Artificial Intelligence for UAV-Enabled Wireless Networks: A Survey

MOHAMED-AMINE LAHMERI (Student Member, IEEE), **MUSTAFA A. KISHK** (Member, IEEE), **AND MOHAMED-SLIM ALOUINI** (Fellow, IEEE)

Computer, Electrical And Mathematical Sciences and Engineering Division, King Abdullah University of Science and Technology,

Thuwal 23955-6900, Saudi Arabia

CORRESPONDING AUTHOR: M. A. KISHK (e-mail: mustafa.kishk@kaust.edu.sa)

ABSTRACT Unmanned aerial vehicles (UAVs) are considered as one of the promising technologies for the next-generation wireless communication networks. Their mobility and their ability to establish line of sight (LOS) links with the users made them key solutions for many potential applications. In the same vein, artificial intelligence (AI) is growing rapidly nowadays and has been very successful, particularly due to the massive amount of the available data. As a result, a significant part of the research community has started to integrate intelligence at the core of UAVs networks by applying AI algorithms in solving several problems in relation to drones. In this article, we provide a comprehensive overview of some potential applications of AI in UAV-based networks. We also highlight the limits of the existing works and outline some potential future applications of AI for UAVs networks.

INDEX TERMS Artificial intelligence, deep learning, federated learning, machine learning, reinforcement learning, UAVs.

I. INTRODUCTION

UNMANNED aerial vehicles, known as UAVs, attracted a lot of research interest in the last decades due to many inherent attributes such as their mobility, their easy deployment, and their ability to establish line of sight (LOS) links with the users [1]–[4]. In general, UAVs can be classified into two main types, namely fixed-wing and rotary-wing UAVs. Each type of UAV is adapted to a specific type of application. For example, fixed-wing UAVs are more appropriate for the type of missions where stationarity is not required, e.g., military applications such as attack and surveillance. However, rotary-wing UAVs have more complex aerodynamics. They also have the ability to remain stationary at a given location, but they cannot carry out long-range missions. For example, rotary-wing UAVs are better suited to provide temporary wireless coverage to ground users.

The involvement of many industries in the manufacture of UAVs has helped to reduce their cost on the markets, making the use of a UAV network no longer a dream or a futuristic idea. In fact, they have been used in many scenarios such as providing wireless connectivity, weather forecasting, disaster management, farming, delivery, surveillance, and traffic control [5]–[12].

Several limitations related to the use of UAVs can be raised, such as their vulnerability to severe weather conditions and the need for visual LOS with the controller to avoid a risky loss of control. Most importantly, the constrained battery and low computational capabilities of drones are considered as their main limitations [13]–[18]. In fact, most commercially available UAVs struggle to hover for more than two hours and must always return to base to recharge their batteries. In addition, complex algorithms requiring high CPU and GPU capacities cannot be run onboard because of the limited computing capacity at the UAV level. Therefore, from a practical point of view, no solution is necessarily adequate for a given UAV-based problem, as complexity may affect its feasibility.

In another context, machine learning (ML) has emerged in the last years as a sub-field of AI. Moreover, its use has become prevalent in scientific research offering a new style usually referred to as the black-box technique where you only care about inputs and outputs. Furthermore, the huge

amount of data available nowadays and the existence of the high performance computing (HPC) and good GPU's helped ML to see the light. As a result, ML is being actively used today in many fields, perhaps more than one would expect.

We can also notice the emergence of several sub-fields from AI such as deep learning (DL), reinforcement learning (RL), and federated learning (FL), each for a specific type of problem. For example, DL is a branch of AI that uses layers of artificial perceptrons to imitate the human mind's thinking. It is massively used in speech recognition, computer vision, and natural language processing. RL is another branch of AI that appeared around 1979 [19] wherein an agent learns the way of making good actions in order to achieve maximum rewards. The learning process is achieved by exploitation and exploration of the different available states. RL is an active field of ML that evolved and matured very quickly. Unlike DL, RL is massively used in robotics for path planning and learning the way to do complex tasks. This does not mean that it is limited only to robotics, it is also used in many other decision-making problems that consider a goal-oriented agent that is interacting with a specific environment.

Another new field of ML is FL, which was proposed by Google in 2016 and designed to support network systems with decentralized data. FL is considered as an ML setting with the objective of training a highly centralized model on devices sharing decentralized data without the need for sending the data to a local shared unit. In other words, it is used to run ML algorithms with decentralized data architecture. This task is performed in a secure manner and when it comes to UAV-based networks, the use of FL is indeed a hot topic.

In short, AI is one of the trending areas that brings intelligence to machines and makes them able to perform tasks even better than a human can do. It is believed that combining the advantages of using AI within UAV networks is a challenging and interesting idea at the same time. Although conventional methods showed a big success in solving many problems in this area, it is still interesting to investigate whether ML and RL could provide more powerful and accurate solutions. While moving from classical approaches to intelligent methods requires sacrificing interpretability and tractability in some scenarios, it is worth opting for AI-assisted solutions given the unprecedented success realized by ML and RL especially in decision-making problems. That being said, we believe that intelligent solutions are not always guaranteed to outperform classical solutions, instead, classical approaches might provide simple and powerful solutions in some scenarios. Undoubtedly, this duality is one of the reasons that motivate investigating the use of AI for the set of special problems related to UAV networks. UAVs were designed originally to be controlled fully manually by humans, however, with the recent emergence of AI, it became a trend to provide smart UAVs in the markets. In this context, AI can use the information gathered by drone sensors to perform several tasks. AI can play an important role in resource management for UAVs to maximize energy efficiency. The design of UAVs' trajectory and deployment are also subject to AI improvement by equipping the UAV with the ability to avoid obstacles and design its trajectory automatically. For instance, in recent days, "Follow me" drones have been a tremendous success in the markets. This type of drone providing great video footage by following and filming its owner is equipped with powerful and intelligent obstacle avoidance and target tracking algorithms. Moreover, a wide range of applications can be improved in this context such as surveillance, traffic management, and landing site detection. Imaging also can be improved for UAVs by applying the existing state of the art related to computer vision for UAVs imaging.

To conclude, UAV-based network performance can be highly improved with the integration of AI algorithms to automate complex tasks and enhance the overall system level of intelligence.

A. PREVIOUS SURVEYS AND TUTORIAL WORKS

With the vast amount of published work linking AI to UAV wireless networks, several tutorials and surveys have attempted to summarize the existing literature. The authors in [4] provided a tutorial for UAV-based wireless communication systems by covering potential applications, challenges and describing the open problems in the field. However, the aforementioned work does not focus on the ML aspect for UAVs. Many other surveys and tutorials do not focus on AI techniques, for instance, a motion planning for UAVs guide is presented in [20] and a survey for UAV traffic monitoring is provided in [21].

In addition to the work mentioned above, there exist other tutorials and surveys that are oriented towards the application of ML tools in wireless communication networks. For example, a tutorial on artificial neural networks (ANNs) for wireless networks is proposed in [20]. In the same context, a review for RL and DL techniques for UAVs summarized some of the works done in this area [22], [23]. However, all the works mentioned above do not consider AI techniques specifically for UAV applications.

Nevertheless, one work can be considered as close to this work, namely the ML study for UAV communication presented in [24]. However, there are still major differences between the two works. First of all, the structure of this survey is different from that of [24] since we review each AI subfield separately and delve into the different UAV-based applications of each AI area, whereas in [24], the survey is based on the application type. We believe that this structure not only allows us to explore each area of AI in-depth, but also to provide a comprehensive overview of each AI area by presenting the most commonly used algorithms in UAV-related problems. Therefore, this survey is aimed at all kinds of readers, including those who have no knowledge of AI and also for researchers who have started to take an interest in AI from a wireless communication perspective. In addition, the work in [24] did not cover FL, which, in our opinion, is a key technique for bringing intelligence to the edge of
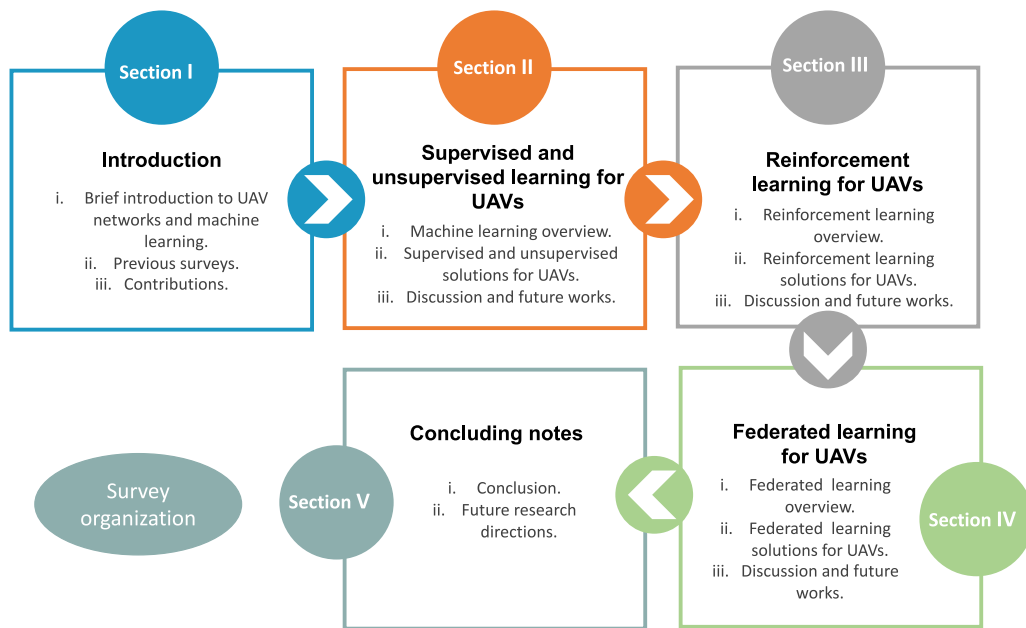
FIGURE 1. Survey organization.

UAV networks in a decentralized and secure way. Given the significant importance of FL, particularly for 6G networks, we covered this area in depth by reviewing the recently published work. We also maintained a unique approach in each AI domain and focused on the most recent publications. For example, one of the most prominent applications covered in this survey are the fusion of RL with the use of intelligent reflective surface (IRS)-equipped UAVs to support millimeter-wave (mmWave) bands, the use of drones to enable virtual reality(VR), and anomaly detection for drones, which are not covered in [24].

B. CONTRIBUTIONS
In this article, we provide a holistic overview of the state of the art research in the area of AI-enabled UAV networks. We also discuss some limitations of the existing research works and outline some potential ideas that could be addressed in the near future. We also study the implementation of intelligence at the edge of UAV networks by reporting some of the works done in FL for UAV-based networks. Furthermore, we provide a comprehensive introduction to each AI area studied in this work so that readers with different backgrounds have the ability to understand this article.

This survey is organized as follows:

- In Section II, we start by reporting the supervised and unsupervised ML works designed for UAV-based networks. A brief overview will cover these two different areas of ML and some typical algorithms and neural network (NN) architectures are provided for the reader's convenience.
- In Section III, we go over the works relating RL with UAV-related problems. We start with a quick overview of RL basics and present a classic example for RL path

planning in order to understand the basic concepts of this area.
- In Section IV, after introducing FL, we outline the key research directions that enable installing intelligence at the edge of a UAV network by reporting some of the works relating FL to UAV-related problems.

All the above-mentioned sections are concluded with a discussion presenting the limitations of the current research work and highlighting some possible future works that could be established.

II. SUPERVISED AND UNSUPERVISED MACHINE LEARNING FOR UAVS
ML is a recent buzzword related to AI. In short, it is the subset of AI that enables a computer to execute tasks accurately based on the experience gained by learning from some previous examples. In fact, ML has been very successful over the last decade because of the large available amount of data and today's powerful computers. This is why research is now oriented towards applying ML in UAV-based problems.

The areas of ML can be divided into different categories of problems, for instance, it might be divided as shown in Fig. 2 to supervised learning problems, unsupervised learning problems, and RL-based problems. In what follows, we" distinguish between the supervised and the unsupervised learning areas to avoid confusion later.

A. SUPERVISED LEARNING OVERVIEW
In supervised learning, the data provided is labeled, in other words, we provide for each data entry the ground-truth value so that the algorithm uses these values to learn how to make a decision for a new unlabeled entry. For example, predicting a UAV price from its characteristics. In this example, you need
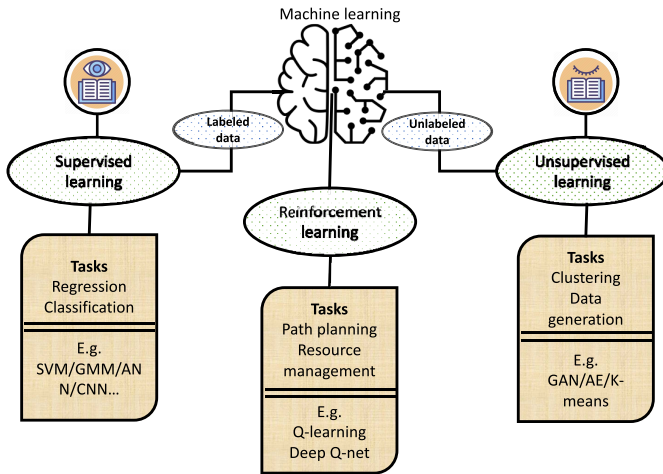
FIGURE 2. Machine learning overview.

to provide the algorithm with a set of training data that contains each UAV characteristics and its associated label (the price). The dataset is usually divided into a training set and a test set. The training set is used to learn the relationship between the input and the output and the test set is used to validate the model by measuring its accuracy. The supervised problems are often divided into either regression problems or classification problems. Regression problems provide continuous output values (e.g., predicting a price). However, classification problems provide discrete values indicating to which class the input belongs (e.g., classify benign or malignant cancer disease). In what follows, we present the most well-known ML algorithms for supervised and unsupervised learning. We also focus on the algorithms that are used to solve the UAV-related problems reported in this survey.

Some Supervised Algorithms and NN Architectures:

1) *Combined Classification and Regression Algorithms:* There are several supervised algorithms that can be used either for classification or regression. For instance, Support Vector Machine (SVM) can do both the tasks, decision trees also can be formulated to solve regression or classification depending on the use case.

2) *Regression Algorithms:* There exist algorithms that perform pure regression tasks by predicting continuous value output. For instance, we can mention two classical algorithms in ML which are linear regression and logistic regression.

3) *Classification Algorithms:* It makes sense to talk about pure classifiers in ML. Although it is mentioned in some references that Naive Bayes classifier with "some modification" can be used for regression, we present it as a pure classifier example since it was derived initially for classification based on the probabilistic Bayes theorem.

4) *Multi Layer Perceptron (MLP):* To imitate the biological human neural networks, ANNs are mathematically formulated for ML. ANNs are built with several partially-connected nodes denoted by perceptrons and grouped into different layers. Each perceptron is responsible for processing information from its input and delivering an output. As shown in Fig. 3, MLP is the simplest form of an ANN that consists of one input layer, one or more hidden layers, and an output layer where a classification or regression task is performed.

5) *Convolutional Neural Networks (CNNs):* CNN is another type of ANN designed initially for computer vision tasks. A CNN usually takes an image as an input, assigns learnable weights and biases that are updated according to a specific algorithm. The CNN architecture is characterized by the convolutional layers which extract high-level features from the image that will be used later. Technical details such as activation functions, pooling layers, and padding operation are beyond the scope of this survey. Fig. 1 shows a typical CNN architecture where feature extraction is performed in the first convolutional layers and classification is performed via a fully connected layer.

6) *Recurrent Neural Networks (RNNs):* When the data is sequential in nature, RNNs take place to solve the problem. For the sake of example, we can mention a text speech, a video, or a sound recording. RNNs are widely used in natural language processing (NLP), in speech recognition, and for generating image description automatically. The RNN architecture is similar to a regular neural network, only it contains a loop that allows the model to carry forward results from previous neurons. RNN in its simplest form is composed of an output containing the prediction and which is denoted by h in Fig. 1 and a hidden state that represents the short-term memory of the system.

B. UNSUPERVISED LEARNING OVERVIEW

Unlike supervised learning, the unsupervised learning does not use labeled data, instead, it looks for some underlying structure or hidden pattern in the data and reveals it. For instance, clustering the data, reducing data dimensionality, and data generation are considered typical tasks for unsupervised learning. In what follows, we present some classical unsupervised algorithms.

Unsupervised Algorithms and NN Architectures:

1) *Clustering Algorithms:* There exists several popular clustering algorithm in ML. We only mention, *K*-means, Gaussian Mixture Modeling (GMM), DBSCAN, and agglomerative Clustering [25]. Some of them are density-based algorithms such as DBSCAN, others implement hard association such as *K*-means. Note that, on the other side, GMM is a probabilistic model that uses soft association rule.

