEEE Veh. Technol. Mag.*, vol. 14, no. 2, pp. 44–52, Jun. 2019.

[44] N. C. Luong, D. T. Hoang, S. Gong, D. Niyato, P. Wang, Y.-C. Liang, and D. I. Kim, "Applications of deep reinforcement learning in communications and networking: A survey," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3133–3174, 4th Quart., 2019.

[45] 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.

[46] Y. Zhao, Y. Li, X. Zhang, G. Geng, W. Zhang, and Y. Sun, "A survey of networking applications applying the software defined networking concept based on machine learning," *IEEE Access*, vol. 7, pp. 95397–95417, 2019.

[47] M. Wang, Y. Cui, X. Wang, S. Xiao, and J. Jiang, "Machine learning for networking: Workflow, advances and opportunities," *IEEE Netw.*, vol. 32, no. 2, pp. 92–99, Mar. 2018.

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

[49] Y. Qian, J. Wu, R. Wang, F. Zhu, and W. Zhang, "Survey on reinforcement learning applications in communication networks," *J. Commun. Inf. Netw.*, vol. 4, no. 2, pp. 30–39, Jun. 2019.

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

[51] J. Wang, C. Jiang, H. Zhang, Y. Ren, K.-C. Chen, and L. Hanzo, "Thirty years of machine learning: The road to Pareto-optimal wireless networks," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 3, pp. 1472–1514, 3rd Quart., 2020.

[52] T. K. Rodrigues, K. Suto, H. Nishiyama, J. Liu, and N. Kato, "Machine learning meets computation and communication control in evolving edge and cloud: Challenges and future perspective," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 1, pp. 38–67, 1st Quart., 2020.

[53] L. Ruan, M. P. I. Dias, and E. Wong, "Enhancing latency performance through intelligent bandwidth allocation decisions: A survey and comparative study of machine learning techniques," *J. Opt. Commun. Netw.*, vol. 12, no. 4, p. B20, 2020.

[54] B. Ma, W. Guo, and J. Zhang, "A survey of online data-driven proactive 5G network optimisation using machine learning," *IEEE Access*, vol. 8, pp. 35606–35637, 2020.

[55] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Jiao, Y.-C. Liang, Q. Yang, D. Niyato, and C. Miao, "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 3, pp. 2031–2063, 3rd Quart., 2020.

[56] F. Dressler and O. Akan, "Bio-inspired networking: From theory to practice," *IEEE Commun. Mag.*, vol. 48, no. 11, pp. 176–183, Nov. 2010.

[57] M. Bkassiny, Y. Li, and S. K. Jayaweera, "A survey on machine-learning techniques in cognitive radios," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 3, pp. 1136–1159, 3rd Quart., 2013.

[58] S. Karunaratne and H. Gacanin, "An overview of machine learning approaches in wireless mesh networks," *IEEE Commun. Mag.*, vol. 57, no. 4, pp. 102–108, Apr. 2019.

[59] S. Yao, Y. Zhao, A. Zhang, S. Hu, H. Shao, C. Zhang, L. Su, and T. Abdelzaher, "Deep learning for the Internet of Things," *Computer*, vol. 51, no. 5, pp. 32–41, May 2018.

[60] S. Torabi, A. Boukhtouta, C. Assi, and M. Debbabi, "Detecting Internet abuse by analyzing passive DNS traffic: A survey of implemented systems," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 3389–3415, 4th Quart., 2018.

[61] J. Park, S. Samarakoon, M. Bennis, and M. Debbah, "Wireless network intelligence at the edge," *Proc. IEEE*, vol. 107, no. 11, pp. 2204–2239, Nov. 2019.

[62] T. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Trans. Cogn. Commun. Netw.*, vol. 3, no. 4, pp. 563–575, Dec. 2017.

[63] T. Wang, C.-K. Wen, H. Wang, F. Gao, T. Jiang, and S. Jin, "Deep learning for wireless physical layer: Opportunities and challenges," *China Commun.*, vol. 14, no. 11, pp. 92–111, Nov. 2017.

[64] S. Haykin, *Neural Networks: A Comprehensive Foundation*. Upper Saddle River, NJ, USA: Prentice-Hall, 2007.

[65] N. Benvenuto, F. Piazza, and A. Uncini, "A neural network approach to data predistortion with memory in digital radio systems," in *Proc. IEEE Int. Conf. Commun. (ICC)*, vol. 1, May 1993, pp. 232–236.

[66] B. E. Watkins, R. North, and M. Tummala, "Neural network based adaptive predistorter for the linearization of nonlinear RF amplifiers," in *Proc. MILCOM*, vol. 1, Nov. 1995, pp. 145–149.

[67] A. A. M. Saleh, "Frequency-independent and frequency-dependent nonlinear models of TWT amplifiers," *IEEE Trans. Commun.*, vol. 29, no. 11, pp. 1715–1720, Nov. 1981.

[68] B. E. Watkins and R. North, "Predistortion of nonlinear amplifiers using neural networks," in *Proc. MILCOM 96 IEEE Mil. Commun. Conf.*, vol. 1, Oct. 1996, pp. 316–320.

[69] M. Ibnkahla, "Neural network predistortion technique for digital satellite communications," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 6, Jun. 2000, pp. 3506–3509.

[70] F. Langlet, H. Abdulkader, D. Roviras, A. Maller, and F. Castanie, "Adaptive predistortion for solid state power amplifier using multilayer perceptron," in *Proc. IEEE Global Telecommun. Conf.*, vol. 1, Nov. 2001, pp. 325–329.

[71] N. Naskas and Y. Papananos, "Neural-Network-Based adaptive baseband predistortion method for RF power amplifiers," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 51, no. 11, pp. 619–623, Nov. 2004.

[72] Y. Qian and F. Liu, "Neural network predistortion technique for nonlinear power amplifiers with memory," in *Proc. 1st Int. Conf. Commun. Netw. China*, Oct. 2006, pp. 1–5.

[73] J. Yang, J. Gao, and F. Jiang, "Neural network predistortion technique for nonlinear RF amplifiers," in *Proc. Int. Conf. Comput. Sci. Softw. Eng.*, vol. 4, Dec. 2008, pp. 769–772.

[74] T. Liu, S. Boumaiza, and F. M. Ghannouchi, "Dynamic behavioral modeling of 3G power amplifiers using real-valued time-delay neural networks," *IEEE Trans. Microw. Theory Techn.*, vol. 52, no. 3, pp. 1025–1033, Mar. 2004.

[75] M. Isaksson, D. Wisell, and W. Rönnow, "Wide-band dynamic modeling of power amplifiers using radial-basis function neural networks," *IEEE Trans. Microw. Theory Techn.*, vol. 53, no. 11, pp. 3422–3428, Nov. 2005.

[76] M. Bahoura and C.-W. Park, "FPGA-implementation of an adaptive neural network for RF power amplifier modeling," in *Proc. IEEE 9th Int. New Circuits Syst. Conf.*, Jun. 2011, pp. 29–32.

[77] L. M. Aguilar-Lobo, J. R. Loo-Yau, J. E. Rayas-Sánchez, S. Ortega-Cisneros, P. Moreno, and J. A. Reynoso-Hernández, "Application of the NARX neural network as a digital predistortion technique for linearizing microwave power amplifiers," *Microw. Opt. Technol. Lett.*, vol. 57, no. 9, pp. 2137–2142, Sep. 2015.

[78] R. Hongyo, Y. Egashira, T. M. Hone, and K. Yamaguchi, "Deep neural network-based digital predistorter for Doherty power amplifiers," *IEEE Microw. Wireless Compon. Lett.*, vol. 29, no. 2, pp. 146–148, Feb. 2019.

[79] J. Bruck and M. Blaum, "Neural networks, error-correcting codes, and polynomials over the binary n-cube," *IEEE Trans. Inf. Theory*, vol. 35, no. 5, pp. 976–987, Sep. 1989.

[80] G. Zeng, D. Hush, and N. Ahmed, "An application of neural net in decoding error-correcting codes," in *Proc. IEEE Int. Symp. Circuits Syst.*, vol. 2, May 1989, pp. 782–785.

[81] W. R. Caid and R. W. Means, "Neural network error correcting decoders for block and convolutional codes," in *Proc. IEEE Global Telecommun. Conf. Exhib. (GLOBECOM)*, vol. 2, Dec. 1990, pp. 1028–1031.

[82] L. G. Tallini and P. Cull, "Neural nets for decoding error-correcting codes," in *Proc. IEEE Tech. Appl. Conf. Workshops. Northcon/95. Conf. Rec.*, Oct. 1995, pp. 89–94.

[83] X.-A. Wang and S. B. Wicker, "An artificial neural net viterbi decoder," *IEEE Trans. Commun.*, vol. 44, no. 2, pp. 165–171, Feb. 1996.

[84] M. E. Buckley and S. B. Wicker, "A neural network for predicting decoder error in turbo decoders," *IEEE Commun. Lett.*, vol. 3, no. 5, pp. 145–147, May 1999.

[85] A. Hamalainen and J. Henriksson, "A recurrent neural decoder for convolutional codes," in *Proc. IEEE Int. Conf. Commun.*, vol. 2, Jun. 1999, pp. 1305–1309.

[86] H. Abdelbaki, E. Gelenbe, and S. E. El-Khamy, "Random neural network decoder for error correcting codes," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, vol. 5, Jul. 1999, pp. 3241–3245.

[87] T. Gruber, S. Cammerer, J. Hoydis, and S. T. Brink, "On deep learning-based channel decoding," in *Proc. 51st Annu. Conf. Inf. Sci. Syst. (CISS)*, Mar. 2017, pp. 1–6.

[88] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006.

[89] E. Nachmani, Y. Be'ery, and D. Burshtein, "Learning to decode linear codes using deep learning," in *Proc. 54th Annu. Allerton Conf. Commun., Control, Comput. (Allerton)*, Sep. 2016, pp. 341–346.

[90] S. Cammerer, T. Gruber, J. Hoydis, and S. ten Brink, "Scaling deep learning-based decoding of polar codes via partitioning," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2017, pp. 1–6.

[91] F. Liang, C. Shen, and F. Wu, "An iterative BP-CNN architecture for channel decoding," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 144–159, Feb. 2018.

[92] W. Lyu, Z. Zhang, C. Jiao, K. Qin, and H. Zhang, "Performance evaluation of channel decoding with deep neural networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–6.

[93] J. Seo, J. Lee, and K. Kim, "Activation functions of deep neural networks for polar decoding applications," in *Proc. IEEE 28th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Oct. 2017, pp. 1–5.

