Journal of Network and Computer Applications 201 (2022) 103341

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Survey on computation offloading in UAV-Enabled mobile edge computing☆

S.M. Asiful Huda, Sangman Moh *

Department of Computer Engineering, Chosun University, Gwangju, 61452, South Korea

ARTICLE INFO

Index Terms:
Computation offloading
Convex optimization
Deep reinforcement learning
Mobile edge computing
Unmanned aerial vehicle

ABSTRACT

With the increasing growth of internet-of-things (IoT) devices, effective computation performance has become a critical issue. Many services provided by IoT devices (e.g., augmented reality, location-tracking, traffic systems, and autonomous driving) require intensive real-time data processing, which demands powerful computational resources. Mobile edge computing (MEC) has been introduced to effectively handle this problem reliably over the internet. The inclusion of a MEC server allows computationally intensive tasks to be offloaded from IoT devices. However, communication overhead and delays are major drawbacks. With the advantages of high mobility and low cost, unmanned aerial vehicles (UAVs) can mitigate this issue by acting as MEC servers. The offloading decisions for such scenarios involve service latency, energy/power consumption, and execution delays. For this reason, this study reviews UAV-enabled MEC solutions in which offloading was the focus of research. We compare the algorithms qualitatively to assess features and performance. Finally, we discuss open issues and research challenges in terms of design and implementation.

1. Introduction

THE increasing growth of smart devices has resulted in a dramatic change in society, which now heavily relies on cellular technologies. Online streaming services and social-networking sites have become popular and useful across all demographics. In 2020, Cisco predicted that the amount of cellular data traffic would increase seven-fold by the end of 2021 (Cisco and Internet, 2020). The resultant data increase has created intense burdens for mobile service providers. Without proper measures for storing and processing such workloads, cellular networks will become even more congested, resulting in deteriorated quality and slower download speeds. Hence, additional computational resources are necessary for mobile devices. Furthermore, the long-term evolution of 5G technology has inspired a wide range of services that require high computational tasks, for which the designated devices are ill-equipped to handle (Diao et al., 2021).

Mobile edge computing (MEC) is a promising solution that leverages cloud servers deployed in support of mobile devices to mitigate computational workloads via process offloading. In 2009, the first edge-computing concept (i.e., cloudlet) was proposed. Cloudlets allow mobile users to take advantage of cloud services, but they require users to swap between Wi-Fi and cellular networks during use (Barbarossa et al., 2014).

Tasks can also be executed locally on mobile devices by leveraging the concept of an *ad hoc* cloud (Yaqoob et al., 2016). It enables multiple user devices to combine their computational resources to process tasks. Notably, offloading to an edge server directly improves the quality of experience and battery lifetime (Zheng et al., 2019). In 2012, Cisco proposed the concept of computation offloading (Bonomi et al., 2012). Hence, any mobile device having constrained resources can wirelessly pass processing tasks to other devices having sufficient resources. Those other devices then complete the tasks and transmit the results back to the mobile devices. Unfortunately, this method continues to fall short of expectations, owing to the characteristics of wireless networks in rural and mountainous areas. Furthermore, cases of emergency response should always take precedence. Thus, even in the most ideal environments, maintaining the quality of experience and energy efficiency while avoiding communication delays is difficult.

☆ This research was supported in part by a National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (No. 2019R1F1A1060501).

* Corresponding author.
E-mail address: smmoh@chosun.ac.kr (S. Moh).

https://doi.org/10.1016/j.jnca.2022.103341
Received 6 October 2021; Received in revised form 29 December 2021; Accepted 24 January 2022
Available online 1 February 2022
1084-8045/

1.1. Motivation of our study

Unmanned aerial vehicle (UAV)-enabled MEC servers have emerged as promising candidates for handling computationally intensive task loads in areas lacking ground infrastructure. UAVs have gained significant research attention since their extensive use in military and civil areas, owing to their fast deployment and low cost. A large number of studies have been carried out in this field and have focused on the establishment of communication protocols and localization solutions (Arafat et al., 2020, 2021; Arafat and Moh, 2019a, 2019b, 2019c, 2021a, 2021b; Habib et al., 2018; Mach and Becvar, 2017). In Ref. (Arafat and Moh, 2019a), the authors studied cluster-based routing protocols and compared them to several qualitative parameters. In Refs. (Arafat and Moh, 2019b; Habib et al., 2018), the authors addressed the problems of designing an effective routing protocol for communications between UAV networks focusing on reinforcement learning (RL)-based routing protocols and those of UAV-based wireless sensor networks. The main advantage of UAVs is that they can be deployed anywhere, which enables them to support recovery operations in disaster-prone areas as well as augment several dynamic network technologies (Gupta et al., 2016). UAVs can send raw images from affected areas to any base station (BS) for fast mobility (Arafat and Moh, 2018). It was recently noted that UAVs can also act as MEC servers or MEC relays. UAV-mounted MEC relay services use line-of-sight (LoS) links to transmit computational tasks to MEC servers situated on the ground as they hover over areas where it is otherwise challenging to set up a cloud or edge computing-based solutions.

