LOCATION PREDICTION AND TRAJECTORY OPTIMIZATION IN MULTI-UAV APPLICATION MISSIONS

A Thesis presented to

the Faculty of the Graduate School

at the University of Missouri

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

---

by

ROUNAK SINGH

Dr. Prasad Calyam, Thesis Supervisor

December 2021

The undersigned, appointed by the Dean of the Graduate School, have examined the dissertation entitled:

LOCATION PREDICTION AND TRAJECTORY
OPTIMIZATION IN MULTI-UAV
APPLICATION MISSIONS

presented by Rounak Singh, a candidate for the degree of Master of Science and hereby certify that, in their opinion, it is worthy of acceptance.

Dr. Prasad Calyam

Dr. Ye Duan

Dr. Sharan Srinivas

ACKNOWLEDGMENTS

I would like to thank Dr. Prasad Calyam for his great support throughout my entire Master's degree. I appreciate the outstanding opportunity given to me by Dr. Calyam to work in Virtualization, Multimedia and Networking (VIMAN) Lab with a talented group of individuals on a state-of-art research area and lab facilities. I would like to express my gratitude towards Dr. Ye Duan and Dr. Sharan Srinivas for their interest to be a part of my thesis committee.

I would like to thank Chengyi Qu and Alicia Esquivel Morel for being great team members and supporting me throughout my research process.

I would like to thank my parents for supporting and showing their confidence in me. Finally, I would like to thank my friends for making this journey exciting and joyful for me.

Rounak Singh

Contents

List of Tables

List of Figures

Figure Page

ABSTRACT

Unmanned aerial vehicles (a.k.a. drones) have a wide range of applications in e.g., aerial surveillance, mapping, imaging, monitoring, maritime operations, parcel delivery, and disaster response management. Their operations require reliable networking environments and location-based services in air-to-air links with cooperative drones, or air-to-ground links in concert with ground control stations. When equipped with high-resolution video cameras or sensors to gain environmental situation awareness through object detection/tracking, precise location predictions of individual or groups of drones at any instant possible is critical for continuous guidance. The location predictions then can be used in trajectory optimization for achieving efficient operations (i.e., through effective resource utilization in terms of energy or network bandwidth consumption) and safe operations (i.e., through avoidance of obstacles or sudden landing) within application missions. In this thesis, we explain a diverse set of techniques involved in drone location prediction, position and velocity estimation and trajectory optimization involving: (i) Kalman Filtering techniques, and (ii) Machine Learning models such as reinforcement learning and deep-reinforcement learning. These techniques facilitate the drones to follow intelligent paths and establish optimal trajectories while carrying out successful application missions under given resource and network constraints. We detail the techniques using two scenarios. The first scenario involves location prediction based intelligent packet transfer between drones in a disaster response scenario using the various Kalman Filtering techniques. The second scenario involves a learning-based trajectory optimization that uses various reinforcement learning models for maintaining high video resolution and effective network performance in a civil application scenario such as aerial monitoring of persons/objects. We conclude with a list of open challenges and future works for intelligent path planning of drones using location prediction and trajectory optimization techniques.

Chapter 1

Introduction

The use of drones has been increasing at a rapid pace for a diverse range of applications in e.g., aerial surveillance, mapping, imaging, monitoring, maritime operations, parcel delivery, and disaster response management. Many applications involve multi-UAV configurations [1], wherein several drones act as either carrier devices to carry supplies [2], or are used for aerial surveillance for intelligent information gathering [3]. They also are deployed as aerial base stations to provide bandwidth and network coverage for ground users in certain applications [4]. An example of air-to-air links with co-operative drones surveying over a designated area is shown in Figure 1.1. These operations require location-aided drone movement and optimal drone paths for reduced energy consumption and efficient resource allocation. We discuss salient challenges in realizing these drone location prediction and trajectory optimization techniques and show their advantages through two scenarios involving: (i) network and video analytics orchestration, and (ii) intelligent packet transfer in a disaster response management scenario. This chapter will illustrate how various predicted location information and intelligent path planning schemes help in achieving efficient performance of application missions.

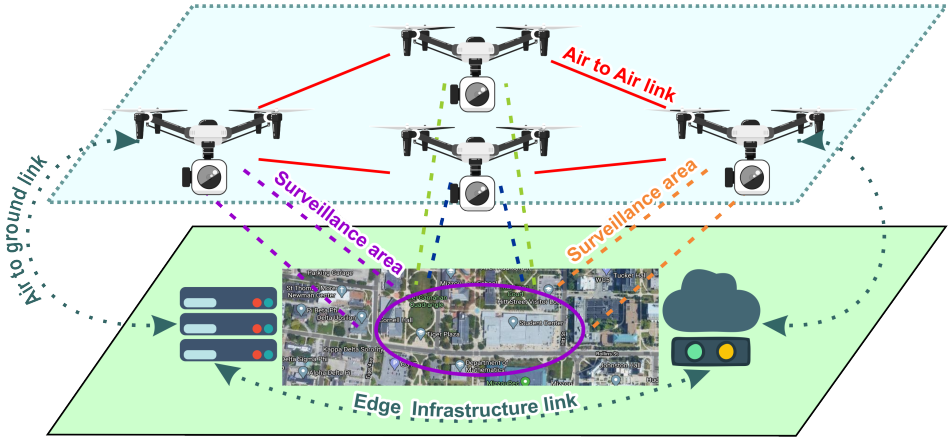


Figure 1.1: Overview of multi-drone setup based on air-to-air and air-to-ground links.

1.1 Use of Drone Location Prediction information in Application Scenarios

To explain the significance of drone location prediction in real-time applications, we consider a multi-drone co-ordination and networking system for a critical application mission such as e.g., a disaster response scenario (DRS) [5, 6]. This scenario involves critical tasks such as monitoring the disaster affected area, search and rescue operations, and providing supplies to victims.

This system features a Flying Ad-Hoc Network Topology (FANET) [7] to support air-to-air, as well as air-to-ground links. The ground control station (GCS) sends requests to the drones to execute certain tasks and the drones send back situational awareness information to the GCS. Such a scenario, however involves challenges related to drone positioning and path planning. Particularly, the *location estimation of drones* is necessary for multi-drone co-operation in order to stay on-course and avoid mid-air collisions. Furthermore, *trajectory planning and optimization* is required to efficiently carry out the application mission considering the limitations of energy and resources. To explicitly understand how these two essential methods impact the performance of drones in application missions, we elaborate them in the following:

1. **Location Estimation and Prediction**: Tracking and predicting the lo-

cations of drones is important in order to get real-time estimates of drone positions for autonomous control and to improve the accuracy of delivery tasks execution in a specific application scenario. It measures how closely the drones are being monitored and also measures the reliability of the path computation algorithm performance. This can be achieved by using motion models of the drone movements, and by using such models within a tracking algorithm or a recursive filter. To get the near-optimal estimates with the motion model, prior works use the Kalman Filter [8] technique which is widely-used for estimation purposes. The popularity of Kalman Filter is due to the fact that this technique takes in the current values as input data (i.e., measurement) along with noises (i.e., measurement noise and process noise) to produce unbiased estimates of system states [9]. Leveraging this state estimation technique can help achieve predicted positions of drones.

2. **Trajectory Optimization** The path that a drone follows during its operation is crucial for effective communication, computation offloading [10], energy consumption and information transfer. A drone's trajectory design unquestionably plays an important role in the application performance enhancement and effectiveness. During its operation, the drone flies over areas which are prone to network and communication vulnerabilities such as signal-loss, cyber-attacks, coverage and range limitations that could severely impact the drones' performance and put the application mission at risk. Machine learning techniques such as model-free reinforcement learning [11] and deep reinforcement learning [12] provide effective reliable solutions for tackling these implications. They use trial-and-error path learning techniques for a drone to establish optimal and intelligent trajectory during its overall flight time during an application mission.

This thesis addresses the concepts of drone position and trajectory optimization techniques related to intelligent path planning. We will first discuss the challenges related to drone location prediction and trajectory optimization. Next,

methods for location prediction will be discussed that involve various Kalman filtering techniques and methods of trajectory optimization using reinforcement and deep reinforcement learning techniques. In this context, we also discuss non-ML-based methods for trajectory optimization. They together provide motivation for localization and intelligent path planning of drones for a given application scenario. Furthermore, we discuss how trajectory optimization of UAVs can aid the operations of public safety networks. These techniques are based on the theoretical and experimental research conducted by the author in the Virtulization, Multimedia and Networking (VIMAN) Lab at University of Missouri Columbia. Lastly, we discuss the main findings of this thesis and list out the open challenges and future works that can be implemented using our approaches to carry out drone-based application missions effectively and efficiently.

1.2 Challenges in Drone-based Application Missions

Since drones are classified under unmanned aerial vehicles, it can be presumed that the navigation, operation and controlling is carried out externally by a ground control station or a ground (human) pilot. In most of the applications today, however the drones flight is increasingly becoming autonomous and may require minimal or almost no external (human) guidance. This is possible due to the variety of sensors on-board that constitute the inertial measurement unit (IMU), global positioning system (GPS), inertial navigation system (INS), gyroscope, accelerometer, barometer and high resolution cameras. These sensors facilitate the autonomous drone flights with high accuracy. Nevertheless, these sensors are prone to external noises that can cause inaccuracies malfunctioning. Anther critical elements on which a drone's fight is dependent is the battery that powers the drones flying mechanism, its flight controller and the above-mentioned sensors. Some of the major challenges pertaining to localization and path-planning relating to the above issues are:

Collision avoidance: Real-world application missions are carried out in complex environments and sometimes, civil applications involving drones are conducted in urban areas. The UAVs are only dependent on their on-board sensor capabilities for their traversal through these environments. It is not always feasible to rely on these sensor readings for navigation and the drones may run into obstacles, hit trees, buildings or other drones mid-operation. Many techniques have been proposed for collision avoidance using decentralized control [13, 14]. The drone has to be aware of the location of its neighbor (drone) and itself at any given instant of time. Leveraging this information can help tackle the problem of mid-air collisions. Object detection using computer vision can help in identifying certain objects by training on datasets of images of common environment obstacles [15]. However, drone's system reliance and communication within the network is usually difficult and challenging in large-scale application missions involving complex environments.

System Security: A wide range of drone-based applications are carried out by the military that operate on highly confidential information gathering within classified missions. Also, many civil applications involve sensitive data collection when drones are deployed as aerial base stations or network providers that handle ground user data (e.g., faces and postures of individuals in crowds). Drones are at risk of cyber-attacks and can be hacked, without the drone being physically captured. The information gathered can become vulnerable and exposed to hackers. Mostly, the camera modules are targeted and video captured is received by hackers which may expose the operations that are carried out in the surveillance area. The work in [16] uses Blockchain technology that encrypts the data being transmitted to base stations. An approach for threat analysis of drone based systems is described in [17]. Countermeasures to security issues in professional drone based networks are shown in [18].

Energy Limitations: Drones require energy for total flight time including hovering over an area for surveillance and data transmission. Additionally, the on-board sensors constantly consume energy to function properly and provide localization of the drones. Energy consumption can also be increased due to attached payloads [19], wind resistance [20] and network issues [21]. The total energy on a drone is limited thus restricting the flight-time of the application mission. The work in [22] provides an energy-aware approach that uses trajectory planning of drones used as mobile anchors to save energy.

Location Awareness and Blockage of Line-of-Sight: In the context of location estimation of drones, blockage of line-of-sight for drones is a very trivial problem that surfaces in the rarest of times [23]. As drones tend to fly long distances based on their application missions, the location awareness becomes essential in order for them to remain in their trajectory and under a predefined network connection for information transfer. It is necessary that they avoid collisions and interference. It becomes a problem if a drone's flight is affected due to external factors and it might become susceptible to unknown attacks. In the worst case scenario, the drone can be thrown off-path and after consuming all its power, it can land or fall in an unknown territory. Thus, it can render itself and the information collected vulnerable, and any expensive sensors or video camera components are subject to expensive damage or loss. Various types of research is being conducted by many groups to realize location awareness [24] of drones.

In the next section, we provide the background for location estimation using various techniques and compare them.