[94] N. Doan, S. A. Hashemi, and W. J. Gross, "Neural successive cancellation decoding of polar codes," in *Proc. IEEE 19th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Jun. 2018, pp. 1–5.

[95] W. Xu, Z. Wu, Y.-L. Ueng, X. You, and C. Zhang, "Improved polar decoder based on deep learning," in *Proc. IEEE Int. Workshop Signal Process. Syst. (SiPS)*, Oct. 2017, pp. 1–6.

[96] S. Speidel, "Neurobeamformer: Adaptive beamforming via neural networks," Naval Ocean Syst. Center, San Diego, CA, USA, Tech. Rep. 1540, 1988.

[97] S. L. Speidel, "Neurobeamformer II: Further exploration of adaptive beamforming via neural networks," Naval Ocean Syst. Center, San Diego, CA, USA, Tech. Rep. 1606, 1989.

[98] B. Quach, H. Leung, T. Lo, and J. Litva, "Hopfield network approach to beamforrning in spread spectrum communication," in *Proc. IEEE 7th SP Workshop Stat. Signal Array Process.*, Jun. 1994, pp. 409–412.

[99] J. Simmers and T. O'Donnell, "Adaptive RBF Neural Beamforming," in *Proc. IEEE Command, Control, Commun. Intell. Technol. Appl. Conf.*, 1992, pp. 94–98.

[100] T. O'Donnell, J. Simmers, and D. Jacavanco, "Neural beamforming for phased-array antennas," in *Proc. Antenna Appl. Symp.*, vol. 1, 1993, pp. 106–115.

[101] J. Simmers, H. Southall, and T. O'Donnell, "Advances in neural beamforming," in *the Proc. 1993 Antenna Appl. Symp., Electromagn. Lab.* Urbana-Champaign, IL, USA: Univ. Illinois, 1993, pp. 203–216.

[102] T. O'Donnell, J. Simmers, and H. Southall, "An introduction to neural beamforming," in *Proc. IEEE Dual-Use Technol. Appl. Conf.*, vol. 1, 1994, pp. 483–492.

[103] H. L. Southall, J. A. Simmers, and T. H. O'Donnell, "Direction finding in phased arrays with a neural network beamformer," *IEEE Trans. Antennas Propag.*, vol. 43, no. 12, pp. 1369–1374, Dec. 1995.

[104] R. Lovato and X. Gong, "Phased antenna array beamforming using convolutional neural networks," in *Proc. IEEE Int. Symp. Antennas Propag. USNC-URSI Radio Sci. Meeting*, Jul. 2019, pp. 1247–1248.

[105] K.-L. Du, A. K. Y. Lai, K. K. M. Cheng, and M. N. S. Swamy, "Neural methods for antenna array signal processing: A review," *Signal Process.*, vol. 82, no. 4, pp. 547–561, Apr. 2002.

[106] A. Rawat, R. N. Yadav, and S. C. Shrivastava, "Neural network applications in smart antenna arrays: A review," *AEU - Int. J. Electron. Commun.*, vol. 66, no. 11, pp. 903–912, Nov. 2012.

[107] A. Goldsmith, "Joint source/channel coding for wireless channels," in *Proc. IEEE 45th Veh. Technol. Conf. Countdown Wireless 21st Century*, vol. 2, Jul. 1995, pp. 614–618.

[108] T. J. O'Shea, T. Erpek, and T. Charles Clancy, "Deep learning based MIMO communications," 2017, *arXiv:1707.07980*. [Online]. Available: http://arxiv.org/abs/1707.07980

[109] S. Dörner, S. Cammerer, J. Hoydis, and S. ten Brink, "Deep learning based communication over the air," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 132–143, Feb. 2018.

[110] A. Felix, S. Cammerer, S. Dörner, J. Hoydis, and S. ten Brink, "OFDM-autoencoder for end-to-end learning of communications systems," in *Proc. IEEE Int. Workshop Signal Proc. Adv. Wireless Commun. (SPAWC)*, Jun. 2018, pp. 1–5.

[111] C. Monteleoni, H. Balakrishnan, N. Feamster, and T. Jaakkola, "Managing the 802.11 energy/performance tradeoff with machine learning," Massachusetts Inst. Technol., Cambridge, USA, Tech. Rep. 8970817, 2004.

[112] N. Samuel, T. Diskin, and A. Wiesel, "Deep MIMO detection," in *Proc. IEEE 18th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Jul. 2017, pp. 1–5.

[113] H. He, C.-K. Wen, S. Jin, and G. Y. Li, "A model-driven deep learning network for MIMO detection," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*, Nov. 2018, pp. 584–588.

[114] H. He, C.-K. Wen, S. Jin, and G. Y. Li, "Deep learning-based channel estimation for beamspace mmwave massive MIMO systems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 5, pp. 852–855, Oct. 2018.

[115] A. K. Nandi and E. E. Azzouz, "Algorithms for automatic modulation recognition of communication signals," *IEEE Trans. Commun.*, vol. 46, no. 4, pp. 431–436, Apr. 1998.

[116] J. Joung, "Machine learning-based antenna selection in wireless communications," *IEEE Commun. Lett.*, vol. 20, no. 11, pp. 2241–2244, Nov. 2016.

[117] A. Balatsoukas-Stimming, O. Castaneda, S. Jacobsson, G. Durisi, and C. Studer, "Neural-network optimized 1-bit precoding for massive MU-MIMO," in *Proc. IEEE 20th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Jul. 2019, pp. 1–5.

[118] H. Ye, G. Y. Li, and B.-H. Juang, "Power of deep learning for channel estimation and signal detection in OFDM systems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 1, pp. 114–117, Feb. 2018.

[119] J. Liu, K. Mei, X. Zhang, D. Ma, and J. Wei, "Online extreme learning machine-based channel estimation and equalization for OFDM systems," *IEEE Commun. Lett.*, vol. 23, no. 7, pp. 1276–1279, Jul. 2019.

[120] H. Huang, J. Yang, H. Huang, Y. Song, and G. Gui, "Deep learning for super-resolution channel estimation and doa estimation based massive MIMO system," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8549–8560, Sep. 2018.

[121] K. Kim, J. Lee, and J. Choi, "Deep learning based pilot allocation scheme (DL-PAS) for 5G massive MIMO system," *IEEE Commun. Lett.*, vol. 22, no. 4, pp. 828–831, Apr. 2018.

[122] H. Q. Ngo and E. G. Larsson, "EVD-based channel estimation in multi-cell multiuser MIMO systems with very large antenna arrays," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2012, pp. 3249–3252.

[123] A. Ahmad, S. Ahmad, M. H. Rehmani, and N. U. Hassan, "A survey on radio resource allocation in cognitive radio sensor networks," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 2, pp. 888–917, 2nd Quart., 2015.

[124] A. Asadi and V. Mancuso, "A survey on opportunistic scheduling in wireless communications," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 4, pp. 1671–1688, 4th Quart., 2013.

[125] D. Scazzoli, A. Kumar, N. Sharma, M. Magarini, and G. Verticale, "Fault recovery in time-synchronized mission critical ZigBee-based wireless sensor networks," *Int. J. Wireless Inf. Netw.*, vol. 24, no. 3, pp. 268–277, Sep. 2017.

[126] D. Scazzoli, A. Kumar, N. Sharma, M. Magarini, and G. Verticale, "A novel technique for ZigBee coordinator failure recovery and its impact on timing synchronization," in *Proc. IEEE 27th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Sep. 2016, pp. 1–5.

[127] F. D. Calabrese, L. Wang, E. Ghadimi, G. Peters, L. Hanzo, and P. Soldati, "Learning radio resource management in RANs: Framework, opportunities, and challenges," *IEEE Commun. Mag.*, vol. 56, no. 9, pp. 138–145, Sep. 2018.

[128] R. Ganti and M. Haenggi, "Spatial and temporal correlation of the interference in ALOHA ad hoc networks," *IEEE Commun. Lett.*, vol. 13, no. 9, pp. 631–633, Sep. 2009.

[129] U. Schilcher, C. Bettstetter, and Brandner, "Temporal correlation of interference in wireless networks with Rayleigh block fading," *IEEE Trans. Mobile Comput.*, vol. 11, no. 12, pp. 2109–2120, Dec. 2012.

[130] M. Ozawa, T. Ohtsuki, W. Jiang, and Y. Takatori, "Interference alignment for time-varying channel with channel and weight predictions based on auto regressive model," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2015, pp. 1–6.

[131] S. Chinchali and S. Tandon. (2012). *Deep Learning for Wireless Interference Segmentation and Prediction*. [Online]. Available: https://www.semanticscholar.org/author/Sandeep-Chinchali/3102622

[132] M. K. Atiq, U. Schilcher, J. F. Schmidt, and C. Bettstetter, "Semi-blind interference prediction in wireless networks," in *Proc. 20th ACM Int. Conf. Modeling, Anal. Simulation Wireless Mobile Syst.*, Nov. 2017, pp. 19–23.

[133] J. Yu, M. Farcasin, and E. Chan-Tin, "Wireless interference prediction for embedded health devices," in *Proc. 26th Int. Conf. Comput. Commun. Netw. (ICCCN)*, Jul. 2017, pp. 1–6.

[134] Q. Li, M. Zhou, Y. Wu, S. Feng, and P. Zhang, "Precise interference estimation for the uplink of LTE heterogeneous networks," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2014, pp. 1235–1240.

[135] Q. Li, C. Li, Y. Wu, S. Feng, and P. Zhang, "An investigation on cooperative interference prediction in LTE–A indoor hotspot scenarios," in *Proc. IEEE 81st Veh. Technol. Conf. (VTC Spring)*, May 2015, pp. 1–6.

[136] F. Diehm and G. Fettweis, "Cooperative interference prediction for enhanced uplink link adaptation under backhaul delays," in *Proc. IEEE 23rd Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2012, pp. 173–177.

[137] K. K. Leung, "Power control by interference prediction for broadband wireless packet networks," *IEEE Trans. Wireless Commun.*, vol. 1, no. 2, pp. 256–265, Apr. 2002.

[138] Y. Cong, X. Zhou, and R. A. Kennedy, "Interference prediction in mobile ad hoc networks with a general mobility model," *IEEE Trans. Wireless Commun.*, vol. 14, no. 8, pp. 4277–4290, Aug. 2015.

[139] C. H. M. de Lima, M. Bennis, and M. Latva-aho, "Statistical analysis of self-organizing networks with biased cell association and interference avoidance," 2013, *arXiv:1301.3601*. [Online]. Available: http://arxiv.org/abs/1301.3601

[140] M. Chen, W. Saad, and C. Yin, "Virtual reality over wireless networks: Quality-of-Service model and learning-based resource management," *IEEE Trans. Commun.*, vol. 66, no. 11, pp. 5621–5635, Nov. 2018.