In military applications, the vital task is to capture information from a given region and send it to the nearest BS to identify or track objects from a distance, which generally is very intricate and burdensome for any human. UAVs can also provide target-tracking geographical data-capture services. A UAV-aided wireless sensor network is another promising paradigm to enhance the energy efficiency of the sensor nodes in collecting data in various areas of commercial applications (Poudel and Moh, 2019) However, transmission delays are hindrances because longer processing times result in late responses. Integrating MEC servers into UAVs effectively eradicates this problem. Furthermore, UAVs can simultaneously operate as computation servers and relay nodes (X. Hu et al., 2019), resulting in faster decision-making during emergency scenarios while increasing efficiency with potential real-time computational capabilities. Notably in such scenarios, UAVs can either process computational tasks locally, or they can offload them to other edge servers.

1.2. Related surveys

Computation offloading is a highly trending topic, and there have been several surveys that emphasized various aspects of offloading in edge-computing environments. In Ref. (F. F. Zhou et al., 2020), the authors provided an overview of recent advances, open issues, and application scenarios of UAVs in MEC environments, and they addressed how UAVs could be utilized as edge servers for offloading. In Ref. (Jiang et al., 2019), the authors provided an overview of a state-of-the-art computational task offloading technique by focusing on performance metrics, such as energy consumption minimization, to ensure quality-of-service while incorporating resource allocation strategies. Offloading based on specific application goals was studied in Ref. (Xu et al., 2018). There, the authors provided a comprehensive summary of traffic and computation offloading tasks based on extant literature. Machine-learning computation offloading approaches were studied in Ref (Shakarami et al., 2020a). wherein the authors compared the literature to the utilized techniques, features, performance metrics, strengths, and weaknesses.

Unlike those works, offloading in dynamic environments (e.g., vehicular) has been investigated to address challenges. Using a taxonomy of communication media and design objectives, modern experiments presented in the literature were summarized in Ref. (de Souza et al., 2020). The authors of Ref (Lin et al., 2020). surveyed basic models from communication, computation, and energy-harvesting perspectives. An overview of current research progress alongside machine-learning techniques was provided in Ref. (Rodrigues et al., 2020). In Ref. (Bhattacharya and De, 2017), the authors analyzed how an offloading system could adapt so that an even more robust and scalable capability could be designed while ensuring the quality of experience. The authors of Ref (Shakarami et al., 2020b). focused particularly on the stochastic behaviors of offloading techniques enabled by mobile-application variances. The authors discussed stochastic offloading mechanisms and their respective environments, including mobile cloud, fog, and edge computing.

A detailed and comprehensive survey was conducted by the authors of Ref. (Mach and Becvar, 2017), in which they reviewed recent developments of edge computing from architectural offloading and computation perspectives. Offloading was further discussed concerning offloading decisions, which are based upon resource allocation and mobility management. A game-theory-based offloading mechanism was surveyed using a novel taxonomy in Ref. (Shakarami et al., 2020c). There, the authors discussed critical factors (e.g., key metrics, application scenarios, implemented techniques, and tools) of the offloading mechanisms in a MEC environment. The authors in (Abrar et al., 2021) provides a brief survey on UAV-enabled MEC network, where they present the energy-efficient techniques on IoT devices in terms of channel access schemes for computation offloading and resource management. UAV-aided relaying was not considered in this study. Machine learning and deep learning based techniques in UAV-enabled MEC networks were reviewed comprehensively by the authors in (Yazid et al., 2021). However, this survey doesn't cover the widely used optimization techniques in UAV-MEC architecture and only focuses on artificial intelligence-based approaches. In Table 1, we provide a summary on the related surveys on computation offloading in edge computing.

The existing surveys discussed above emphasize the computation

Table 1
Existing surveys on computation offloading in edge computing.

Article	Year	Focused area
Ref. (Mach and Becvar, 2017)	2017	Discusses the mobile edge computing architecture and computation offloading concerning offloading decision, resource allocation, and mobility
Ref. (Bhattacharya and De, 2017)	2017	Discusses the adaptation approaches that are widely used in task offloading problems
Ref. (Xu et al., 2018)	2018	Investigates traffic and computation offloading, and compares them in terms of different characteristic
Ref. (Jiang et al., 2019)	2019	Surveys several offloading aspects in edge computing
Ref. (Shakarami et al., 2020a)	2020	Surveys machine learning-based offloading techniques in the mobile edge computing environment
Ref. (de Souza et al., 2020)	2020	Surveys the challenges faced in a vehicular environment from offloading perspective
Ref. (Lin et al., 2020)	2020	Reviews and classifies the offloading approaches in terms of the channel model, communication model, computation model, and energy harvesting model
Ref. (Rodrigues et al., 2020)	2020	Addressed the challenges to guarantee the quality of service in an edge computing environment.
Ref. (Shakarami et al., 2020b)	2020	Surveys stochastic offloading mechanisms in terms of strength and weakness
Ref. (Shakarami et al., 2020c)	2020	Reviews the game theory-based offloading techniques in mobile edge computing
Ref (Abrar et al., 2021)	2021	Discusses energy-efficient resource management techniques in IoT devices using UAV-enabled MEC networks
Ref (Yazid et al., 2021)	2021	Reviews machine learning and deep learning techniques used in the applications of UAV-enabled MEC architecture