2) *Dimensionality Reduction Algorithms:* Dimensionality reduction is a typical technique in ML consisting of transforming data from a high-dimensional space representation to a lower-dimensional space. In this
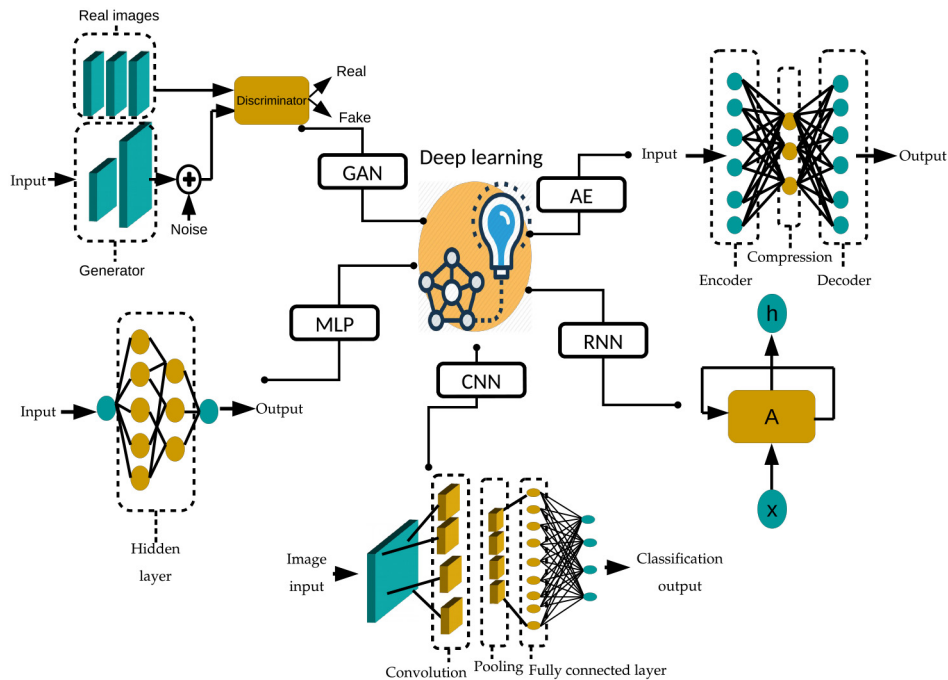
FIGURE 3. Neural network architectures.

context, we mention some spectral-based techniques such as autoencoders (AEs) which are a type of neural networks used to learn a representation of the data and encode it. Surprisingly, the architecture of an AE is extremely simple as the Fig. 1 shows. We also mention another spectral-based algorithm which is principal component analysis (PCA) as a popular dimensionality reduction technique [26].

3) *Generative Adversarial Networks (GANs):* GANs are algorithmic architectures that use two neural networks in order to generate new, synthetic instances of data that can pass for real data. They are used widely in image generation, video generation, and voice generation.

C. SUPERVISED AND UNSUPERVISED SOLUTIONS FOR UAVS-BASED PROBLEMS
1) THE POSITIONING AND DEPLOYMENT OF UAVS

Authors in [12] investigate the optimal deployment of aerial base stations to offload terrestrial base stations by minimizing the power consumption of the drones. The provided solution is considered ML-assisted due to the fact that UAVs are not required to continuously change their positions, instead, they are placed temporarily by predicting the congestion in the wireless network. The wireless traffic is predicted based on the GMM which is a probabilistic model that belongs to the set of unsupervised ML defined previously. It assumes that the data distribution can be modeled by the Gaussian distribution. First, a K-means algorithm divides the users into K clusters and then a weighted expectation maximization algorithm is performed on the K clusters in order to find the optimal parameters for the GMM

model. The next step is to deduce the optimal deployment by formulating a power minimization problem for UAVs. The numerical results show that the ML-assisted approach outperforms the classical solution by reducing the mobility and the power needed for downlink purposes. Although the work done is of great importance by combining ML with optimization techniques, using a K-means algorithm to classify the users brings the question of how to choose manually the value of the number of clusters K and also how to initialize the cluster center positions.

In the same context, the authors in [27] investigate an optimal placement of UAVs acting as aerial base stations by building a structured radio map. Due to the nature of the complex terrain and the difficulty of exploiting such radio map, the authors propose a joint clustering and regressing problem using a maximum likelihood approach that is formulated based on the K-segment ray-tracing model. ML is also used in predicting the channel in order to reconstruct the radio map.

The optimal deployment of aerial base stations using UAVs is also studied in [28]. ML techniques are used to predict download traffic using weighted expectation maximization. This ML technique has been compared in terms of performance to a baseline expectation maximization algorithm and the K-means algorithm. In addition, contract theory is used to ensure that the downlink demand is satisfied by selecting the appropriate UAV for each hotspot.

In [29] the authors investigate generating waypoints for the UAV flight path based on previous flight records. The problem is that any error committed by pilots in the past will influence the future waypoints of the UAV. To this end, the authors employed an unsupervised learning technique based

on AE to remove pilot errors and reconstruct the image generated from the flight record. The proposed method is compared to generating the waypoints throughout the K-means algorithm and has proven its efficiency.

In [30] the communication efficiency between a UAV and a base station is improved by predicting the location of the UAV given its past locations. In fact, while offloading a terrestrial base station, a UAV can be subject to wind perturbation which will result in a certain degree of offset and hence a loss in capacity. To solve this issue, the authors propose a RNN-assisted framework where the next elevation and horizontal angles of the UAV with reference to the base station are predicted using the past angles. This method leads to predicting the specific location for a high-speed movement UAV. The authors keep tuning the RNN parameters such as the number of hidden nodes and the number of hidden layers and then study their impact on the prediction accuracy. Numerical results have shown that a high accuracy could be achieved for a 4 layer RNN with 16 hidden nodes.

Further alternatives for UAV path planning have been proposed in the open literature. In [31] an unsupervised solution has been proposed to enable motion prediction for a group of heterogeneous flying UAVs. In addition to predicting the future locations of the UAVs, the algorithm is designed to classify the network nodes based on their motion properties. In addition, the authors in [32] use ANN to predict the optimal location of UAVs that are used as relays in a Flying Ad Hoc Network (FANET) setup.

2) CHANNEL ESTIMATION

One could think about how ML can improve the well-established empirical models used to estimate and model the complex UAV-to-UAV and ground-to-UAV channels. In this context, the work in [33] investigates the prediction of the UAV-to-UAV path loss. The predictions generated by the K Nearest Neighbors (KNN) and the Random Forest algorithms are compared to empirical results. The path loss is predicted starting from several parameters such as the propagation distance, transmitter altitude, receiver altitude, and elevation angle. The comparison of the results with the data generated by the ray-tracing software shows that ML performs well in these prediction tasks.

As mmWave bands are being exploited for next-generation cellular systems to improve the communication capacity, the authors in [34] use a generative neural network to predict the channel state between a UAV and two types of terrestrial base stations: (i) terrestrial street-level base stations and (ii) aerial roof-mounted base stations. A first neural network classifies the link type (LOS/NLOS/outage) and feeds this information to a second neural network to generate the different channel parameters. The authors in [35] use GAN to estimate the air-to-ground channel characteristics for mmWave communication within a UAV wireless network. A distributed GAN architecture is proposed to enable distributed learning for the UAV network. The learning process is based on distributed datasets from channel measurement.

Supervised ML is used in other research works to predict channel quality between a UAV and ground nodes. For instance, ANNs are used to predict the path loss in [36]–[39]. The authors in [36] use ANNs to predict the signal strength of the UAV and estimate the channel propagation. A shallow ANN is proposed to analyze the effect of several natural phenomena on the signal such as: diffraction, reflection, and scattering. The input layer is composed of parameters like the distance to the UAV, altitude, frequency, and path loss. This exciting work may be impeded by the large processing time of the data by the ANN which raises the question of whether this solution is adequate for real-time applications or not. In the same context, in [39] the signal strength between the UAV and ground nodes is estimated using ANN. The authors consider an urban environment where the signal strength data is used as an input to feed an ANN. Using this data, the channel parameters are estimated accurately. In addition to ANN, other supervised ML algorithms are used in [40] to predict the received signal strength received at a flying unit from a cellular base station.

In contrast, in [41] unsupervised learning is used for a 3D channel modeling between UAVs and ground mobile users. The popular K-means algorithm is used in this work to classify the links into LOS and NLOS. Aside from ANN, the SVM algorithm can be used for regression in addition to classification. The work in [42] proposes a method for path loss prediction in urban outdoor environment using support vector regression (SVR) algorithm and compares the obtained results with the empirical ones.

In short, we have reviewed several ML techniques used for the same purpose, which is predicting path loss for UAVs. Among these techniques, we mention the use of KNN, Random forest, ANN, and SVR. Many similarities and differences exist between these different algorithms as they can all be considered as non-linear classifiers in some scenarios. On the one hand, KNN is known as a simple non-linear ML classifier, but it could do robust prediction tasks, it is shown to have 4.56 dB mean error in [33]. On the other hand, Random Forest, which is an ensemble learning method that uses multiple learners to make predictions, is shown to be more effective in predicting path loss. Random Forest is shown to have 2.26 dB mean error in [33]. Moreover, ANN is a more complex algorithm since the model includes many parameters, layers, activation functions, optimizer, and data. ANNs are also less tractable and less expensive in terms of computation depending on the size of the network. However, for path loss prediction, they have shown relative success since they have reached 2.72 dB in terms of mean average error [38]. The SVR algorithm on the other side is also tested for the path loss prediction problem and different kernels have been used, such as Gaussian, polynomial and Laplacian kernels. It has been shown in [42] that the performance of ANN is comparable to that of the SVR with the Laplacian kernel, however, it is believed that the SVR is not only faster than ANN but also simpler since initialization problems could be avoided. To conclude, we think that

KNN performs poorly on path loss prediction and that the SVR-Laplacian and ANN seem to have better performances. That being said, the choice between SVR and ANN creates a compromise between accuracy and complexity that should be taken into account.

3) UAVS FOR VIRTUAL REALITY (VR)

Virtual Reality (VR) refers to the creation of a virtual environment for multiple purposes, such as education or entertainment. This technology has begun to gain significant interest in the last few years and we have recently seen the emergence of an exciting research direction linking the use of VR technology to UAV networks. In this context, AI is used to overcome the various VR application challenges since such technology requires high transmission standards, for instance, low latency and high data rate. In [43], VR networks are studied by optimizing transmission and content caching. In the proposed framework, UAVs are used to collect VR data from the users and transmit it to small base stations serving the users. Therefore, an optimization problem for joint transmission and content caching is formulated and solved via the use of deep learning techniques combining liquid state machine spiking neural networks and echo state networks. In another context, authors in [44] provide a UAV VR simulation platform used to assess the performance of several DL-based solutions for UAVs such as autonomous path planning and obstacle avoidance. The constructed VR platform can generate images and data for UAV navigation that could be used in the training phase. The authors in [45] investigates the use of UAV-IoT network to enable a remote VR immersion experience. UAVs are positioned at a given scene of interest, capture different points of view and transmit them to a dedicated aggregation point. The aggregation is then performed to enable the VR session at the user's location. The objective is to maximize the fidelity of the user's immersion under certain constraints. RL techniques have been used in this work to derive the optimal UAV placement so that immersion fidelity is maximized. In [43], mentioned earlier, where the transmission and caching of content is optimized by DL techniques, RL is compared to the proposed DL-based algorithm. However, the numerical results show that DL outperformed the Q-learning algorithm.

4) ANOMALY DETECTION FOR UAVS

UAV networks are extremely sensitive to any malfunctioning or anomalies that might occur during the drones' operations. To avoid such scenarios, it is rational to use the data provided by the drone sensors to keep track of the flight safety level. In this context, anomaly detection is considered a typical task that consists of detecting data samples that do not adhere to the normal functioning of the system. In [46], the authors propose an unsupervised algorithm to detect unusual events happening in the drone field of view. The proposed DL architecture is trained on bird-view images and GPS data to perform this task. Authors in [47] propose an anomaly

detection algorithm to identify and isolate malfunctioning UAVs. To this end, Kullback-Leibler Divergence is used to detect any problem based on the external sensors such as the humidity rate and wind speed. On the other side, an ANN is used to classify the internal sensor data. Numerical results show an acceptable accuracy for the proposed data. Furthermore, kernel PCA (KPCA) is used in [48] to perform anomaly detection based on the drone sensors' data. KCPA is used to reduce the dimension of the data taking into account the non-linear relationships between the samples. We kindly refer interested readers to [49] covering the literature and the current trends related to unsupervised anomaly detection for UAVs. In another context, authors in [50] investigate learning the law of the rotor flow field of a UAV. This complex task is achieved through the use of GAN where the flow field features are learned by the proposed model. Numerical results show that the proposed method outperformed the computational fluid dynamics model.

5) UAV DETECTION

Since UAVs are used extensively not only by the military but also by civilians, several applications need to be supervised by the authorities because UAVs could be used for espionage or even as a lethal weapon. Therefore, the detection and tracking of UAVs is very important to limit these dangers. In this field, numerous research projects have proposed different solutions, which we divide into image-based solutions and sound-based solutions.

Starting with sound-based detection problems, the authors in [51] present a real-time UAV detection system based on analyzing the sound data coming from the drone. For this purpose, two ML methods have been applied and compared in terms of accuracy. The first step consists of detecting potential existing UAV by analyzing the frequency and then check whether the sound exceeds a predefined threshold for drones. The first ML method used is Plotted Image Machine Learning (PIL). This method uses the visualized Fast Fourier Transform (FFT) graph generated from the data sound to compare average image similarity with a reference FFT target. The second method is based on the KNN algorithm applied to the FFT and measures the average distance with the target. The simulation results show that the PIL method outperforms the KNN methods and succeeded to provide good results. At this level, we point out that even if the visual drone detection can be limited by the quality and the resolution of the input image, sound data also can be highly affected by ambient noise in real applications. It also is not obvious if all the UAVs will have the same FFT profile used as a predefined target in the problem. Moreover, KNN algorithm is a simple and straightforward algorithm in ML, and hence trying more sophisticated algorithms will be beneficial for the problem. In another line of research, the authors in [52] use a spiral microphone array to detect and track the position of a flying UAV. Several spectrograms and filters are applied to the input sound to perform feature detection before feeding the concurrent neural networks architecture

TABLE 1. Imaging for UAVs.

Reference	Model	Application	Date of publication
[63]	Faster R-CNN	Car detection	Late 2017
[64]	Nazr-CNN	Damage detection	Late 2017
[65]	CNN+SVM=CSVM	General object detection	2018
[66]	modified region-based CNN	Electrical equipment defect detection	2018
[67]	Faster R-CNN+ Region proposal network(RPN)	Pedestrians detection	2018
[68]	Semantic segmentation+CNN	UAV geolocalization	2018
[69]	CNN	Car detection	2018
[70]	CNN	Building crack detection	2018
[71]	Faster R-CNN+Yolov3+RetinaNet	Tree detection	2019
[72]	CNN+Digital Surface Model(DSM)	Surface classification	2019
[73]	CNN with semi-supervised learning	Agricultural detection (soybean leaf and herbivorous pest)	2019
[9]	CNN	Rice-grain estimation	2019
[11]	Yolov3	Weed location	2020
[10]	CNN	Single tree detection	2020
[74]	Yolov2	Green mangoes detection	2020
[75]	Faster R-CNN	Maize Tassels detection	2020
[76]	CNN	Counting and locating citrus trees	2020

to perform the task. The scientific literature is still rich with regard to the recognition of the acoustic signature of UAVs [53]–[55]. Other research works use different features to detect the UAV presence such as the WiFi traffic [56] or the UAV radio frequency signal [57]–[60].

Another approach to address UAV detection problem is through images instead of sound. In this context, different deep ML architectures are compared at the visual UAV detection task using a Pan-Tilt-Zoom camera in [61]. Moreover, a recent work in [62] reviewed some of the computer vision techniques used in UAV detection problems.

Finally, we highlight the fact that making a hybrid system that uses image, sound, and radio UAV transmission signals at the same time will be a very interesting futuristic idea. In this context, we refer interested readers to the recent work in [51].

6) IMAGING FOR UAVS

Although computer vision is beyond the scope of this survey, several research works relating imaging to UAVs exist in the literature, for example, authors in [77] investigate the detection of a forced (emergency) safe landing site. The detection is converted into a classification problem where two known classifiers (GMM and SVM) are tested. The classifier converts the real map into a safe or unsafe grid map. A filter is applied later to remove unsafe spots and keep the potential landing sites. The main reason why these types of problems are not considered in this study is that they can be treated as pure computer vision problems, and the application to UAVs does not change the nature of the task, except that the images are taken from a given altitude. In other words, the same techniques are applied to UAVs imaging, such as CNN, feature extractors, and edge detectors. For the reader convenience, we summarize a number of the recent works in Table 1. We also refer the readers interested in more UAV imaging problems to [22, Table 1].

D. DISCUSSIONS AND FUTURE WORKS
1) PRACTICAL ISSUES OF ML IMPLEMENTATION

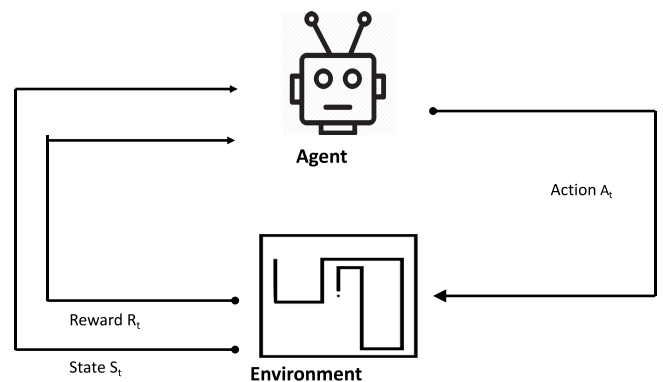The application of ML techniques in UAV-based networks can be hampered by the limited computing capacity onboard.

FIGURE 4. Reinforcement learning elements.

In fact, most commercially available UAVs are not equipped with the sophisticated processors needed to execute heavy ML algorithms. Even if you intend to equip a drone with a powerful CPU and GPU, you must first take into account its weight and power consumption. As a result, the same problem will persist due to the power constraints of UAVs. One solution to this problem is to use the cloud to train models and make inferences at the UAV level. However, this solution will increase the communication costs, which in turn will bring us back to the energy constraint problem, because the UAV will have to communicate back and forth with the cloud. Therefore, another interesting solution to the problem is to run the ML onboard, but this time adjusting the ML algorithms to the UAVś limited capacity. This technique leads us to a new field usually referred to as on-device learning dedicated to constrained devices. A recently published study [78] has examined device learning by addressing lightweight ML algorithms and discussing the different ML and DL algorithms in terms of complexity and resource consumption. We propose a final solution to address the execution of ML onboard, namely FL. It consists of executing ML in a decentralized way by sending model updates over networks instead of sharing raw data. We intend to cover and discuss this technique at Section IV of this survey.