Chapter 2

Methods for Drone Location Estimation and Prediction

In our DRS application the drone environment is considered to be a 2D dynamic and non-linear horizontal plane. As discussed in subsection 1.1, we assume that all the drones are connected forming a FANET. They communicate the mapping and monitoring information over the same network to the delivery drones in order to carry out a delivery task. Consequently, the network topology of the multi-drone system keeps on changing based on the mobility of the drones. The position estimation of the drones must be performed in very short intervals of time using the new coordinates being updated rapidly within the FANET. Each drone in the FANET is considered to have a GPS module and an IMU to record its current location. This information is broadcast to the FANET so that the other drones in the vicinity are recognized for packet or information transfers when needed. We get

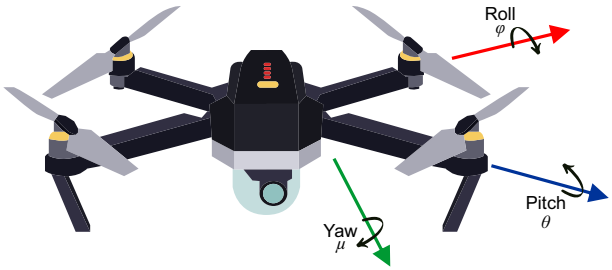


Figure 2.1: Motion angles of a drone responsible for movement with six degrees of freedom controlled by the gyroscope and flight controller.

the initial measurement data of the drone using GPS and other on-board sensors such as gyroscope, barometer, accelerometer and magnetometer that are all part of the IMU. The drone's rotational movement angles observed and controlled by a gyroscope and rotary movements, for stability are shown in Figure 2.1. The accelerations and rotations of the drone can be observed over time to give an estimated position by learning the next measurement values for different time-steps.

The position, velocity, acceleration and heading of a UAV are considered as dynamic states at a given time-step. In order to get the location prediction of an UAV, a state estimator is required to get the true values along with a prediction of these states for the next time-step. Kalman filter can be used to observe state estimates over time along with process noise and measurement noise from sensors to give estimates on which drone position state estimates are closer to true values that cannot be calculated directly [25]. Since the inception of Kalman filter in 1960, it has evolved over time, and the most popular Kalman filters for UAV location estimation are the original Kalman filter, the extended Kalman filter (EKF) [26] and the unscented Kalman filter (UKF) [27].

2.1 State Estimation of Drone Parameters using Kalman Filter

The functionality of Kalman filter relies on consecutive iterations of prediction and filtering i.e., it follows a sequence of prediction and update equations. Along with the inertial navigation system (INS) data, a predefined motion model of the drones' movement is given as input to the Kalman filter. The motion model is basically a state transition matrix having time-periods of the states i.e., x and y coordinate, acceleration and angular velocity. The prediction equations give priori estimates and the update equations give posterior estimates. The update equations take up the previous state's mean and noise covariance and produce the updated

mean and noise covariance values for the next state. The filter then combines the predicted states and noisy measurements to produce unbiased estimates of drone system states. In this process, data with process noise and measurement noise from sensors is used as input, and the Kalman filter produces a statistically optimal estimate of the underlying state by recursively acting on the series of observed inputs.

2.2 Extended Kalman Filtering for Non-linear Drone State Estimation

The major limitation of a Kalman filter is that it can only process estimates of linear systems, and it suffers from linearization when operated on nonlinear models. Drone flight operation is generally non-linear and time varying and system parameters with a dynamic motion model cannot be measured directly with on-board sensors because they may be subject to noise and malfunctioning. To overcome this non-linearity issue of drone position estimation, one of the widely used filter for non-linear state estimation, i.e the extended Kalman filter (EKF) is used. It uses Taylor series expansion and linearizes and approximates the state estimates of a non-linear function around the conditional mean. EKF can be reliable while estimating the drone positions using the drones' dynamic state parameters.

The dynamic motion model is solved by learning the non-linear transition of measurement noise covariance and process noise covariance along with the change in states to give an optimal estimate of the UAV position. The EKF also follows a series of prediction and update equations. The priori estimates calculated during the prediction process are updated to give the posterior estimates and their covariance. Additionally, Jacobians of dynamic functions are used with respect to system state of the UAV to map its states to observations. Additionally, by recursive operations, the covariance of the estimated error is minimized. Hence,

the EKF can be used to get the more accurate positions of the drones through prediction of future positions with insignificant errors, when compared to the original Kalman filter. The work in [29] shows the non-linear estimation of drone's state along with sensor data for localization and [30] shows an approach for determining the locations of drones using inter-drone distances in 2D co-ordinates.

2.3 Unscented Kalman Filter for Position Estimation and Orientation Tracking of UAVs

The EKF is computationally complex but its accuracy is reliable in real-time. Though the EKF is widely used, another advancement of the Kalman filter called the unscented Kalman filter (UKF) is used for the same applications requiring higher accuracy. It is a deterministic sampling approach involving sampling of distributions using a Gaussian random variable. It employs the unscented transform method to select a set of samples called sigma points around the mean to calculate the mean and covariance of the estimation that eradicates the requirement of using Jacobians, as in the EKF. This preserves the linear update structure of the original Kalman of estimates filter unlike the EKF. Table 2.1 shows the comparison of various Kalman filtering schemes used for location estimation of UAVs; for a detailed comparison, readers can refer to [31].

In drone localization application, the system dynamics is expanded as the drone's cartesian location i.e., position, velocity and acceleration. These provide a non-liner relationship between the system states and measurements, and thereby the implementation becomes simpler. The orientation tracking of a drone is also carried out using the UKF [32] by considering rigid body dynamics using various types of measurements like acceleration, angular velocity and magnetic field strength. It uses quaternions and UKF, thus proving its computational effectiveness of tracking. Another approach for position estimation using UKF samples

Table 2.1: Comparison of Kalman filtering scheme variants for location estimation and prediction of UAVs.

Type of Filter	**Type of System**	**Accuracy**	**Model Design**	**Computation Time**
Kalman Filter	Linear	Least accurate	Least Complex	Slowest
Linearized Kalman Filter	Non-Linear	Moderately accurate	Moderately Complex	Slow
Extended Kalman Filter (EKF)	Non-Linear	Accurate	Most Complex	Fastest
Unscented Kalman Filter (UKF)	Non-Linear	Most Accurate	Complex	Fast

images uses a visual target. It uses weights (difference of observed value and estimated value of vision sensor) for observations to prevent divergence in estimated values by UKF [33]. Table 2.1 shows the comparison between different Kalman filtering techniques.

2.4 Sensor Fusion for UAV Localization

Multi-sensor fusion is another technique that shows the importance of using data from distinct sensors to predict the dynamic state estimates of drones for aerial applications. The work in [34] shows how data collected from the GPS, IMU, and INS are fused together for UAV localization using state-dependent Ricatti-equation non-linear filter along with a UKF. Drone path planning involves navigating the drone to a desired destination travelling over a predefined path that constitutes obstacles and other environment constraints. The work in [35] shows how the sensor fusion along with real-time kinematic GPS sensors is used to accurately calculate the altitude and position of the drone. They generate a data-set using instantaneous positions of the drone in different directions along with the roll, pitch and yaw angles. Further, they compare this data with the output of the sensor fusion model estimations that are carried out using an EKF to produce position and altitude estimates of drones.

Table 2.2 summarizes how different methods of location prediction of drones have been proposed in prior works to achieve goals in different application missions.

Table 2.2: Methods and Applications of Location Estimation of Drones.

Case Study	**Method**	**Solution**	**Application**	**Goal**
Xiong et al. [25]	Kalman Filter	Linear Estimation	State estimation	Autonomous Flight
Mao et al.[30]	Extended Kalman Filter	Non-Linear Estimation	Localization of UAVs	Localization without GPS
Kraft et al.[32]	Unscented Kalman Filter	Linearized Estimation	Localization of UAVs	Orientation Computation
Abdelfatah et al. [35]	Sensor Fusion	Non-Linear Estimation	Localization of UAVs	Altitude, Position Estimation

Chapter 3

Methods for Drone Trajectory Optimization using Machine Learning

In context of drone trajectory optimization, we consider an area that is prone to signal-losses, cyber-attacks and potential obstacles like trees, buildings, tall-standing structures which affect the drones' performance and cause hindrance in the application mission. An overview of a drone's trajectory during an application is shown in Figure 3.1. To overcome these problems there is a need for intelligent path planning that can enable the drones follow an optimal trajectory, flying in areas free of all the impediments and attacks.

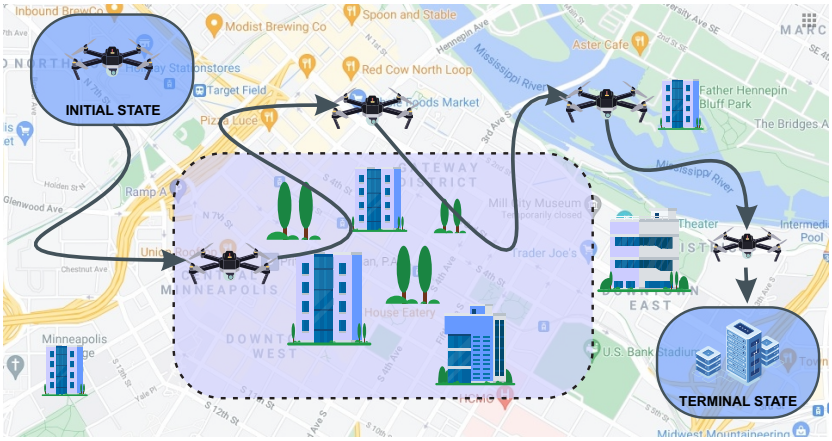


Figure 3.1: Overview of drone's trajectory in a learning based environment comprising of potential obstacles.

The details of the salient methods used to optimize the drones' trajectories while operating in an application are described in the following:

Reinforcement Learning: Path planning of drones is a crucial aspect of research in drone-based applications because the efficiency of missions is dependent on the traversal of the drones in a given area. It correlates with autonomy and has a profound impact on guidance, operation and endurance of the drones. Most drone based application missions are defined in unknown environments. Therefore, Markov Decision Process (MDP) is employed to solve such environments and the Q-Learning algorithm is used that follows the Markov property [36]. It is a model-free reinforcement learning algorithm that puts emphasis on an agent to learn actions under given circumstances to handle problems with stochastic transitions. For any finite MDP, the Q-learning algorithm finds an optimal policy by maximizing the expected value of cumulative rewards over successive actions taken in given states, starting from a current state. There has been a wide usage of reinforcement learning algorithms in varied areas of drone-based application research where drones are allowed to directly and continuously interact with the environment.

Deep Reinforcement learning (DRL): This concept can be considered as a combination of deep learning and reinforcement learning. It employs a deep neural network (DNN) to estimate the Q funtion $Q(s, a)$ for a given set of state-action pairs. Often reinforcement learning requires the state space and the action space to be fixed and discrete, and the agent learns to make decisions by using a trial and error method. It basically involves employing a Q learning algorithm that maintains a record of values of what actions have been taken in given state spaces and also the rewards associated with the corresponding states and actions in a limited format where the state space is predefined. The DRL method allows the agent to act in an environment that has a continuous and mostly undefined

state space. It also uses a set of discrete or continuous actions which are given as a stack of inputs in contrast to the single inputs in case of a simple reinforcement learning. In other words, the DRL makes sure that the agent performs well with extensive input data coming from a large state space to optimize the given objective of any application e.g., it uses pixels as input data in Atari games [37]. The DNN approximates the Q function which estimates the cummulative reward for each state-action pair. A DNN may often suffer with divergence, so it uses a set of experience replay memory and target network to overcome this issue. DQN based RL solutions for drones are necessary because a drone's operation in a given environment is considered as a continuous state space and multi drone scenarios require more robust algorithms such as the multi-agent DQN [38] and the actor-critic [39] networks, which also employ DNNs to generate an optimal policy solution.

3.1 Q-learning

Q-learning is a type of model-free reinforcement learning as described in [40], which is used to solve MDP based problems with dynamic programming. The Q-learning algorithm creates a table (i.e., Q-table) containing the corresponding values of each state-action pair and keeps updating them along with the reward values. The scores obtained in the Q-table are represented as the values of the Q-function $Q(s_t, a_t)$, and are given by -

$$Q(s_t, a_t) = E[\sum_k \gamma^k R_{t+k+1} | (s_t, a_t)] \tag{3.1}$$

where t is the time step and k is the episode. The Q-function is updated for each episode when the agent performs certain actions in a given state to maximize its cumulative reward using the Bellman's equation [41], which is given as -

$$Q(s_{t+1}, a_{t+1}) \leftarrow (1-\alpha)Q(s_t, a_t)$$
$$+ \alpha[R_t + \gamma.max_a Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)]; \quad (3.2)$$

The algorithm converges when maximum reward is reached. The policy encourages the agent to choose optimal actions and receive greater scores in an iterative fashion, which results in the model rendering high Q-values. The interaction of the agent with the environment to generate rewards and to establish a policy is shown in Figure 3.2. The output of the Q-learning is the drone trajectory update guidance that is used to keep the drones as much as possible in the optimal trajectory.