offloading techniques in an edge computing environment from various perspectives. Because the location of the ground MEC server is usually fixed (Y. Wang et al., 2020) and it cannot be shifted following mobile users, this is a major drawback of the traditional MEC server. In such context, UAV can be regarded as a promising solution by acting as a potential MEC server providing communication and computation resources, where a terrestrial MEC network does not exist or has been destroyed due to natural disaster. Owing to its mobility, easy deployment, and low cost, UAVs can be easily utilized in emergency rescue operations (Poudel and Moh, 2021), military reconnaissance, and desert areas. Thus, it is very interesting and essential to have a comprehensive review of the computation offloading techniques on the UAV-MEC networks. Our main objective is to extensively discuss state-of-the-art offloading techniques with their respective advantages and limitations. In addition, we also point out the lessons learned from this study and conclude the study by highlighting the future research directions for motivating further research in this field. A detailed summary of the contribution of this study is presented in the following subsection.

1.3. Contribution

The major contributions of this study are summarized as follows.

- First, we provide a brief overview of the background information on UAVs, the difference between single-UAV and multi-UAV systems, and the interaction between UAV and the MEC so that general readers can obtain a notion of how UAVs can be integrated into the MEC system.
- Second, we study the application scenarios of UAV-enabled MEC systems. In addition, we provide a case study where UAV is deployed for crowd surveillance. A comparison is provided for readers to understand the design considerations for such applications.
- Third, we discuss the design issues of offloading algorithms for UAV-enabled MEC systems in the perspectives of computation, security, cost, and mobility.
- Fourth, we provide an overview of the basic network architecture of UAV-enabled MEC systems consisting of system layer, UAV layer and device layer.
- Fifth, we derive a taxonomy for classifying the existing state-of-the-art offloading techniques in terms of offloading decision. The offloading decision includes binary offloading, partial offloading, and load relaying. Each category of the offloading algorithms is also classified into different subcategories in terms of the type of the respective algorithm. We extensively review and discuss a total of twenty-eight state-of-the-art offloading algorithms in terms of the basic operation principles, characteristics, advantages, and limitations.
- Sixth, we qualitatively compare the offloading algorithms in terms of the design approach, main idea, and operational characteristics. In addition, we provide sufficient discussion based on the comparison to provide an idea of selecting the appropriate algorithm.
- Seventh, we provide lessons learned where we point out several factors which researchers must consider before designing an offloading algorithm for ensuring the desired performance.
- Finally, we conclude our study with crucial open issues and research challenges for motivating further enhancement and development in the field of UAV-enabled MEC systems.

1.4. Organization of the study

This survey comprises eight sections, as shown in Fig. 1. We provide background information on the UAV and its interaction with MEC in Section 2. In Section 3, we discuss application scenarios of UAV-enabled MEC systems and provide a case study where UAV is deployed for crowd surveillance. In Section 4, we provide some design issues for offloading in the UAV–MEC environment. We then provide the basic network architecture of UAV-enabled MEC systems in Section 5. In Section 6, the classification of computation offloading algorithms in terms of the offloading decision is discussed comprehensively. In Section 7, we qualitatively compare existing techniques in terms of objectives, focused parameters, and performance. In Section 8, we discuss lessons-learned and future research directions. Finally, we conclude this study in Section 9.

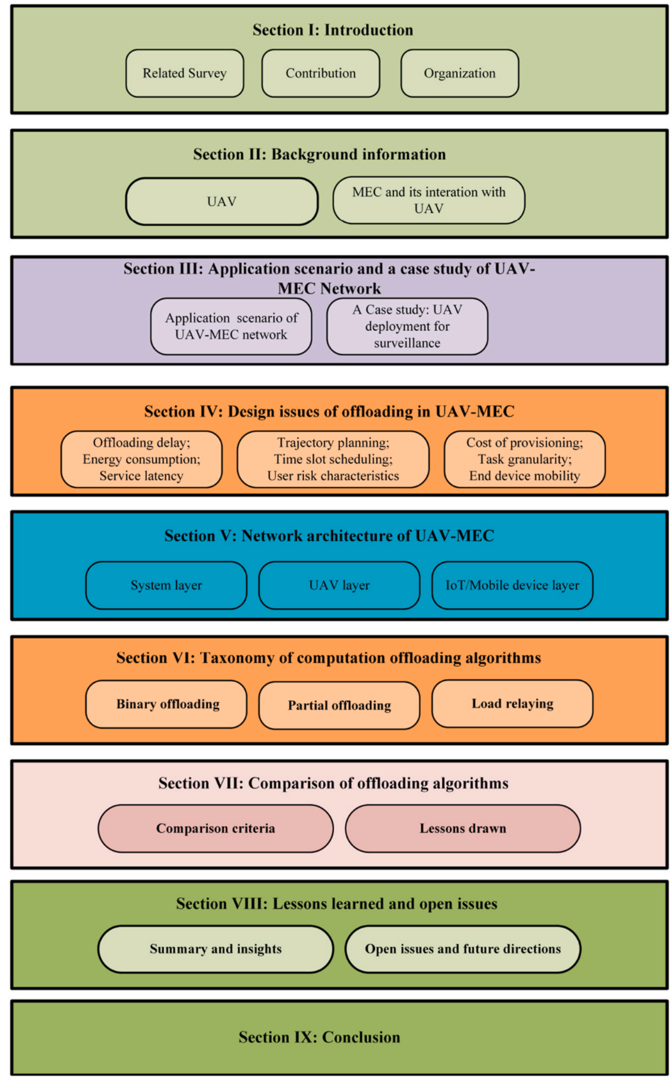

Fig. 1. Outline of this survey.