[141] M. Chen, W. Saad, and C. Yin, "Liquid state machine learning for resource and cache management in LTE-U unmanned aerial vehicle (UAV) networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 3, pp. 1504–1517, Mar. 2019.

[142] L. Ding, Y. Wang, P. Wu, and J. Zhang, "Position-based user-centric radio resource management in 5G UDN for ultra-reliable and low-latency vehicular communications," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, May 2019, pp. 1–6.

[143] A. Roy, V. Borkar, P. Chaporkar, and A. Karandikar, "Low complexity online radio access technology selection algorithm in LTE-WiFi HetNet," *IEEE Trans. Mobile Comput.*, vol. 19, no. 2, pp. 376–389, Feb. 2020.

[144] M. Chen, M. Mozaffari, W. Saad, C. Yin, M. Debbah, and C. S. Hong, "Caching in the sky: Proactive deployment of cache-enabled unmanned aerial vehicles for optimized Quality-of-Experience," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 5, pp. 1046–1061, May 2017.

[145] K. Witrisal and C. Antón-Haro, "New localization methods for 5G wireless systems and the Internet-of-Things, white paper," Aalborg Univ., Aalborg, Denmark, COST Action CA15104, 2018.

[146] H. Wymeersch, G. Seco-Granados, G. Destino, D. Dardari, and F. Tufvesson, "5G mmWave positioning for vehicular networks," *IEEE Wireless Commun.*, vol. 24, no. 6, pp. 80–86, Dec. 2017.

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

[148] R. Combes, I. E. Bouloumi, S. Senecal, and Z. Altman, "The association problem in wireless networks: A policy gradient reinforcement learning approach," 2013, *arXiv:1306.2554*. [Online]. Available: https://arxiv.org/abs/1306.2554

[149] D. S. Wickramasuriya, C. A. Perumalla, K. Davasligoglu, and R. D. Gitlin, "Base station prediction and proactive mobility management in virtual cells using recurrent neural networks," in *Proc. IEEE 18th Wireless Microw. Technol. Conf. (WAMICON)*, Apr. 2017, pp. 1–6.

[150] S.-M. Senouci, A.-L. Beylot, and G. Pujolle, "Call admission control in cellular networks: A reinforcement learning solution," *Int. J. Netw. Manage.*, vol. 14, no. 2, pp. 89–103, Mar. 2004.

[151] N. Morozs, T. Clarke, and D. Grace, "A novel adaptive call admission control scheme for distributed reinforcement learning based dynamic spectrum access in cellular networks," in *Proc. Int. Symp. Wireless Commun. Syst. (ISWCS)*, Aug. 2013, pp. 1–5.

[152] S.-M. Senouci, A.-L. Beylot, and G. Pujolle, "A dynamic Q-learning-based call admission control for multimedia cellular networks," in *Proc. IEEE Int. Conf. Mobile Wireless Commun. Netw. (MWCN)*, Recife, Brazil, Aug. 2001, pp. 1–8. [Online]. Available: https://hal.archives-ouvertes.fr/hal-01570731

[153] N. Morozs, T. Clarke, and D. Grace, "A novel adaptive call admission control scheme for distributed reinforcement learning based dynamic spectrum access in cellular networks," in *Proc. 10th Int. Symp. Wireless Commun. Syst. (ISWCS)*, Aug. 2013, pp. 1–5.

[154] L. Ghouti, T. R. Sheltami, and K. S. Alutaibi, "Mobility prediction in mobile ad hoc networks using extreme learning machines," *Procedia Comput. Sci.*, vol. 19, pp. 305–312, 2013.

[155] J. Capka and R. Boutaba, "Mobility prediction in wireless networks using neural networks," in *Management of Multimedia Networks and Services*, J. Vicente and D. Hutchison, Eds. Berlin, Germany: Springer, 2004, pp. 320–333.

[156] Y.-H. Chang, T. Ho, and L. Pack Kaelbling, "Mobilized ad-hoc networks: A reinforcement learning approach," in *Proc. Int. Conf. Autonomic Comput.*, May 2004, pp. 240–247.

[157] M. R. Hassan, M. S. M. Haque, M. I. Hossain, M. M. Hassan, and A. Alelaiwi, "A novel cascaded deep neural network for analyzing smart phone data for indoor localization," *Future Gener. Comput. Syst.*, vol. 101, pp. 760–769, Dec. 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X19308696

[158] A. Coluccia and A. Fascista, "A review of advanced localization techniques for crowdsensing wireless sensor networks," *Sensors*, vol. 19, no. 5, p. 988, Feb. 2019.

[159] J. A. del Peral-Rosado, R. Raulefs, J. A. Lopez-Salcedo, and G. Seco-Granados, "Survey of cellular mobile radio localization methods: From 1G to 5G," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 2, pp. 1124–1148, 2nd Quart., 2018.

[160] N. Saeed, H. Nam, T. Y. Al-Naffouri, and M. Alouini, "A state-of-the-art survey on multidimensional scaling-based localization techniques," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3565–3583, 4th Quart., 2019.

[161] J. A. del Peral-Rosado, J. A. Lopez-Salcedo, and G. Seco-Granados, "Impact of frequency-hopping NB-IoT positioning in 4G and future 5G networks," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, May 2017, pp. 815–820.

[162] S. Ezpeleta, J. Claver, J. Pérez-Solano, and J. Martí, "RF-based location using interpolation functions to reduce fingerprint mapping," *Sensors*, vol. 15, no. 10, pp. 27322–27340, Oct. 2015.

[163] W. Zhang, K. Liu, W. Zhang, Y. Zhang, and J. Gu, "Deep neural networks for wireless localization in indoor and outdoor environments," *Neurocomputing*, vol. 194, pp. 279–287, Jun. 2016.

[164] Y. Tifani, B. Lee, and E. Jeong, "A patient's indoor positioning algorithm using artificial neural network and SVM," *J. Theor. Appl. Inf. Technol.*, vol. 95, no. 16, 2017.

[165] B. Berruet, O. Baala, A. Caminada, and V. Guillet, "DelFin: A deep learning based CSI fingerprinting indoor localization in IoT context," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Sep. 2018, pp. 1–8.

[166] Z. Chen, H. Zou, J. Yang, H. Jiang, and L. Xie, "WiFi fingerprinting indoor localization using local feature-based deep LSTM," *IEEE Syst. J.*, vol. 14, no. 2, pp. 3001–3010, Jun. 2020.

[167] H. Malik, H. Pervaiz, M. Mahtab Alam, Y. Le Moullec, A. Kuusik, and M. Ali Imran, "Radio resource management scheme in NB-IoT systems," *IEEE Access*, vol. 6, pp. 15051–15064, 2018.

[168] M. Al-Imari, P. Xiao, M. A. Imran, and R. Tafazolli, "Low complexity subcarrier and power allocation algorithm for uplink OFDMA systems," *EURASIP J. Wireless Commun. Netw.*, vol. 2013, no. 1, p. 98, Apr. 2013.

[169] S. Chinchali, P. Hu, T. Chu, M. Sharma, M. Bansal, R. Misra, M. Pavone, and S. Katti, "Cellular network traffic scheduling with deep reinforcement learning," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 1–9.

[170] G. Alnwaimi, S. Vahid, and K. Moessner, "Dynamic heterogeneous learning games for opportunistic access in LTE-based Macro/Femtocell deployments," *IEEE Trans. Wireless Commun.*, vol. 14, no. 4, pp. 2294–2308, Apr. 2015.

[171] B. Bojović, E. Meshkova, N. Baldo, J. Riihijärvi, and M. Petrova, "Machine learning-based dynamic frequency and bandwidth allocation in self-organized LTE dense small cell deployments," *EURASIP J. Wireless Commun. Netw.*, vol. 2016, no. 1, p. 183, Aug. 2016.

[172] F. Bernardo, R. Agusti, J. Pérez-Romero, and O. Sallent, "Intercell interference management in OFDMA networks: A decentralized approach based on reinforcement learning," *IEEE Trans. Syst., Man, Cybern., C (Appl. Rev.)*, vol. 41, no. 6, pp. 968–976, Nov. 2011.

[173] S. Imtiaz, H. Ghauch, M. M. U. Rahman, G. Koudouridis, and J. Gross, "Learning-based resource allocation scheme for TDD-based 5G CRAN system," in *Proc. 19th ACM Int. Conf. Modeling, Anal. Simulation Wireless Mobile Syst.*, New York, NY, USA, Nov. 2016, pp. 176–185.

[174] M. Khan, M. Alam, Y. Moullec, and E. Yaacoub, "Throughput-aware cooperative reinforcement learning for adaptive resource allocation in Device-to-Device communication," *Future Internet*, vol. 9, no. 4, p. 72, Nov. 2017.

[175] W. van Norden, J. de Jong, F. Bolderheij, and L. Rothkrantz, "Intelligent task scheduling in sensor networks," in *Proc. 7th Int. Conf. Inf. Fusion*, vol. 2, 2005, pp. 1351–1358.

[176] Q. Liao, "Dynamic uplink/downlink resource management in flexible duplex-enabled wireless networks," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, May 2017, pp. 625–631.

[177] H. Malik, M. M. Alam, H. Pervaiz, Y. L. Moullec, A. Al-Dulaimi, S. Parand, and L. Reggiani, "Radio resource management in NB-IoT systems: Empowered by interference prediction and flexible duplexing," *IEEE Netw.*, vol. 34, no. 1, pp. 144–151, Jan. 2020.

[178] P. Pirinen, "Challenges and possibilities for flexible duplexing in 5G networks," in *Proc. IEEE 20th Int. Workshop Comput. Aided Model. Design Commun. Links Netw. (CAMAD)*, Sep. 2015, pp. 6–10.

[179] Z. Zhao, W. Chen, X. Wu, P. C. Y. Chen, and J. Liu, "LSTM network: A deep learning approach for short-term traffic forecast," *IET Intell. Transp. Syst.*, vol. 11, no. 2, pp. 68–75, Mar. 2017.

[180] S. Chinchali, P. Hu, T. Chu, M. Sharma, M. Bansal, R. Misra, M. Pavone, and S. Katti, "Cellular Network Traffic Scheduling With Deep Reinforcement Learning," in *Proc. AAAI*, 2018, pp. 1–9.

[181] C. Zhang and P. Patras, "Long-term mobile traffic forecasting using deep spatio-temporal neural networks," in *Proc. 18th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, New York, NY, USA, Jun. 2018, pp. 231–240, doi: 10.1145/3209582.3209606.