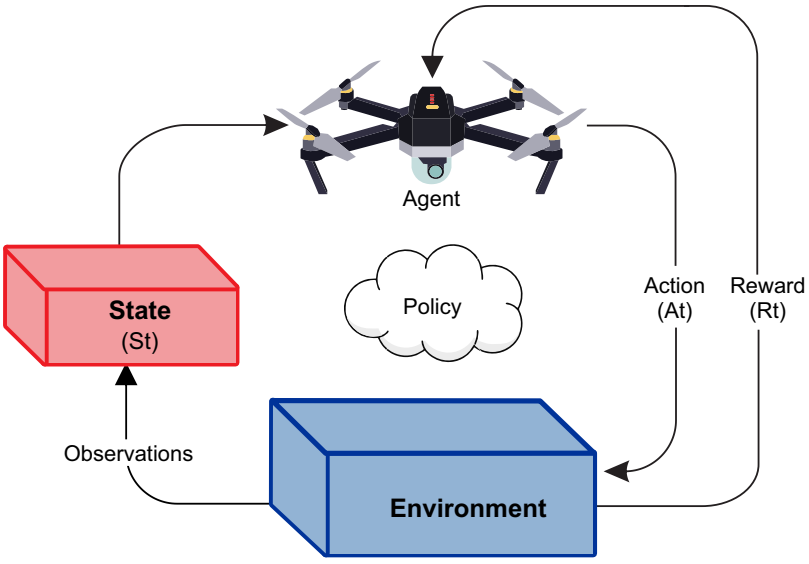


Figure 3.2: Overview of Reinforcement learning showing the agent's interaction with the environment corresponding to given states and actions to generate a policy.

Ensuing the design of the drone's optimal trajectory selection scenario using an MDP, we can evaluate the overall performance by tuning the values of the discount factor γ for obtaining the optimal policy $\pi_t^* : S_t \rightarrow A_t$, which maps the state space with best suitable actions.

3.2 Deep Q-Network

To implement the Q-Learning based algorithm that render optimal trajectories of the drones, we choose a DQN that allows for maximum exploration and exploitation [42] of the learning environment by the agent. The actions in this case are dependent on the weights of the primary DNN, which adds flexibility in the overall learning process i.e., as the weights update, the rewards update accordingly. The intelligent trajectory learning application for DRS scenario renders network performance in terms of throughput and the video quality scores (i.e., rewards) obtained in the process of learning. The DQN is trained using a experience replay, which is memory buffer that stores the sequence of state-action pairs from previous episodes. The process of utilizing replay memory to gain experience by random sampling is called experience replay.

The DQN utilizes the mini-batch from experience replay with the observed state transition samples to update its DNNs after each episode during the training process. Thereby, it breaks any correlation made using sequential state-action pairs in the previous episodes. Sometimes, drones are used as swarms in application missions that are connected via wireless links. For any broken link, the drones have to position themselves to make up the broken link to maintain the same QoS requirements. The work in [43] gives an approach that uses DQN to determine optimal links between drones in swarms and to localize the drones to improve overall network performance of the swarm's wireless network.

3.3 Double Deep Q Network

The Deep Q Network has a single action value function and while updating the primary DNN, same values are used for selection and evaluation of actions. This in turn leads to overestimation that renders over optimistic action value estimates. To avoid this issue, Double Deep Q Learning decouples the selection and evaluation of value function using two separate DNNs (primary and target). It employs two

value functions that learn by selecting random experiences that produce two set of weights [44]. It aims to get the most out of Double Q learning with slight increase in computation. For civil and military based application missions, Double Deep Q Network (DDQN) is used for 3 Dimensional path planning of drones using greedy exploitation strategy to improve learning in complex environments [45].

3.4 Dueling Deep Q Network

The Dueling Deep Q Network (Dueling DQN) is another form of a deep reinforcement learning algorithm. It consists of two separate estimators (DNNs) for state value function and action value function. It is used to overcome the impact caused by similar action values in multiple episodes [46]. Some application missions involve multi-drone connections using cellular networks with each drone acting as a base station. To improve the connectivity over the cellular network, Dueling DQN is used to provide trajectory optimization and coverage-aware navigation for radio mapping [47]. Also in other dynamic environments with unrealized threats, Dueling DQN can provide intelligent path-planning using epsilon greedy policy to render optimal trajectories of the drones [48].

3.5 Actor Critic Networks

Some of the most recent and popular reinforcement learning algorithms are the actor critic networks that aim to achieve optimal policies using low-gradient estimates. The actor network is a DNN that takes in the current environment state and computes continuous actions and the critic judges the performance of the actor network with respect to the input states. It also provides feedback to determine the best possible actions that render higher rewards [49, 50]. An approach to achieve efficient communication and band allocation in the drone network involves determining their 3D trajectory under energy constraints using deep deterministic policy gradient (DDPG) [51] actor-critic networks as shown in [52].

Chapter 4

Non-ML-based Trajectory Optimization Techniques for Drones

Although machine learning is gaining traction in solutions for autonomous vehicles, trajectory optimization of UAVs in real-time scenarios is challenging because it is a non-convex optimization problem. There have been advances in drone trajectory planning and optimization techniques for single-UAV, dual-UAV and multi-UAV based applications. A survey for long-distance trajectory optimization of small UAVs is given in [61], and a survey of techniques involving joint trajectory optimization with resource allocation is given in [62]. An approach to perform joint trajectory and communication co-design can be found in [63]. Advances in path-planning techniques feature techniques that are quite different from learning-based methods. To provide high-mobility and flexibility in FANETs, many techniques have been proposed. However, there are several open challenges when it comes to path planning of UAVs. A series of latest works that try to solve the open challenges are as follows:

4.1 Quantization Theory-Lagrangian Approach

An approach to provide optimal UAV positions in static networks under spatial user density is described in [64]. This approach uses uniform distribution of ground terminals at zero altitude and determines optimal placement of UAVs in static environments along with ways to reduce power consumption. The optimizations for the static case are done by considering the UAVs at varying altitudes, followed by characterizing optimal UAV deployments in dynamic scenarios. These optimizations are performed by varying ground terminal density in any given dimension for a fixed number of UAVs which are placed at moderate distances from each other. Two two cases are considered: (i) UAVs with no movement, and (ii) UAVs with unlimited movement. This approach aims to achieve lowest possible average power consumption followed by providing a Lagrangian-based descent trajectory optimization technique. The Lagrangian technique is similar to Voronoi based coverage control algorithms and is based on time discretization.

4.2 Joint Optimization of UAV 3D Placement and Path Loss

An approach in [65] aims to fill the gaps of joint aerial base station (ABS) deployments and path loss compensation for ABS placements at certain heights. It puts stress on the power control mechanism needed to establish reliable communication, and on the propagation path-loss that hinders the overall communication performance. The 3D UAV placement procedure involves altitude optimization for maximum coverage along with horizontal position optimization for 2D placement that uses a modified K-means algorithm for aerial base station height with a compensation factor.

4.3 Flexible Path Discretization and Path Compression

This technique considers a piecewise-linear continuous trajectory of a UAV whose path comprises of consecutive line segments connected through a finite number of points in 3D called way-points. It provides a solution to render an optimal path by using a flexible path discretization technique to optimize number of way-points in the path to reduce the complexity in the design of the UAV trajectory [66]. The variables that tend to solve the path-planning are considered in two sets of design-able and non-design-able way-points. The way-points are generated using their sub-path representations that ensure a desired trajectory discretization accuracy. They also help to obtain utility and constraint functions that retain accuracy in e.g., aerial data harvesting using distributed sensors. Following this, a path compression technique is performed that takes the 3D UAV trajectory and decomposes it into a 1D (sub-path) signal to further reduce the path-design complexity.

4.4 Connectivity Constrained Trajectory Optimization

This technique provides a solution to optimize an UAV's trajectory in an energy and connectivity constrained application to reduce the overall mission completion time. It uses graph theory and convex optimizations to achieve high-quality solutions in various scenarios involving: (i) altitude mask constraints, (ii) coordinated multi point (CoMP)-based cellular enabled UAV communications, (iii) QoS requirements based communication using UAVs, and (iv) non-LoS channel model. The degree of freedom of UAV movement is exploited to increase the design flexibility of UAV trajectories with respect to the locations of GCS, and ground users for effective communication. By applying structural properties, effective bounding and approximation techniques, the non-convex trajectory problem is converted into a simple shortest path problem between two vertices and solved

using two graph theory based algorithms [67]. A similar technique involving effective trajectory planning under connectivity constraints using graph theory is shown in [68].

4.5 3D Optimal Surveillance and Trajectory Planning

Public safety is another crucial application domain for designing drone based communication systems. Prior works such as [69] have proposed approaches to solve challenges for the public safety application domain. Specifically, a swarm optimization based trajectory planner is provided with surveillance-area-importance updating apparatus. The apparatus aims to derive 3D surveillance trajectories of several monitoring drones along with a multi-objective fitness function. The fitness function is used as a metric for various factors of the trajectories generated by the planner such as energy consumption, area priority and flight risk. This approach renders collision-free UAV trajectories with high fitness values and exhibits dynamic environment adaptability and preferential important area selection for multiple drones. Table 4.1 summarizes how different methods of trajectory estimation and optimization have been proposed to achieve certain goals in various applications.

Table 4.1: Methods and Applications of Trajectory Optimization of Drones.

Case Study	**Objective**	**Solution**	**Method**	**Performance**
Koushik et al. [43]	Node Positioning	MHQ-PRP Queing	Deep Q-Network	Increased throughput of dynamic UAV swarming network
Zhao et al. [45]	3D Path Planning	Greedy Exploration (DRL)	Double Deep Q-Network	Better convergence compared to DQN and DDQN
Yan et al. [48]	Real-time path planning	STAGE scenario	Dueling DDQN	Efficient dynamic environment path planning
Ding et al. [52]	3D Trajectory Planning	DDPG (DRL)	Actor-Critic Network	Increased throughput under fairness conditions
Saxena et al. [56]	Traffic-aware UAV trajectories	Leveraging UAV Base Station Network (UAVBSN)	Deep Reinforcement Learning	Three fold increase in throughput of UAVBSN
Nguyen et al. [58]	UAV Trajectory Optimization	Energy Harvesting Time Scheduling	Deep Deterministic Policy Gradient	Efficient resource allocation under energy and flight-time constraints
Xu wt al. [63]	2D Trajectory Planning	Semi-definite Programming	Monotonic Optimization (various)	Significant power saving
Koyuncu et al. [64]	Multi-UAV Trajectory Optimization	Lagrangian Approach (1D)	Quantization Theory	Minimized power consumption
Shakoor et al. [65]	3D Placement and Path-Loss	Placement Compensation Factor	Various Optimization techniques	Improved coverage and performance
Guo et al. [66]	3D Tajectory Design	Flexible Path Discretization and Path Compression	Graph Theory (Shortest path)	Reduced path design complexity
Zhang et al. [67]	3D UAV Trajectory Design	Graph Theory	Optimization - (various)	Improved connectivity
Yang et al. [68]	3D UAV Trajectory Design	Graph Theory, Inequality property	Optimization - (various)	Improved connectivity
Teng et al. [69]	3D Optimal Trajectory Planning	Particle Swarm Optimization	Trajectory Planner	Improved dynamic environment adaptability

Chapter 5

How can trajectory optimization aid UAV-assisted PSNs?

Public safety networks (PSNs) are established for public welfare and safety. They are essential means of communication for first responders, security agencies and healthcare facilities. Nowadays, PSNs have been widely relying on wireless technologies such as long range WiFi networks, mobile communication and broadband services that use satellite-aided communication links. In addition, PSNs operate extensively during natural disasters, during times when there is a threat to national security such as terrorist attacks, and any large-scale hazards caused due to human activities. As wireless communication is the backbone of PSNs, advanced and efficient communication technologies such as LTE and 5G-based communications can help establish broadband services that provide improved situational awareness with security and reliability characteristics in the network. In this section, we will discuss how UAVs could be a choice for public safety networks in terms of various use cases, provide case studies on trajectory optimization and localization for UAV-assisted PSNs, and discuss open challenges in UAV-assisted PSNs. Figure 5.1 provides an overview of multi-UAV operations spanning diverse applications ranging from civil applications to public safety networks.

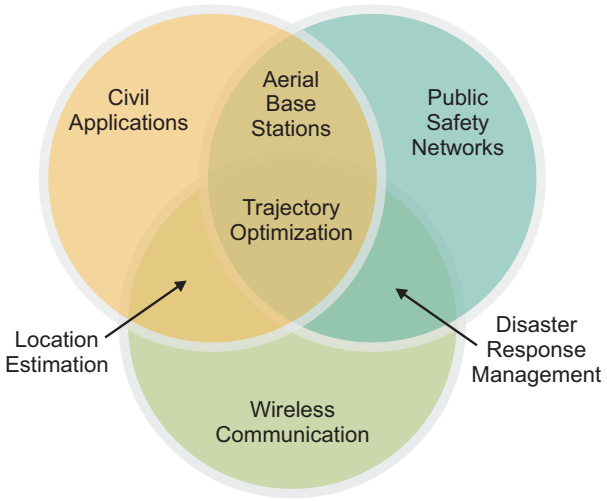


Figure 5.1: Overview of multi-UAV operations across various applications ranging from civil applications to public safety networks.