In addition to the hardware and software limitations of UAVs mentioned above, the practical use of ML in UAV

networks still faces other significant hurdles related to existing rules and regulations. Although research is aimed at partially or even fully autonomous UAV applications, most existing regulations do not allow such operations in real life. For example, the U.S. Federal Aviation Administration (FAA), in its latest regulation published in December 2020, did not specify a single point concerning autonomous UAVs. Instead, it focuses on regulations dedicated to the human operators who control a drone. That being said, it is important to mention that there is still hope for autonomous UAVs to see the light of day. In fact, unlike the FAA, the European Aviation Safety Agency (EASA), in its latest regulation published in December 2020, admits the existence of autonomous UAV operations by including them and classifying them in different classes according to the risk level of the application. This will certainly offer new opportunities for innovative UAV solutions based on ML and AI in general. In conclusion, it is important to harmonize and unify the UAV regulations around the world, as this will motivate future research in this area.

2) LITERATURE DISCUSSIONS

Although we have tried to criticize objectively some of the work we have covered previously, in this section we intend to present our thoughts on the use of ML in wireless communication problems.

Firstly, it can be noted that, in the literature, ML tools are often used to solve problems that could be solved in a simpler and more deterministic way, giving the impression that the need to use ML is not well justified, which could lead in many cases to an ML misapplication.

Moreover, we remarked that in all works that we have covered so far, ML results always appear better than empirical results in the numerical simulation. This fact raises the question of whether is it true that ML tools are always outperforming the classical methods or not. At this point, we should mention that we are neither doubting the major success of ML nor questioning its efficiency in solving many problems, instead, we are highlighting the fact that in some cases, choosing the data plays an important role in assessing the accuracy of an ML model. To clarify the idea, let's put in place a concrete example. Imagine that you are working on a computer vision object detection problem and the goal is to detect a UAV in the images. Then, if you do not provide a good quantity of non-UAV images to the model, you will find out that the CNN is providing good accuracy on UAV images and fail in non-UAV images. Moreover, if the test set for example is biased and has some similarity with the training set, you will end up with a good accuracy, but in reality, the model will fail to predict new unseen examples. Hence, the quality and the quantity of the data plays a big role in assessing the accuracy of the model, and neglecting this point will lead to a fake ML accuracy.

In the same context, we know that data plays an indispensable role as the learning algorithm is used to discover and learn knowledge or properties from the data. That is

why we strongly believe that the wireless communication community should accredit more importance to providing open-source high-quality data as we remarked that there is not a sufficient quantity of data online dedicated for wireless purposes compared to the amount of data available for computer vision tasks. In this regard, we mention some techniques that might help in overcoming this problem such as data augmentation if the problem is related to the size of the dataset and transfer learning in the case where data is inexistent. The latter technique consists of using a pre-trained model on another dataset of the same problem type.

Another ML drawback is remarkable in some works where methods are compared in terms of performance. For instance for CNN architecture comparison, you may notice that when comparing two ML models, there is no mathematical explanation as to why such model is better than the other one or why such a NN architecture outperforms another NN architecture. This point illustrates the "black box" aspect of ML, in other words, it is a matter of tuning parameters and evaluating the result and no further explanation could be provided. Consequently, for a given problem we sometimes cannot predict which model will perform well and which model is not promising.

However, ML remains an interesting alternative and a promising tool for UAV-related problems in particular. Therefore, we believe that several ideas can be addressed in the future. In fact, more complex ML models can be tested on some UAV-related problems, for example, in path loss prediction more regression tools can be tested on this problem. Also for UAV detection problems, we noticed that it is solved either via sound detection or via image detection by converting it into a computer vision problem. Instead, we think that a complex hybrid system that uses different types of inputs (e.g., sound FFT, image, radio transmission.) is feasible by ML, where an adequate NN (e.g., a type of CNN for images and a given type of RNN for sound and radio) will provide a score for each type of input and then a final NN is used to classify the output. Moreover, we noticed that there is a tendency to use supervised learning algorithms in solving UAV-based problems. For this reason, we believe that unsupervised learning techniques, such as clustering and dimensionality reduction, could be further explored in the future for drone-related problems.

To conclude, ML supervised and unsupervised frameworks have often successfully overcome many challenges by providing intelligent solutions for various problems involving UAVs.

III. REINFORCEMENT LEARNING SOLUTIONS FOR UAVS
A. RL OVERVIEW

Like the supervised and unsupervised learning areas of ML, RL is another area of ML dedicated to making decisions in a well-defined environment. Formally, a reinforcement problem always has 5 main elements as shown in the Fig. 4:

1) *The Agent:* An entity that can take an action denoted by A_t and receives a reward R_t accordingly.

2) *The Environment:* A representation of the real-world in which the agent operates.

3) *The Policy:* It is the mapping of each state S_t to an action A_t. We usually denote a policy by π.

4) *The Reward Signal:* The feedback that the agent receives after performing an action. It is denoted on the Fig. 4 by R_t.

5) *The Value Function:* It represents how good a state is, hence it is the total expected future rewards starting from a given state. A value function is usually denoted by $V(s)$ where s is the state that we are interested in. Mathematically, it is formulated as follows: $V(s) = \mathbf{E}(G_t)$, where G_t is the discounted sum of future rewards: $G_t = \sum_t \gamma^{t-1} R_t$, $\gamma \in [0, 1]$.

The goal is to decide correct actions (or policy) in a way that maximizes a predefined reward function, which should be adapted to the type of problem. In addition to the 5 elements of RL mentioned above, another element can be considered in some cases, which is the model. Depending on its presence or not, RL problems can be divided into two main categories which are the model-based RL and the model-free RL. In what follows, we differentiate between these two areas.

1) MODEL-BASED RL

As its name indicates, the model-based RL problem uses a model as a sixth element to mimic the behavior of the environment to the agent. Consequently, the agent becomes able to predict the state and the action for time T+1 given the state and the action at time T. At this level, supervised learning could be a powerful tool to do the prediction work. Thus, unlike the model-free RL, in model-based RL, the update of the value function is based on the model and not on any experience.

2) MODEL-FREE RL

In model-free RL problems, the agent cannot predict the future and this is the main difference with the model-based RL framework explained previously. Instead, the actions are based on the so-called "trials and errors" method where the agent, for instance, can search over the policy space, evaluate the different rewards, and pick finally an optimal reward. A well-known classic example for model-free RL is the Q-learning method where it estimates the optimal Q-values of each action and reward and chooses the action having the highest Q-value for the current state. To summarize, differentiating between model-based and model-free RL problems is an easy task. Just ask yourself the following question: Is the agent able to predict the next state and action If the answer is yes then you are dealing with a model-based RL, otherwise, it is more likely a model-free RL problem.

3) DEEP REINFORCEMENT LEARNING (DRL) OVERVIEW

Although RL had great success in solving a variety of decision-making problems, it showed a limited performance in solving complex problems, especially when using large action and state space. Thus, DRL started to gain a big momentum in solving complex problems, especially after beating humans in many complex games, for instance, chess and Go. The idea that lies behind the success of DRL consist of approximating the states by the use of neural networks. This is what also makes the agent able to deal efficiently even with unseen situations, contrary to the classical RL approach. Among the many algorithms proposed in the literature, in the following section, we go over the most used ones. For the reader convenience, the two algorithms, Deep Q Network (DQN) and Deep Deterministic Policy Gradient (DDPG), are going to be covered briefly. Thus, we kindly refer readers interested in their deep technical details to [79] for DQN and to [80] for DDPG.

Deep Q Network (DQN): DQN is the first algorithm proposed in the context of DRL by Mnih *et al.* in [79]. To understand the key concepts of DQN, a basic knowledge of Q-learning algorithm is recommended, hence we refer interested readers to Section III-B1. It is worth to mention that DQN is proposed as an improvement to Q-learning which uses a discrete state and action space in order to build the Q-table. In contrast, the Q-values of the DQN are approximated using ANN by stocking all the previous agent experience in a dataset and then feeding it to the ANN to generate the actions based on minimizing a predefined loss function derived from the Bellman equation. We should also mention the fact that the idea of DQN is inspired by Neural Fitted Q-learning (NFQ) proposed in 2005 [81] that was suffering from overestimation problems and instabilities in the convergence. There exist many other improved variations of DQN such as double DQN, dueling DQN, and distributional DQN. Despite the remarkable success of DQN, especially when it was historically tested on ATTARI games, it has its own limitations such as the fact that it cannot deal with continuous space action and cannot use stochastic policies.

Deep Deterministic Policy Gradient (DDPG): To overcome the restriction of discrete actions, Deterministic Policy Gradient (DPG) algorithm was first proposed in Deepmind's publication in 2014 [80] based on an Actor-Critic off policy approach. We refer readers that are not familiar with Actor-Critic RL methods to [19, Chapter 13]. For the sake of simplicity, let's keep in mind that Actor-Critic methods are generally composed mainly of two parts: a Critic that estimates either the action-value or the state-value and an Actor that updates the policy in the direction proposed by the Critic. Later on, in 2015, and based on the DPG algorithm, Deepmind proposed a new DRL algorithm called the Deep Deterministic Policy Gradient (DDPG) algorithm. DDPG is a model-free, off-policy method that is based on Actor-Critic algorithm. In short, DDPG is a DRL algorithm that helps the agent to find an optimal strategy by maximizing the reward return signal. The main advantage of such deep algorithm is that it performs well on high-dimensional/infinite continuous action space.
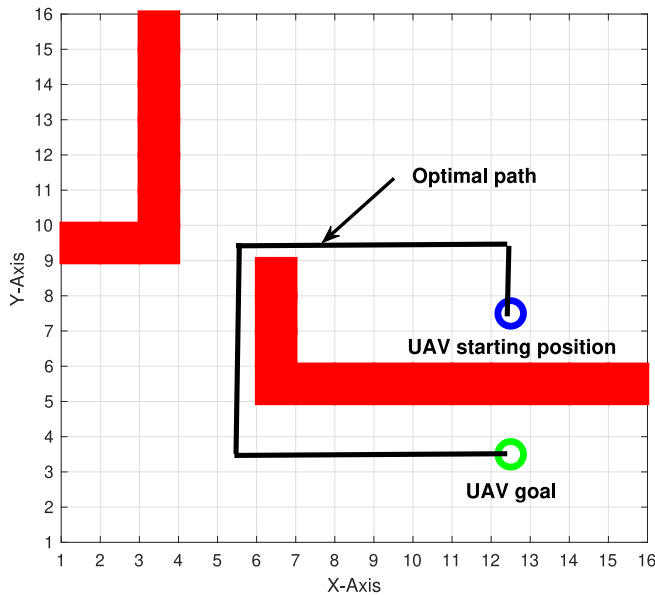
FIGURE 5. Grid map.

FIGURE 6. Exploration/Exploitation dilemma.

B. CASE STUDY

Motivated by its popularity among RL algorithms, we introduce Q-learning which is a classical free-model RL algorithm. We intend to provide a comprehensive and practical explanation to the reader on how RL could be used in UAV path planning problem. Readers with a basic knowledge of RL could skip this section. We stick to a basic example where a UAV flying at a fixed altitude learn how to reach a given target while avoiding obstacles in the map shown in Fig. 5.

1) Q-LEARNING OVERVIEW

Q-learning algorithm is based on the Q-table used to select actions for the agent at each step. The table is composed of the combination of every state with every possible action and hence its dimension is $|States| \times |Actions|$. The Q-table is used to store and update the maximum future reward referred to by Q($state_i$, $action_j$) which is the (i_{th}, j_{th}) entry of the Q-table. This Q-table is of great importance to the Q-learning algorithm simply because it is used to determine which action should the agent take such that the expected future reward is maximized.

2) UPDATE RULE

The update of the Q-table is done using a fundamental equation in RL which is the Bellman equation:

$$Q_{new}(s_t, a_t) = (1 - \alpha)Q_{old}(s_t, a_t)$$
$$+ \alpha(R_{t+1} + \gamma max_a(Q(s_{t+1}, a)), \quad (1)$$

where s_t, a_t are respectively the state and the action taken at time t, α is the learning rate, which allows the old value of the Q-table to influence current updates, γ is the discount factor, which is a measure 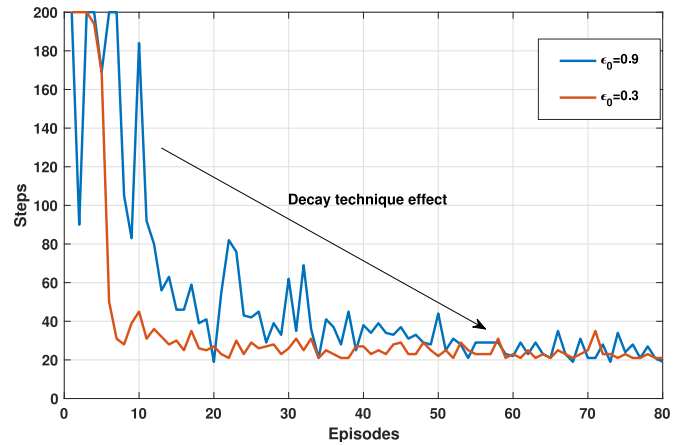of how future rewards will affect the system. After every taken action, the agent updates its Q-table values using (1), then, at a given state, it selects the action having the highest Q-value.

3) THE EXPLORATION/EXPLOITATION DILEMMA

One fundamental concept for RL, which is visible also in Q-learning, is the exploration/exploitation dilemma. To explain this duality, let's discover how the agent will succeed in reaching its goal. First, the agent makes a random step in the environment, then it starts updating the Q table (initialized with zeros for example) according to (1). However, if the agent only uses the Bellman equation, it is possible that it is stuck in a good state forever, while better states exist on the map. It is similar to a case of an optimization process that is stuck in a local minimum or maximum while better solutions still exist by exploring the environment. To solve the last problem, the exploitation/exploration dilemma is proposed. This duality introduces randomness into the system so that the agent at each step could either exploit the environment by selecting actions that maximize the Q-values of the Q-table, or explore the system by selecting some random actions. The parameter that usually refers to the probability threshold for exploration is denoted by ϵ. In our implementation, we used a decay technique that decreases the value for epsilon at each episode so that we encourage exploration at the beginning of the process, usually known as early exploration, and then prioritize exploitation so that the agent can use the learned paths. Fig. 6 shows the effect of the initial value of ϵ, denoted by ϵ_0, on the convergence of the system. The red line, corresponding to a low ϵ value, converges more rapidly since the exploration probability is low and hence the system will rapidly use the optimal values from the Q-table to take actions. However, it is clear that for the blue line using early exploration, additional randomness is introduced at the beginning of the process due to the high starting value of ϵ. We also remark that the number of steps is decreasing due to the fact that the UAV has already found its optimal path, shown by a solid black line in Fig. 5, starting from episode 20 approximately.

RL is considered a promising framework for the UAV network in many scenarios. In what follows, we cover part of its applications based on the existing literature

C. RL SOLUTIONS FOR UAV-BASED PROBLEMS
1) RL FOR AUTONOMOUS NAVIGATION

A network of UAVs can no longer be controlled in a classical way by manually controlling the navigation of each UAV from the network separately. It is highly recommended nowadays to equip UAVs with the ability to make intelligent decisions by implementing a high level of control. Achieving such a high autonomy for UAVs is a challenging task due to the continuous changes in the UAVs environment and to the different constraints related to UAV navigation (e.g., battery, UAV dynamics).

In [82] a high-level control method of UAVs is implemented for uncertain or unknown environments. Although model-based algorithms are generally not suitable for real-time applications due to the expensive computation needed for learning the model and deciding actions to take, the paper uses a modified model-based RL by implementing Texplore algorithm to perform path planning task for a UAV. The advantage of Texplore is that it separates the action selection, model learning, and planning by performing them in a parallel manner. Simulation results show that Texplore algorithm outperforms the classical Q-learning method by avoiding exhaustive exploration of the environment. The work done so far is interesting but still limited to a 2D problem with a simplified map and hence could be extended in the future to a more complex and challenging 3D environment where the UAV can adjust its height in order to avoid potential obstacles.

Beyond pure path planning, the work in [83] investigates providing coverage for ground users by studying the deployment and the trajectory design of a network of UAVs in order to meet several performance metrics such as; coverage, minimum interference, and best quality of experience (QoE). The proposed model-free RL-assisted framework enables dynamic tracking of users' movement by adjusting the UAV's location accordingly. The idea starts by clustering the users using a classical GAK-means algorithm (which is a modified version of the K-means algorithm explained briefly in Section II). It is worth mentioning that deploying the UAV at the cluster center does not meet optimality simply because the performance metrics adopted in the problem are not only related to the euclidean distance to users, instead, they are related to other parameters such as the altitude of the UAV and the LOS presence. Consequently, a Q-learning algorithm is proposed to firstly deploy UAVs in a sub-optimal way and decide their trajectory later. The work done so far is of great importance, however, some assumptions made may be far from reality, such as the fact that users, when moving around, are not supposed to mix with other clusters. Moreover, positioning the UAVs initially using the K-means results might be better than selecting a random location in terms of fast convergence to the sub-optimal positions.

More other works in the literature have focused on providing coverage for ground users. Since the problem setup is composed of several UAVs, some of these works investigated using multi-agent RL techniques. For instance, in [84] optimal coverage is studied through a distributed RL algorithm based on Multi Agent Reinforcement Learning (MARL) which consists of applying RL techniques for the type of problems where multiple agents are operating in the same environment. Each agent has the objective of maximizing its own future rewards by interacting with the environment. Moreover, authors in [85] investigate using multi-agent Q-learning to design a trajectory for a network of UAVs. The objective is to maximize the information rate while minimizing the power consumption. In addition to the multi-agent Q-learning algorithm that is used to determine the optimal positions for UAVs, an echo state network is used to predict the users' movement. In [86], the authors prove that double Q-learning could be an improvement for the classical Q-learning for problems related to wireless coverage. A new framework for trajectory design is proposed to maximize the number of users having a good QoE.

So far, we covered classical RL solutions for UAVs path planning problems, however, a more complex autonomous navigation solutions could be provided via DRL. In what follows, we go over the most relevant works coupling DRL with autonomous navigation for UAVs.