2. Background information

Here, we provide a brief overview of UAV networks and their interactions with MEC.

2.1. UAVs

UAVs are airborne systems that can be controlled remotely or autonomously. Owing to their versatility, easy deployment, and low cost, UAVs are used widely for both civil and military applications. In military arenas, they are typically used for surveillance, reconnaissance, and strike missions. The type of platform varies depending on mission characteristics. Mission objectives must account for the complete area of operations, the precise locations of targets, mission durations, and payloads. For real-time video surveillance operations, multiple collaborating UAVs may be used (Gupta et al., 2016). Hence, the planning and coordination challenges are vast. In Table 2, we present a comparison between single- and multi-UAV systems.

Although UAV networks have several unique characteristics, many studies consider them to be *ad hoc* networks. This is not entirely accurate. First, UAV network architectures have not been well-studied, and in multi-UAV scenarios, multiple topologies are applied (Gupta et al., 2016). Furthermore, airborne networks can be stationary, slow-moving, or rapidly mobile. In contrast to most *ad hoc* networks, UAV networks rely heavily on infrastructures. Additionally, UAVs can be designated as clients or servers (Gupta et al., 2016), depending on the nature of the exercise. Furthermore, link connectivity must be well-managed because multiple simultaneous agent movements pose significant challenges.

2.2. UAV–MEC interaction

To address the quality-of-service of data-hungry mobile applications, several MEC solutions have emerged. However, establishing an edge server is not always possible in remote, mountainous, and disaster-prone areas. Hence, UAVs have entered the equation. As long as line of sight (LoS) communications can be ensured, UAVs can be utilized to provide task offloading and better downloading performance.

Because UAV-enabled MEC research is still in its early stages, very few studies have investigated this area in detail. In Ref. (Mach and Becvar, 2017), the authors pointed out crucial hotspots, war zones, and natural disaster scenarios wherein UAVs with MEC capabilities can be beneficial. The authors also discuss the potential roles in integrating with MEC server as shown in Fig. 2 which is one of the most fundamental questions in designing an UAV enabled MEC system. As discussed, UAV-enabled MEC designs depend largely on the application scenario in which the UAV is deployed. To build a large-scale collaborative UAV-based MEC system, relay nodes can significantly enhance network performance.

As such, we can conclude that two key factors play a critical role in deciding how to best utilize UAV in such a situation which are as follows: "What kind of UAV is being considered?" and "What are the task requirements?" Notably, there are many types of UAVs, and they can carry multifarious payloads.

3. Application scenarios and a case study of UAV-MEC networks

In this section, first we provide several application scenarios in which the UAV-MEC system can be deployed to meet the intensive demand of computational processing in the absence of a terrestrial MEC infrastructure. Then we provide a case study of UAV-enabled MEC network where we consider a scenario in which UAV is acting as a MEC server.

3.1. Application scenarios of UAV-MEC networks

i. UAV-MEC in next-generation wireless networks

With the drastic increase of IoT devices and emerging applications (e.g., the Internet of everything, multi-agent robots, and space tourism) are experiencing the drawbacks of the fifth-generation (5G) mobile networks in terms of processing power, latency, bandwidth, reliability, and scalability (Alwis et al., 2021). The major drawback with the introduction of next-generation wireless networks is going to be limited battery lifetime and computation resource of the mobile users for performing the task with low latency requirements, which are introduced by the services and applications such as virtual reality, telesurgery, autonomous driving, and UAVs (Yilmaz and Ozbek, 2021). UAV-MEC systems can comfortably handle such emergent situations where terrestrial MEC servers are out of service or overloaded (Nouri et al., 2021b). In (Z. Yang et al., 2021), the authors consider a non-orthogonal multiple access (NOMA)-enabled UAV-MEC system where UAVs with the computational capability are grouped with mobile users having heterogeneous channel conditions, and thus can provide a computational resource to the mobile users exploiting NOMA. The authors conclude that, to meet the stringent delay restriction of applications and various services, the UAV-MEC system can also be integrated into a sixth-generation (6G) wireless network by utilizing UAV-NOMA-MEC, reconfigurable intelligent surface, and millimeter-wave communication. An intelligent UAV-aided offloading scheme is provided in (R. R. Liu et al., 2021), where the authors aim to minimize the energy consumption of IoT devices which frequently communicate with connected vehicles in the Internet of vehicles under a 6G network.

Table 2
Comparison of Single- and Multi-UAV systems.