5.1 UAV-assisted PSNs

Since wireless communications play a fundamental role in PSN operations, their effectiveness and responsiveness to emergency situations becomes critical [70]. A few issues that affect the functioning of PSNs include: communication equipment deployment costs, spectrum availability, network coverage and quality of service (QoS). A few of these issues can be solved by improving the ground-based communication systems by fully exploiting the potential of situational awareness and enabling advanced tracking, navigation and localization services [71]. However, to eradicate these issues of PSNs as a whole, UAVs with enhanced functionalities that can operate as aerial base stations with high-end communication equipment can be used to amplify the effectiveness of communication, improve coverage, reliability, and energy efficiency of wireless networks. In such UAV-assisted PSNS, UAVs are operated by acting as flying mobile terminals within a cellular network or broadband service while monitoring the area, simultaneously. The other advantage on UAV-assisted PSNs is that the UAV base stations are faster and easier to deploy, which provides effectively on cost and can be flexibly reconfigured based on mobility.

5.2 Trajectory Optimization and Path-Planning for UAV-Assisted PSNs

Trajectory Optimization and localization of UAVs can significantly impact the 3D-deployment of the aerial base-stations serving non-stationary users. Optimal path planning can help strengthen the carrier channel transmitting and receiving characteristics. The cellular networks involving aerial base stations can be converted to FANETs, which can help to establish efficient wireless communication in the PSNs. A case study in [72] used path-planning for UAVs in a disaster resilient network. They showed how drones can be used in an wireless infrastructure, allowing a large number of users to establish line-of-sight links for communication. Another approach in [73] uses fast K-means based user cluster model for joint optimization of UAV deployment and resource allocation along with joint optimal power and time transfer allocation for restoring network connectivity during a disaster response scenario. Similarly, research in [74] discusses the role of UAVs in PSNs in terms of energy efficiency and provides a multi-layered architecture that involves UAVs to establish efficient communication by considering the energy consumption considerations.

5.3 Open Challenges in UAV-assisted PSNs

As we can observe from the previous subsections, UAVs when used as aerial base stations can significantly improve the performance and operation of PSNs. However, there are sill open challenges that need to be resolved. For example, the monitoring of moving objects/target-users becomes an issue after deployment in a disaster scenario. Few challenges such as traffic estimation, frequency allocation and cell association are addressed in [75]. An approach in [76] propose a disaster resilient communications architecture that facilitates edge-computing by providing a UAV cloudlet layer to aid emergency services communication links. Another

approach in [77] has a uplink/downlink architecture for a Full-Duplex UAV relay to facilitate ground base stations around the UAVs. The UAVs communicate to distant ground users using non-orthogonal multiple access (NOMA) assisted networks.

Another important concern raised with UAV-based PSNs is security (see Section 1.2). In most cases, These PSNs are handling confidential information and may become vulnerable. They can also be subject to cyber and physical attacks. A variety of security concerns and challenges in drone-assisted PSNs are addressed in [78] such as: WiFi attacks, channel-jams, grey hole attacks, GPS spoofing and other issues relating to interruption, modification, interception and fabrication of information along with procedures to handle them.

Chapter 6

Online Learning in Multi-UAV Application Missions

This section aims to illustrate the location prediction and trajectory optimization schemes for the application mission explained in Section 1.1 using two scenarios that involve location prediction and trajectory optimization of drones. As discussed earlier, we assume that all the drones are connected forming a FANET. By this, they communicate the mapping and monitoring information over the same network to the delivery drones in order to carry out a delivery task. Consequently, the network topology of the multi-drone system keeps on changing based on the mobility of the drones. Firstly, our multi-drone coordination and networking scheme features a location-aided prediction algorithm coupled with a packet forwarding algorithm for drone-to-ground network establishment. Our novelty is in the approach of using Reinforcement Learning (RL) to estimate future drones' trajectories based on their coordination status and their on-board sensors information. Specifically, once the intermediate drone gets the accurate predicted position information of the destination drone, then a list of preliminary decisions on where to forward the packets is made. Secondly, the performance in the network links across multi-drone FANETs vary due to certain factors such as, application requirements, weather conditions, obstacles in the path, etc that cause frequent or

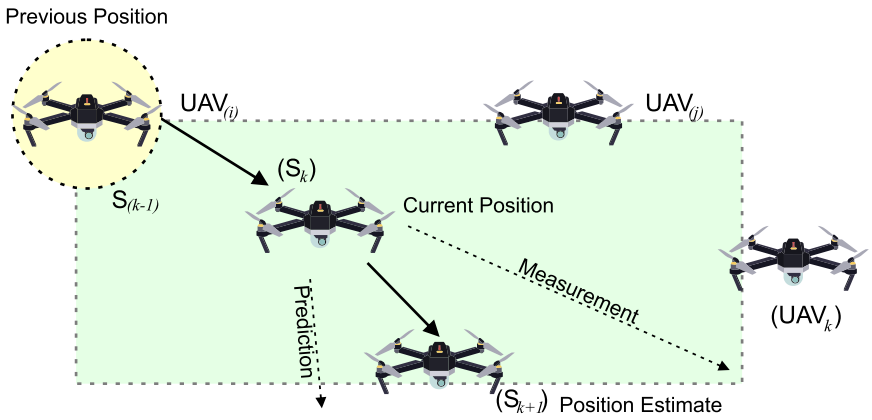


Figure 6.1: Overview of drone location prediction using Extended Kalman Filter.

intermittent outages in transmission and reception of crucial information inside the FANET. This could also affect the drone's video analytics, when used for civil applications or aerial surveillance. Our proposed orchestration process solves the network links and video analytics disruption by employing an online learning based technique. It analyzes each drone's position and trajectory during the flight, and find ways to optimize the drone's path and even the video quality by selection of pertinent network protocol and video properties during the drone flight.

For simplicity, the drone environment in the FANET is considered to be a 2D dynamic and non-linear horizontal plane, assuming the drones are deployed at a fixed altitude. For location prediction, we use EKF with inter-drone distance measurements and a fixed motion model to obtain optimal position estimates of a drone. In our application mission, we get the initial measurement data using a GPS. The accelerations and rotations of the drone can be observed over time to give an estimated position by learning the next measurement values for different time-steps. The position estimate of the drone is the average of the predictions made using previous positions and new measurements obtained as shown in Figure 6.1.

Predicting the trajectory of drones is crucial for improving the global performance. The non-linear, dynamic motion model for a single drone is defined by the discrete-time state vector:

$$S = [x_t, y_t, \phi_t, v]^T$$

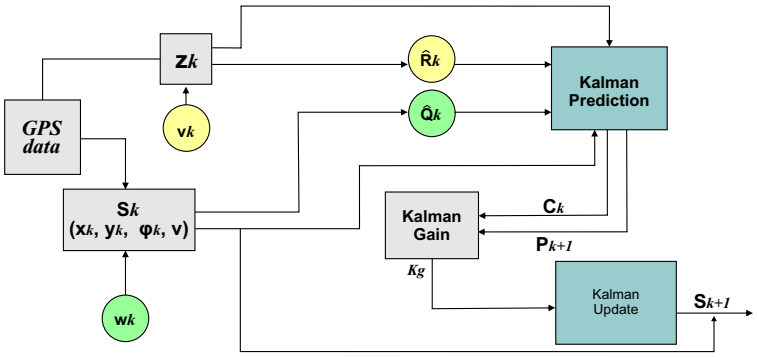


Figure 6.2: Workflow diagram of Extended Kalman Filter based drone state estimation.

where, (x_t, y_t) are coordinates of the drone, ϕ_t is the drone's heading, and v is the constant drone's velocity. The workflow diagram of an EKF to produce dynamic state estimates is shown in Figure 6.2.

The motion model remains the same for all the drones, and the measurement equation $h[S]$ gives the relative distance between the drone's current position and next position estimate:

$$h[S] = \sqrt{(\hat{x_k} - x_t)^2 + (\hat{y_k} - y_t)^2}$$

where $(\hat{x_K}, \hat{y_k})$ are the predicted coordinates of drones for k^{th} time-step. The state prediction of a drone for k^{th} time-step is given as:

$$S_k = g[S_{k-1}] + w_k$$

$$z_k = h[S_k] + v_k$$

where z_k is the observation vector, w_k is process noise, and v_k is the measurement noise used to model the distance measurements (both are uncorrelated sequences of white Gaussian noise).

The dynamic function g maps the measurement to the state, and it is modeled using a_t, which is the drone's acceleration as follows:

$$g = \begin{bmatrix} a_t \cdot v\cos(\phi) + x \\ a_t \cdot v\sin(\phi) + y \\ \phi \\ v \end{bmatrix}$$

The process and measurement noise covariances are given by Q and R respectively:

$$Q = \begin{bmatrix} a_t & 0 & 0 & 0 \\ 0 & a_t & 0 & 0 \\ 0 & 0 & \phi & 0 \\ 0 & 0 & 0 & v \end{bmatrix}, R = \begin{bmatrix} \sigma(\phi) & 0 \\ 0 & \sigma(\phi) \end{bmatrix}$$

The position estimates from the EKF are obtained via a two-step process that involves: (i) performing a series of predictions, and (ii) updating the produced estimate data. The dynamic motion model is solved by learning the non-linear transition of Q and R along with change in states to give optimal estimates of the position data. The state prediction covariance is computed as:

$$\hat{P}_{k+1} = g[S_k] \cdot P_k \cdot J_g[S_k] + \hat{Q}_k$$
$$\hat{z}_{k+1} = h[S_{k+1}]$$

while the predicted measurement covariance is given by:

$$\hat{C}_k = J_h[S_k] \cdot \hat{P}_{k+1} \cdot J_h[S_k]^T + \hat{R}_k$$

where J_g and J_h are the Jacobians of dynamic functions g and h with respect to S_k that map states along with uncertainties, to observations. The prediction step involves process covariance $\hat{Q}_k$ estimation. The measurement step produces measurement noise covariance $\hat{R}_k$, which is also required by the prediction step.The priori estimates calculated during the prediction process are updated to give the posterior estimates and their covariance. The corresponding updated equations

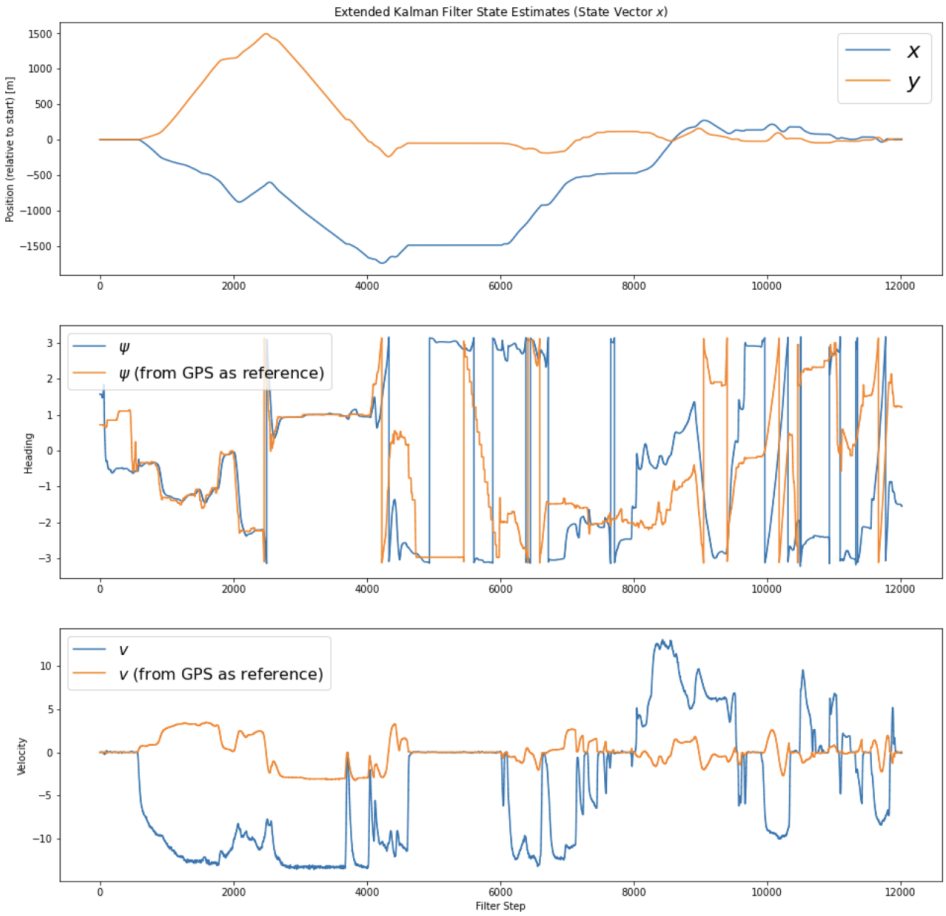


Figure 6.3: Drone position, heading and velocity estimation using EKF.

are given as follows:

$$K_g = \hat{P}_{k+1} \cdot J_h[S_k]^T \cdot [\hat{C}_k]^{-1}$$

$$S_{k+1} = S_k + K_g(z_{k+1} - \hat{z}_{k+1})$$

where K_g, $(0 \leq K_g \leq 1)$ is the Kalman gain, S_{k+1} is the next state estimated, and the updated state covariance matrix is given by:

$$P_{k+1} = (I - K_g \cdot J_h[S_k]) \cdot \hat{P}_{k+1}$$

The optimal position estimates are produced by recursively operating on previous values of noisy data. Figure 6.3 shows the position, heading and velocity estimation of the drone.