[182] A. Azzouni and G. Pujolle, "NeuTM: A neural network-based framework for traffic matrix prediction in SDN," in *Proc. IEEE/IFIP Netw. Oper. Manage. Symp. (NOMS)*, Apr. 2018, pp. 1–5.

[183] M. Chincoli, A. A. Syed, D. C. Mocanu, and A. Liotta, "Predictive power control in wireless sensor networks," in *Proc. IEEE 1st Int. Conf. Internet-Things Design Implement.*, Apr. 2016, pp. 309–312.

[184] E. Ghadimi, F. Davide Calabrese, G. Peters, and P. Soldati, "A reinforcement learning approach to power control and rate adaptation in cellular networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2017, pp. 1–7.

[185] C. Pandana and K. J. R. Liu, "Near-optimal reinforcement learning framework for energy-aware sensor communications," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 4, pp. 788–797, Apr. 2005.

[186] Y. Li, H. Hu, Y. Wen, and J. Zhang, "Learning-based power prediction for data centre operations via deep neural networks," in *Proc. 5th Int. Workshop Energy Efficient Data Centres (E2DC)*, New York, NY, USA, 2016, p. 6.

[187] H. Feng and Y. Shu, "Study on network traffic prediction techniques," in *Proc. Int. Conf. Wireless Commun., Netw. Mobile Comput.*, vol. 2, Sep. 2005, pp. 1041–1044.

[188] U. Paul, A. P. Subramanian, M. M. Buddhikot, and S. R. Das, "Understanding traffic dynamics in cellular data networks," in *Proc. IEEE INFOCOM*, Apr. 2011, pp. 882–890.

[189] F. Xu, Y. Li, H. Wang, P. Zhang, and D. Jin, "Understanding mobile traffic patterns of large scale cellular towers in urban environment," *IEEE/ACM Trans. Netw.*, vol. 25, no. 2, pp. 1147–1161, Apr. 2017.

[190] C. Bontu and E. Illidge, "DRX mechanism for power saving in LTE," *IEEE Commun. Mag.*, vol. 47, no. 6, pp. 48–55, Jun. 2009.

[191] H. Holma, A. Toskala, and J. Reunanen, *LTE-Advanced Evolution in Releases 12–13*. London, U.K.: Wiley, 2015, pp. 29–43.

[192] A. K. Sultania, P. Zand, C. Blondia, and J. Famaey, "Energy modeling and evaluation of NB-IoT with PSM and eDRX," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2018, pp. 1–7.

[193] R. J. Perlman, "Network layer protocols with Byzantine robustness," Ph.D. dissertation, Dept. Elect. Eng. Comput. Sci., Massachusetts Inst. Technol., Cambridge, MA, USA, 1988.

[194] A. Adas, "Traffic models in broadband networks," *IEEE Commun. Mag.*, vol. 35, no. 7, pp. 82–89, Jul. 1997.

[195] A. Feldmann, A. Greenberg, C. Lund, N. Reingold, and J. Rexford, "NetScope: Traffic engineering for IP networks," *IEEE Netw.*, vol. 14, no. 2, pp. 11–19, Mar. 2000.

[196] R. Li, Z. Zhao, X. Zhou, J. Palicot, and H. Zhang, "The prediction analysis of cellular radio access network traffic: From entropy theory to networking practice," *IEEE Commun. Mag.*, vol. 52, no. 6, pp. 234–240, Jun. 2014.

[197] W. Juan, S. Min, Z. Yan, W. Xijun, and L. Yuzhou, "Traffic characteristics based dynamic radio resource management in heterogeneous wireless networks," *China Commun.*, vol. 11, no. 1, pp. 1–11, Jan. 2014.

[198] F. Morales, L. Gifre, F. Paolucci, M. Ruiz, F. Cugini, P. Castoldi, and L. Velasco, "Dynamic core VNT adaptability based on predictive metroflow traffic models," *IEEE/OSA J. Opt. Commun. Netw.*, vol. 9, no. 12, pp. 1202–1211, Dec. 2017.

[199] F. Morales, M. Ruiz, L. Gifre, L. M. Contreras, V. López, and L. Velasco, "Virtual network topology adaptability based on data analytics for traffic prediction," *J. Opt. Commun. Netw.*, vol. 9, no. 1, pp. A35–A45, Jan. 2017.

[200] E. D. Osborne and A. Simha, *Traffic Engineering With MPLS*. Indianapolis, IN, USA, Cisco Press, 2002.

[201] P. Kumar, Y. Yuan, C. Yu, N. Foster, R. Kleinberg, and R. Soulé, "Kulfi: Robust traffic engineering using semi-oblivious routing," 2016, *arXiv:1603.01203*. [Online]. Available: http://arxiv.org/abs/1603.01203

[202] M. Suchara, D. Xu, R. Doverspike, D. Johnson, and J. Rexford, "Network architecture for joint failure recovery and traffic engineering," in *Proc. ACM SIGMETRICS Joint Int. Conf. Meas. Modeling Comput. Syst. (SIGMETRICS)*, New York, NY, USA, 2011, pp. 97–108, doi: 10.1145/1993744.1993756.

[203] F. Tang, B. Mao, Z. M. Fadlullah, N. Kato, O. Akashi, T. Inoue, and K. Mizutani, "On removing routing protocol from future wireless networks: A real-time deep learning approach for intelligent traffic control," *IEEE Wireless Commun.*, vol. 25, no. 1, pp. 154–160, Feb. 2018.

[204] K. Adachi, M. Li, P. H. Tan, Y. Zhou, and S. Sun, "Q-learning based intelligent traffic steering in heterogeneous network," in *Proc. IEEE 83rd Veh. Technol. Conf. (VTC Spring)*, May 2016, pp. 1–5.

[205] Y. Azar, E. Cohen, A. Fiat, H. Kaplan, and H. Räcke, "Optimal oblivious routing in polynomial time," *J. Comput. Syst. Sci.*, vol. 69, no. 3, pp. 383–394, 2004. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0022000004000625

[206] M. Chiesa, G. Rétvári, and M. Schapira, "Lying your way to better traffic engineering," in *Proc. 12th Int. Conf. Emerg. Netw. Exp. Technol.*, New York, NY, USA, Dec. 2016, pp. 391–398, doi: 10.1145/2999572.2999585.

[207] B. Fortz and M. Thorup, "Increasing Internet capacity using local search," *Comput. Optim. Appl.*, vol. 29, no. 1, pp. 13–48, Oct. 2004, doi: 10.1023/B:COAP.0000039487.35027.02.

[208] A. Valadarsky, M. Schapira, D. Shahaf, and A. Tamar, "A machine learning approach to routing," 2017, *arXiv:1708.03074*. [Online]. Available: http://arxiv.org/abs/1708.03074

[209] N. Feamster, J. Borkenhagen, and J. Rexford, "Guidelines for inter-domain traffic engineering," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 33, no. 5, pp. 19–30, Oct. 2003, doi: 10.1145/963985.963988.

[210] N. Kato, Z. M. Fadlullah, B. Mao, F. Tang, O. Akashi, T. Inoue, and K. Mizutani, "The deep learning vision for heterogeneous network traffic control: Proposal, challenges, and future perspective," *IEEE Wireless Commun.*, vol. 24, no. 3, pp. 146–153, Jun. 2017.

[211] J. T. Moy, *OSPF: Anatomy of an Internet Routing Protocol*. Reading, MA, USA: Addison-Wesley, 1998.

[212] B. Fortz and M. Thorup, "Internet traffic engineering by optimizing OSPF weights," in *Proc. IEEE Conf. Comput. Commun. 19th Annu. Joint Conf. IEEE Comput. Commun. Societies (INFOCOM)*, vol. 2, Mar. 2000, pp. 519–528.

[213] I. Kassabalidis, M. A. El-Sharkawi, R. J. Marks, P. Arabshahi, and A. A. Gray, "Swarm intelligence for routing in communication networks," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, vol. 6, Nov. 2001, pp. 3613–3617.

[214] J. Kennedy, "Swarm intelligence," in *Handbook of Nature-Inspired and Innovative Computing*. Boston, MA, USA: Springer, 2006, pp. 187–219.

[215] E. Bonabeau, M. Dorigo, D. R. D. F. Marco, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*. no. 1. New York, NY, USA: Oxford Univ. Press, 1999.

[216] G. Di Caro, F. Ducatelle, and L. M. Gambardella, "Swarm intelligence for routing in mobile ad hoc networks," in *Proc. IEEE Swarm Intell. Symp. (SIS)*, Jun. 2005, pp. 76–83.

[217] C.-C. Shen and C. Jaikaeo, "Ad hoc multicast routing algorithm with swarm intelligence," *Mobile Netw. Appl.*, vol. 10, nos. 1–2, pp. 47–59, Feb. 2005. [Online]. Available: http://dl.acm.org/citation.cfm?id=1046430.1046435

[218] M. Salani, C. Rottondi, and M. Tornatore, "Routing and spectrum assignment integrating Machine-Learning-Based QoT estimation in elastic optical networks," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Apr. 2019, pp. 1738–1746.

[219] D. Subramanian, P. Druschel, and J. Chen, "Ants and reinforcement learning: A case study in routing in dynamic networks," in *Proc. IJCAI*, vol. 2, 1997, pp. 832–839.

[220] R. Schoonderwoerd, O. E. Holland, J. L. Bruten, and L. J. M. Rothkrantz, "Ant-based load balancing in telecommunications networks," *Adapt. Behav.*, vol. 5, no. 2, pp. 169–207, Jan. 1997, doi: 10.1177/105971239700500203.

[221] L. Pelusi, A. Passarella, and M. Conti, "Opportunistic networking: Data forwarding in disconnected mobile ad hoc networks," *IEEE Commun. Mag.*, vol. 44, no. 11, pp. 134–141, Nov. 2006.

[222] S. Biswas and R. Morris, "ExOR: Opportunistic multi-hop routing for wireless networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 35, no. 4, pp. 133–144, Oct. 2005, doi: 10.1145/1090191.1080108.

[223] S. Jain, K. Fall, and R. Patra, "Routing in a delay tolerant network," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 34, no. 4, pp. 145–158, Aug. 2004, doi: 10.1145/1030194.1015484.

[224] D. K. Sharma, S. K. Dhurandher, I. Woungang, R. K. Srivastava, A. Mohananey, and J. J. P. C. Rodrigues, "A machine learning-based protocol for efficient routing in opportunistic networks," *IEEE Syst. J.*, vol. 12, no. 3, pp. 2207–2213, Sep. 2018.

[225] T.-K. Huang, C.-K. Lee, and L.-J. Chen, "PRoPHET+: An adaptive PRoPHET-based routing protocol for opportunistic network," in *Proc. 24th IEEE Int. Conf. Adv. Inf. Netw. Appl.*, Apr. 2010, pp. 112–119.