Feature	Single UAV Systems	Multi-UAV systems
Mission time	Longer mission time	Shorter mission time
Coverage	Provides coverage in a large area	Can only cover a small area
Cost	High	Low
Control	Simple	Complex
Security	Medium	Low
Transmission power	Low	High
Directional antenna	Absent	Present

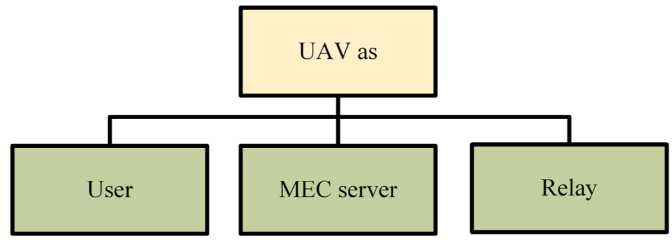

Fig. 2. Roles in UAV-enabled MEC systems.

ii. *UAV-MEC in pandemic*

Since the outbreak of the COVID-19 pandemic, more than 20 million people have been infected worldwide as well as taking a huge toll on the global economy. To fight such a pandemic, UAV-MEC can be an appealing paradigm to address various challenges brought by the pandemic. Contact tracing is one of the challenges where the authorities need to process a huge amount of data daily. In such a use case, UAV-MEC systems can monitor patients from a certain distance, measure body temperature, and even detect the person using face recognition. To process the data, a MEC server at the base station is deployed, and UAVs will be able to establish direct communication with the MEC server using 5G technology to perform those tasks and send the data to the MEC server for further processing (Siriwardhana et al., 2020).

iii. *Reconnaissance using UAV-MEC*

Reconnaissance refers to surveilling a certain area by military forces for obtaining information about the enemy or region, or for other activities. In such cases, human intervention usually is extremely risky due to the life risk or other events that may endanger human lives. In the scenarios, UAV-MEC is a promising solution since it enables UAV to patrol a certain area for collecting various information (e.g., imagery and body temperature). Afterward, image processing and computer vision techniques are applied. Because such tasks need a long battery life and sending this data to a remote cloud is not feasible, an edge server may perform the rest of the task to save the UAV resource (Sacco et al., 2021b). In (J. J. Chen et al., 2021), the authors provide a similar use case where they consider a coalition of UAVs providing a computational resource to jointly optimize the computational offloading and the UAV deployment.

iv. *Surveillance of property*

In order to monitor an important infrastructure or facility against specific activities such as human intrusion, UAVs can be deployed and perform the duty of continuous monitoring. Upon noticing any unusual activities, a UAV sends the data to the nearest edge server for further processing. Upon the completion of the data processing, the information is can be sent to the law enforcement agencies for further steps (Sacco et al., 2021b).

v. UAV-enabled target tracking

Tracking any visual target is another paradigm of UAV-enabled MEC networks as shown in Fig. 3. In such a scenario, a deep learning module with a pre-trained model is deployed at the UAV for identifying a target. The whole process consists of two phases called transient phase and steady phase (Deng et al., 2021). In the first phase, the UAV takes off and localizes the mobile target which is denoted as the transient phase. Then, the UAV captures video streams to keep track of the moving target with the onboard camera which is called the steady phase. Owing to the huge computational resource needed for such a use case, UAV alone cannot accomplish the task by itself. Thus, a MEC server performs the duty of processing the video data to get the location information of the desired target. In such a scenario, however, there is a significant number of challenges introduced by the computational and communication resource of the UAV, the processing delay, and the quality of data. All these issues must be considered while considering such a use case (B. Yang et al., 2021).

After any disaster, the whole network communication can be heavily damaged, which results in communication failure even for several days. This affects the rescue operation. As a result, casualties and property loss are becoming a very common scenario in many cases. In addition, due to any network failure in backbone stations, communication failure can also happen in the span of a large area. This might take a huge time to get back to normal state. The flexibility and dynamic mobility of the UAV enable to provide a wide range of edge services in such affected areas. The authors in (Xu et al., 2020) investigated strategies regarding to task management in a post disaster scenario as shown in Fig. 4. The authors consider UAVs as an edge node carrier, which maintains communication by using long range wide area networking to allow low-power transmission for transmitting control information to achieve fast communication.

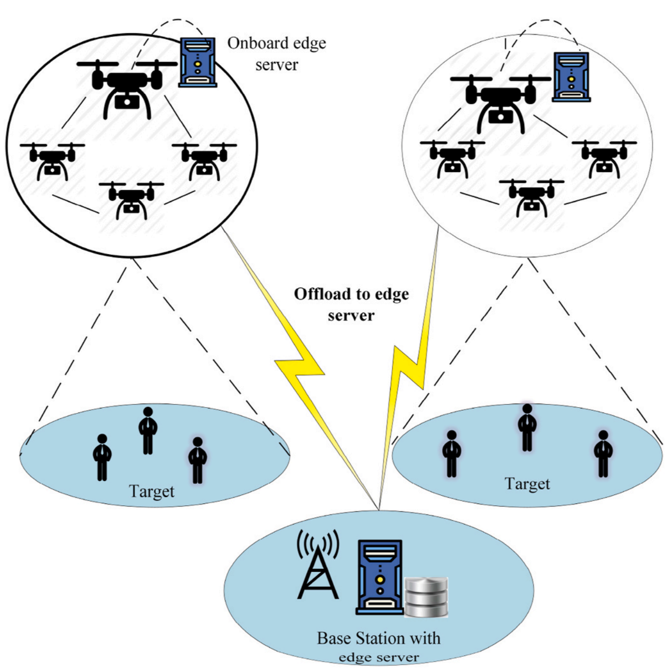

Fig. 3. An example of UAV- MEC application in target tracking scenario. UAVs have certain onboard capacity for performing computation. Due to the heavy computation and limited battery, however, UAVs may also choose to offload the task to the ground edge server situated at the base station.