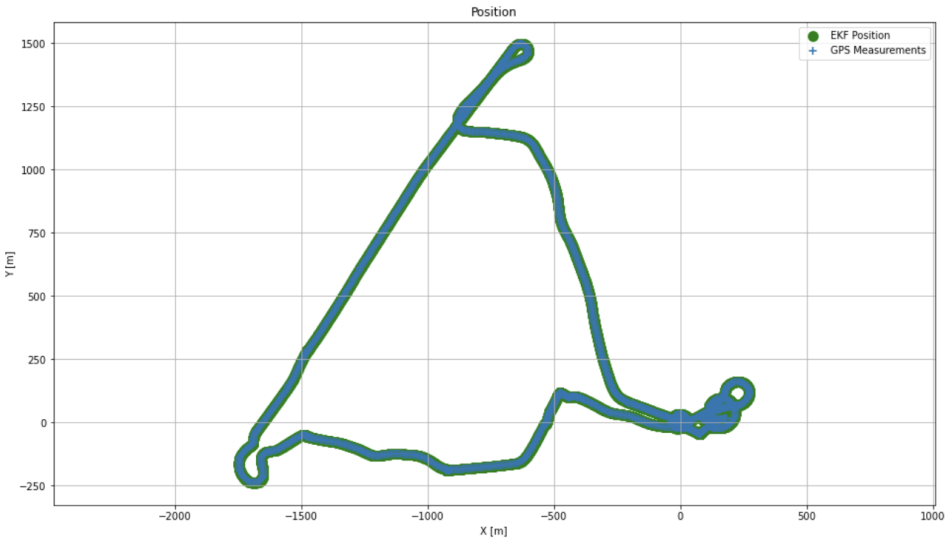


Figure 6.4: Drone location prediction using Extended Kalman Filter.

Additionally, by recursive operations, the covariance of the estimated error is minimized. Thus, the EKF can be used to get the most accurate position of the drone through prediction of future positions with insignificant errors as shown in Figure 6.4.

Thus, location prediction information of all the drones in the FANET can be collected and and fed to the online trajectory learning algorithm to make advance decisions by using future location information of the mapping drones, monitoring drones and the delivery drones in the FANET. Thus, the FANET in the DRS scenario can utilize theses location estimation techniques to facilitate efficient packet transfer.

6.1 Online-based Multi-Drone Approach for Intelligent Packet Transfer using A3C Network

We allot a hexagonal area as in [60] of side C for each drone as shown in Figure 6.5, in order to maneuver inside them and change their heading to any direction to efficiently cover a surveillance area and discover POIs. For initiating a packet transfer between the drones in the environment, we consider the location of the drones as $P_n = (x_n, y_n, h_n)$, where x_n, y_n are the location coordinates, and h_n

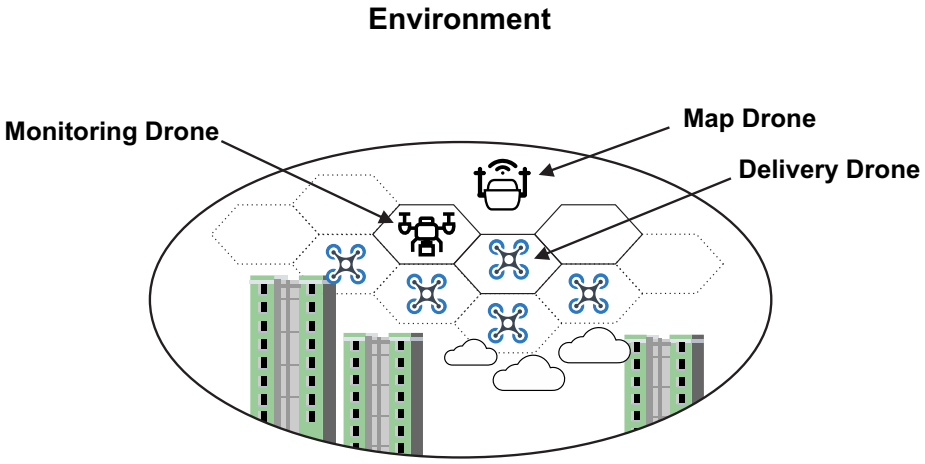


Figure 6.5: Deployment of multi-drones in the intelligent packet transfer application scenario.

denotes the altitude. Moreover, D_{ij} is the distance among any two drones $i \neq j$.

Due to the potential uncertainties in the environment that affect the drones' localization and orientation, the location prediction of the drones can be formulated as a Partially-Observable Markov Decision Process (POMDP) defined by the tuple $[S, A, P, Z, O, R]$, where S, A, P, Z, O, and R denote the states, actions, probability of transition, probability distribution function for observing states, observations, and rewards, respectively.

The environment states are defined as $s_t = (P_i, \phi_i, P_j, \phi_j, D_{ij})$ where P_i and P_j are the positions, while ϕ_i and ϕ_j are the headings of the packet transferring and receiving drones. The actions performed by the agents are defined as a set of 7 flight operations a_t, i.e., hover, forward, backward, up, down, yaw-left, and yaw-right. Moreover, the rewards R_t are defined as follows:

$$R_t = \begin{cases} +10 & \text{every drone action} \\ -50 & \text{going out of Hexagonal cell} \\ -100 & \text{collision with obstacles/drones} \\ +100 & \text{successful packet transfer} \end{cases}$$

Let T be the episode in which the drone performs action in the state space. The

drone does not track the exact states $s_t \in S$, but uses the observations $o \in O$ in any given episode T. Therefore, it has to rely on the history of actions and observations $S_t = (a_t, o_t; a_{t-1}, o_{t-1}; \ldots; a_0, o_0)$ to perform intelligent actions that allow higher rewards. However, this history S_t exponentially grows with every action taken and every state observed. The drone rather chooses to utilize the belief states b which are single valued and represent the probability distribution $Z = p(o_t|s_t, a_t)$ over all possible states s_t in a given episode. These belief states are a sufficient measure of history, and given the current belief state b_t, the POMDP aims to find an optimal policy π^* that maximizes a value function V^π while following a sequence of actions and observations. The value function V^π is thus given by -

$$V^\pi(b) = \mathbb{E}[\sum_{i=0}^{\infty} \gamma^{i-t} R_i|b, \pi]. \tag{6.1}$$

Our approach, aims to solve the POMDP problem using a multi-drone Asynchronous Advantage Actor Critic (A3C) Network such that the best possible actions are chosen in given states, and the cumulative reward G_t (i.e., the accumulated discounted return) gets maximized, as follows:

$$G_t = \mathbb{E}[\sum_{t=0}^{\infty} \gamma^t R(b_t, a_t)] \tag{6.2}$$

where γ is the discount factor. Finally, the action value function is given by:

$$Q_\pi(b_t, a_t) = \mathbb{E}[\sum_{n=0}^{\infty} \gamma^n R(b_t, a_t)|b_t, a_t] \tag{6.3}$$

Each episode in the A3C network stochastically progresses, and each corresponding action is probabilistically sampled. The *actor* and related *critic* networks within the A3C network are deep neural networks (DNNs) and have target networks. In the following, we detail the proposed A3C network functioning.

A master network has an actor network that generates a policy $\pi(a_t|b_t; \theta_a) = P(a_t|b_t; \theta_a)$ and a critic network that generates the state value function to test the expected return under belief state $V_\pi(b_t; \theta_v)$, where, b_t is the belief state of an

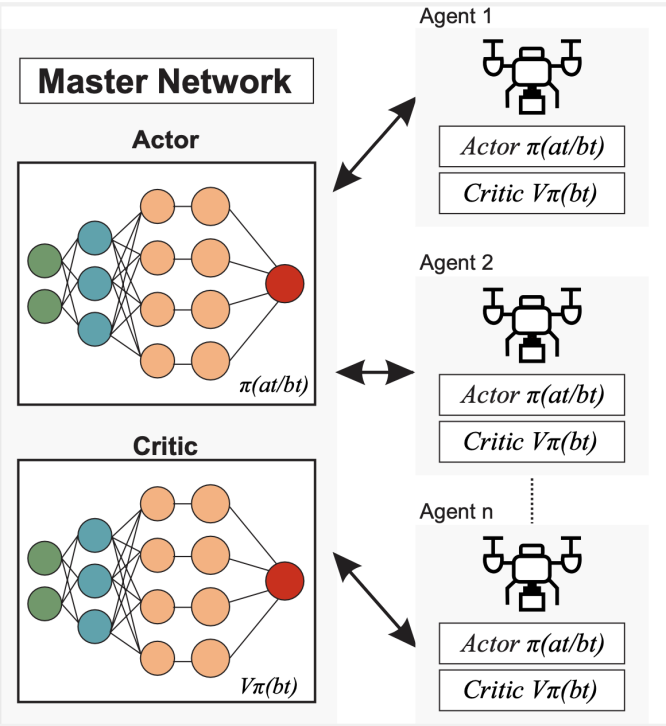


Figure 6.6: Multi-agent Asynchronous Advantage Actor Critic (A3C) Network

episode, θ_a denotes weights of actor network, and θ_v denotes the weight of critic network. The weights in these cases are updated using back-propagation. Each copy of the Master Network is sent to the agents as shown in Figure 6.6. The actor network is trained with the loss function:

$$L(\theta_a) = \frac{1}{N_{batch}} \sum_{i=1}^{n} [-Q(b_t(i), \pi_b(i); \theta_a)]^2 \tag{6.4}$$

where, N_{batch} is the batch size and the critic network is trained with the loss function given by:

$$L(\theta_v) = \frac{1}{N_{batch}} \sum_{i=1}^{n} [y_t(i) - Q(b_t(i), a_t(i); \theta_v)]^2 \tag{6.5}$$

where $y_t(i) = R_t + \gamma \cdot Q^{'}(b_{t+1}, \pi^{'}(b_{t+1}; \theta_a^{'})|\theta_v^{'})$ and $\pi^{'}(b|\theta_a^{'})$ and $Q^{'}(b, a|\theta_v^{'})$ are target networks of actor and critic networks. Due to the fact that gradient methods are used to optimize the network weights, there are chances of high variance occurrence in the critic network. Therefore, we use employ an advantage function $\Omega(b_t, a_t) = Q(b, a) - V^{\pi}(b_t, \theta_v)$ to overcome this high variance problem. Upon solving the

POMDP using the proposed A3C network, we get a policy $\pi : b \rightarrow a$ that maps the actions to belief states b. The optimal policy obtained from the actor and critic network operations is given by -

$$\pi^*(b) = argmax_b \cdot V^{\pi}(b) \tag{6.6}$$

The A3C network allows multiple drones to interact in a parallel fashion with the environment in order to generate individual policies that are the outputs. Thus, we can use the RL based location prediction A3C method to enhance the packet forwarding performance in the presence of environment obstacles, while using the residual energy from the forwarding drone candidates. Based on the drone position prediction information, we can obtain the local relative distance between the drones in advance.