[226] L. Zhi-yuan, W. Ru-chuan, and B. Jun-lei, "A multipath routing algorithm based on traffic prediction in wireless mesh networks," in *Proc. 5th Int. Conf. Natural Comput.*, vol. 6, Aug. 2009, pp. 115–119.

[227] J. A. Boyan and M. L. Littman, "Packet routing in dynamically changing networks: A reinforcement learning approach," in *Proc. Adv. Neural Inf. Process. Syst.*, 1994, pp. 671–678.

[228] L. Peshkin and V. Savova, "Reinforcement learning for adaptive routing," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, vol. 2, May 2002, pp. 1825–1830.

[229] S. P. Choi and D.-Y. Yeung, "Predictive Q-routing: A memory-based reinforcement learning approach to adaptive traffic control," in *Proc. Adv. Neural Inf. Process. Syst.*, 1996, pp. 945–951.

[230] L. C. Baird III and A. W. Moore, "Gradient descent for general reinforcement learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 1999, pp. 968–974.

[231] Y. Shilova, M. Kavalerov, and I. Bezukladnikov, "Full echo Q-routing with adaptive learning rates: A reinforcement learning approach to network routing," in *Proc. IEEE NW Russia Young Researchers Electr. Electron. Eng. Conf. (EIConRusNW)*, Feb. 2016, pp. 341–344.

[232] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, nos. 3–4, pp. 279–292, May 1992, doi: 10.1007/BF00992698.

[233] M. Kavalerov, Y. Likhacheva, and Y. Shilova, "A reinforcement learning approach to network routing based on adaptive learning rates and route memory," in *Proc. SoutheastCon*, Mar. 2017, pp. 1–6.

[234] J. Dowling, E. Curran, R. Cunningham, and V. Cahill, "Using feedback in collaborative reinforcement learning to adaptively optimize MANET routing," *IEEE Trans. Syst., Man, Cybern., A, Syst. Humans*, vol. 35, no. 3, pp. 360–372, May 2005.

[235] F. Tang, B. Mao, Z. M. Fadlullah, and N. Kato, "On a novel deep-learning-based intelligent partially overlapping channel assignment in SDN-IoT," *IEEE Commun. Mag.*, vol. 56, no. 9, pp. 80–86, Sep. 2018.

[236] G. Choudhury, D. Lynch, G. Thakur, and S. Tse, "Two use cases of machine learning for SDN-enabled IP/Optical networks: Traffic matrix prediction and optical path performance prediction [Invited]," *J. Opt. Commun. Netw.*, vol. 10, no. 10, pp. D52–D62, Oct. 2018.

[237] B. Mao, Z. M. Fadlullah, F. Tang, N. Kato, O. Akashi, T. Inoue, and K. Mizutani, "Routing or computing? The paradigm shift towards intelligent computer network packet transmission based on deep learning," *IEEE Trans. Comput.*, vol. 66, no. 11, pp. 1946–1960, Nov. 2017.

[238] S. T. V. Pasca, S. S. P. Kodali, and K. Kataoka, "AMPS: Application aware multipath flow routing using machine learning in SDN," in *Proc. 23rd Nat. Conf. Commun. (NCC)*, Mar. 2017, pp. 1–6.

[239] P. Wang, S.-C. Lin, and M. Luo, "A framework for QoS-aware traffic classification using semi-supervised machine learning in SDNs," in *Proc. IEEE Int. Conf. Services Comput. (SCC)*, Jun. 2016, pp. 760–765.

[240] M. Zorzi, A. Zanella, A. Testolin, M. De Filippo De Grazia, and M. Zorzi, "Cognition-based networks: A new perspective on network optimization using learning and distributed intelligence," *IEEE Access*, vol. 3, pp. 1512–1530, 2015.

[241] A. Mestres, A. Rodriguez-Natal, J. Carner, P. Barlet-Ros, E. Alarcón, M. Solé, V. Muntés-Mulero, D. Meyer, S. Barkai, M. J. Hibbett, G. Estrada, K. Ma'ruf, F. Coras, V. Ermagan, H. Latapie, C. Cassar, J. Evans, F. Maino, J. Walrand, and A. Cabellos, "Knowledge-defined networking," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 47, no. 3, pp. 2–10, Sep. 2017, doi: 10.1145/3138808.3138810.

[242] Z. M. Fadlullah, F. Tang, B. Mao, J. Liu, N. Kato, O. Akashi, T. Inoue, and K. Mizutani, "On intelligent traffic control for large-scale heterogeneous networks: A value matrix-based deep learning approach," *IEEE Commun. Lett.*, vol. 22, no. 12, pp. 2479–2482, Dec. 2018.

[243] J. Rexford and C. Dovrolis, "Future Internet architecture: clean-slate versus evolutionary research," *Commun. ACM*, vol. 53, no. 9, pp. 36–40, Sep. 2010, doi: 10.1145/1810891.1810906.

[244] D. Kreutz, F. M. V. Ramos, P. Esteves Verissimo, C. Esteve Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proc. IEEE*, vol. 103, no. 1, pp. 14–76, Jan. 2015.

[245] B. A. A. Nunes, M. Mendonca, X.-N. Nguyen, K. Obraczka, and T. Turletti, "A survey of software-defined networking: Past, present, and future of programmable networks," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 3, pp. 1617–1634, 3rd Quart., 2014.

[246] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "OpenFlow: Enabling innovation in campus networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, Mar. 2008, doi: 10.1145/1355734.1355746.

[247] S. Gangadhar and J. P. G. Sterbenz, "Machine learning aided traffic tolerance to improve resilience for software defined networks," in *Proc. 9th Int. Workshop Resilient Netw. Design Modeling (RNDM)*, Sep. 2017, pp. 1–7.

[248] J. R. Quinlan, "Induction of decision trees," *Mach. Learn.*, vol. 1, no. 1, pp. 81–106, 1986.

[249] N. Friedman, D. Geiger, and M. Goldszmidt, "Bayesian network classifiers," *Mach. Learn.*, vol. 29, no. 2, pp. 131–163, Nov. 1997.

[250] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995.

[251] N. M. M. K. Chowdhury and R. Boutaba, "A survey of network virtualization," *Comput. Netw.*, vol. 54, no. 5, pp. 862–876, Apr. 2010. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128609003387

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

[253] K. Han, S. Li, S. Tang, H. Huang, S. Zhao, G. Fu, and Z. Zhu, "Application-driven End-to-End slicing: When wireless network virtualization orchestrates with NFV-based mobile edge computing," *IEEE Access*, vol. 6, pp. 26567–26577, 2018.

[254] M. Liyanage, I. Ahmed, J. Okwuibe, M. Ylianttila, H. Kabir, J. L. Santos, R. Kantola, O. L. Perez, M. U. Itzazelaia, and E. M. De Oca, "Enhancing security of software defined mobile networks," *IEEE Access*, vol. 5, pp. 9422–9438, 2017.

[255] H. Kim and N. Feamster, "Improving network management with software defined networking," *IEEE Commun. Mag.*, vol. 51, no. 2, pp. 114–119, Feb. 2013.

[256] I. Ahmad, S. Namal, M. Ylianttila, and A. Gurtov, "Towards software defined cognitive networking," in *Proc. 7th Int. Conf. New Technol., Mobility Secur. (NTMS)*, Jul. 2015, pp. 1–5.

[257] Z. A. Qazi, J. Lee, T. Jin, G. Bellala, M. Arndt, and G. Noubir, "Application-awareness in SDN," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 43, no. 4, pp. 487–488, Sep. 2013, doi: 10.1145/2534169.2491700.

[258] J. Rischke, P. Sossalla, H. Salah, F. H. P. Fitzek, and M. Reisslein, "QR-SDN: Towards reinforcement learning states, actions, and rewards for direct flow routing in software-defined networks," *IEEE Access*, vol. 8, pp. 174773–174791, 2020.

[259] D. D. Clark, C. Partridge, J. C. Ramming, and J. T. Wroclawski, "A knowledge plane for the Internet," in *Proc. Conf. Appl., Technol., Archit., Protocols Comput. Commun. (SIGCOMM)*, New York, NY, USA, 2003, pp. 3–10, doi: 10.1145/863955.863957.

[260] R. W. Thomas, D. H. Friend, L. A. DaSilva, and A. B. MacKenzie, *Cognitive Networks*. Dordrecht, The Netherlands: Springer, 2007, pp. 17–41.

[261] I. Ahmad, S. Namal, M. Ylianttila, and A. Gurtov, "Security in software defined networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2317–2346, 4th Quart., 2015.

[262] T. Abar, A. Ben Letaifa, and S. Elasmi, "Enhancing QoE based on machine learning and DASH in SDN networks," in *Proc. 32nd Int. Conf. Adv. Inf. Netw. Appl. Workshops (WAINA)*, May 2018, pp. 258–263.

[263] L. Fawcett, S. Scott-Hayward, M. Broadbent, A. Wright, and N. Race, "Tennison: A distributed SDN framework for scalable network security," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 12, pp. 2805–2818, Dec. 2018.

[264] V. Varadharajan, K. Karmakar, U. Tupakula, and M. Hitchens, "A policy-based security architecture for software-defined networks," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 4, pp. 897–912, Apr. 2019.

[265] S. Nanda, F. Zafari, C. DeCusatis, E. Wedaa, and B. Yang, "Predicting network attack patterns in SDN using machine learning approach," in *Proc. IEEE Conf. Netw. Function Virtualization Softw. Defined Netw. (NFV-SDN)*, Nov. 2016, pp. 167–172.

[266] A. Ahmad, E. Harjula, M. Ylianttila, and I. Ahmad, "Evaluation of machine learning techniques for security in SDN," in *Proc. IEEE Globecom Workshops*, Oct. 2020, pp. 1–6.

[267] L. Yang and H. Zhao, "DDoS attack identification and defense using SDN based on machine learning method," in *Proc. 15th Int. Symp. Pervas. Syst., Algorithms Netw. (I-SPAN)*, Oct. 2018, pp. 174–178.

[268] M. Latah and L. Toker, "Towards an efficient anomaly-based intrusion detection for software-defined networks," *IET Netw.*, vol. 7, no. 6, pp. 453–459, Nov. 2018.

[269] Z. Zhou, T. Zhang, and A. Kwatra, "NFV closed-loop automation experiments using deep reinforcement learning," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Apr. 2019, pp. 696–701.

[270] Z. Zhou and T. Zhang, "Applying machine learning to service assurance in network function virtualization environment," in *Proc. 1st Int. Conf. Artif. Intell. Industries (AI4I)*, Sep. 2018, pp. 112–115.