In the context of providing coverage for ground users, DRL can play an important role in building efficient solutions. In such a setup, the UAVs are usually deployed as flying base stations or relays. For instance, the authors in [87] investigated applying DRL to a multiple input multiple output (MIMO)-based UAV network where each UAV is equipped with a single antenna. The proposed DRL solution is based on DQN where the Signal-to-Interference-Plus-Noise Ratio (SINR) was used as a metric for the quality of the channel and based on which the reward signal is defined. The UAV maximizes the expected reward calculated based on the received signal strength. Consequently, the UAV maximizes its coverage efficiency measured based on a predefined coverage score. The proposed solution was finally compared with other DRL methods and proved its superiority in some setups. However, we think that comparing it to a different type of DRL algorithms such as DRL-based energy efficient control for coverage and connectivity (DRL-EC^3) proposed in [88] is not a fair comparison since the latter solution is based on DDPG where the action space and the state space are continuous. In addition, the state space considered in the solution provided in [87] is limited to 3 possible cases related to the received signal strength.

In addition to providing ground wireless connectivity, there exist a plethora of areas where UAVs could be used efficiently, such as drone delivery. In this frame, achieving drone delivery tasks through DRL was investigated in [89]. The authors used double DQN to propose a path planning algorithm for UAVs having an objective to reach a destination in an obstacle-impaired environment. The proposed

solution is an improvement to the authors' previous work in [90], where three DRL algorithms were tested which are the DQN, double DQN, and duel DQN. As double DQN gave the best results, in [89] the same algorithm was used and the depth information deduced from the image of the UAV stereo-vision front camera was used as an input. More futuristic ideas are discussed in the literature, such as using drones to serve food and drinks in restaurants [91].

To this point, we have mainly focused on research works that are based on Q-learning methods, either using a classical RL or DQN. However, policy gradient methods could be applied to a wider range of RL problems. For instance, DDPG algorithm, which belongs to the set of policy gradient methods, is more suitable for complex problems, especially when dealing with continuous action space. To make it simpler, let's consider a UAV path planning problem. To apply DQN you need to discretize the action space and accordingly, the UAV will have a well-defined set of movement directions that it can perform. As a solution, policy gradient methods could be easily implemented to perform continuous action. For instance, in the context of UAV motions, actions could be related to the speed values and direction angles. In what follows, we are going to cover some of the relevant works that apply this type of methods to solve path planning tasks for UAVs.

Back to providing wireless connectivity for ground users, the authors in [88] used an Actor-Critic based method to solve a multi-objective control problem where the UAVs tend to minimize their energy consumption and maximize their coverage range. Unlike the previously discussed DQN-based solutions, this work is based on continuous action space formed by the UAV direction and the flying distance for each UAV. Moreover, the authors take into consideration coverage fairness which is an important indicator since maximizing coverage could fall into covering a small subset of ground users. As a solution for the defined multi-objective problem, the authors adjusted the DDPG algorithm accordingly and called it DRL-EC^3. The new algorithm is compared to two baseline methods and proved its superiority in terms of coverage score and energy consumption.

In [92] the environment considered is a complex large-scale three-dimensional map. In other words, the map is crowded with obstacles, where all directions are possible for the UAV, and it is also dynamic. Those types of maps are quite challenging for RL path planning due to the fact that it is very difficult to rely on methods that use maps to represent the environment for the agent. The solution proposed is based on modeling the navigation problem using partially observable Markov decision process (POMDP) and then solving it by applying a DRL-based algorithm called Fast-RDPG. It is worth mentioning that 1) POMDP is an extension of Markov decision process (MDP) and that 2) recurrent deterministic policy gradient algorithm (RDPG) belongs to another set of DRL algorithms that are based on DPG.

In [93], DDPG algorithm, briefly introduced in Section III-A3, is used to train the UAV to navigate in a 3D environment while avoiding obstacles. The proposed solution considers a continuous action space which explains the use of a DDPG-based approach. The authors us transfer learning to accelerate the learning by using the weights learned by the UAV after being trained in a free space environment. An urban environment is simulated by adding obstacles in specific locations and penalizing the UAV for any crash that occurred while reaching its target location. Numerical results show that the UAV succeeded in reaching its target while avoiding all obstacles in the way. However, the success rate of the UAV decreases with the complexity of the environment by adding more obstacles. The lacking of precision is explained by the fact that using infinite continuous action space makes it hard to reach full accuracy.

DDPG is also used differently in [94] to jointly design a path for a network of UAVs in order to maximize its throughput. The proposed idea consists of formulating the problem as an MDP where the reward is related to the throughput and the constraints are related to total transmission power and channel availability. The actions taken by the UAV are related to adjusting both the 3D location and the transmission control. Due to the fact that the actions and the states are continuous, DDPG framework is used in 3 three different setups. For each setup, the reward function is changed to achieve a given control objective.

Several other works existing in the literature are quite interesting. For instance, environment exploration and obstacle avoidance problems for UAVs are solved via different RL methods with both continuous and discrete space action in [95]–[106]. In [107] the optimal deployment of UAVs that minimizes several parameters such as transmission power, caching, and the number of UAVs, is achieved through RL. Other works tested RL for assisting a UAV in a landing operation [108]–[111]. In [112] an anomaly detection is performed via RL in order to detect abnormality in the functioning of the motor and launch the landing procedure immediately.

All the previously discussed works only focus on a specific type of RL application which is path planning. The Table 2 summarizes all works in the literature up to the date of writing this survey. The cited works are oriented towards trajectory planning solutions for UAV-based networks. Most of the cited works are recent and are classified according to the application's main objective and to the different parameters used. We insist on the fact that the classification of dimensionality in 3D and 2D works is made on the basis of the movement of the UAV and not on the nature of the environment, in other words, a work where a 3D environment is considered and where the UAV flies at a fixed altitude is classified as a 2D work. In addition, the classification according to battery level parameters is marked if the UAV's limited energy is considered as a constraint to the problem, so any work where the UAV's energy consumption is minimized without imposing a limit on the battery level will not be marked. To summarize, we have so far covered several research works for path planning

TABLE 2. Reinforcement learning solutions for path planning UAV-based problems.

Paper	RL technique	Application Objective	3D/2D	Single/Multiple UAVs	Wireless communication based parameters	Obstacles	LOS/NLOS	Battery	UAV dynamics	Users movement
					Problem Parameters					
Discrete action space (Classical RL)										
[82]	Q-Learning based Texplore	Reaching a target	2D	Single	✗	✗	✗	✓	✓	✗
[114]	DRL ESN-Based	Multiple objective optimization	2D	Multiple	Interference/ Wireless latency/ Transmit power	✗	✗	✗	✗	✗
[83]	K-means+Q-learning	Ground coverage	3D	Multiple	QoE at the users	✗	✗	✗	✗	✓
[85]	Q-learning+ESN+GAK-means/ Power control	Ground coverage	3D	Multiple	Information rate and transmit power	✗	✓	✗	✗	✓
[115]	Q-learning vs NN-based Q-learning	Ground coverage	2D	Single	Transmission rate	Single	✓	✗	✗	✗
[116]	Q-learning	Ground coverage	2D	Single	QoE at the users	Single	✗	✓	✗	✗
[97]	Q-learning	Obstacle avoidance	2D	Single	✗	Multiple	✗	✗	✗	✗
[86]	Double Q-learning	Ground coverage	2D	Multiple	User satisfaction	✗	✓	✗	✗	✗
[117]	Q-learning	Ground coverage	3D	Multiple	Throughput/Interference	Multiple	✓	✗	✗	✓
[118]	S-MGD	Ground coverage	3D	Multiple	Downlink	✗	✓	✓	✗	✓
[119]	Enhanced Multi agent Q-learning	Sensing task	3D	Multiple	✗	✗	✓	✗	✗	✗
[120]	Multi agent RL (MAXQ)	Collision avoidance	3D	Multiple	✗	Multiple	✗	✗	✓	✗
[121]	Q-learning	Maximize ground user throughput	2D	Single	NOMA throughput	Multiple	✓	✗	✗	✓
[122]	Q-learning	Collision avoidance	2D	Multiple	Path loss/ SNR	Multiple	✗	✗	✗	✗
[123]	enhanced multi-agent Q-learning	Target localization	2D	Multiple	Channel modeling	✗	✓	✓	✓	✗
[124]	Sarsa vs Q-learning	Data collection	2D	Single	Channel gain/ Data rate	✗	✗	✗	✗	✗
Discrete action space (DRL)										
[125]	DDQN	Data harvesting	2D	Multiple	SNR/Information rate/ Throughput	Multiple	✓	✓	✓	✓
[126]	DQN	Reaching a target	3D	Single	✗	Multiple	✗	✗	✗	✗
[127]	DQN	Ground coverage	3D	Single	Spectral efficiency	✗	✓	✗	✗	✗
[87]	DQN	Ground coverage	2D	Multiple	MIMO technique/ interference	✗	✗	✗	✗	✗
[128]	Multi agent DQN	QoS at users	3D	Multiple	SNR/Throughput	✗	✓	✗	✗	✗
[129]	Q-learning+DQN	Ground coverage	2D	Both	Path loss/ Interference/ Coverage and fairness score	✗	✓	✓	✗	✗
[130]	DQN	Data collection	2D	Single	Path loss/ SNR	✗	✗	✗	✗	✗
[131]	DQN	Combat mission	3D	Single	✗	✗	✗	✗	✓	✗
[132]	Decaying DQN	multi-objective optimization	2D	Single	NOMA channel modeling/Bandwidth/SINR	✗	✓	✗	✗	✗
[133]	Mutual DQN	Cellular offloading	3D	Multiple	NOMA/Data rate/Interference	✗	✓	✗	✗	✓
[134]	DQN	Reaching Target	3D	Single	✗	✗	✗	✗	✓	✗
[135]	DQN	Pursuer-evader problem	3D	Single	✗	Multiple	✗	✗	✓	✗
[136]	Double DQN	Cellular offloading	2D	Single	✗	✗	✗	✓	✗	✓
Continuous action space (DRL)										
[137]	DDPG	Ground target tracking	2D	Single	✗	Multiple	✓	✗	✓	✗
[93]	DDPG	Reaching a target	3D	Single	✗	Multiple	✗	✗	✓	✗
[88]	DDPG (DRL-EC)	Multi-objective optimization	2D	Multiple	Coverage score/Energy	✗	✗	✗	✗	✗
[92]	Fast-RDPG	Obstacle avoidance	3D	Single	✗	Multiple	✗	✗	✓	✗
[138]	DRL (LwH)	Reaching a goal	2D	Single	✗	Multiple	✗	✗	✓	✗
[139]	Compound action Actor-Critic (CA2C)	Cooperative sensing	2D	Multiple	Transmit power/ SNR/ Data rate	✗	✓	✗	✗	✗
[140]	Multi agent DDPG	Fair ground coverage	2D	Multiple	Data rate	✗	✗	✗	✗	✗
[141]	DRL (Actor-Critic)	Long-term coverage	2D	Multiple	Coverage fairness score	✗	✗	✗	✗	✗
[142]	DDPG	Network utility	3D	Single	SINR / Path loss	✗	✓	✗	✓	✗
[143]	DDPG	Obstacle avoidance	3D	Single	✗	Multiple	✗	✗	✓	✗
[144]	DDPG (EEFC-TDBA)	Ground coverage	3D	Single	Bandwidth/Data rate/Throughput	✗	✓	✓	✓	✗
[145]	DDPG	Ground coverage	2D	Multiple	Data rate/ Bandwidth	✗	✗	✓	✗	✗
[146]	jPPO+ConvNTM	Data collection	2D	Multiple	✗	✓	✗	✓	✗	✗

applications. Obviously, the different approaches used in this regard could be classified into discrete and continuous methods. Q-learning-based solutions use a computationally expensive process called discretization, as the action space and the state space are finite. This is why policy gradient methods seem more realistic since the action and state space are continuous, which leads to a wider range of possible actions. It is important to point out that policy gradient
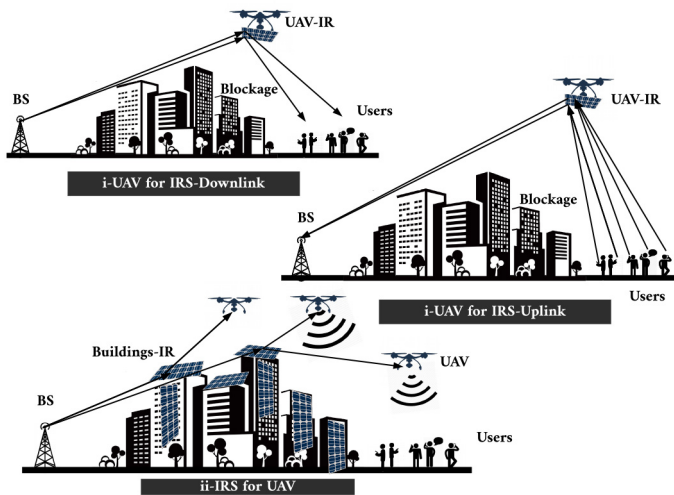
FIGURE 7. UAV-IR use case scenarios.

methods do not always perform better than Q-learning-based methods, instead, they have their drawbacks such as low convergence and the high variance problem in estimating the reward expectation. For instance, the work [113] investigates some failure problems with DDPG in deterministic environments with sparse reward. In the coming section, we will cover more interesting potential applications of RL such as event scheduling and resource allocation.

2) RL FOR IRS-ENABLED UAV NETWORKS

In this section, we cover a promising technology for 6G networks through the use of intelligent reflective surfaces (IRS). This next-generation technology consists of using a set of passive reflective elements to control the wireless propagation and direct the signal to a target direction. In recent literature, some works have considered equipping UAVs with IRSs to increase spectrum efficiency. Other works have investigated the use of the IRS to reflect the signal towards UAVs that are acting as flying base stations. In Fig. 7, we visualize these two different scenarios where the IRSs are coupled with UAVs: (i) UAVs for IRSs, where the UAVs carrying the reflective surfaces can act as a passive relay in both uplink and downlink communications between terrestrial BSs and ground users and (ii) IRSs for UAVs, where the IRS-equipped buildings assist the UAV's communication. In what follows, we review the work done for these different scenarios, including support for mmWave communications.

In the context of supporting mmWave technology, the authors in [147] propose an efficient deployment for the IRS-equipped UAV to serve a moving ground user that does not have a LOS with the terrestrial base station. In addition to reflecting the mmWave signals, IRS is also used to harvest energy at the UAV and Deep Q network is used to set the UAV location and the reflecting parameters of the IRS so that the downlink capacity is maximized. The same authors in [148] consider overcoming the blockage induced by the buildings through IRS-equipped UAV but for multiple users.

To this end, distributional RL is used to optimize the UAV's location, the precoding matrix at the base station, and the reflection coefficients.

Unlike [147], the work in [132] does not take into account energy harvesting, but rather focuses on minimizing the UAV energy consumption. The authors study the improvement of the quality of service of a UAV-carried intelligent reflector (UAV-IR) by optimizing its location, the phase shift of the IRS and, its power allocation to mobile users on the ground. To further improve the efficiency of downlink communications, a non-orthogonal multiple access technique is used. Decaying DQN is used to dynamically adjust the position of the UAV and the algorithm is found to converge and avoid oscillations compared to the classical Q-learning algorithm. Although D-DQN has proven its efficiency compared to the classical Q-learning, it still interesting to see some comparison with other DQN-based algorithms.

From an uplink perspective, the authors of [149] study the use of the UAV-IR as a passive relay for the transmission of IoT devices to a terrestrial base station. DRL is used to optimize the UAV location, the IRS phase shift, and the transmission scheduling in order to minimize the average age of the information.

Unlike the above works, where IRS-equipped UAVs are studied, in [150], improving the channel condition between a UAV and a set of ground users is considered by assisting the UAV with IRSs that are mounted on top of the surrounding buildings. In this configuration, several parameters are optimized, such as data throughput, UAV trajectory, and IRS phase shifts. In terms of the RL algorithm used, the authors adopt two different types of solutions, a discrete action space solution based on DQN and a continuous action space solution using the DDPG algorithm.

3) RL FOR SCHEDULING AND RESOURCE MANAGEMENT

Beyond path planning for smart UAVs, one can think about smart event scheduling for a drone network. In this context, the authors in [151] propose a spatio-temporal scheduling framework for autonomous UAVs. The proposed RL solution is model-free based and uses the popular Q-learning algorithm. The algorithm is handling the unexpected events iteratively by checking at every time slot their existence and updating the UAVs schedule accordingly. After that, the trajectory of the UAV is updated according to the Q-learning strategy. There are multiple parameters taken into account for every event (e.g., starting time, processing time, location, priority). The considered work is interesting for many reasons; it takes into account multiple factors such as dealing with unexpected events efficiently, it also considers the battery level, and works within a cooperative UAV environment. However, it is still not clear how to select some parameters optimally. For instance, the time discretization parameter enables a trade-off between complexity and time efficiency. In other words, deciding in an optimal way the next event will inevitably result in an increased time processing. This

will certainly affect the coverage rate of the UAV badly. Moreover, the author could have considered a more realistic case where multiple docking stations are available instead of considering only one station for the whole network, and therefore the UAV should always consider moving to the nearest station if needed.

In [152] a UAV network is managed by ensuring the UAV connectivity given the available bandwidth and energy. The set of drones are charged by a wind-powered station that enables a green wireless power transfer. The number of UAVs authorized to take off is managed through classical RL by solving a system of Bellman optimality equations in order to extract the optimal policy. The authors focused on the physical implementation of the charging station and the drone receiving pads by going through the different technical details of the wireless power transfer system. Among the assumptions made throughout the work is the fact that the charging time is constant, which could be hampered by several factors. First of all, while establishing wireless power transfer, the UAV could face several problems such as losing LOS connection with the station or some misalignment issues. Secondly, the fact that the charging station uses wind power makes it subject to variability in the harvested power. To solve the latter problem, the authors propose using adaptive current control.