vi. Disaster management

3.2. *A case study: UAV deployment for surveillance*

Most of the recent works have focused on the use cases of UAVs aiding either for ground IoT devices (Asheralieva and Niyato, 2020; Guo and Liu, 2020; Seid et al., 2021b; Zhan et al., 2021) or for mobile users (J. Hu et al., 2019; Hu et al., 2021; Luo et al., 2021; Ning et al., 2021a; Ren et al., 2021; L. Wang et al., 2021b), where the devices intend to adopt latency-critical applications such as augmented reality or virtual reality. In this section, we consider a simple case study to validate UAV-enabled MEC application, in which UAV is acting as the user with a computation-intensive task such as video processing for crowd surveillance. This use case is very common and is required in various purposes such as security, finding lost citizens, health monitoring, and target tracking. Especially, in urban areas, the crime rate has increased drastically such as terrorism, vandalism, and street crimes. The deployment of UAVs for such tasks is considered as a promising solution due to the dynamic mobility and low cost because conventional human patrol system requires a large number of guards as well as human effort to ensure safety for the people. Unlike the tasks generated from IoT and mobile devices, the video surveillance task consists of several phases before initiating the task. For example, UAVs need to be deployed with a certain topology and locate the people. In case of lost citizen case, it may need to match any data with the database stored in the ground MEC server. Thus, efficient communication must be ensured to avoid delay for transmitting the collected data and receiving the result. In an urban area, there can be dynamic obstacles and, thus, UAV needs to avoid the obstacles to perform the mission. Furthermore, the quality of picture may be full of noises, which needs a lot of pre-processing beforehand. Hence, the roles of UAVs and MEC must be designed carefully. These issues are critical for conducting a successful surveillance mission in a large area. In this section, we investigate a UAV-MEC system which aims to provide face recognition service by acting collaboratively with a ground MEC server (Motlagh et al., 2017). The authors demonstrate the impact that offloading the task has on the overall processing delay and the energy consumption.

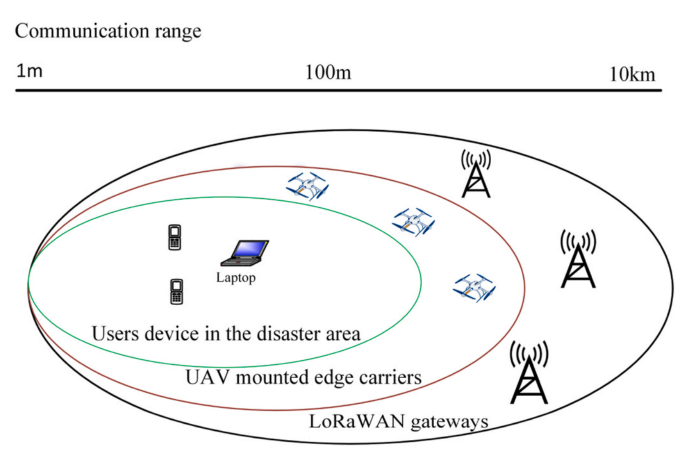

Fig. 4. Scenario of UAV-enabled MEC using LoRaWAN

i. *System model*

The authors in (Motlagh et al., 2017) consider a UAV which is integrated with gimbal for capturing video and is connected to a ground control station in the Aalto University campus via long-term evolution (LTE) cellular network as shown in Fig. 5. For enabling efficient completion of the task, an edge computing resource is co-located with the LTE-base station. This enables to access the edge resources with very low latency. The UAV used in this scenario is a hexacopter which consists of a LTE modem, a gimbal being able to capture high resolution picture and onboard computing resources. There is also a module for controlling the flight for stable control of the flight which is integrated with accelerometer, gyroscopes, and barometer. For handling unexpected situation, it has a safe landing option. In the scenario, the safety guards are the only person who have access to the control station and surveil the people. Upon noticing any strange behavior, command is sent to the UAV to surveil the person or a group of people and utilize face recognition technique to check if the person have any prior criminal records.