[271] R. Shi, J. Zhang, W. Chu, Q. Bao, X. Jin, C. Gong, Q. Zhu, C. Yu, and S. Rosenberg, "MDP and machine learning-based cost-optimization of dynamic resource allocation for network function virtualization," in *Proc. IEEE Int. Conf. Services Comput.*, Jun. 2015, pp. 65–73.

[272] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Hoboken, NJ, USA: Wiley, 2014.

[273] D. J. White, "Markov decision processes," in *Encyclopedia of Statistical Sciences*, vol. 7. 2004.

[274] L. He, C. Xu, and Y. Luo, "VTC: Machine learning based traffic classification as a virtual network function," in *Proc. ACM Int. Workshop Secur. Softw. Defined Netw. Netw. Function Virtualization (SDN-NFV Secur.)*, New York, NY, USA: ACM, 2016, pp. 53–56, doi: 10.1145/2876019.2876029.

[275] T. Wen, H. Yu, G. Sun, and L. Liu, "Network function consolidation in service function chaining orchestration," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2016, pp. 1–6.

[276] D. Bhamare, R. Jain, M. Samaka, and A. Erbad, "A survey on service function chaining," *J. Netw. Comput. Appl.*, vol. 75, pp. 138–155, Nov. 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804516301989

[277] G. Sun, G. Zhu, D. Liao, H. Yu, X. Du, and M. Guizani, "Cost-efficient service function chain orchestration for low-latency applications in NFV networks," *IEEE Syst. J.*, vol. 13, no. 4, pp. 3877–3888, Dec. 2019.

[278] I. Sutskever, G. E. Hinton, and G. W. Taylor, "The recurrent temporal restricted Boltzmann machine," in *Proc. Adv. Neural Inf. Process. Syst.*, D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, Eds. Red Hook, NY, USA: Curran Associates, Inc., 2009, pp. 1601–1608. [Online]. Available: http://papers.nips.cc/paper/3567-the-recurrent-temporal-restricted-boltzmann-machine.pdf

[279] M. F. Bari, R. Boutaba, R. Esteves, L. Z. Granville, M. Podlesny, M. G. Rabbani, Q. Zhang, and M. F. Zhani, "Data center network virtualization: A survey," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 2, pp. 909–928, 2nd Quart., 2013.

[280] M. R. Rahman and R. Boutaba, "SVNE: Survivable virtual network embedding algorithms for network virtualization," *IEEE Trans. Netw. Service Manage.*, vol. 10, no. 2, pp. 105–118, Jun. 2013.

[281] A. Blenk, A. Basta, J. Zerwas, M. Reisslein, and W. Kellerer, "Control plane latency with SDN network hypervisors: The cost of virtualization," *IEEE Trans. Netw. Service Manage.*, vol. 13, no. 3, pp. 366–380, Sep. 2016.

[282] J. Costa-Requena, M. Liyanage, M. Ylianttila, E. M. de Oca, J. L. Santos, V. F. Guasch, K. Ahokas, G. Premsankar, S. Luukkainen, O. L. Perez, M. U. Itzazelaia, and I. Ahmad, "SDN and NFV integration in generalized mobile network architecture," in *Proc. Eur. Conf. Netw. Commun. (EuCNC)*, Jun. 2015, pp. 154–158.

[283] J. Matias, J. Garay, N. Toledo, J. Unzilla, and E. Jacob, "Toward an SDN-enabled NFV architecture," *IEEE Commun. Mag.*, vol. 53, no. 4, pp. 187–193, Apr. 2015.

[284] A. Basta, W. Kellerer, M. Hoffmann, H. J. Morper, and K. Hoffmann, "Applying NFV and SDN to LTE mobile core gateways, the functions placement problem," in *Proc. 4th Workshop Things Cellular: Oper., Appl., Challenges (AllThingsCellular)*, New York, NY, USA, 2014, pp. 33–38, doi: 10.1145/2627585.2627592.

[285] W. Kellerer, P. Kalmbach, A. Blenk, A. Basta, M. Reisslein, and S. Schmid, "Adaptable and data-driven softwarized networks: Review, opportunities, and challenges," *Proc. IEEE*, vol. 107, no. 4, pp. 711–731, Apr. 2019.

[286] P. Kalmbach, J. Zerwas, P. Babarczi, A. Blenk, W. Kellerer, and S. Schmid, "Empowering self-driving networks," in *Proc. Afternoon Workshop Self-Driving Netw. (SelfDN)*, 2018, pp. 8–14, doi: 10.1145/3229584.3229587.

[287] J. Zerwas, P. Kalmbach, L. Henkel, G. Rétvári, W. Kellerer, A. Blenk, and S. Schmid, "NetBOA: self-driving network benchmarking," in *Proc. Workshop Netw. Meets AI ML (NetAI)*, 2019, pp. 8–14, doi: 10.1145/3341216.3342207.

[288] Q.-V. Pham, F. Fang, V. N. Ha, M. J. Piran, M. Le, L. B. Le, W.-J. Hwang, and Z. Ding, "A survey of multi-access edge computing in 5G and beyond: Fundamentals, technology integration, and State-of-the-Art," *IEEE Access*, vol. 8, pp. 116974–117017, 2020.

[289] P. G. Lopez, "Edge-centric computing: Vision and challenges," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 45, no. 5, pp. 37–42, 2015.

[290] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, Oct. 2016.

[291] A. Rasheed, P. H. J. Chong, I. W. H. Ho, X. J. Li, and W. Liu, "An overview of mobile edge computing: Architecture, technology and direction," Korean Soc. Internet Inf., Seoul, South Korea, Tech. Rep. 4849, 2019.

[292] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the Internet of Things," in *Proc. 1st Ed. MCC Workshop Mobile Cloud Comput. (MCC)*, New York, NY, USA, 2012, pp. 13–16, doi: 10.1145/2342509.2342513.

[293] F. Bonomi, R. Milito, P. Natarajan, and J. Zhu, "Fog computing: A platform for Internet of Things and analytics," in *Big Data Internet Things: A Roadmap for Smart Environments*. Cham, Switzerland: Springer, 2014, pp. 169–186.

[294] P. Porambage, J. Okwuibe, M. Liyanage, M. Ylianttila, and T. Taleb, "Survey on multi-access edge computing for Internet of Things realization," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 2961–2991, 4th Quart., 2018.

[295] Y. C. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, "Mobile edge computing—A key technology towards 5G," Eur. Telecommun. Standards Inst., Sophia-Antipolis, France, Tech. Rep. 11, 2015, pp. 1–16.

[296] *Nokia–Multi-access Edge Computing (MEC)*. Accessed: Nov. 2020. [Online]. Available: https://networks.nokia.com/solutions/multi-access-edge-computing

[297] *Getting Close to You: MEC@CloudEdge*. Accessed: Nov. 2020. [Online]. Available: https://www-file.huawei.com/-/media/CORPORATE/PDF/publications/communicate/82/10-solutions-getting-close-to-you-meccloudedge.pdf

[298] *OpenStack*. Accessed: Nov. 2020. [Online]. Available: https://www.openstack.org/

[299] *Saguna–Edge Cloud Computing*. Accessed: Nov. 2020. [Online]. Available: https://www.saguna.net/

[300] M. Chen, Y. Hao, Y. Li, C.-F. Lai, and D. Wu, "On the computation offloading at ad hoc cloudlet: Architecture and service modes," *IEEE Commun. Mag.*, vol. 53, no. 6, pp. 18–24, Jun. 2015.

[301] P. Patel, M. Intizar Ali, and A. Sheth, "On using the intelligent edge for IoT analytics," *IEEE Intell. Syst.*, vol. 32, no. 5, pp. 64–69, Sep. 2017.

[302] R. G. Aryal and J. Altmann, "Dynamic application deployment in federations of clouds and edge resources using a multiobjective optimization AI algorithm," in *Proc. 3rd Int. Conf. Fog Mobile Edge Comput. (FMEC)*, Apr. 2018, pp. 147–154.

[303] S. B. Calo, M. Touna, D. C. Verma, and A. Cullen, "Edge computing architecture for applying AI to IoT," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2017, pp. 3012–3016.

[304] J. Xu, L. Chen, and S. Ren, "Online learning for offloading and autoscaling in energy harvesting mobile edge computing," *IEEE Trans. Cognit. Commun. Netw.*, vol. 3, no. 3, pp. 361–373, Sep. 2017.

[305] J. Wen, C. Ren, and A. K. Sangaiah, "Energy-efficient Device-to-Device edge computing network: An approach offloading both traffic and computation," *IEEE Commun. Mag.*, vol. 56, no. 9, pp. 96–102, Sep. 2018.

[306] S. Wang, M. Chen, X. Liu, C. Yin, S. Cui, and H. V. Poor, "A machine learning approach for task and resource allocation in mobile edge computing based networks," *IEEE Internet Things J.*, pp. 1–1, 2020.

[307] H. Wu, Z. Zhang, C. Guan, K. Wolter, and M. Xu, "Collaborate edge and cloud computing with distributed deep learning for smart city Internet of things," *IEEE Internet Things J.*, vol. 7, no. 9, pp. 8099–8110, Sep. 2020.

[308] M. Nduwayezu, Q.-V. Pham, and W.-J. Hwang, "Online computation offloading in NOMA-based multi-access edge computing: A deep reinforcement learning approach," *IEEE Access*, vol. 8, pp. 99098–99109, 2020.

[309] W. Jiang, G. Feng, S. Qin, and Y. Liu, "Multi-agent reinforcement learning based cooperative content caching for mobile edge networks," *IEEE Access*, vol. 7, pp. 61856–61867, 2019.

[310] Z. Cao, P. Zhou, R. Li, S. Huang, and D. Wu, "Multiagent deep reinforcement learning for joint multichannel access and task offloading of mobile-edge computing in industry 4.0," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6201–6213, Jul. 2020.

[311] I. Khan, X. Tao, G. M. S. Rahman, W. U. Rehman, and T. Salam, "Advanced energy-efficient computation offloading using deep reinforcement learning in MTC edge computing," *IEEE Access*, vol. 8, pp. 82867–82875, 2020.

[312] R. Dong, C. She, W. Hardjawana, Y. Li, and B. Vucetic, "Deep learning for hybrid 5G services in mobile edge computing systems: Learn from a digital twin," *IEEE Trans. Wireless Commun.*, vol. 18, no. 10, pp. 4692–4707, Oct. 2019.

[313] Q. Cui, Z. Gong, W. Ni, Y. Hou, X. Chen, X. Tao, and P. Zhang, "Stochastic online learning for mobile edge computing: Learning from changes," *IEEE Commun. Mag.*, vol. 57, no. 3, pp. 63–69, Mar. 2019.