The authors in [153] investigate the dynamic management of information exchange in a UAV network by studying two different information sharing schemes: the dynamic channel model and the dynamic time slot model. In the first scheme, the channel is shared for the exchange of information in the same time slot, while in the second model, time slots are shared according to the priority level. To solve the problem of dynamic management in the two proposed schemes, the authors used a DQN network coupled with a Long Short Term Memory network (LSTM). The purpose of using the LSTM network is to speed up the convergence of the DQN model since this network helps to predict future states of the environment.

In [154], UAVs are considered as remote edge computing systems that could offload ground users when they are excessively overwhelmed with computation. Hence, the authors studied the best way to assign a UAV to execute a given task in order to minimize the overall mission time. They formulated this resource allocation problem as a Markov decision process and proposed an Actor-Critic based RL technique (A2C) as a solution. In addition to offloading terrestrial users through UAVs, it is possible to perform the inverse and thus offloading the UAVs through Mobile Edge Computing(MEC) server. In this context, the work in [155] investigates offloading the UAV network via MEC servers to minimize the processing time of the drones and save their energy. The complex task of associating each UAV with its corresponding task based on its available energy, and of choosing the optimal MEC server to offload, is carried out on the basis of two Q-learning models. In the same direction, authors in [156] investigate offloading a network of UAVs

via MEC where a central control system has the ability to turn the UAV computing units on and off and to decide whether a given task should be performed by another UAV. This management problem is solved through RL techniques.

Resource allocation is another potential problem that can be solved with RL. The work in [157] is among the few works that go beyond UAV deployment or trajectory design, instead, it focuses on resource allocation for a network of multiple UAVs that communicates with ground users. The solution provided is based on Multi Agent Reinforcement Learning (MARL) and the problem formulation is based on stochastic game theory. The authors investigate sub-channel selection, user selection, and also power allocation for each user. Several parameters are taken into account such as the SINR, LOS and NLOS conditions with the users. The work described is of great importance, especially when considering the scarcity of publications in this particular application. Some other works investigated resource management for UAVs such as in [158] where the number of handoffs that occur when a ground user does not receive its minimal signal from its serving UAV is minimized. The solution used is based on RL. More specifically, it uses the Upper Confidence Bound (UCB) algorithm. In [159] UAV cluster task scheduling problem is addressed by solving the channel allocation problem using DRL. Moreover, the authors in paper [160] propose a comparison between Swarm intelligence and the Q-learning algorithm of RL. The objective is to enable autonomous swarm coordination for a network of high-altitude platform stations.

D. DISCUSSIONS AND FUTURE WORKS
1) RL LIMITATIONS

Unlike supervised learning, RL is the area of ML that does not use the power of data to learn a task. Instead, RL uses the so-called "trial and error" methodology based on an agent's past experiences. Surprisingly, this fact makes RL an extremely potent tool, especially for UAV-based problems such as path planning, resource management, and scheduling, where data is sometimes elusive. On the other hand, RL might resemble supervised learning in one single point which is the goal of achieving full autonomy within a UAV network by equipping UAVs with the ability to autonomously make decisions in a real-time situation. Generally speaking, RL has proven its efficiency by excelling in many tasks and games, such as beating the world's top chess grandmasters. However, many still doubt the applicability of RL in real-world problems, especially for autonomous flying or autonomous driving tasks. Not only is it difficult to have a perfect perception of the environment from the agent's perspective, but it is also extremely difficult to perform the exploration/exploitation dilemma explained earlier in Section III-B. To be more specific, let's take the example of planning a path for a UAV to reach a target in a fully autonomous control mode. Then, in order to apply RL, the UAV, which is the agent technically speaking, needs to perform exploration to discover its surroundings and learn how
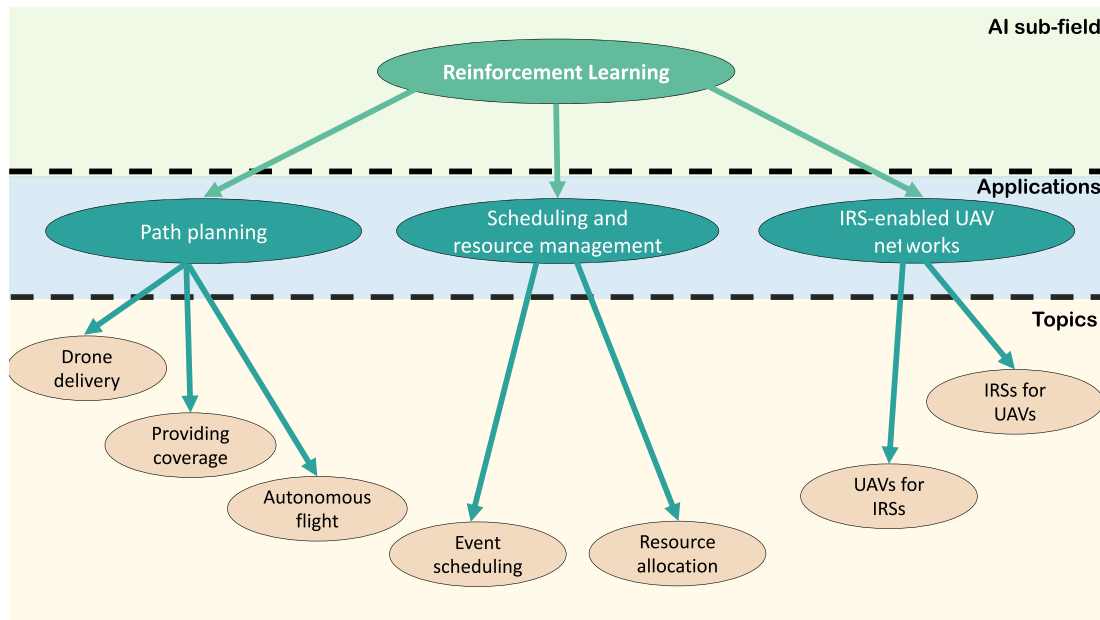
FIGURE 8. RL taxonomy.

to react. But this is almost impossible, especially for a high-varying environment. In other words, the exploration task is somehow impeded by the complexity of the environment and the cost of a UAV crash.

In terms of regulations, and as mentioned in Section II, many countries do not allow the use of autonomous UAVs. For example, the use of UAVs for delivery has been completely excluded in the latest FAA regulations, as the new rules require that the UAV must always be in the operator's field of sight, which is by definition in contradiction with UAV delivery applications. Hence, such rules can impede the progress done so far in RL for different UAV applications.

Interested readers are referred to [161] for more information on practical challenges that may arise in the real-world application of the RL, such as system delays, cases with limited samples, and more. On the bright side, several large companies and research labs have been working on producing alternatives to RL such as Evolution Strategies (ES) proposed by OpenAI [162]. To sum up, even if RL is not the ultimate solution for all UAV-based problems, it should be for some of them, which is confirmed by the numerical results of many papers covered so far in this survey. In what follows, we present our observations and criticisms of the current state of the art.

2) LITERATURE DISCUSSIONS

Based on the recent literature related to RL for UAV- related problems, we would like to offer the following observations. One can easily notice that the big majority of the published works are focusing on path planning applications for UAVs. More specifically, we remarked that the majority of papers tend to use a Q-learning approach to propose autonomous

path planning for the UAVs. Although Q-learning is a classical algorithm and an interesting way to start solving such problems, it is somewhat impeded by the need for full knowledge of the map which is not trivial in reality, especially when considering a high movement speed of UAVs. Added to that the fact that Q-learning might be slow if optimality is needed. Consequently, a trade-off between both complexity and optimality must be carefully studied. To sum up, DRL techniques such as Q-learning neural networks and DDPG, are more promising in terms of path planning and should gain more interest in the future.

In addition, we also noted that most research contributions use a discrete approach for path planning problems. While solutions with a discrete set of actions and states is a classic approach to address RL problems, they do not reflect a real situation where actions could be infinite as in real-world trajectory planning. Although solutions with a continuous state/action space are more difficult, solving them can only bring significant benefits to the area.

Furthermore, we noticed that most of the existing works are focusing on traditional centralized approaches for RL solutions which raises several challenges related to complexity and time management. That is why we strongly believe that distributed RL is an interesting technique to solve UAVs' real-time applications such as the distributed Q-learning algorithm. This type of RL technique is well suited for UAV networks where multi agents are subject to collaborative decisions.

Besides that, other potential applications such as resource allocation and event scheduling are not well covered by the literature which has created an unbalanced research content oriented towards path planning problems. The actual works looking into these topics are quite a few, and hence future
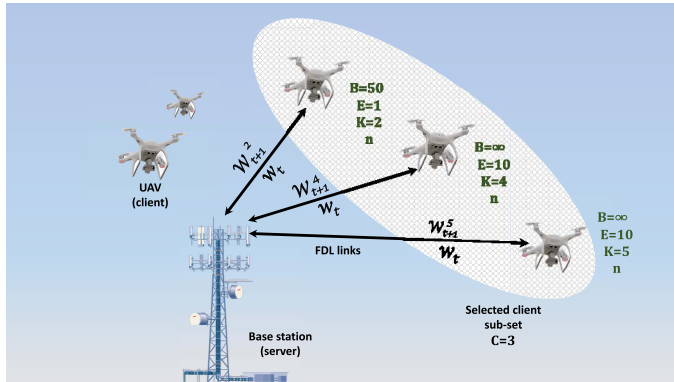
FIGURE 9. Federated learning principle.

works can be directed towards applying other RL-based approaches to solve these problems.

IV. FEDERATED LEARNING FOR UAVS

So far, we have covered a large number of techniques that could contribute to the development of intelligent UAV networks, ranging from supervised learning to unsupervised learning to RL. However, some of the algorithms covered previously do not go hand in hand with some UAV constraints. Specifically, we highlight the limited computing capacity on-board. Consequently, we question the applicability of AI in UAV networks in a practical situation. In response to this last question, Google has recently implemented what is called FL, envisioning a practical way to implement ML algorithms in constrained networks [163], [164]. FL is based on executing ML algorithms in a decentralized manner without the need to download the training set to a central node (or server). It is not specifically designed for a UAV network, but for any type of network composed of a central server (e.g., a base station in our configuration) and a number of clients (e.g., UAVs, mobile users).

A. FL PRINCIPLE

Without loss of generality, we provide a comprehensive explanation for FL algorithm for a setup of a network of UAVs that are served by a terrestrial base station. As a typical task, we suppose that the UAVs are processing different ground images. We also assume that the optimization of the loss function is done through a simple stochastic gradient descent (SGD) algorithm. As illustrated in Fig. 9, the central server, which is the base station in our case, shares the current update of the global model, denoted by w_t, with a sub-set of the users. The subset size denoted by C, is randomly selected by the server. Once the client UAV receives the current update of the global model, it uses its local training data to compute a local update of the global model. We should mention that several parameters are related to each UAV as indicated in Fig. 9. Those parameters are the mini-batch size denoted by B which indicates the amount of the local data used per each UAV, the index

k of the UAV, and the number of training passes each client makes over its local dataset on each round, which is denoted by E. After performing the update, the UAV only communicates the update, denoted by w_{t+1}^k, to the base station. For an SGD-based optimization, the update is calculated as follows:

$$w_{t+1}^k = w_t - \eta \nabla l(w_t, B), \qquad (2)$$

where η is the learning rate and l is the loss function.

For example, the UAV ($k = 4$) on Fig. 9 performs a full batch update and hence uses all its local data since $B = \infty$. Then it repeats the (2) ten times since $E = 10$ and delivers the output w_{t+1}^k to the base station. Once the local update w_{t+1}^k is received by the base station, it improves the global model and then removes these updates because they are no longer needed.

B. FL ADVANTAGES

At the ML discussion section, we have already mentioned FL as a promising solution for constrained networks where exhaustive calculation could not be done onboard. It allows decoupling the model training and the access to the raw data due to the fact that UAVs are not required to share any data with the server, instead, they only transmit their local update as explained previously. First and foremost, FL reduces privacy and security issues by minimizing data traffic over the network. As a result, it is considered a key solution for confidential systems where data does not need to be shared. We may consider a recommender system as an example of ML application where it is recommended that raw data will not be shared between the clients. In fact, some of the clients do not want others to reveal their preferences, therefore FL preserves this privacy by keeping the local data of each user private and sharing model updates only. Second, FL is suitable for applications where data is unbalanced. In other words, one client may be outside the region of interest and therefore have a small amount of data compared to other clients. Let's take the example of detecting a car using a UAV's camera, so even if one of the UAVs is misplaced in a given location where cars don't often pass by, that UAV will effectively detect a car when it is in the field of its camera. This is because the other UAVs communicating with the server have participated in the training of the misplaced drone. In addition, the learning process within the FL can be active even if one of the nodes is in the idle state, i.e., even if one of the UAVs has to perform charging, an emergency landing or encounters a connectivity failure, the learning process continues and the UAV can restore the updates when it reconnects to the network. Lastly, FL performs well on non-independent and identically distributed data, for example, the data partition observed by a single drone cannot be representative of the overall data of the system simply because the UAV can only visualize part of a given process.

C. FL SOLUTIONS FOR UAVS
1) FL FOR RESOURCE ALLOCATION AND SCHEDULING

In [165], an optimization problem is formulated to design joint power allocation and scheduling for a UAV swarm network. The considered network is composed of one leader UAV and a group of following UAVs. Each one of the following UAVs runs a FL algorithm on its local data and then sends the update to the leader drone. The leader aggregates all the local updates to perform a global update to the global model. While exchanging the updates between the UAVs, several natural phenomena will affect the wireless transmissions such as fading, wind, transmission delay, antenna gain and deviation, and interference. In the same mentioned work, the impact of these wireless transmission factors on the performance of the FL algorithm is analyzed. Finally, the effect of multiple wireless parameters on the convergence of the FL is studied.

Some of the work devoted to vehicle networks can be considered as a source of inspiration for UAV networks because there is a remarkable similarity between these two types of networks. But that's not all, the results of UAV networks are more promising because of the higher probability of establishing LOS links. In this context, the authors in [166], [167] investigate enabling ultra-reliable low latency communication for a network of vehicular users. In short, the authors propose a joint power and resource allocation framework for ultra-reliable and low-latency network of vehicles based on FL. Surprisingly, the FL-based method achieves the same performance with a centralized method but with a significant reduction of data transfer that reaches 79%. This fact shows how FL can enable privacy in the network with the same performance as classical algorithms.

In the same context, authors in [168] have investigated task and resource allocation for high-altitude balloon networks. It is worth mentioning that this type of network has a lot of similarities with a UAVs network as the HABs are operating as a wireless base station. The authors formulate an optimization problem for a mobile edge computing-enabled balloon network by minimizing energy and time consumption. However, to solve this problem, the user association with the balloons needs to be specified first. To solve the latter issue, a distributed SVM-based FL model is proposed to determine to which HAB every user connects. As usual, the FL technique will guarantee privacy by minimizing data sharing across the network.

To improve the efficiency of an Internet of vehicles network, the authors in [169] suggest the use of UAVs as relays in order to overcome the problem of communication failure while executing a FL task. To do so, the authors propose the formation of a coalition of UAVs in order to facilitate the training process by improving the communication efficiency level. Each UAV in a coalition will participate in the training in a sequential way, in other words, after completing the maximum number of iterations by the nearest UAV, the second nearest UAV takes over and continues

training the model and so on until all required iterations are done. A reward will be received to the UAV depending on the number of iterations performed. In the same context, an auction is designed for the UAVs to find the optimal allocation that maximizes the profits of the drones. So far, we covered the works that use FL to enable resource management within a UAV network, in what follows, we go over FL solutions for autonomous UAV navigation problems.

2) FL FOR UAV PATH CONTROL
The authors in [170] investigate the control of a massive number of UAVs starting from a source point and aiming to reach a destination point. The motion of the drones is perturbed by the wind which is the main source of randomness in the problem. This disturbance can lead to a fatal collision between the UAVs and, as a solution, trajectory control is proposed to avoid this scenario. The authors used the mean-field game framework to control the UAV path. However, within this framework, complex differential equations must be solved analytically, which is not possible for real-time applications and, in particular, for constrained networks. That is why approximated solutions are proposed based on two ANNs for each one of the two differential equations. At this level, even approximating the solution via DL is not enough for the convergence of the mean-field game framework. Thus, FL is used as a solution to share model parameters of the two NN between UAVs, and, as a result, UAVs will be able to take into account the effect of locally non-observable samples by a UAV for learning.

3) FL FOR FLYING AD-HOC NETWORK (FANET) SECURITY
FANET is a decentralized communication architecture composed of a group of UAVs where one of them at least is connected to ground base stations or a satellite. In recent years, a significant number of research works investigated the performance and the security level of such a setup. For instance, in [171], FANET security is studied in depth. This type of network is vulnerable to jamming attacks disrupting the communication at the receiver. To avoid such a scenario, the authors propose a FL-assisted solution for jamming attack detection. Many reasons stand behind selecting FL as a potential solution for FANET. First, FANETs are usually heterogeneous networks in terms of power consumption constraints and communication range. Secondly, the data available at the different nodes is unbalanced and, lastly, because the number of interacting nodes is huge. As we have already mentioned previously, FL performs well on this type of setup. Moreover, the FL technique is enhanced by a client selection algorithm based on Dempster-Shafer algorithm. This technique enables user group prioritization mechanism allowing selecting better clients for calculating the global update to the model. The numerical results, based on two different datasets, show that FL always outperforms distributed learning in many different setups. Furthermore, due to the different values of latency and bandwidth available

at each UAV, the client selection-based FL model outperformed the traditional FL algorithm. In another area of research, the authors of [172] have proposed a defense strategy to deal with jamming attacks on FANETs. The jamming detection strategy is based on federated RL, more precisely, on the Q-learning algorithm. In view of the numerical results, the authors proved that the proposed defense architecture, which combines RL and FL without a model, outperformed the distributed approach.