To demonstrate the performance comparison between local computing and offloading the face recognition task to the edge server, the authors present a testbed consisting of a Raspberry Pi considered as local processing unit installed onboard in the UAV and a laptop which acts as both the MEC server and ground control station. Commands such as turning the camera off or on and to process the task locally or in the edge server.

ii. *Experiment*

To recognize the faces, they used local binary pattern histogram (LBPH) algorithm provided by OpenCV framework on a database which consists of 40 samples. Of the 40 samples, there is 10 different samples for the same person with various characteristics (e.g., open eyes, closed eyes, and smiling). The simulation code is written using the Python programming language. 10 videos of different length with a group of people was considered. A digital resolution multi-meter was utilized for measuring the amount of energy consumption. Processing delay and energy consumption are the two metrics considered for evaluating the performance of the study. After evaluating the performance, the authors conclude that a considerable amount of energy is consumed, and battery is drained by the local processing of the video data. On the other hand, the offloading process can significantly reduce the energy consumption by a large margin. To understand the effect of the number of suspected people to the energy consumption and the processing delay, they varied the number of people in another experiment. It turned out that the performance of the edge server remains the same regardless of the number of people in the video whereas the energy consumption of local processing increases linearly with the increase of suspected people.

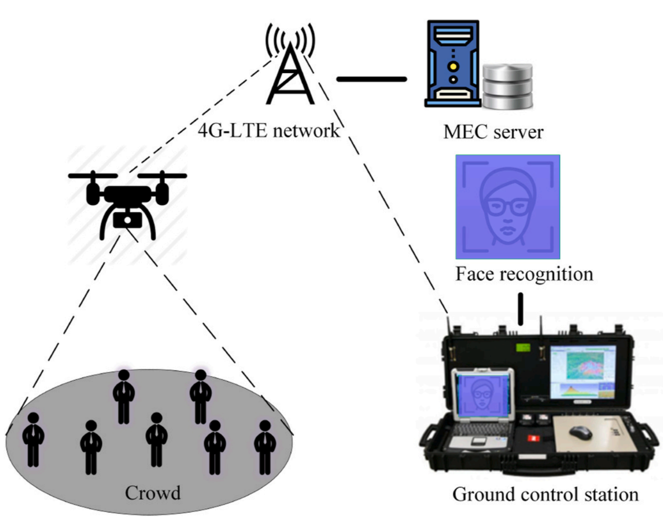

Fig. 5. Envisioned experiment scenario.

iii. *Advantages and limitations*

The authors conclude that computation offloading has significant benefit. This is mainly because it enhances the overall system performance in detecting suspicious activities when there are longer videos and the number of people is varied. Due to the limited capacity onboard, the performance of UAV often falls short during performing such task, and this bottleneck can be overcome by deploying a ground MEC server. However, because the authors only consider a single UAV system, in larger areas and more complex scenario with hundreds of people, only a single UAV based system will face many difficulties while performing such mission.

iv. *Comparison among similar application scenarios*

We compare the above-discussed case study with similar but varying offloading types. In (Sacco et al., 2021b), the authors consider a multi-UAV aided edge computing scenario which can be used for multi-purpose applications such as military reconnaissance, rescue people in harsh environments using image processing technique, and property or human intrusion surveillance. They consider a swarm of UAVs. LTE as well as Wi-Fi network is considered as wireless communication mode. As the offloading mode, binary offloading is considered in the study, where the UAV either executes the task by itself or offloads the task to the edge server. To formulate the offloading decision, the authors proposed a multi-agent reinforcement learning algorithm, where each agent tries to maximize the cumulative reward. The authors in (Bai et al., 2019) consider both partial offloading and binary offloading, where they assume a similar surveillance case to detect the face of the criminals using face recognition technique. They assume that UAV-MEC system might encounter potential eavesdropper during data transmission and, thus, they consider three different eavesdropping cases which are active eavesdropper with both channel state and location information, passive eavesdropper with known location information and unknown information, and lastly passive eavesdropper without any location or channel state information. The system model is derived for the three offloading scenarios as well as the case of both active and passive eavesdroppers. As for the offloading strategy, the convexity of the formulated problem is proved and validated through detailed simulation. In (Q. Zhang et al., 2020), the authors proposed a two-hop UAV scenario to transmit the crucial video streaming information to the control station as soon as possible. They consider a swarm of small rotary wing UAVs denoted as B-UAV using its camera to provide video streaming service for various purposes (such as fire, survivor, or other target detection) and send them to another fixed wing UAV denoted as T-UAV. The T-UAV receives the video via 5G millimeter wave link at the receiver side. The total video frame is then compressed by keeping only the crucial and useful frames by the MEC servers in T-UAV to reduce the communication overhead. The compressed video can then be transmitted to the control center via backhaul links. Table 3 presents a summary of the works discussed above.

It can be observed that energy consumption and task completion time are the most crucial performance metrics for designing offloading algorithm for such scenario. The offloading type depends on the UAV flying time, computational capacity of both UAV-aided edge server and the ground edge server. However, most existing studies assume that ground edge server has sufficient energy to execute the offloaded task, which is often not a practical assumption (Sacco et al., 2021b). In addition, in practical scenarios, single UAV based MEC system will not be able to perform the surveillance task because face recognition task involves deep learning layers with heavy computation. To free UAV from sensing, computation, and communication overhead, a two-hop

Table 3
Comparison of computation offloading algorithms for surveillance application scenario.