[314] T. Subramanya, D. Harutyunyan, and R. Riggio, "Machine learning-driven service function chain placement and scaling in MEC-enabled 5G networks," *Comput. Netw.*, vol. 166, Jan. 2020, Art. no. 106980. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128619310254

[315] B. Cao, L. Zhang, Y. Li, D. Feng, and W. Cao, "Intelligent offloading in multi-access edge computing: A State-of-the-Art review and framework," *IEEE Commun. Mag.*, vol. 57, no. 3, pp. 56–62, Mar. 2019.

[316] Y.-L. Lee, P.-K. Tsung, and M. Wu, "Techology trend of edge AI," in *Proc. Int. Symp. VLSI Design, Autom. Test (VLSI-DAT)*, Apr. 2018, pp. 1–2.

[317] M. Chen, Z. Yang, W. Saad, C. Yin, H. Vincent Poor, and S. Cui, "A joint learning and communications framework for federated learning over wireless networks," 2019, *arXiv:1909.07972*. [Online]. Available: http://arxiv.org/abs/1909.07972

[318] N. Shlezinger, M. Chen, Y. C. Eldar, H. V. Poor, and S. Cui, "Federated learning with quantization constraints," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 8851–8855.

[319] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. AISTATS*, 2017, pp. 1273–1282.

[320] M. Chen, H. Vincent Poor, W. Saad, and S. Cui, "Convergence time optimization for federated learning over wireless networks," 2020, *arXiv:2001.07845*. [Online]. Available: http://arxiv.org/abs/2001.07845

[321] M. Chen, H. V. Poor, W. Saad, and S. Cui, "Convergence time minimization of federated learning over wireless networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2020, pp. 1–6.

[322] J. Konečný, H. B. McMahan, D. Ramage, and P. Richtárik, "Federated optimization: Distributed machine learning for on-device intelligence," 2016, *arXiv:1610.02527*. [Online]. Available: https://arxiv.org/abs/1610.02527

[323] B. Yang, X. Cao, X. Li, Q. Zhang, and L. Qian, "Mobile-Edge-Computing-Based hierarchical machine learning tasks distribution for IIoT," *IEEE Internet Things J.*, vol. 7, no. 3, pp. 2169–2180, Mar. 2020.

[324] J. Partala, L. Lovén, E. Peltonen, P. Porambage, M. Ylianttila, and T. Seppänen, "EdgeAI: A vision for privacy-preserving machine learning on the edge," in *Proc. 10th Nordic Workshop Syst. Netw. Optim. Wireless (SNOW)*, Apr. 2019.

[325] T. Nishio and R. Yonetani, "Client selection for federated learning with heterogeneous resources in mobile edge," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2019, pp. 1–7.

[326] A. Celesti, D. Mulfari, M. Fazio, M. Villari, and A. Puliafito, "Exploring container virtualization in IoT clouds," in *Proc. IEEE Int. Conf. Smart Comput. (SMARTCOMP)*, May 2016, pp. 1–6.

[327] *Kubernetes–Production-Grade Container Orchestration*. Accessed: Nov. 2020. [Online]. Available: https://kubernetes.io

[328] I. Ahmad, S. Shahabuddin, T. Kumar, J. Okwuibe, A. Gurtov, and M. Ylianttila, "Security for 5G and beyond," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3682–3722, 4th Quart., 2019.

[329] C. E. Landwehr, "Cybersecurity and artificial intelligence: From fixing the plumbing to smart water," *IEEE Secur. Privacy Mag.*, vol. 6, no. 5, pp. 3–4, Sep. 2008.

[330] E. Bursztein, S. Bethard, C. Fabry, J. C. Mitchell, and D. Jurafsky, "How good are humans at solving CAPTCHAs? A large scale evaluation," in *Proc. IEEE Symp. Secur. Privacy*, May 2010, pp. 399–413.

[331] J. Frank, "Artificial intelligence and intrusion detection: Current and future directions," in *Proc. 17th Nat. Comput. Secur. Conf.*, vol. 10. Baltimore, MD, USA, 1994, pp. 1–12.

[332] B. Morel, "Artificial intelligence and the future of cybersecurity," in *Proc. 4th ACM Workshop Secur. Artif. Intell. (AISec)*, New York, NY, USA, 2011, pp. 93–98, doi: 10.1145/2046684.2046699.

[333] P. Sharma, H. Liu, H. Wang, and S. Zhang, "Securing wireless communications of connected vehicles with artificial intelligence," in *Proc. IEEE Int. Symp. Technol. Homeland Secur. (HST)*, Apr. 2017, pp. 1–7.

[334] A. W. Moore and D. Zuev, "Internet traffic classification using Bayesian analysis techniques," *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 33, no. 1, pp. 50–60, Jun. 2005, doi: 10.1145/1071690.1064220.

[335] J. Erman, A. Mahanti, and M. Arlitt, "QRP05-4: Internet traffic identification using machine learning," in *Proc. IEEE Globecom*, Nov. 2006, pp. 1–6.

[336] L. Caviglione, M. Gaggero, J.-F. Lalande, W. Mazurczyk, and M. Urbanski, "Seeing the unseen: Revealing mobile malware hidden communications via energy consumption and artificial intelligence," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 4, pp. 799–810, Apr. 2016.

[337] M. Botha, R. von Solms, K. Perry, E. Loubser, and G. Yamoyany, "The utilization of artificial intelligence in a hybrid intrusion detection system," in *Proc. Annu. Res. Conf. South Afr. Inst. Comput. Scientists Inf. Technologists Enablement Through Technol.*, 2002, pp. 149–155.

[338] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. London, U.K.: Pearson, 2016.

[339] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. Mcmahan, S. Patel, D. Ramage, A. Segal, and K. Seth, "Practical secure aggregation for privacy-preserving machine learning," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, New York, NY, USA, Oct. 2017, pp. 1175–1191, doi: 10.1145/3133956.3133982.

[340] M. Esmalifalak, L. Liu, N. Nguyen, R. Zheng, and Z. Han, "Detecting stealthy false data injection using machine learning in smart grid," *IEEE Syst. J.*, vol. 11, no. 3, pp. 1644–1652, Sep. 2017.

[341] D. Grzonka, A. Jakóbik, J. Kołodziej, and S. Pllana, "Using a multi-agent system and artificial intelligence for monitoring and improving the cloud performance and security," *Future Gener. Comput. Syst.*, vol. 86, pp. 1106–1117, Sep. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X17310531

[342] TensorFlow. *Google*. Accessed: Nov. 2020. [Online]. Available: https://www.tensorflow.org/

[343] L.-D. Chou, C.-W. Tseng, M.-S. Lai, W.-Y. Chen, K.-C. Chen, C.-K. Yen, T.-F. Ou, W.-H. Tsai, and Y.-H. Chiu, "Classification of malicious traffic using TensorFlow machine learning," in *Proc. Int. Conf. Inf. Commun. Technol. Converg. (ICTC)*, Oct. 2018, pp. 186–190.

[344] H. Debar, M. Dacier, and A. Wespi, "A revised taxonomy for intrusion-detection systems," *Annales des Télécommun.*, vol. 55, no. 7, pp. 361–378, Jul. 2000, doi: 10.1007/BF02994844.

[345] C. Sinclair, L. Pierce, and S. Matzner, "An application of machine learning to network intrusion detection," in *Proc. 15th Annu. Comput. Secur. Appl. Conf. (ACSAC)*, Dec. 1999, pp. 371–377.

[346] P. García-Teodoro, J. Díaz-Verdejo, G. Maciá-Fernández, and E. Vázquez, "Anomaly-based network intrusion detection: Techniques, systems and challenges," *Comput. Secur.*, vol. 28, nos. 1–2, pp. 18–28, Feb. 2009. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167404808000692

[347] S. Pang, Y. Peng, T. Ban, D. Inoue, and A. Sarrafzadeh, "A federated network online network traffics analysis engine for cybersecurity," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2015, pp. 1–8.

[348] J. Suomalainen, A. Juhola, S. Shahabuddin, A. Mammela, and I. Ahmad, "Machine learning threatens 5G security," *IEEE Access*, vol. 8, pp. 190822–190842, 2020.

[349] M. Barreno, B. Nelson, A. D. Joseph, and J. D. Tygar, "The security of machine learning," *Mach. Learn.*, vol. 81, no. 2, pp. 121–148, Nov. 2010, doi: 10.1007/s10994-010-5188-5.

[350] M. Abadi and D. G. Andersen, "Learning to protect communications with adversarial neural cryptography," 2016, *arXiv:1610.06918*. [Online]. Available: http://arxiv.org/abs/1610.06918

[351] G. J. Mendis, J. Wei, and A. Madanayake, "Deep learning-based automated modulation classification for cognitive radio," in *Proc. IEEE Int. Conf. Commun. Syst. (ICCS)*, Dec. 2016, pp. 1–6.

[352] F. N. Khan, C. Lu, and A. P. T. Lau, "Joint modulation format/bit-rate classification and signal-to-noise ratio estimation in multipath fading channels using deep machine learning," *Electron. Lett.*, vol. 52, no. 14, pp. 1272–1274, Jul. 2016.

[353] R. Boisguene, S.-C. Tseng, C.-W. Huang, and P. Lin, "A survey on NB-IoT downlink scheduling: Issues and potential solutions," in *Proc. 13th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, Jun. 2017, pp. 547–551.

[354] P. Andres-Maldonado, P. Ameigeiras, J. Prados-Garzon, J. Navarro-Ortiz, and J. M. Lopez-Soler, "Narrowband IoT data transmission procedures for massive machine-type communications," *IEEE Netw.*, vol. 31, no. 6, pp. 8–15, Nov. 2017.

[355] D. L. Tennenhouse and D. J. Wetherall, "Towards an active network architecture," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 26, no. 2, pp. 5–17, Apr. 1996, doi: 10.1145/231699.231701.

[356] A. Greenberg, G. Hjalmtysson, D. A. Maltz, A. Myers, J. Rexford, G. Xie, H. Yan, J. Zhan, and H. Zhang, "A clean slate 4D approach to network control and management," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 35, no. 5, pp. 41–54, Oct. 2005, doi: 10.1145/1096536.1096541.

[357] M. Casado, M. J. Freedman, J. Pettit, J. Luo, N. McKeown, and S. Shenker, "Ethane: Taking control of the enterprise," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 4, pp. 1–12, Oct. 2007, doi: 10.1145/1282427.1282382.