6.2 Online-based Trajectory Optimization with Network and Video Orchestration

When the drones are operating in the environment, the time intervals in which the drones carry out surveillance are considered to be irregular, and we refer to each of these time steps as an episode. During each episode, the drones ($n \neq 0$) hover over specific regions in discrete time steps. At each time step Δt where $t = 0, 1, 2, 3, ..n$, the drones enter states $s(n)_t$, performs a set of actions $a(n)_t$, receives rewards r_t and learns where to move depending on the reward values in the observed state. This observed state forms the basis for the next set of actions. The states in the learning environment are defined below -

$$s(n)_t = \begin{cases} S_{0(random\_area)} \\ S_{1(secure\_area)} \\ S_{2(precarious\_area)} \\ S_{3(terminal\_state)} \end{cases} \tag{6.7}$$

To ensure that the near optimal actions a_t can be chosen at every time step Δt, we consider the following actions that the drone can perform -

$$a(n)_t = \begin{cases} A_{0(hover)} \\ A_{1(move\_rapidly)} \\ A_{2(land\_to\_recharge)} \end{cases} \tag{6.8}$$

The state-action pairs result in various rewards. The rewards are independent of observed states as the probabilities of state transitions are different for each action for a given state. The reward function is constructed using two distinct rewards in this scenario. In our proposed approach, there are both positive and negative rewards associated with the actions that the drones perform in the defined state space. Let C be the cost associated with the network protocol and video parameters selection, and i be the task of switching the video codec along with resolution i.e., H.265 HEVC and H.264 AVC between 720p, 1080p and 2K. The immediate costs of actions $a_{t(i)}$ are given by -

$$\psi(s(n)_t, a(n)_{t(i)}) = \begin{cases} C_{switch}(s(n)_t, a(n)_{t(i)}), if a(n)_{t(i)} \geq 1 \\ C_{stable}(s(n)_t, a(n)_{t(i)}), if a(n)_{t(i)} = 0 \end{cases} \tag{6.9}$$

Where C_{switch} represents the cost of switching resolution of the video stream due to increased traffic in a particular state, and C_{stable} is the cost related to the change in network performance in the stable state. The total long term cost is the expected sum of all the components' immediate costs, given by -

$$\psi^{\pi}(s(n)) = \sum_{i \in \xi} \psi(s(n)_t, a(n)_{t(i)}) \tag{6.10}$$

Let d be the distance that the drones travel in the secure area, λ is the network wavelength and β is the bandwidth of the network and frequency lies in the range (2.4 GHz to 5.8 GHz). It is assumed that the best network conditions are available when drones hover in S_1, and transmit high resolution video. As the drones keep taking actions to reach the secure state, there is a possibility of a large number

of drones accumulating in the same space. This creates a traffic and β is over-utilized. This results in frame stalling, distortion and blurring of the video [59]. To effectively use β, the agent has to remain in the secure region and simultaneously continue to use the network protocol.

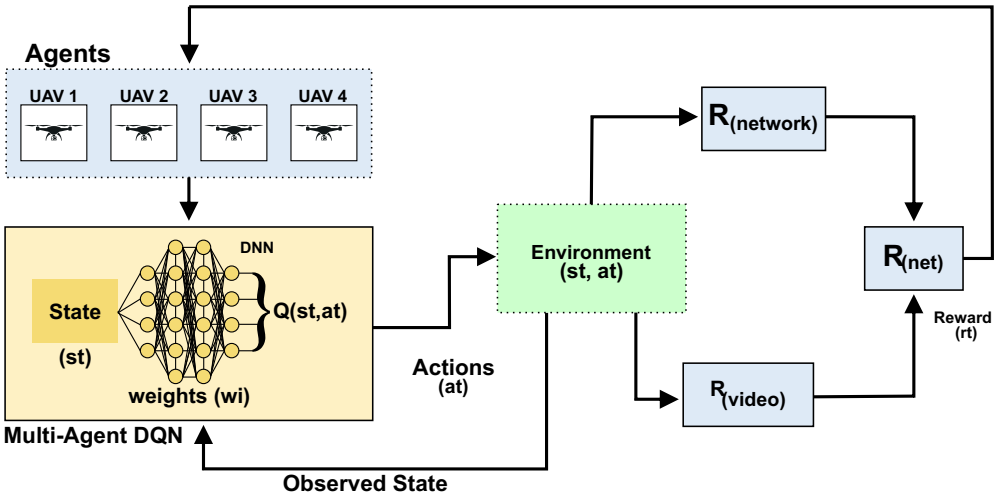


Figure 6.7: Deep Q Network based Intelligent Trajectory Optimization Design

The reward associated with the reliable connection establishment after entering a new state at cost C_{stable} is given by -

$$R[s(n)_t, a(n)_t]_{network} = \frac{\psi^{\pi}(s(n))}{\lambda} log_{10}(\beta) \tag{6.11}$$

The reward associated with the agent hovering over the secure state that allows for highest quality video resolution at cost C_{switch} is given by -

$$R[s(n)_t, a(n)_t]_{video} = \alpha \left[1 - [\frac{d}{d_{max}}]^{0.4}\right] . \psi^{\pi}(s(n)) \tag{6.12}$$

where α is the security parameter associated with S_1. The net reward function of a state-action episode is given as the sum of the two intermediate rewards.

$$R[s(n)_t, a(n)_t]_{net} = R[s(n)_t, a(n)_t]_{video} + R[s(n)_t, a(n)_t]_{network} \tag{6.13}$$

The trajectory learning of the drones occurs by maximizing the gain G_t along their path which is a function of expected cumulative discounted rewards.

$$G_t = E[\sum_{m=0}^{\infty} \gamma^m R_{net}(s(n)_t, a(n)_t)] \tag{6.14}$$

where γ is the discount factor ($0 \leq \gamma \leq 1$). Each action change may only produce a small reward. Thus, we require the value of the discount factor γ to be such that it maximizes the cumulative reward. From our empirical observations, our proposed DQN best converges at $\gamma = 0.8$. The optimal policy which maps the state-space and actions after the algorithm converges is obtained as -

$$\pi_t^* = argmin_\pi \sum_{i \in \xi} \psi^\pi(s(n)_t, a(n)_{t(i)}) \tag{6.15}$$

The optimal Q function is given as -

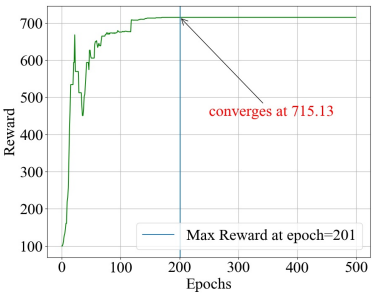
$$Q_\pi^*[s(n)_t, a(n)_t] = E[\sum_k \gamma^k (R_{net}|s(n), a(n))] \tag{6.16}$$

The optimal policy governs the convergence of the algorithm and leads the drones in intelligent traversals during their operational path in their optimal trajectory.

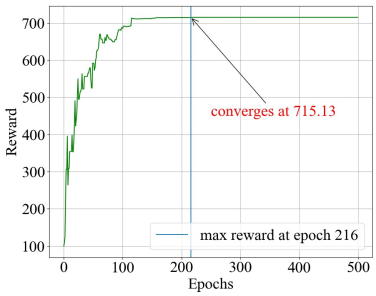
6.3 Online RL-based Model Evaluation

We have used a single-agent Q-learning algorithm to optimize the trajectory of the drone considering a discrete learning environment of states having secure and precarious surveillance area conditions in terms of network packet loss, video analytics impairments during a drone flight. We then implement a multi-agent Deep Q Network (DQN) that using similar policy by considering a continuous state space that allows the drones to explore and exploit the learning environment to the largest extent possible.

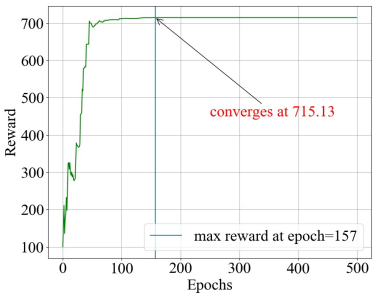
Our DQN has pre-defined weights that take state space values $(s(n)_t)$ of a drone as input, forward passes the values and generates optimal action value function $Q[s(n)_t, a(n)_t]$, and compares it with the optimal values of action value function $Q^*_\pi[s(n)_t, a(n)_t]$. While back-propagating, updates are performed to the weights of the neurons so that in the later iterations the output values come close to the optimal value. Once the optimal value is reached, our DQN algorithm converges. This way the agent learns to take actions and generates the optimal policy, following which the intelligent trajectory of the drone is obtained. This applies to all the drones involved in the operation. The DQN is modelled using a custom environment created using OpenAI's `gym` and is solved using the `keras-rl2` library that provides the DQN agent, Boltzmann Q policy and sequential memory, which is used as a replay memory to store the state-action pairs along with rewards for reinforcement learning based simulations. The constituent neural network model is created using Tensorflow. The model is sequential with Adam optimizer and the mean absolute error (MAE) is chosen as the loss function. Figures 6.8 shows the performance of the RL-based drone trajectory optimization algorithm for all the three possible state spaces for a drone to reach the terminal state during operation: (i) Random Area, (ii) Secure Area, and (iii) Precarious Area. The rewards $R(s_t, a_t)_{network}$ and $R(s_t, a_t)_{video}$ are calculated to give the network performance and video quality scores as $R(s_t, a_t)_{net}$. The terminal state is where the score becomes maximum and the algorithm converges. Since the Secure area has the strongest network performance, the drone reaches the terminal state quickly (in relatively less number of episodes) as compared to the Random state. In the Random state, the network performance is relatively weaker, and hence the drone reaches the terminal state later (in relatively higher number of episodes). The Precarious state is prone to network losses which causes poor video quality, and hence the drone reaches the terminal state much later (after several episodes) when compared to the other two states. Considering e.g., the drone flying over a network frequency of 4 GHz and supposing the maximum distance d_{max} travelled



(a) Random area



(b) Precarious area



(c) Secure area

Figure 6.8: Rewards obtained by a drone in various surveillance regions

is 5000m considering adequate available network bandwidth, the maximum score obtained at the terminal state is 715 (convergence level) for all the three state spaces.

Using a continuous state space that includes all three state spaces (s_t) i.e., Secure area, Random area and Precarious area and discrete actions (a_t) with different discount factors (γ) by trial and error method, we obtain the scores in the same manner as single-drone Q-learning method as shown in Figure 6.8. We observe that the scores are less at $\gamma = 0.6$ and subsequently the scores reach a maximum at $\gamma = 0.8$ for the multi-drone DQN as shown in Figure 6.9.

Thus, a simple Q-learning algorithm which generates a policy on finite MDP does not help. Given the complexity of our environment that accounts for both network and video protocols, the algorithm becomes non-convex and temporal credit assignment becomes hard which results in the algorithm yielding immediate extremely high or low rewards consecutively. The continuous state algorithm does not converge easily because any given drone keeps moving between different states

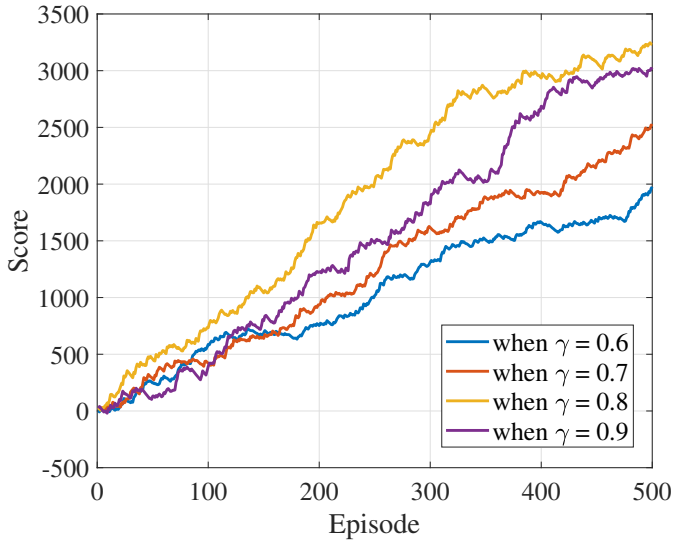


Figure 6.9: Incremental reward distribution with different discount factors (γ)

within the state space (s_t) throughout its trajectory. This phenomenon of lack of convergence can be overcome using the DQN. To evaluate the benefit of DQN, we trained the DQN, and its variants i.e., Double-DQN, and Dueling-DQN, each for 50,000 steps. We then tested the agent's parameters performance for every 500 steps, and saved the resulting network parameters for the best performance as shown in Figure 6.10. Although we find the performance of the three algorithms to be comparable with each other, the proposed DQN renders slightly higher scores. Therefore, we can conclude that DQN and its variants provide better trajectory optimization than the Q-learning approach in the *online* learning based approach. As we can observe from Figure 6.11, it is clear that - with the DQN model, trace-based experiments can achieve better throughput performance when employed in various application missions involving multi-drone co-ordination.