4) FL FOR CONTENT CACHING

To address one of the 5G drawbacks, which is the increased delay caused by the significant activity and congestion at the backhaul links, the 6G networks employ content caching at the small-cell base stations. Those small-cells can be in some scenarios UAVs acting as base stations. As a result, content caching is considered a promising alternative regarding the limited capacity of the UAVs in terms of computing capacity and memory. In this context, the work in [173] investigates an intelligent caching technique for a 6G heterogeneous aerial-terrestrial network composed of heterogeneous base stations such as UAVs and terrestrial remote radio heads. The proposed solution is based on FL techniques and hence users are no longer required to share explicitly their reporting and content preference. Instead, a heterogeneous computing platform (HCP), will accurately predict the content cached to the different base stations depending on mobile users' preferences. In the above-mentioned setup, the HCP plays the role of the server and the different nodes of the network will only share updates to the global model in a secure manner. A CNN was used so that the HCP learns the most popular files to cache in the heterogeneous base stations, and the optimization of the loss function is done via SGD as described previously in Section IV-A. The HCP-based FL solution was tested on different data-sets and proved its efficiency compared to other baseline methods.

5) FL FOR UAV SENSING

Authors in [174] investigate predicting air quality index throughout combining vision-based and sensor-based air quality sensing. Within the proposed framework, aerial sensing is performed by a network of UAVs deployed to learn from haze images, and ground sensing is performed via a wireless sensor network on the ground. DL models were used to provide inference from the terrestrial and aerial networks. The authors propose a visual model based on Dense-MobileNet CNN and a Spatio-temporal inference model based on LSTM networks and learning is established through the use of FL techniques. The accuracy of the proposed framework is evaluated on a real data set and is found to be higher than conventional approaches.

6) CLIENT SELECTION STRATEGIES FOR UAVS

Many published works related to FL are made on an optimistic assumption that all the clients will unconditionally participate in FL whenever they are called by the server.

However, it is obvious that deciding which nodes (UAVs) should participate in the learning is a sensitive task for FL that could influence the overall accuracy. Thus, in this section, we will cover some client selection technique and some participation strategies that could be of great importance to FL.

Authors in [175] propose a contract-matching solution based on which the UAV will get a reward according to its type. The contract proposed by the authors is multi-dimensional so that it takes into account the different sources of heterogeneity in the UAV types. After setting the contracts, a matching-based algorithm will assign the optimal UAVs to each region. The UAV parameters considered while designing the contracts are the sensing model, computation model, and transmission model. The proposed method enabled selecting the UAVs with the lowest costs to the target sub-region.

D. DISCUSSIONS AND FUTURE WORKS

We would like to emphasize the fact that FL is not necessarily applied only for UAV or mobile user networks, instead, it is being used successfully in many daily applications. For example, Google's Gboard implements the FL to learn a RNN to predict your next word when you start typing on the keyboard. However, we would like to point out that it is not clear how to select certain parameters in the FL algorithm as defined in [163]. For example, the client selection process has been defined as random, which raises the question of whether there is a better way to assign clients in each round of the FL algorithm. This last issue needs to be studied in depth for UAV networks where several parameters could affect the client selection process. From a wireless communication perspective, channel quality, LOS condition, available data, and battery state are crucial factors that could significantly affect the client selection process. To be specific, those parameters could make a subset of users more suitable to be selected for the FL training.

In addition, and in the meantime, while a large part of the scientific community asserts that the primary purpose of FL is data confidentiality, others doubt this assumption and argue that even sharing only updates over the wireless network is not safe. Unfortunately, the FL could be subject to a poisoning attack threatening the integrity of the model. These types of attacks are known in the ML community by backdoor attacks and are typically carried out either by a single node or by a group of nodes injecting poisoned data into the model to adversely influence it. More importantly, even FL remains vulnerable to this category of attacks not by poisoning the data but by poisoning the model itself by some malicious clients [176], [177]. In the same vein, as a futuristic solution to the unreliability of FL systems, we propose to support UAV networks using Blockchain techniques to increase the integrity of local models at each drone. The fusion of Blockchain and FL is considered a hot topic and a number of recent works have started to study this field [178]–[183]. It has been found that, in addition to

the increased level of stability and integrity, the Blockchain technique can increase the users motivation to participate in training by accurately rewarding them for their contribution, perhaps using cryptographic currency. Even for UAV-based networks, the research community has recently begun to apply the concept of a Blockchain coupled to FL to propose solutions for UAV networks. For example, the authors in [184] proposed a secure FL framework for a mobile crowdsensing application assisted by a UAV-network. The local exchanges of the FL algorithm were secured on the basis of a Blockchain architecture. In short, we highlight the potential of combining Blockchain and FL in future work.

In addition to security issues, more attention should be paid to the convergence of a FL algorithm which is not always guaranteed. Convergence depends on the specific type of problem, such as the convexity of the loss function and the number of updates performed on the model. For example, if there is a poor selection of clients where the selective nodes are not available or do not have enough data, the optimization of the overall model will fail. One can notice that this issue overlaps with the client selection problem mentioned previously, however, it is not only related to client selection but also to the type of the loss function.

As we proposed FL as a solution to running ML on the edge at the ML discussion section, we should mention that the massive exchange of updates across the network will result in huge communication loads in the training phase, especially for neural networks, which will induce a scalability problem for FL. Several CNN architectures require a massive number of parameters to be updated at each round, for instance, Alexnet contains 61 million parameters whereas VGG16 involves 138 million trainable parameters. Added to that the fact that UAV networks are usually characterized by a limited battery capacity and constrained bandwidth, which makes the UAVs unable to support all these communication loads. To face this issue, many researchers have been working on alternatives and solutions that could improve memory consumption and communication efficiency by proposing compression techniques and reducing the number of communication rounds [185]–[189].

Another drawback of FL arises when operating in a heterogeneous UAV network formed by various types of UAVs, rotary or fixed-wing, with different processing capabilities and different GPUs. These differences mean that some UAVs will have fast response times while others will experience severe delays. Therefore, since the FL algorithm is expected to receive the required model updates at each communication round, these induced delays will cause a major problem by considerably slowing down convergence. In some works, network nodes with large response delays are called "stragglers". In [190] a distributed computation scheme has been proposed to mitigate the impact of slow nodes on convergence for gradient methods. In addition, the quality of connectivity can affect the convergence of the FL algorithm since several network nodes may encounter an unexpected failure when transmitting their local updates.

These interruptions can also degrade the overall efficiency of the FL by slowing convergence.

To sum up, even with all the above-mentioned issues related to FL, it remains a good alternative for UAV-based networks. However, there are still some open problems that are worth investigation. For instance, the application of FL with supervised learning techniques, which were discussed in Section II-A, is still an open problem.

V. OPEN ISSUES SUMMARY

The use of AI for UAV networks has led to the introduction of several innovative and smart solutions for a wide range of problems. In what follows, we briefly summarize the most important open issues mentioned previously for UAV-based problems. Firstly, concerning the use of ML, we proposed (a) addressing lightweight ML algorithms for on-board running, (b) designing hybrid UAV detection solutions based on different types of data including sound, image, and radar, (c) working on expanding and unifying UAV drone regulations worldwide. Secondly, based on the unbalanced RL research content, we suggested (d) exploring other types of applications rather than path planning such as resource management and scheduling, (e) addressing realistic and complex trajectory planning problems with continuous action/state spaces, and (f) exploring DRL and distributed RL solutions for autonomous UAV networks instead of classical Q-learning-based methods. Lastly, in the FL section, we mentioned (g) addressing secure FL-based solutions, (h) incorporating Blockchain techniques with FL to increase the integrity level of each drone local model, (i) investigating client-selection strategies and convergence issues for UAV networks, and (j) providing communication-aware FL solutions taking into account possible congestions that could occur due to the need for excessive information exchange within dense UAV networks.

VI. CONCLUSION

Motivated by a wide set of new applications that can benefit from UAV networks, such as smart cities and aerial BS deployment, in this paper, we have explored in detail a new research direction where ML techniques are used to enhance the performance of UAV networks. We started by providing an extensive overview of unsupervised and supervised ML techniques that have been applied in UAV networks. Then, we briefly introduced the RL technique and discussed a number of the relevant works that implemented this ML technique for UAVs. Finally, we discussed a set of research works where FL techniques are used in the area of UAV networks. For each of the three considered techniques, we provided a set of concluding remarks that discuss the current limitations and challenges as well as a set of interesting open problems.

REFERENCES

[1] B. Li, Z. Fei, and Y. Zhang, "UAV communications for 5G and beyond: Recent advances and future trends," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2241–2263, Apr. 2019.

[2] Y. Zeng, Q. Wu, and R. Zhang, "Accessing from the sky: A tutorial on UAV communications for 5G and beyond," *Proc. IEEE*, vol. 107, no. 12, pp. 2327–2375, Dec. 2019.

[3] X. Cao, P. Yang, M. Alzenad, X. Xi, D. Wu, and H. Yanikomeroglu, "Airborne communication networks: A survey," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 1907–1926, Sep. 2018.

[4] M. Mozaffari, W. Saad, M. Bennis, Y.-H. Nam, and M. Debbah, "A tutorial on UAVs for wireless networks: Applications, challenges, and open problems," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2334–2360, 3rd Quart., 2019.

[5] G. Jiangchun, D. Guoru, X. Yitao, W. Haichao, and W. Qihui, "Proactive optimization of transmission power and 3D trajectory in UAV-assisted relay systems with mobile ground users," *Chin. J. Aeronaut.*, vol. 34, no. 3, pp. 129–144, 2021.

[6] G. Ding, Q. Wu, L. Zhang, Y. Lin, T. A. Tsiftsis, and Y.-D. Yao, "An amateur drone surveillance system based on the cognitive Internet of Things," *IEEE Commun. Mag.*, vol. 56, no. 1, pp. 29–35, Jan. 2018.

[7] B. Zhang, L. Tang, and M. Roemer, "Probabilistic weather forecasting analysis for unmanned aerial vehicle path planning," *J. Guid. Control Dyn.*, vol. 37, no. 1, pp. 309–312, 2014.

[8] A. Puri, *A Survey of Unmanned Aerial Vehicles (UAV) for Traffic Surveillance*, Dept. Comput. Sci. Eng., Univ. South Florida, Tampa, FL, USA, pp. 1–29, 2005.

[9] Q. Yang, L. Shi, and L. Lin, "Plot-scale rice grain yield estimation using UAV-based remotely sensed images via CNN with time-invariant deep features decomposition," in *Proc. IGARSS IEEE Int. Geosci. Remote Sens. Symp.*, 2019, pp. 7180–7183.

[10] G. T. Miyoshi *et al.*, "A novel deep learning method to identify single tree species in UAV-based hyperspectral images," *Remote Sens.*, vol. 12, no. 8, p. 1294, 2020.

[11] R. Zhang *et al.*, "Weed location and recognition based on UAV imaging and deep learning," *Int. J. Precision Agricult. Aviation*, vol. 3, no. 1, pp. 23–29, 2020.

[12] Q. Zhang, M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Machine learning for predictive on-demand deployment of UAVs for wireless communications," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, 2018, pp. 1–9.

[13] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless communications with unmanned aerial vehicles: Opportunities and challenges," *IEEE Commun. Mag.*, vol. 54, no. 5, pp. 36–42, May 2016.

[14] M. Lahmeri, M. A. Kishk, and M.-S. Alouini, "Stochastic geometry-based analysis of airborne base stations with laser-powered UAVs," *IEEE Commun. Lett.*, vol. 24, no. 1, pp. 173–177, Jan. 2020.

[15] O. M. Bushnaq, M. A. Kishk, A. Celik, M. S. Alouini, and T. Y. Al-Naffouri, "Optimal deployment of tethered drones for maximum cellular coverage in user clusters," *IEEE Trans. Wireless Commun.*, vol. 20, no. 3, pp. 2092–2108, Mar. 2021.

[16] Y. Qin, M. A. Kishk, and M. S. Alouini, "Performance evaluation of UAV-enabled cellular networks with battery-limited drones," *IEEE Commun. Lett.*, vol. 24, no. 12, pp. 2664–2668, Dec. 2020.

[17] M. A. Kishk, A. Bader, and M.-S. Alouini, "On the 3-D placement of airborne base stations using tethered UAVs," *IEEE Trans. Commun.*, vol. 68, no. 8, pp. 5202–5215, Aug. 2020.

[18] M. Kishk, A. Bader, and M. S. Alouini, "Aerial base station deployment in 6G cellular networks using tethered drones: The mobility and endurance tradeoff," *IEEE Veh. Technol. Mag.*, vol. 15, no. 4, pp. 103–111, Dec. 2020.

[19] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT press, 2018.

[20] 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.

[21] C. Goerzen, Z. Kong, and B. Mettler, "A survey of motion planning algorithms from the perspective of autonomous UAV guidance," *J. Intell. Robot. Syst.*, vol. 57, nos. 1–4, p. 65, 2010.

[22] A. Carrio, C. Sampedro, A. Rodriguez-Ramos, and P. Campoy, "A review of deep learning methods and applications for unmanned aerial vehicles," *J. Sensors*, vol. 2017, Aug. 2017, Art. no. 3296874.

[23] P. V. Klaine, R. D. Souza, L. Zhang, and M. Imran, "An overview of machine learning applied in wireless UAV networks," *5G Ref: The Essential 5G Reference Online*. Hoboken, NJ, USA: Wiley, 2019, pp. 1–15.

[24] P. S. Bithas, E. T. Michailidis, N. Nomikos, D. Vouyioukas, and A. G. Kanatas, "A survey on machine-learning techniques for UAV-based communications," *Sensors*, vol. 19, no. 23, p. 5170, 2019.

[25] G. Gan, C. Ma, and J. Wu, *Data Clustering: Theory, Algorithms, and Applications*. Philadelphia, PA, USA: SIAM, 2020.

[26] I. K. Fodor, "A survey of dimension reduction techniques," Lawrence Livermore Nat. Lab., Livermore, CA, USA, Rep. UCRL-ID-148494, 2002.

[27] J. Chen, U. Yatnalli, and D. Gesbert, "Learning radio maps for UAV-aided wireless networks: A segmented regression approach," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2017, pp. 1–6.

[28] Q. Zhang, W. Saad, M. Bennis, X. Lu, M. Debbah, and W. Zuo, "Predictive deployment of UAV base stations in wireless networks: Machine learning meets contract theory," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 637–652, Jan. 2021.

[29] J. Kwak and Y. Sung, "Autoencoder-based candidate waypoint generation method for autonomous flight of multi-unmanned aerial vehicles," *Adv. Mech. Eng.*, vol. 11, no. 6, pp. 1–16, 2019.

[30] K. Xiao, J. Zhao, Y. He, and S. Yu, "Trajectory prediction of UAV in smart city using recurrent neural networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2019, pp. 1–6.

[31] H. Peng, A. Razi, F. Afghah, and J. Ashdown, "A unified framework for joint mobility prediction and object profiling of drones in UAV networks," *J. Commun. Netw.*, vol. 20, no. 5, pp. 434–442, 2018.

[32] J. Jailton, T. Carvalho, J. Araújo, and R. Francês, "Relay positioning strategy for traffic data collection of multiple unmanned aerial vehicles using hybrid optimization systems: A FANET-based case study," *Wireless Commun. Mobile Comput.*, vol. 2017, Dec. 2017, Art. no. 2865482.

[33] Y. Zhang, J. Wen, G. Yang, Z. He, and X. Luo, "Air-to-air path loss prediction based on machine learning methods in urban environments," *Wireless Commun. Mobile Comput.*, vol. 2018, Jun. 2018, Art. no. 8489326.

[34] W. Xia *et al.*, "Generative neural network channel modeling for millimeter-wave UAV communication," [Online]. Available: arxiv.abs/2012.09133.

[35] Q. Zhang, A. Ferdowsi, W. Saad, and M. Bennis. (2021). *Distributed Conditional Generative Adversarial Networks (GANs) for Data-Driven Millimeter Wave Communications in UAV Networks*. [Online]. Available: https://arxiv.org/abs/2102.01751

[36] S. Alsamhi, O. Ma, and S. Ansari, "Predictive estimation of the optimal signal strength from unmanned aerial vehicle over Internet of Things using ANN," [Online]. Available: arxiv.abs/1805.07614.

[37] Y. Egi and C. E. Otero, "Machine-learning and 3D point-cloud based signal power path loss model for the deployment of wireless communication systems," *IEEE Access*, vol. 7, pp. 42507–42517, 2019.

[38] S. K. Goudos, G. V. Tsoulos, G. Athanasiadou, M. C. Batistatos, D. Zarbouti, and K. E. Psannis, "Artificial neural network optimal modeling and optimization of UAV measurements for mobile communications using the L-SHADE algorithm," *IEEE Trans. Antennas Propag.*, vol. 67, no. 6, pp. 4022–4031, Jun. 2019.

[39] P. Ladosz, H. Oh, G. Zheng, and W.-H. Chen, "A hybrid approach of learning and model-based channel prediction for communication relay UAVs in dynamic urban environments," *IEEE Robot. Autom. Lett.*, vol. 4, no. 3, pp. 2370–2377, Jul. 2019.

[40] S. K. Goudos and G. Athanasiadou, "Application of an ensemble method to UAV power modeling for cellular communications," *IEEE Antennas Wireless Propag. Lett.*, vol. 18, no. 11, pp. 2340–2344, Nov. 2019.

[41] J.-L. Wang, Y.-R. Li, A. B. Adege, L.-C. Wang, S.-S. Jeng, and J.-Y. Chen, "Machine learning based rapid 3D channel modeling for UAV communication networks," in *Proc. 16th IEEE Annu. Consum. Commun. Netw. Conf. (CCNC)*, 2019, pp. 1–5.

[42] R. D. Timoteo, D. C. Cunha, and G. D. Cavalcanti, "A proposal for path loss prediction in urban environments using support vector regression," in *Proc. Advanced Int. Conf. Telecommun*, 2014, pp. 1–5.

[43] M. Chen, W. Saad, and C. Yin, "Deep learning for 360 content transmission in UAV-enabled virtual reality," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2019, pp. 1–6.