[358] S. Das, G. Parulkar, and N. McKeown, "Rethinking IP core networks," *IEEE/OSA J. Opt. Commun. Netw.*, vol. 5, no. 12, pp. 1431–1442, Dec. 2013. [Online]. Available: http://jocn.osa.org/abstract.cfm?URI=jocn-5-12-1431

[359] C. Monsanto, J. Reich, N. Foster, J. Rexford, and D. Walker, "Composing software defined networks," in *Proc. NSDI*, vol. 13, 2013, pp. 1–13.

[360] Z. Jiang, A. Balu, C. Hegde, and S. Sarkar, "Collaborative deep learning in fixed topology networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5904–5914.

[361] V. Smith, C.-K. Chiang, M. Sanjabi, and A. S. Talwalkar, "Federated multi-task learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 4424–4434.

IJAZ AHMAD (Member, IEEE) received the M.Sc. and Ph.D. degrees in wireless communications from the University of Oulu, Finland. He is currently with VTT Technical Research Centre of Finland. He was a Visiting Scientist with Aalto University, Finland, in 2018. In 2019, he visited TU Vienna, Austria, to work with Prof. T. Sauter as a Visiting Scientist. His research interests include security in 5G and 6G, SDN, the IoT, and the application of machine learning in wireless networks. He has received several awards, including the Nokia Foundation, Tauno Tönning and Jorma Ollila grant awards, and two IEEE best paper awards.

SHARIAR SHAHABUDDIN received the M.Sc. degree (Hons.) from the University of Oulu, in 2012, for which he received the Best Master's Thesis Award, and the Ph.D. degree from the Centre for Wireless Communications, University of Oulu, Finland, in 2019. In 2015, he worked with the Professor Christoph Studer's group, Computer Systems Laboratory, Cornell University, USA. He works at Nokia, Finland, as a SoC Specialist. His Ph.D. research interest includes machine learning applications for wireless communications. He received several scholarships and grants, such as Nokia Foundation Scholarship, the University of Oulu Scholarship Foundation Grant, and Taunu Tonningen Foundation Grant.

HASSAN MALIK (Member, IEEE) received the B.E. degree from the National University of Sciences and Technology (NUST), Pakistan, in 2009, the M.Sc. degree from the University of Oulu, Finland, in 2012, and the Ph.D. degree from the 5G Innovation Centre (5GIC), University of Surrey, U.K., in 2017. He worked as a Research Assistant with the Centre for Wireless Communication (CWC), Oulu, Finland, from 2011 to 2013. From 2017 to 2020, he worked as a Researcher with the Thomas Johann Seebeck Department of Electronics, Tallinn University of Technology, Estonia. He is currently working as a Senior Lecturer with the Computer Science Department, Edge Hill University, U.K. His current research interests include wireless communication protocols, the IoT, URLLC, LPWAN technologies, blockchain-based IoT, and embedded systems.

ERKKI HARJULA (Member, IEEE) received the D.Sc. degree in communications engineering from the University of Oulu, Finland, in 2016, where he is currently an Assistant Professor (tenure track) with the Centre for Wireless Communications Research Group. He also has a long experience as a Research Project Manager. His research interests include edge computing, IoT, healthcare ICT, distributed ledger technologies, artificial intelligence, and machine learning.

TEEMU LEPPÄNEN (Senior Member, IEEE) received the D.Sc. degree from the University of Oulu, Finland, in 2018. He is currently a Research Scientist with the Center for Ubiquitous Computing, University of Oulu. His research interests include the IoT edge computing realworld infrastructures with a focus on distributed AI methods that target energy effciency and human machine interactions. He visited the Institute of Industrial Science, The University of Tokyo, Japan, from 2012 to 2013. He has authored over 40 peer-reviewed papers in scientific journals, research books, and international conferences, where he has received three best paper awards.

LAURI LOVÉN (Senior Member, IEEE) is currently pursuing the Ph.D. degree with the Center for Ubiquitous Computing (UBICOMP), University of Oulu, Finland, with a background in mathematical statistics and ICT business. He has coauthored two patents and coauthor of 30 research papers in international peer-reviewed journals, conferences, and workshops. His current research interests include distributed AI, machine learning and data analysis methods, and applications in edge computing.

ANTTI ANTTONEN (Senior Member, IEEE) received the D.Sc. (Tech.) degree from the University of Oulu, Finland, in 2011. He is a Senior Scientist with VTT Technical Research Centre of Finland Ltd. He has made longer research collaboration visits to Lucent Technologies, Murray Hill, NJ, USA, in 2000, the University of Hannover, Germany, in 2008, and the University of Leuven, Belgium, in 2014. He has participated in numerous national and European research projects related to wireless communications. He has also authored several scientific journal articles related to energy efficiency, energy sustainability, and delay sensitivity. His main interests include energy efficient transmission algorithms and cooperative smart resource management for personal area and cellular networks.

ALI HASSAN SODHRO (Member, IEEE) received the Ph.D. degree in computer applications technology from the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences (SIAT, CAS), Shenzhen, China, and University of Chinese Academy of Sciences (UCAS), Beijing, China, in May 2016. He is currently working as a University Lecturer with the Department of Computer and System Science, Mid-Sweden University, Östersund, Sweden. Before that, he worked as a Postdoctoral Research Fellow with Computer and Information Science Department, Linköping University, Linköping, Sweden, from 2018 to 2020. He is also working as an Assistant Professor at Electrical Engineering Department, Sukkur IBA University, Sukkur, Sindh, Pakistan. He has published 70 research articles in top-notch journals [IEEE INTERNET OF THINGS JOURNAL, *IEEE Wireless Communications Magazine*, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE TIS, IEEE SENSORS JOURNAL, and *Information Fusion* (Elsevier)] and conference proceedings (IEEE ICC, IEEE VTC). His research areas include energy-efficient and battery-aware communication systems for healthcare applications, AI, 5G, IoT-device authentication, and sensor networks. He was a recipient of very prestigious Erasmus Mundus's Postdoctoral Research Fellow Award at University Lumiere Lyon 2, Lyon, France, from 2017 to 2018, and the CAS-PIFI 2019–2020 Visiting Research Scientist Award, China.

MUHAMMAD MAHTAB ALAM (Senior Member, IEEE) received the M.Sc. degree in electrical engineering from Aalborg University, Denmark, in 2007, and the Ph.D. degree in signal processing and telecommunication from the INRIA Research Center, University of Rennes 1, France, in 2013. He joined the Swedish College of Engineering and Technology, Pakistan, in 2013, as an Assistant Professor. He did his postdoctoral research at the Qatar Mobility Innovation Center, Qatar, from 2014 to 2016. In 2016, he joined as the European Research Area Chair and an Associate Professor with the Thomas Johann Seebeck Department of Electronics, Tallinn University of Technology, where he was elected as a Professor, in 2018. In 2019, he was the Leader of Communication Systems Research Group. His research interests include the fields of wireless communications—connectivity, NB-IoT 5G/B5G services and applications, and low-power wearable networks for SmartHealth.

MARKKU JUNTTI (Fellow, IEEE) received the Dr.Sc. (electrical engineering) degree from the University of Oulu, Oulu, Finland, in 1997. He was with the University of Oulu, from 1992 to 1998. In Academic Year 1994–1995, he was a Visiting Scholar with Rice University, Houston, TX, USA. From 1999 to 2000, he was a Senior Specialist with Nokia Networks. He has been a Professor of Communications Engineering with the Centre for Wireless Communications (CWC), University of Oulu, since 2000, where he also serves as the Head of CWC–Radio Technologies Research Unit. His research interests include signal processing for wireless networks as well as communication and information theory. He is also an Adjunct Professor at Rice University. He is an Editor of IEEE TRANSACTIONS ON COMMUNICATIONS.

ANTTI YLÄ-JÄÄSKI (Member, IEEE) received the Ph.D. degree from ETH Zürich, in 1993. From 1994 to 2009, he was with Nokia, in several research and research management positions, with a focus on future Internet, mobile networks, applications, services, and service architectures. Since 2004, he has been a tenured Professor with the Department of Computer Science, Aalto University. His current research interests include mobile cloud computing, mobile multimedia systems, pervasive computing and communications, indoor positioning and navigation, energy efficient communications and computing, and the Internet of Things.

THILO SAUTER (Fellow, IEEE) received the Ph.D. degree in electrical engineering from TU Wien, Vienna, Austria, in 1999. He is currently a Professor of Automation Technology at TU Wien. He was the Founding Director of the Center for Integrated Sensor Systems, Danube University Krems, Wiener Neustadt, Austria. His expertise includes smart sensors and automation networks with a focus on real-time, security, interconnection, and integration issues. He has been working in fieldbus standardization for more than 20 years, and had leading positions in several national and international research projects concerned with industrial communication and enterprise integration.

ANDREI GURTOV (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees in computer science from the University of Helsinki, Finland, 2000 and 2004, respectively. He is currently a Professor of Computer Science with Linköping University, Sweden. Previously, he was with the University of Oulu (for three years) and Aalto University (for six years) and visiting the International Computer Science Institute, Berkeley, multiple times. He coauthored over 200 publications, including four books, five IETF RFCs, and over 60 journal and 110 conference papers, and holds six patents. He supervised 15 Ph.D. theses. His research interests include network protocols, security of vehicular, airborne, industrial systems, mobile, wireless, and IoT networks, and SmartGrids.

MIKA YLIANTTILA (Senior Member, IEEE) received the Ph.D. degree in communications engineering from the University of Oulu, in 2005. He is currently a full-time Associate Professor (tenure track) with the Centre for Wireless Communications (CWC), University of Oulu, Finland. He has coauthored more than 170 international peer-reviewed articles. His research interests include edge computing, network security, network virtualization, and software-defined networking. He is an Editor of *Wireless Networks* journal. He leads the Network Security and Softwarization Research Group (NSOFT), CWC Networks and Systems Research Unit, which studies and develops secure, scalable, and resource-efficient techniques for 5G and beyond 5G systems.

JUKKA RIEKKI (Member, IEEE) received the D.Sc. (Tech.) degree from the University of Oulu, Finland, in 1999. His Ph.D. thesis was on the reactive task execution of mobile robots. Since 2001, he has been a Professor of Software Architectures for Embedded Systems with the University of Oulu, where he has been the Dean of the Faculty of Information Technology and Electrical Engineering, since 2014. He visited the Electrotechnical Laboratory, Intelligent Systems Division, Tsukuba, Japan, from 1994 to 1996. He has been an Advisor of nine Ph.D. theses. He has authored 40 journal articles, six book chapters, and over 160 conference papers. His research interests include the Internet of Things and edge computing, emphasizing efficient application of artificial intelligence, and semantic Web technologies in distributed and resource limited systems.

• • •