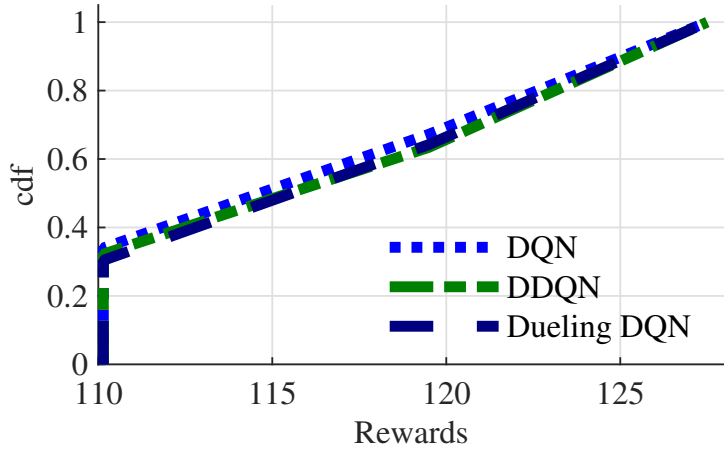


Figure 6.10: Cumulative rewards distribution of DQN, DDQN and Dueling DQN.

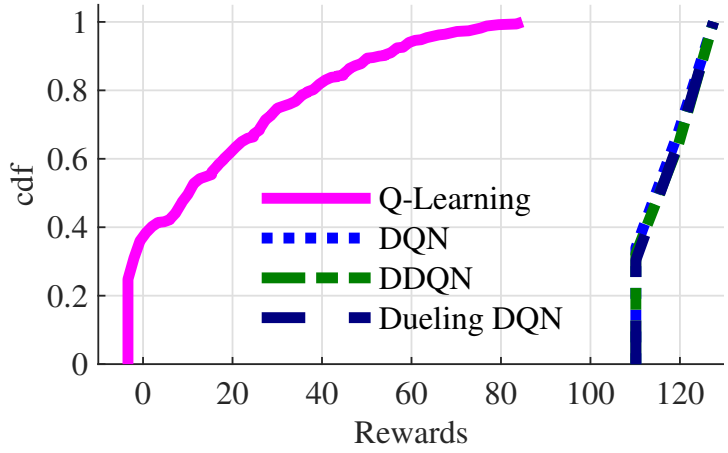


Figure 6.11: Comparision of Q Learning rewards with cumulative rewards distribution of DQN, DDQN and Dueling DQN.

Chapter 7

Conclusion

In this thesis, we have presented multi-UAV co-operation applications and explained how drone location prediction and trajectory optimization can be performed. We have learnt how location estimation prediction and trajectory optimization of drones can be beneficial in diverse application missions such as disaster response and other civil applications relating e.g., transportation. Various challenges in drone localization, path-planning and trajectory prediction were detailed.

To cope up with the challenges of localization of drones in application scenarios, we studied how techniques such as non-linear dynamic parameter state estimation of drones using distinct Kalman filtering techniques and sensor fusion can solve the drone localization and position prediction problem. We have also seen how Kalman filter can be used for position and velocity estimation of drones followed by location prediction with inter-drone distances and sensory measurements using the Extended Kalman filter. To cope up with sensory malfunctions and other inconsistencies of the filtering techniques, we detailed various machine learning techniques such as reinforcement learning and deep reinforcement learning. Furthermore, to cope up with the challenges of collision avoidance, trajectory optimization and path planning as well as handling of energy constraints, we have seen how a variety of reinforcement and deep reinforcement learning techniques can be used to realize the potential of multi-UAV co-operation.

Further, we presented a scenario corresponding to online orchestration and learning of network and video analytics for civil applications using multi-agent reinforcement learning techniques. These techniques feature prominent mechanisms that can be used for the 2-D and 3-D path-planning of UAVs along with network and resource allocation under bandwidth and energy constraints. Moreover, we discussed non-ML-based trajectory optimization techniques and explained how UAV-based applications can aid public safety networks.

The Road Ahead to More Open Challenges: We conclude this thesis with a list of more open challenges for multi-drone co-ordination in application missions. Addressing these challenges is essential for a variety of multi-drone applications such as aerial surveillance, deployment of UAVs as base stations and aerial mapping and monitoring that are relevant for location estimation and path planning. Few approaches such as [79] shows how joint positioning of UAVs as aerial base stations is done to provide a smart backhaul-fronthaul connectivity network. Other issues are shown in the following-

- **Excessive movements during flight with no hovering:** When the drones are in complex environments or unknown territories with unrealized threats, they tend to fly more rapidly and in different directions in a short span of time. This may be a result of collision avoidance of obstacles in the path or ineffectual attempts to explore the environment to learn threats. This leads to increased energy consumption and affects the battery capacity of drones, thus shortening their overall flight time. To avoid this issue, dynamic programming and scheduling algorithm could be useful if the drone flight plan in the mission is known apriori. The work in [80] provides two cases that show how data services using UAVs is maximized using hover time management for resource allocation, where the optimal hover time can be derived using service load requirements of ground users.

- **Air-resistance due to strong winds:** Severe wind gusts can throw the drones off-course and deviate a drone from following its optimal path. The

on-board sensors are subject to vibrations during severe wind conditions and can produce noisy data that may lead to inaccurate estimates of drones parameters. Unexpected wind resistance can also hinder the trajectory learning of the drone using DRL techniques. This hindrance is possible when the drone traversing in optimal path may change course due to the impacts wind. Further research on EKF and UKF based state estimation of gryoscope readings to study the effects of wind could help in developing suitable solutions. The approach in [81] addresses the altitude control problem of UAVs in presence of wind gusts and proposes a control strategy along with stability analysis to solve the issue of air-resistance.

- **Combining LSTMs with Kalman Filters and DQN:** The non-linear state etsimation of drone's dynamic parameters is done using individual time-steps of data by on-board sensors and use of the Kalman filter. Also, for the DRL techniques, the drone (agent) takes actions in a given state in independent episodes. Long short term memorys (LSTMs) can be used to utilize the information of previous time-steps of drones instead of just one time step or one episode to make predictions. This way LSTM based Kalman Filtering mechanisms and LSTMs based DRL mechanisms can use past information of the drone(s) and make much accurate predictions. There are works that show how coupling a Kalman Filter with LSTM network improves performance and provides faster convergence of algorithms for various application purposes [82, 83].

- **Multi-drone Co-ordination under energy constraints:** In missions involving a drone swarm or a fleet of drones, it is difficult to monitor each of the drones' parameters. Factors such as malfunctioning or loss of one drone due to total battery utilization can affect the operation of other drones and compromise the overall application mission. Off-line path planning along with online path-planning can help UAVs find the nearest base stations with recharge units and help alleviate this issue and support multi-drone

co-ordination even under available energy limitations. One such approach to solve the issue of multi-drone coordination under energy constraints is detailed in [84].

Bibliography

[1] Yoo, S., Kim, K., Jung, J., Chung, A. Y., Lee, J., Lee, S. K., ... & Kim, H. (2015, September). Poster: A multi-drone platform for empowering drones' teamwork. In Proceedings of the 21st Annual International Conference on Mobile Computing and Networking (pp. 275-277).

[2] Sorbelli, F. B., Corò, F., Das, S. K., & Pinotti, C. M. (2020). Energy-constrained delivery of goods with drones under varying wind conditions. IEEE Transactions on Intelligent Transportation Systems.

[3] Abiodun, T. F. (2020). Usage of Drones or Unmanned Aerial Vehicles (UAVs) for Effective Aerial Surveillance, Mapping System and Intelligence Gathering in Combating Insecurity in Nigeria. African Journal of Social Sciences and Humanities Research, 3(2), 29-44.

[4] Bor-Yaliniz, R. I., El-Keyi, A., & Yanikomeroglu, H. (2016, May). Efficient 3-D placement of an aerial base station in next generation cellular networks. In 2016 IEEE international conference on communications (ICC) (pp. 1-5). IEEE.

[5] Mayor, V., Estepa, R., Estepa, A., & Madinabeitia, G. (2019). Deploying a reliable UAV-aided communication service in disaster areas. Wireless Communications and Mobile Computing, 2019.

[6] Mishra, B., Garg, D., Narang, P., & Mishra, V. (2020). Drone-surveillance for search and rescue in natural disaster. Computer Communications, 156, 1-10.

[7] Kim, G. H., Nam, J. C., Mahmud, I., & Cho, Y. Z. (2016, July). Multi-drone control and network self-recovery for flying Ad Hoc Networks. In 2016 Eighth International Conference on Ubiquitous and Future Networks (ICUFN) (pp. 148-150). IEEE.

[8] Ribeiro, M. I. (2004). Kalman and extended kalman filters: Concept, derivation and properties. Institute for Systems and Robotics, 43, 46.

[9] Kalman, R. E. (1960). A new approach to linear filtering and prediction problems.

[10] Qu, C., Calyam, P., Yu, J., Vandanapu, A., Opeoluwa, O., Gao, K., ... & Palaniappan, K. (2021). DroneCOCoNet: Learning-based edge computation offloading and control networking for drone video analytics. Future Generation Computer Systems, 125, 247-262.

[11] Strehl, A. L., Li, L., Wiewiora, E., Langford, J., & Littman, M. L. (2006, June). PAC model-free reinforcement learning. In Proceedings of the 23rd international conference on Machine learning (pp. 881-888).

[12] Li, Y. (2017). Deep reinforcement learning: An overview. arXiv preprint arXiv:1701.07274.

[13] Liu, W., Gu, W., Sheng, W., Meng, X., Wu, Z., & Chen, W. (2014). Decentralized multi-agent system-based cooperative frequency control for autonomous microgrids with communication constraints. IEEE Transactions on Sustainable Energy, 5(2), 446-456.

[14] M. Abouheaf, W. Gueaieb, and F. Lewis, "Online model-free reinforcement learning for the automatic control of a flexible wing aircraft," IET Control Theory & Applications, vol. 14, no. 1, pp. 73–84, 2020.

[15] Zhu, P., Wen, L., Bian, X., Ling, H., & Hu, Q. (2018). Vision meets drones: A challenge. arXiv preprint arXiv:1804.07437.

[16] Rana, T., Shankar, A., Sultan, M. K., Patan, R., & Balusamy, B. (2019, January). An intelligent approach for UAV and drone privacy security using blockchain methodology. In 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence) (pp. 162-167). IEEE.

[17] Samland, F., Fruth, J., Hildebrandt, M., Hoppe, T., & Dittmann, J. (2012, January). AR. Drone: security threat analysis and exemplary attack to track persons. In Intelligent Robots and Computer Vision XXIX: Algorithms and Techniques (Vol. 8301, p. 83010G). International Society for Optics and Photonics.

[18] Rodday, N. M., Schmidt, R. D. O., & Pras, A. (2016, April). Exploring security vulnerabilities of unmanned aerial vehicles. In NOMS 2016-2016 IEEE/IFIP Network Operations and Management Symposium (pp. 993-994). IEEE.

[19] Di Franco, C., & Buttazzo, G. (2015, April). Energy-aware coverage path planning of UAVs. In 2015 IEEE international conference on autonomous robot systems and competitions (pp. 111-117). IEEE.

[20] Ware, J., & Roy, N. (2016, May). An analysis of wind field estimation and exploitation for quadrotor flight in the urban canopy layer. In 2016 IEEE International Conference on Robotics and Automation (ICRA) (pp. 1507-1514). IEEE.

[21] Artemenko, O., Dominic, O. J., Andryeyev, O., & Mitschele-Thiel, A. (2016, August). Energy-aware trajectory planning for the localization of mobile devices using an unmanned aerial vehicle. In 2016 25th international conference on computer communication and networks (ICCCN) (pp. 1-9). IEEE.

[22] Kouroshnezhad, S., Peiravi, A., Haghighi, M. S., & Jolfaei, A. (2020). An energy-aware drone trajectory planning scheme for terrestrial sensors localization. Computer Communications, 154, 542-550.

[23] Ivancic, W. D., Kerczewski, R. J., Murawski, R. W., Matheou, K., & Downey, A. N. (2019, April). Flying drones beyond visual line of sight using 4g LTE: Issues and concerns. In 2019 Integrated Communications, Navigation and Surveillance Conference (ICNS) (pp. 1-13). IEEE.

[24] Kato, N., Kawamoto, Y., Aneha, A., Yaguchi, Y., Miura, R., Nakamura, H., ... & Kitashima, A. (2019). Location awareness system for drones flying beyond visual line of sight exploiting the 400 MHz frequency band. IEEE Wireless Communications, 26(6), 149-155.

[25] Xiong, J. J., & Zheng, E. H. (2015). Optimal kalman filter for state estimation of a quadrotor UAV. Optik, 126(21), 2862-2868.

[26] Fujii, K. (2013). Extended kalman filter. Refernce Manual, 14-22.

[27] Julier, S. J., & Uhlmann, J. K. (1997, July). New extension of the Kalman filter to nonlinear systems. In Signal processing, sensor fusion, and target recognition VI (Vol. 3068, pp. 182-193). International Society for Optics and Photonics.