[44] S. Wang, J. Chen, Z. Zhang, G. Wang, Y. Tan, and Y. Zheng, "Construction of a virtual reality platform for UAV deep learning," in *Proc. Chin. Autom. Congr. (CAC)*, 2017, pp. 3912–3916.

[45] J. Chakareski, "UAV-IoT for next generation virtual reality," *IEEE Trans. Image Process.*, vol. 28, no. 12, pp. 5977–5990, Dec. 2019.

[46] I. Bozcan and E. Kayacan, "UAV-AdNet: Unsupervised anomaly detection using deep neural networks for aerial surveillance," 2020. [Online]. Available: arxiv.abs/2011.02853.

[47] C. Titouna, F. Nait-Abdesselam, and H. Moungla, "An online anomaly detection approach for unmanned aerial vehicles," in *Proc. Int. Wireless Commun. Mobile Comput. (IWCMC)*, pp. 469–474.

[48] D. Yong, Z. Yuanpeng, X. Yaqing, P. Yu, and L. Datong, "Unmanned aerial vehicle sensor data anomaly detection using kernel principle component analysis," in *Proc. 13th IEEE Int. Conf. Electron. Meas. Instrum. (ICEMI)*, 2017, pp. 241–246.

[49] S. Khan, C. F. Liew, T. Yairi, and R. McWilliam, "Unsupervised anomaly detection in unmanned aerial vehicles," *Appl. Soft Comput.*, vol. 83, Oct. 2019, Art. no. 105650.

[50] S. Wen *et al.*, "Single-rotor UAV flow field simulation using generative adversarial networks," *Comput. Electron. Agricult.*, vol. 167, Dec. 2019, Art. no. 105004.

[51] J. Kim, C. Park, J. Ahn, Y. Ko, J. Park, and J. C. Gallagher, "Real-time UAV sound detection and analysis system," in *Proc. IEEE Sensors Appl. Symp. (SAS)*, 2017, pp. 1–5.

[52] C. Dumitrescu, M. Minea, I. M. Costea, I. Cosmin Chiva, and A. Semenescu, "Development of an acoustic system for UAV detection," *Sensors*, vol. 20, no. 17, p. 4870, 2020.

[53] B. Yang, E. T. Matson, A. H. Smith, J. E. Dietz, and J. C. Gallagher, "UAV detection system with multiple acoustic nodes using machine learning models," in *Proc. IEEE Int. Conf. Robot. Comput. (IRC)*, 2019, pp. 493–498.

[54] S. Jeon, J.-W. Shin, Y.-J. Lee, W.-H. Kim, Y. Kwon, and H.-Y. Yang, "Empirical study of drone sound detection in real-life environment with deep neural networks," in *Proc. IEEE Eur. Signal Process. Conf. (EUSIPCO)*, 2017, pp. 1858–1862.

[55] S. Li *et al.*, "Convolutional neural networks for analyzing unmanned aerial vehicles sound," in *Proc. ICCAS*, 2018, pp. 862–866.

[56] A. Alipour-Fanid, M. Dabaghchian, N. Wang, P. Wang, L. Zhao, and K. Zeng, "Machine learning-based delay-aware UAV detection over encrypted Wi-Fi traffic," in *Proc. IEEE Conf. Commun. Netw. Security (CNS)*, 2019, pp. 1–7.

[57] R. Amorim, J. Wigard, H. Nguyen, I. Z. Kovacs, and P. Mogensen, "Machine-learning identification of airborne UAV-UEs based on LTE radio measurements," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, 2017, pp. 1–6.

[58] H. Zhang, C. Cao, L. Xu, and T. A. Gulliver, "A UAV detection algorithm based on an artificial neural network," *IEEE Access*, vol. 6, pp. 24720–24728, 2018.

[59] M. Ezuma, F. Erden, C. K. Anjinappa, O. Ozdemir, and I. Guvenc, "Micro-UAV detection and classification from RF fingerprints using machine learning techniques," in *Proc. IEEE Aerosp. Conf.*, 2019, pp. 1–13.

[60] M. F. Al-Sad, A. Al-Ali, A. Mohamed, T. Khattab, and A. Erbad, "RF-based drone detection and identification using deep learning approaches: An initiative towards a large open source drone database," *Future Gener. Comput. Syst.*, vol. 100, pp. 86–97, Nov. 2019.

[61] J. Park, D. H. Kim, Y. S. Shin, and S.-H. Lee, "A comparison of convolutional object detectors for real-time drone tracking using a PTZ camera," in *Proc. IEEE Int. Conf. Control Autom. Syst. (ICCAS)*, 2017, pp. 696–699.

[62] L. I. Ivanov, N. A. Obukhova, and P. S. Baranov, "Review of modern UAV detection algorithms using methods of computer vision," in *Proc. IEEE Conf. Russian Young Res. Elect. Electron. Eng. (EIConRus)*, 2020, pp. 322–325.

[63] Y. Xu, G. Yu, Y. Wang, X. Wu, and Y. Ma, "Car detection from low-altitude UAV imagery with the faster R-CNN," *J. Adv. Transp.*, vol. 2017, Aug. 2017, Art. no. 2823617.

[64] N. Attari, F. Ofli, M. Awad, J. Lucas, and S. Chawla, "Nazr-CNN: Fine-grained classification of UAV imagery for damage assessment," in *Proc. IEEE Int. Conf. Data Sci. Adv. Anal. (DSAA)*, 2017, pp. 50–59.

[65] Y. Bazi and F. Melgani, "Convolutional SVM networks for object detection in UAV imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 6, pp. 3107–3118, Jun. 2018.

[66] M. Lan, Y. Zhang, L. Zhang, and B. Du, "Defect detection from UAV images based on region-based CNNs," in *Proc. IEEE Int. Conf. Data Min. Workshops (ICDMW)*, 2018, pp. 385–390.

[67] T. Liu and T. Stathaki, "Faster R-CNN for robust pedestrian detection using semantic segmentation network," *Frontiers Neurorobot.*, vol. 12, p. 64, 2018. [Online]. Available: https://www.frontiersin.org/article/10.3389/fnbot.2018.00064

[68] A. Nassar, K. Amer, R. ElHakim, and M. ElHelw, "A deep CNN-based framework for enhanced aerial imagery registration with applications to UAV geolocalization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2018, pp. 1513–1523.

[69] C. Kyrkou, G. Plastiras, T. Theocharides, S. I. Venieris, and C.-S. Bouganis, "DroNet: Efficient convolutional neural network detector for real-time UAV applications," in *Proc. IEEE Design Autom. Test Europe Conf. Exhibit. (DATE)*, 2018, pp. 967–972.

[70] J. Vazquez-Nicolas, E. Zamora, I. Gonzalez-Hernandez, R. Lozano, and H. Sossa, "Towards automatic inspection: Crack recognition based on quadrotor UAV-taken images," in *Proc. IEEE Int. Conf. Unmanned Aircraft Syst. (ICUAS)*, 2018, pp. 654–659.

[71] A. A. d. Santos *et al.*, "Assessment of CNN-based methods for individual tree detection on images captured by RGB cameras attached to UAVs," *Sensors*, vol. 19, no. 16, p. 3595, 2019.

[72] H. A. Al-Najjar *et al.*, "Land cover classification from fused DSM and UAV images using convolutional neural networks," *Remote Sens.*, vol. 11, no. 12, p. 1461, 2019.

[73] E. C. Tetila *et al.*, "Detection and classification of soybean pests using deep learning with UAV images," *Comput. Electron. Agr.*, vol. 179, 2020, Art. no. 105836.

[74] J. Xiong *et al.*, "Visual detection of green mangoes by an unmanned aerial vehicle in orchards based on a deep learning method," *Biosyst. Eng.*, vol. 194, pp. 261–272, Jun. 2020.

[75] Y. Liu, C. Cen, Y. Che, R. Ke, Y. Ma, and Y. Ma, "Detection of maize tassels from UAV RGB imagery with faster R-CNN," *Remote Sens.*, vol. 12, no. 2, p. 338, 2020.

[76] L. P. Osco *et al.*, "A convolutional neural network approach for counting and geolocating citrus-trees in UAV multispectral imagery," *ISPRS J. Photogrammetry Remote Sens.*, vol. 160, pp. 97–106, Feb. 2020.

[77] K. Mukadam, A. Sinh, and R. Karani, "Detection of landing areas for unmanned aerial vehicles," in *Proc. IEEE Int. Conf. Comput. Commun. Control Autom. (ICCUBEA)*, 2020, pp. 1–5.

[78] S. Dhar, J. Guo, J. Liu, S. Tripathi, U. Kurup, and M. Shah, "On-device machine learning: An algorithms and learning theory perspective," [Online]. Available: arxiv.abs/1911.00623.

[79] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[80] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," in *Proc. ICML*, 2014, pp. 387–395.

[81] M. Riedmiller, "Neural fitted Q iteration—First experiences with a data efficient neural reinforcement learning method," in *Proc. Eur. Conf. Mach. Learn.*, 2005, pp. 317–328.

[82] N. Imanberdiyev, C. Fu, E. Kayacan, and I.-M. Chen, "Autonomous navigation of UAV by using real-time model-based reinforcement learning," in *Proc. IEEE 14th Int. Conf. Control Autom. Robot. Vis. (ICARCV)*, 2016, pp. 1–6.

[83] X. Liu, Y. Liu, and Y. Chen, "Reinforcement learning in multiple-UAV networks: Deployment and movement design," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 8036–8049, Aug. 2019.

[84] H. X. Pham, H. M. La, D. Feil-Seifer, and A. Nefian, "Cooperative and distributed reinforcement learning of drones for field coverage," [Online]. Available: arxiv.abs/1803.07250.

[85] X. Liu, Y. Liu, Y. Chen, and L. Hanzo, "Trajectory design and power control for multi-UAV assisted wireless networks: A machine learning approach," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 7957–7969, Aug. 2019.

[86] X. Liu, M. Chen, and C. Yin, "Optimized trajectory design in UAV based cellular networks for 3D users: A double Q-learning approach," *J. Commun. Inf. Netw.*, vol. 4, no. 1, pp. 24–32, Mar. 2019.

[87] H. Huang, Y. Yang, H. Wang, Z. Ding, H. Sari, and F. Adachi, "Deep reinforcement learning for UAV navigation through massive MIMO technique," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 1117–1121, Jan. 2020.

[88] C. H. Liu, Z. Chen, J. Tang, J. Xu, and C. Piao, "Energy-efficient UAV control for effective and fair communication coverage: A deep reinforcement learning approach," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 2059–2070, Sep. 2018.

[89] G. Muñoz, C. Barrado, E. Çetin, and E. Salami, "Deep reinforcement learning for drone delivery," *Drones*, vol. 3, no. 3, p. 72, 2019.

[90] K. Kersandt, G. Munoz, and C. Barrado, "Self-training by reinforcement learning for full-autonomous drones of the future," in *Proc. IEEE/AIAA 37th Digit. Avion. Syst. Conf. (DASC)*, 2018, pp. 1–10.

[91] E. Camci and E. Kayacan, "Waitress quadcopter explores how to serve drinks by reinforcement learning," in *Proc. IEEE Region Conf. (TENCON)*, 2016, pp. 28–32.

[92] C. Wang, J. Wang, Y. Shen, and X. Zhang, "Autonomous navigation of UAVs in large-scale complex environments: A deep reinforcement learning approach," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2124–2136, Mar. 2019.

[93] O. Bouhamed, H. Ghazzai, H. Besbes, and Y. Massoud, "Autonomous UAV navigation: A DDPG-based deep reinforcement learning approach," [Online]. Available: arxiv.abs/2003.10923.

[94] M. Zhu, X.-Y. Liu, and X. Wang, "Deep reinforcement learning for unmanned aerial vehicle-assisted vehicular networks," [Online]. Available: arxiv.abs/1906.05015.

[95] S.-Y. Shin, Y.-W. Kang, and Y.-G. Kim, "Obstacle avoidance drone by deep reinforcement learning and its racing with human pilot," *Appl. Sci.*, vol. 9, no. 24, p. 5571, 2019.

[96] A. Singla, S. Padakandla, and S. Bhatnagar, "Memory-based deep reinforcement learning for obstacle avoidance in UAV with limited environment knowledge," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 1, pp. 107–118, Jan. 2021.

[97] I. Kim, S. Shin, J. Wu, S.-D. Kim, and C.-G. Kim, "Obstacle avoidance path planning for UAV using reinforcement learning under simulated environment," in *Proc. 3rd Int. Conf. Electron. Elect. Eng. Comput. Sci. (IASER)*, 2017, pp. 34–36.

[98] S.-Y. Shin, Y.-W. Kang, and Y.-G. Kim, "Reward-driven U-net training for obstacle avoidance drone," *Exp. Syst. Appl.*, vol. 143, Apr. 2020, Art. no. 113064.

[99] Z. Ma, C. Wang, Y. Niu, X. Wang, and L. Shen, "A saliency-based reinforcement learning approach for a UAV to avoid flying obstacles," *Robot. Auton. Syst.*, vol. 100, pp. 108–118, Feb. 2018.

[100] X. Han, J. Wang, J. Xue, and Q. Zhang, "Intelligent decision-making for 3-dimensional dynamic obstacle avoidance of UAV based on deep reinforcement learning," in *Proc. IEEE 11th Int. Conf. Wireless Commun. Signal Process. (WCSP)*, 2019, pp. 1–6.

[101] K. Wan, X. Gao, Z. Hu, and G. Wu, "Robust motion control for UAV in dynamic uncertain environments using deep reinforcement learning," *Remote Sens.*, vol. 12, no. 4, p. 640, 2020.

[102] L. He, N. Aouf, J. F. Whidborne, and B. Song, "Deep reinforcement learning based local planner for UAV obstacle avoidance using demonstration data," [Online]. Available: arxiv.abs/2008.02521.

[103] O. Walker, F. Vanegas, F. Gonzalez, and S. Koenig, "A deep reinforcement learning framework for UAV navigation in indoor environments," in *Proc. IEEE Aerosp. Conf.*, 2019, pp. 1–14.

[104] F. Bin, F. XiaoFeng, and X. Shuo, "Research on cooperative collision avoidance problem of multiple UAV based on reinforcement learning," in *Proc. IEEE 10th Int. Conf. Intell. Comput. Technol. Autom. (ICICTA)*, 2017, pp. 103–109.

[105] G. Kahn, A. Villaflor, V. Pong, P. Abbeel, and S. Levine, "Uncertainty-aware reinforcement learning for collision avoidance," [Online]. Available: arxiv.abs/1702.01182.

[106] B. G. Maciel-Pearson, L. Marchegiani, S. Akcay, A. Atapour-Abarghouei, J. Garforth, and T. P. Breckon, "Online deep reinforcement learning for autonomous UAV navigation and exploration of outdoor environments," [Online]. Available: arxiv.abs/1912.05684.

[107] H. Dai, H. Zhang, B. Wang, and L. Yang, "The multi-objective deployment optimization of UAV-mounted cache-enabled base stations," *Phys. Commun.*, vol. 34, pp. 114–120, Jun. 2019.

[108] A. Rodriguez-Ramos, C. Sampedro, H. Bavle, P. De La Puente, and P. Campoy, "A deep reinforcement learning strategy for UAV autonomous landing on a moving platform," *J. Intell. Robot. Syst.*, vol. 93, nos. 1–2, pp. 351–366, 2019.

[109] R. Polvara *et al.*, "Autonomous quadrotor landing using deep reinforcement learning," [Online]. Available: arxiv.abs/1709.03339.

[110] A. Rodriguez-Ramos, C. Sampedro, H. Bavle, I. G. Moreno, and P. Campoy, "A deep reinforcement learning technique for vision-based autonomous multirotor landing on a moving platform," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2018, pp. 1010–1017.

[111] R. Polvara *et al.*, "Toward end-to-end control for UAV autonomous landing via deep reinforcement learning," in *Proc. Int. Conf. Unmanned Aircraft Syst. (ICUAS)*, 2018, pp. 115–123.

[112] H. Lu, Y. Li, S. Mu, D. Wang, H. Kim, and S. Serikawa, "Motor anomaly detection for unmanned aerial vehicles using reinforcement learning," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2315–2322, Aug. 2018.

[113] G. Matheron, N. Perrin, and O. Sigaud, "The problem with DDPG: Understanding failures in deterministic environments with sparse rewards," 2019. [Online]. Available: arxiv.abs/1911.11679.

[114] U. Challita, W. Saad, and C. Bettstetter, "Interference management for cellular-connected UAVs: A deep reinforcement learning approach," *IEEE Trans. Wireless Commun.*, vol. 18, no. 4, pp. 2125–2140, Apr. 2019.

[115] H. Bayerlein, P. De Kerret, and D. Gesbert, "Trajectory optimization for autonomous flying base station via reinforcement learning," in *Proc. IEEE 19th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, 2018, pp. 1–5.

[116] S. Colonnese, F. Cuomo, G. Pagliari, and L. Chiaraviglio, "Q-SQUARE: A *Q*-learning approach to provide a QoE aware UAV flight path in cellular networks," *Ad Hoc Netw.*, vol. 91, Aug. 2019, Art. no. 101872.

[117] P. V. Klaine, J. P. Nadas, R. D. Souza, and M. A. Imran, "Distributed drone base station positioning for emergency cellular networks using reinforcement learning," *Cogn. Comput.*, vol. 10, no. 5, pp. 790–804, 2018.

[118] J. Lu, S. Wan, X. Chen, Z. Chen, P. Fan, and K. B. Letaief, "Beyond empirical models: Pattern formation driven placement of UAV base stations," *IEEE Trans. Wireless Commun.*, vol. 17, no. 6, pp. 3641–3655, Jun. 2018.

[119] J. Hu, H. Zhang, and L. Song, "Reinforcement learning for decentralized trajectory design in cellular UAV networks with sense-and-send protocol," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6177–6189, Aug. 2019.