[28] Wu, Z., Li, J., Zuo, J., & Li, S. (2018). Path planning of UAVs based on collision probability and Kalman filter. IEEE Access, 6, 34237-34245.

[29] Abdelkrim, N., Aouf, N., Tsourdos, A., & White, B. (2008, June). Robust nonlinear filtering for INS/GPS UAV localization. In 2008 16th Mediterranean Conference on Control and Automation (pp. 695-702). IEEE.

[30] Mao, G., Drake, S., & Anderson, B. D. (2007, February). Design of an extended kalman filter for uav localization. In 2007 Information, Decision and Control (pp. 224-229). IEEE.

[31] St-Pierre, M., & Gingras, D. (2004, June). Comparison between the unscented Kalman filter and the extended Kalman filter for the position estimation module of an integrated navigation information system. In IEEE Intelligent Vehicles Symposium, 2004 (pp. 831-835). IEEE.

[32] Kraft, E. (2003, July). A quaternion-based unscented Kalman filter for orientation tracking. In Proceedings of the Sixth International Conference of Information Fusion (Vol. 1, No. 1, pp. 47-54). IEEE Cairns, Queensland, Australia.

[33] Tang, S. H., Kojima, T., Namerikawa, T., Yeong, C. F., & Su, E. L. M. (2015, July). Unscented Kalman filter for position estimation of UAV by using image information. In 2015 54th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE) (pp. 695-700). IEEE.

[34] Nemra, A., & Aouf, N. (2010). Robust INS/GPS sensor fusion for UAV localization using SDRE nonlinear filtering. IEEE Sensors Journal, 10(4), 789-798.

[35] Abdelfatah, R., Moawad, A., Alshaer, N., & Ismail, T. (2021, May). UAV Tracking System Using Integrated Sensor Fusion with RTK-GPS. In 2021 International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC) (pp. 352-356). IEEE.

[36] Gurvits, L., & Ledoux, J. (2005). Markov property for a function of a Markov chain: A linear algebra approach. Linear algebra and its applications, 404, 85-117.

[37] Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., & Riedmiller, M. (2013). Playing atari with deep reinforcement learning. arXiv preprint arXiv:1312.5602.

[38] Tampuu, A., Matiisen, T., Kodelja, D., Kuzovkin, I., Korjus, K., Aru, J., ... & Vicente, R. (2017). Multiagent cooperation and competition with deep reinforcement learning. PloS one, 12(4), e0172395.

[39] Sutton, R. S., & Barto, A. G. (2018). Reinforcement learning: An introduction. MIT press.

[40] Roderick, M., MacGlashan, J., & Tellex, S. (2017). Implementing the deep q-network. arXiv preprint arXiv:1711.07478.

[41] González, R. L. V., & Aragone, L. S. (2000). A Bellman's equation for minimizing the maximum cost. Indian Journal of Pure and Applied Mathematics, 31(12), 1621-1632.

[42] Osband, I., Blundell, C., Pritzel, A., & Van Roy, B. (2016). Deep exploration via bootstrapped DQN. Advances in neural information processing systems, 29, 4026-4034.

[43] Koushik, A. M., Hu, F., & Kumar, S. (2019). Deep Q-Learning-Based Node Positioning for Throughput-Optimal Communications in Dynamic UAV Swarm Network. IEEE Transactions on Cognitive Communications and Networking, 5(3), 554-566.

[44] Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., ... & Kavukcuoglu, K. (2016, June). Asynchronous methods for deep reinforcement learning. In International conference on machine learning (pp. 1928-1937). PMLR.

[45] Zhao, L., Ma, Y., & Zou, J. (2020, October). 3D Path Planning for UAV with Improved Double Deep Q-Network. In Chinese Intelligent Systems Conference (pp. 374-383). Springer, Singapore.

[46] Wang, Z., Schaul, T., Hessel, M., Hasselt, H., Lanctot, M., & Freitas, N. (2016, June). Dueling network architectures for deep reinforcement learning. In International conference on machine learning (pp. 1995-2003). PMLR.

[47] Zeng, Y., Xu, X., Jin, S., & Zhang, R. (2021). Simultaneous navigation and radio mapping for cellular-connected UAV with deep reinforcement learning. IEEE Transactions on Wireless Communications.

[48] Yan, C., Xiang, X., & Wang, C. (2020). Towards real-time path planning through deep reinforcement learning for a UAV in dynamic environments. Journal of Intelligent & Robotic Systems, 98(2), 297-309.

[49] Grondman, I., Busoniu, L., Lopes, G. A., & Babuska, R. (2012). A survey of actor-critic reinforcement learning: Standard and natural policy gradients. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 42(6), 1291-1307.

[50] Konda, V. R., & Tsitsiklis, J. N. (2000). Actor-critic algorithms. In Advances in neural information processing systems (pp. 1008-1014).

[51] Hou, Y., Liu, L., Wei, Q., Xu, X., & Chen, C. (2017, October). A novel DDPG method with prioritized experience replay. In 2017 IEEE international conference on systems, man, and cybernetics (SMC) (pp. 316-321). IEEE.

[52] Ding, R., Gao, F., & Shen, X. S. (2020). 3D UAV trajectory design and frequency band allocation for energy-efficient and fair communication: A deep reinforcement learning approach. IEEE Transactions on Wireless Communications, 19(12), 7796-7809.

[53] Zhao, Y. J., Zheng, Z., Zhang, X. Y., & Liu, Y. (2017). Q learning algorithm based UAV path learning and obstacle avoidence approach. In 2017 36th Chinese Control Conference (CCC) IEEE.

[54] Coggan, M. (2004). Exploration and exploitation in reinforcement learning. Research supervised by Prof. Doina Precup, CRA-W DMP Project at McGill University.

[55] Sutton, R. S., & Barto, A. G. (2018). Reinforcement learning: An introduction. MIT press.

[56] Saxena, V., Jaldén, J., & Klessig, H. (2019). Optimal UAV base station trajectories using flow-level models for reinforcement learning. IEEE Transactions on Cognitive Communications and Networking, 5(4), 1101-1112.

[57] Yin, S., Zhao, S., Zhao, Y., & Yu, F. R. (2019). Intelligent trajectory design in UAV-aided communications with reinforcement learning. IEEE Transactions on Vehicular Technology, 68(8), 8227-8231.

[58] Nguyen, K. K., Vien, N. A., Nguyen, L. D., Le, M. T., Hanzo, L., & Duong, T. Q. (2020). Real-time energy harvesting aided scheduling in UAV-assisted D2D networks relying on deep reinforcement learning. IEEE Access, 9, 3638-3648.

[59] Ramisetty, R. R., Qu, C., Aktar, R., Wang, S., Calyam, P., & Palaniappan, K. (2020, January). Dynamic computation off-loading and control based on occlusion detection in drone video analytics. In Proceedings of the 21st International Conference on Distributed Computing and Networking (pp. 1-10).

[60] Sujit, P. B., & Ghose, D. (2004). Search using multiple UAVs with flight time constraints. IEEE Transactions on Aerospace and Electronic Systems, 40(2), 491-509.

[61] Langelaan, J. (2007, August). Long distance/duration trajectory optimization for small UAVs. In AIAA Guidance, Navigation and Control Conference and Exhibit (p. 6737).

[62] Lakew, D. S., Masood, A., & Cho, S. (2020, January). 3D UAV Placement and Trajectory Optimization in UAV Assisted Wireless Networks. In 2020 International Conference on Information Networking (ICOIN) (pp. 80-82). IEEE.

[63] Xu, D., Sun, Y., Ng, D. W. K., & Schober, R. (2020). Multiuser MISO UAV communications in uncertain environments with no-fly zones: Robust trajectory and resource allocation design. IEEE Transactions on Communications, 68(5), 3153-3172.

[64] Koyuncu, E., Shabanighazikelayeh, M., & Seferoglu, H. (2018). Deployment and trajectory optimization of UAVs: A quantization theory approach. IEEE Transactions on Wireless Communications, 17(12), 8531-8546.

[65] Shakoor, S., Kaleem, Z., Do, D. T., Dobre, O. A., & Jamalipour, A. (2020). Joint optimization of UAV 3D placement and path loss factor for energy efficient maximal coverage. IEEE Internet of Things Journal.

[66] Guo, Y., You, C., Yin, C., & Zhang, R. (2021). UAV trajectory and communication co-design: Flexible path discretization and path compression. IEEE Journal on Selected Areas in Communications.

[67] Zhang, S., Zeng, Y., & Zhang, R. (2018). Cellular-enabled UAV communication: A connectivity-constrained trajectory optimization perspective. IEEE Transactions on Communications, 67(3), 2580-2604.

[68] Yang, D., Dan, Q., Xiao, L., Liu, C., & Cuthbert, L. (2021). An efficient trajectory planning for cellular-connected UAV under the connectivity constraint. China Communications, 18(2), 136-151.

[69] Teng, H., Ahmad, I., Msm, A., & Chang, K. (2020). 3D optimal surveillance trajectory planning for multiple UAVs by using particle swarm optimization with surveillance area priority. IEEE Access, 8, 86316-86327.

[70] Fantacci, R., Gei, F., Marabissi, D., & Micciullo, L. (2016). Public safety networks evolution toward broadband: Sharing infrastructures and spectrum with commercial systems. IEEE Communications Magazine, 54(4), 24-30.

[71] Laoudias, C., Moreira, A., Kim, S., Lee, S., Wirola, L., & Fischione, C. (2018). A survey of enabling technologies for network localization, tracking, and navigation. IEEE Communications Surveys & Tutorials, 20(4), 3607-3644.

[72] S. A. R. Naqvi, S. A. Hassan, H. Pervaiz, and Q. Ni, "Drone-aided communication as a key enabler for 5g and resilient public safety networks," IEEE Communications Magazine, vol. 56, no. 1, pp. 36–42, 2018.

[73] Do-Duy, T., Nguyen, L. D., Duong, T. Q., Khosravirad, S., & Claussen, H. (2021). Joint Optimisation of Real-time Deployment and Resource Allocation for UAV-Aided Disaster Emergency Communications. IEEE Journal on Selected Areas in Communications.

[74] Shakoor, S., Kaleem, Z., Baig, M. I., Chughtai, O., Duong, T. Q., & Nguyen, L. D. (2019). Role of UAVs in public safety communications: Energy efficiency perspective. IEEE Access, 7, 140665-140679.

[75] Mozaffari, M., Saad, W., Bennis, M., Nam, Y. H., & Debbah, M. (2019). A tutorial on UAVs for wireless networks: Applications, challenges, and open problems. IEEE communications surveys & tutorials, 21(3), 2334-2360.

[76] Kaleem, Z., Yousaf, M., Qamar, A., Ahmad, A., Duong, T. Q., Choi, W., & Jamalipour, A. (2019). UAV-empowered disaster-resilient edge architecture for delay-sensitive communication. IEEE Network, 33(6), 124-132.

[77] Do, D. T., Nguyen, T. T. T., Le, C. B., Voznak, M., Kaleem, Z., & Rabie, K. M. (2020). UAV relaying enabled NOMA network with hybrid duplexing and multiple antennas. IEEE Access, 8, 186993-187007.

[78] He, D., Chan, S., & Guizani, M. (2017). Drone-assisted public safety networks: The security aspect. IEEE Communications Magazine, 55(8), 218-223.

[79] Shehzad, M. K., Ahmad, A., Hassan, S. A., & Jung, H. (2021). Backhaul-Aware Intelligent Positioning of UAVs and Association of Terrestrial Base Stations for Fronthaul Connectivity. IEEE Transactions on Network Science and Engineering.

[80] Mozaffari, M., Saad, W., Bennis, M., & Debbah, M. (2017). Wireless communication using unmanned aerial vehicles (UAVs): Optimal transport theory for hover time optimization. IEEE Transactions on Wireless Communications, 16(12), 8052-8066.

[81] Shi, D., Wu, Z., & Chou, W. (2018). Super-twisting extended state observer and sliding mode controller for quadrotor uav attitude system in presence of wind gust and actuator faults. Electronics, 7(8), 128.

[82] Pérez-Ortiz, J. A., Gers, F. A., Eck, D., & Schmidhuber, J. (2003). Kalman filters improve LSTM network performance in problems unsolvable by traditional recurrent nets. Neural Networks, 16(2), 241-250.

[83] Coskun, H., Achilles, F., DiPietro, R., Navab, N., & Tombari, F. (2017). Long short-term memory kalman filters: Recurrent neural estimators for pose regularization. In Proceedings of the IEEE International Conference on Computer Vision (pp. 5524-5532).

[84] Scherer, J., & Rinner, B. (2016, August). Persistent multi-UAV surveillance with energy and communication constraints. In 2016 IEEE international conference on automation science and engineering (CASE) (pp. 1225-1230). IEEE.