[120] Y. Yan, H. Wang, and X. Chen, "Collaborative path planning based on MAXQ hierarchical reinforcement learning for manned/unmanned aerial vehicles," in *Proc. IEEE 39th Chin. Control Conf. (CCC)*, 2020, pp. 4837–4842.

[121] Y. Huang, X. Mo, J. Xu, L. Qiu, and Y. Zeng, "Online maneuver design for UAV-enabled NOMA systems via reinforcement learning," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, 2020, pp. 1–6.

[122] Y.-H. Hsu and R.-H. Gau, "Reinforcement learning-based collision avoidance and optimal trajectory planning in UAV communication networks," *IEEE Trans. Mobile Comput.*, early access, Jun. 19, 2020, doi: 10.1109/TMC.2020.3003639.

[123] Y.-J. Chen, D.-K. Chang, and C. Zhang, "Autonomous tracking using a swarm of UAVs: A constrained multi-agent reinforcement learning approach," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 13702–13717, Nov. 2020.

[124] J. Cui, Z. Ding, Y. Deng, A. Nallanathan, and L. Hanzo, "Adaptive UAV-trajectory optimization under quality of service constraints: A model-free solution," *IEEE Access*, vol. 8, pp. 112253–112265, 2020.

[125] H. Bayerlein, M. Theile, M. Caccamo, and D. Gesbert. (2020). *Multi-UAV Path Planning for Wireless Data Harvesting With Deep Reinforcement Learning*. [Online]. Available: https://arxiv.org/abs/2010.12461

[126] H. X. Pham, H. M. La, D. Feil-Seifer, and L. V. Nguyen, "Autonomous UAV navigation using reinforcement learning," [Online]. Available: arxiv.abs/1801.05086.

[127] J. Wu, P. Yu, L. Feng, F. Zhou, W. Li, and X. Qiu, "3D aerial base station position planning based on deep *Q*-network for capacity enhancement," in *Proc. IFIP/IEEE Symp. Integr. Netw. Service Manag. (IM)*, 2019, pp. 482–487.

[128] F. Wu, H. Zhang, J. Wu, and L. Song, "Cellular UAV-to-device communications: Trajectory design and mode selection by multi-agent deep reinforcement learning," *IEEE Trans. Commun.*, vol. 68, no. 7, pp. 4175–4189, Jul. 2020.

[129] H. V. Abeywickrama, Y. He, E. Dutkiewicz, B. A. Jayawickrama, and M. Mueck, "A reinforcement learning approach for fair user coverage using UAV mounted base stations under energy constraints," *IEEE Open J. Veh. Technol.*, vol. 1, pp. 67–81, 2020.

[130] K. Li, W. Ni, E. Tovar, and M. Guizani, "Joint flight cruise control and data collection in UAV-aided Internet of Things: An onboard deep reinforcement learning approach," *IEEE Internet Things J.*, early access, Aug. 24, 2020, doi: 10.1109/JIOT.2020.3019186.

[131] Q. Yang, J. Zhang, G. Shi, J. Hu, and Y. Wu, "Maneuver decision of UAV in short-range air combat based on deep reinforcement learning," *IEEE Access*, vol. 8, pp. 363–378, 2019.

[132] X. Liu, Y. Liu, and Y. Chen, "Machine learning empowered trajectory and passive beamforming design in UAV-RIS wireless networks," *IEEE J. Sel. Areas Commun.*, early access, Dec. 2, 2020, doi: 10.1109/JSAC.2020.3041401.

[133] R. Zhong, X. Liu, Y. Liu, and Y. Chen, "Multi-agent reinforcement learning in NOMA-aided UAV networks for cellular offloading," [Online]. Available: arxiv.abs/2010.02749.

[134] S. Zhou, B. Li, C. Ding, L. Lu, and C. Ding, "An efficient deep reinforcement learning framework for UAVs," in *Proc. IEEE 21st Int. Symp. Qual. Electron. Design (ISQED)*, 2020, pp. 323–328.

[135] D. Chen, Y. Wei, L. Wang, C. S. Hong, L.-C. Wang, and Z. Han, "Deep reinforcement learning based strategy for quadrotor UAV pursuer and evader problem," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, 2020, pp. 1–6.

[136] Q. Liu, L. Shi, L. Sun, J. Li, M. Ding, and F. Shu, "Path planning for UAV-mounted mobile edge computing with deep reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 69, no. 5, pp. 5723–5728, May 2020.

[137] B. Li and Y. Wu, "Path planning for UAV ground target tracking via deep reinforcement learning," *IEEE Access*, vol. 8, pp. 29064–29074, 2020.

[138] C. Wang, J. Wang, J. Wang, and X. Zhang, "Deep reinforcement learning-based autonomous UAV navigation with sparse rewards," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6180–6190, Jul. 2020.

[139] J. Hu, H. Zhang, L. Song, R. Schober, and H. V. Poor, "Cooperative Internet of UAVs: Distributed trajectory design by multi-agent deep reinforcement learning," *IEEE Trans. Commun.*, vol. 68, no. 11, pp. 6807–6821, Nov. 2020.

[140] L. Wang, K. Wang, C. Pan, W. Xu, N. Aslam, and L. Hanzo, "Multi-agent deep reinforcement learning based trajectory planning for multi-UAV assisted mobile edge computing," *IEEE Trans. Cogn. Commun. Netw.*, vol. 7, no. 1, pp. 73–84, Mar. 2021.

[141] C. H. Liu, X. Ma, X. Gao, and J. Tang, "Distributed energy-efficient multi-UAV navigation for long-term communication coverage by deep reinforcement learning," *IEEE Trans. Mobile Comput.*, vol. 19, no. 6, pp. 1274–1285, Jun. 2020.

[142] N. Zhao, Y. Cheng, Y. Pei, Y.-C. Liang, and D. Niyato, "Deep reinforcement learning for trajectory design and power allocation in UAV networks," in *Proc. IEEE Int. Conf. Commun.*, 2020, pp. 1–6.

[143] D. Han, Q. Yang, and R. Wang, "Three-dimensional obstacle avoidance for UAV based on reinforcement learning and RealSense," *J. Eng.*, vol. 2020, no. 13, pp. 540–544, 2020.

[144] R. Ding, F. Gao, and X. Shen, "3D UAV trajectory design and frequency band allocation for energy-efficient and fair communication: A deep reinforcement learning approach," *IEEE Trans. Wireless Commun.*, vol. 19, no. 2, pp. 7796–7809, Dec. 2020.

[145] M. S. Shokry, D. Ebrahimi, C. Assi, S. Sharafeddine, and A. Ghrayeb, "Leveraging UAVs for coverage in cell-free vehicular networks: A deep reinforcement learning approach," *IEEE Trans. Mobile Comput.*, early access, Apr. 29, 2020, doi: 10.1109/TMC.2020.2991326.

[146] C. H. Liu, C. Piao, and J. Tang, "Energy-efficient UAV crowdsensing with multiple charging stations by deep learning," in *Proc. IEEE INFOCOM Conf. Comput. Commun.*, 2020, pp. 199–208.

[147] Q. Zhang, W. Saad, and M. Bennis, "Reflections in the sky: Millimeter wave communication with UAV-carried intelligent reflectors," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, 2019, pp. 1–6.

[148] Q. Zhang, W. Saad, and M. Bennis, "Reflections in the sky: Millimeter wave communication with UAV-carried intelligent reflectors," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, 2019, pp. 1–6.

[149] M. Samir, M. Elhattab, C. Assi, S. Sharafeddine, and A. Ghrayeb, "Optimizing age of information through aerial reconfigurable intelligent surfaces: A deep reinforcement learning approach," [Online]. Available: arxiv.2011.04817.

[150] L. Wang, K. Wang, C. Pan, W. Xu, and N. Aslam, "Joint trajectory and passive beamforming design for intelligent reflecting surface-aided UAV communications: A deep reinforcement learning approach," [Online]. Available: arxiv.abs/2007.08380.

[151] O. Bouhamed, H. Ghazzai, H. Besbes, and Y. Massoud, "A generic spatiotemporal scheduling for autonomous UAVs: A reinforcement learning-based approach," *IEEE Open J. Veh. Technol.*, vol. 1, pp. 93–106, 2020.

[152] G. Faraci, A. Raciti, S. A. Rizzo, and G. Schembra, "Green wireless power transfer system for a drone fleet managed by reinforcement learning in smart industry," *Appl. Energy*, vol. 259, Feb. 2020, Art. no. 114204.

[153] Y. Lin, M. Wang, X. Zhou, G. Ding, and S. Mao, "Dynamic spectrum interaction of UAV flight formation communication with priority: A deep reinforcement learning approach," *IEEE Trans. Cogn. Commun. Netw.*, vol. 6, no. 3, pp. 892–903, Sep. 2020.

[154] S. Zhu, L. Gui, N. Cheng, Q. Zhang, F. Sun, and X. Lang, "UAV-enabled computation migration for complex missions: A reinforcement learning approach," *IET Commun.*, vol. 14, no. 5, pp. 2472–2480, 2020.

[155] K. Kim, Y. M. Park, and C. S. Hong, "Machine learning based edge-assisted UAV computation offloading for data analyzing," in *Proc. IEEE Int. Conf. Inf. Netw. (ICOIN)*, 2020, pp. 117–120.

[156] G. Faraci, C. Grasso, and G. Schembra, "Design of a 5G network slice extension with MEC UAVs managed with reinforcement learning," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 10, pp. 2356–2371, Oct. 2020.

[157] J. Cui, Y. Liu, and A. Nallanathan, "Multi-agent reinforcement learning-based resource allocation for UAV networks," *IEEE Trans. Wireless Commun.*, vol. 19, no. 2, pp. 729–743, Feb. 2020.

[158] Q. Li, M. Ding, C. Ma, C. Liu, Z. Lin, and Y.-C. Liang, "A reinforcement learning based user association algorithm for UAV networks," in *Proc. IEEE 28th Int. Telecommun. Netw. Appl. Conf. (ITNAC)*, 2018, pp. 1–6.

[159] J. Yang *et al.*, "Application of reinforcement learning in UAV cluster task scheduling," *Future Gener. Comput. Syst.*, vol. 95, pp. 140–148, Jun. 2019.

[160] O. Anicho, P. B. Charlesworth, G. S. Baicher, A. Nagar, and N. Buckley, "Comparative study for coordinating multiple unmanned HAPS for communications area coverage," in *Proc. IEEE Int. Conf. Unmanned Aircraft Syst. (ICUAS)*, 2019, pp. 467–474.

[161] G. Dulac-Arnold, D. Mankowitz, and T. Hester, "Challenges of real-world reinforcement learning," [Online]. Available: arxiv.abs/1904.12901.

[162] T. Salimans, J. Ho, X. Chen, S. Sidor, and I. Sutskever, "Evolution strategies as a scalable alternative to reinforcement learning," [Online]. Available: arxiv.abs/1703.03864.

[163] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Artif. Intell. Stat.*, 2017, pp. 1273–1282.

[164] K. Bonawitz *et al.*, "Towards federated learning at scale: System design," [Online]. Available: arxiv.abs/1902.01046.

[165] T. Zeng, O. Semiari, M. Mozaffari, M. Chen, W. Saad, and M. Bennis, "Federated learning in the sky: Joint power allocation and scheduling with UAV swarms," [Online]. Available: arxiv.abs/2002.08196.

[166] S. Samarakoon, M. Bennis, W. Saad, and M. Debbah, "Federated learning for ultra-reliable low-latency V2V communications," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, 2018, pp. 1–7.

[167] S. Samarakoon, M. Bennis, W. Saad, and M. Debbah, "Distributed federated learning for ultra-reliable low-latency vehicular communications," *IEEE Trans. Commun.*, vol. 68, no. 2, pp. 1146–1159, Feb. 2020.

[168] S. Wang *et al.*, "Federated learning for task and resource allocation in wireless high altitude balloon networks," [Online]. Available: arxiv.abs/2003.09375.

[169] J. S. Ng *et al.*, "Joint auction-coalition formation framework for communication-efficient federated learning in UAV-enabled Internet of Vehicles," [Online]. Available: arxiv.abs/2007.06378.

[170] H. Shiri, J. Park, and M. Bennis, "Communication-efficient massive UAV online path control: Federated learning meets mean-field game theory," [Online]. Available: arxiv.abs/2003.04451.

[171] N. I. Mowla, N. H. Tran, I. Doh, and K. Chae, "Federated learning-based cognitive detection of jamming attack in flying ad-hoc network," *IEEE Access*, vol. 8, pp. 4338–4350, 2019.

[172] N. I. Mowla, N. H. Tran, I. Doh, and K. Chae, "AFRL: Adaptive federated reinforcement learning for intelligent jamming defense in FANET," *J. Commun. Netw.*, vol. 22, no. 3, pp. 244–258, 2020.

[173] Z. M. Fadlullah and N. Kato, "HCP: Heterogeneous computing platform for federated learning based collaborative content caching towards 6G networks," *IEEE Trans. Emerg. Topics Comput.*, early access, Apr. 8, 2020, doi: 10.1109/TETC.2020.2986238.

[174] Y. Liu *et al.*, "Federated learning in the sky: Aerial-ground air quality sensing framework with UAV swarms," *IEEE Internet Things J.*, early access, Sep. 1, 2020, doi: 10.1109/JIOT.2020.3021006.

[175] B. Lim *et al.*, "Towards federated learning in UAV-enabled Internet of Vehicles: A multi-dimensional contract-matching approach," Apr. 2020. [Online]. Available: arxiv.abs/2004.03877.

[176] K. Bonawitz *et al.*, "Practical secure aggregation for federated learning on user-held data," [Online]. Available: arxiv.abs/1611.044822.

[177] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, "How to backdoor federated learning," in *Proc. IEEE Int. Conf. Artif. Intell. Stat.*, 2020, pp. 2938–2948.

[178] I. Martinez, S. Francis, and A. S. Hafid, "Record and reward federated learning contributions with blockchain," in *Proc. IEEE Int. Conf. Cyber Enabled Distrib. Comput. Knowl. Disc. (CyberC)*, 2019, pp. 50–57.

[179] Y. J. Kim and C. S. Hong, "Blockchain-based node-aware dynamic weighting methods for improving federated learning performance," in *Proc. 20th Asia–Pac. Netw. Oper. Manag. Symp. (APNOMS)*, 2019, pp. 1–4.

[180] U. Majeed and C. S. Hong, "FLchain: Federated learning via MEC-enabled blockchain network," in *Proc. 20th Asia–Pac. Network Oper. Manag. Symp. (APNOMS)*, 2019, pp. 1–4.

[181] H. Kim, J. Park, M. Bennis, and S.-L. Kim, "Blockchained on-device federated learning," *IEEE Commun. Lett.*, vol. 24, no. 6, pp. 1279–1283, Jun. 2020.

[182] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, "Blockchain and federated learning for privacy-preserved data sharing in industrial IoT," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 4177–4186, Jun. 2020.

[183] Y. Qu *et al.*, "Decentralized privacy using blockchain-enabled federated learning in fog computing," *IEEE Internet Things J.*, vol. 7, no. 6, pp. 5171–5183, Jun. 2020.

[184] Y. Wang, Z. Su, N. Zhang, and A. Benslimane, "Learning in the air: Secure federated learning for UAV-assisted crowdsensing," *IEEE Trans. Netw. Sci. Eng.*, early access, Aug. 5, 2020, doi: 10.1109/TNSE.2020.3014385.

[185] S. Wang, F. Roosta, P. Xu, and M. W. Mahoney, "GIANT: Globally improved approximate Newton method for distributed optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 2332–2342.

[186] J. Konečnỳ, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," [Online]. Available: arxiv.abs/1610.05492.

[187] W. Luping, W. Wei, and L. Bo, "CMFL: Mitigating communication overhead for federated learning," in *Proc. IEEE 39th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, 2019, pp. 954–964.

[188] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding," 2015. [Online]. Available: arxiv.abs/1510.00149.

[189] A. T. Suresh, X. Y. Felix, S. Kumar, and H. B. McMahan, "Distributed mean estimation with limited communication," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 3329–3337.

[190] S. Li, S. M. M. Kalan, A. S. Avestimehr, and M. Soltanolkotabi, "Near-optimal straggler mitigation for distributed gradient methods," in *Proc. IEEE Int. Parallel Distrib. Process. Symp. Workshops (IPDPSW)*, 2018, pp. 857–866.

MOHAMED-AMINE LAHMERI (Student Member, IEEE) was born in Tunis, Tunisia. He received the National Engineering Diploma degree from Ecole Polytechnique de Tunisie. He is currently pursuing the M.S./Ph.D. degrees with the Communication Theory Lab, King Abdullah University of Science and Technology, Thuwal, Saudi Arabia. His current research interests include UAV-enabled communication systems, and machine learning for wireless communication.
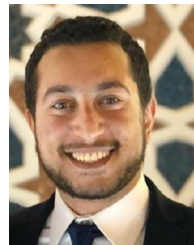
MUSTAFA A. KISHK (Member, IEEE) received the B.Sc. and M.Sc. degrees from Cairo University, Giza, Egypt, in 2013 and 2015, respectively, and the Ph.D. degree from Virginia Tech, Blacksburg, VA, USA, in 2018. He is currently a Postdoctoral Research Fellow with the Communication Theory Lab, King Abdullah University of Science and Technology, Thuwal, Saudi Arabia. His current research interests include stochastic geometry, energy harvesting wireless networks, UAV-enabled communication systems, and satellite communications.

MOHAMED-SLIM ALOUINI (Fellow, IEEE) was born in Tunis, Tunisia. He received the Ph.D. degree in electrical engineering from the California Institute of Technology, Pasadena, CA, USA, in 1998. He served as a Faculty Member with the University of Minnesota, Minneapolis, MN, USA, then with the Texas A&M University at Qatar, Doha, Qatar, before joining King Abdullah University of Science and Technology, Thuwal, Saudi Arabia, as a Professor of Electrical Engineering in 2009. His current research interests include the modeling, design, and performance analysis of wireless communication systems.