Literature	Year	Number of UAVs	Focused performance metrics	Offloading type	UAV control	Main contribution	Advantages	Limitations
Case study Ref. (Motlagh et al., 2017)	2017	Single	Processing delay Energy consumption	Binary offloading	Centralized	Studies the offloading of a video data processing in an edge node compared to local processing in UAV	Offloading the video data to the edge node consumes less energy and processing time.	Single UAV aided scenario cannot be considered in larger areas with a large number of people
Ref. (Bai et al., 2019)	2019	Single	Energy consumption Computational load.	Binary offloading Partial offloading	Centralized	Proposes an energy efficient offloading solution that can perform under potential eavesdroppers	Secure transmission is possible in the presence of potential eavesdropper.	In a larger area, single UAV system will be very inefficient during performing face recognition task.
Ref. (Q. Zhang et al., 2020)	2020	Multiple	Response delay Successful transmission probability	Full offloading	Centralized, Distributed	Proposes a 3D Poisson point process model for jointly optimizing the communication and computational resource in a two-hop UAV scenario	Can perform under unpredictable environment and sufficiently larger network	A much higher task generation probability would cause longer task queue at the T-UAV end.
Ref. (Sacco et al., 2021b)	2021	Multiple (50–200)	Task computation rate Energy consumption Task completion time	Binary offloading	Distributed	Proposes a distributed multi-agent reinforcement learning algorithm that dynamically offload task from UAV to the edge server	Compared to the benchmark algorithms, the proposed algorithm achieves less energy consumption and task completion time.	Assumes the energy consumption at the edge cloud negligible and does not consider the task waiting time at the edge server

UAV scenario is considered in (Q. Zhang et al., 2020), where a group of rotary-wing UAVs capture the information such as video streams, and then transmits to another fixed-wing UAV which compresses and transmits the information to the ground control station. Furthermore, compared to rotary wing UAV, fixed-wing UAV offers extended coverage, prolonged flying duration, and enhanced computing ability. This is why the authors consider rotary-wing UAV for sensing and fixed-wing UAV as a flying MEC server. Thus, such design consideration can be effective for surveillance scenario in larger areas.

4. Design issues of UAV–MEC offloading

In this section, we discuss the potential design issues related to offloading data to/from UAV–MEC servers. Many researchers have striven to enhance the offloading performance of both UAVs and edge servers. Furthermore, the variety of decisions that can occur in an offloading scenario comprises the most crucial metrics that affect overall performance.

4.1. Offloading delay

Offloading delays are incurred when sending a task to either the UAV or the edge server. Delays have several causes, such as inefficient UAV deployments, obstacles, low-frequency channels, and task size. To deal with offloading delays, the UAV must be positioned in a highly optimized orbit.

4.2. Energy consumption

Owing to the limited energy of an independent UAV platform, energy consumption is the most important metric for consideration. UAVs consume energy through their flight mechanisms as well as their computation and communication equipment (Tun et al., 2020). Most studies have focused on minimizing total energy consumption under several realistic constraints.

4.3. Service latency

To maximize UAV offloading services, sufficient onboard computational capacity is required. However, the amount of ancillary equipment is restricted. Thus, depending on the UAV capacity, task size, and loiter area, service throughput may differ. A higher latency causes system overhead, which severely deteriorates offloading performance.

4.4. Trajectory Planning

UAV flight planning is crucial to network performance. Hence, environmental obstacles, including buildings, must be avoided or mitigated. Furthermore, static UAV positions may not always minimize system energy consumption, owing to the nature of distributed high-density IoT devices when there are delays that could be mitigated by proximity. Hence, not only flight planning but scenario planning is needed (L. Wang et al., 2021a).

4.5. Time-slot scheduling

Before task execution, task scheduling predominates. Tasks can be interdependent. Hence, it is necessary to execute tasks in order of not the only priority but also integration potential. In a multi-UAV scenario with partial offloading, this becomes very challenging because UAVs must cooperate. Hence, workflow capabilities based on task priority and task interdependency are needed.

4.6. User risk characteristics

Notably, different UAVs may be owned and operated by different providers. Furthermore, offloaded data may involve sensitive or confidential information. Hence, eavesdropping is a risk. Therefore, security and privacy issues must be addressed during the design of UAV-enabled MEC networks.

4.7. Operational costs of provisioning

Provisioning refers to the management and distribution of object data associated with tasks. For computational services, the UAVs must be able to arrange task-related data, objects, and machine-learning models as they are needed to serve users. This requires efficient planning and considers several metrics such as UAV position, communication, and computation resource allocation, and task portioning decision (Xue et al., 2020).

4.8. Task granularity

During partial offloading, jobs are executed in parallel at the UAV and the edge server. In such cases, partitioning must be planned so that the right level of granularity can be achieved. Dynamic task sizing makes the design of partial offloading systems even more challenging.

4.9. Mobility of end devices

Mobility considerations are very important design aspects when building a system to support offloading decisions. There are two types of issues that complicate this problem: static and mobile conditions. Considering the high variability of network states, offloading becomes rapidly complex when the nodes move. Deploying the UAV without considering the mobility of end devices can not only affect the overall system performance but also fail to consume the privileges provided by the UAV's flexibility (Liao et al., 2021).

5. UAV–MEC network architecture

Stable network architecture is a prerequisite to the determination of a task being executed locally or offloaded to an edge server. The major challenge is reconciling the UAVs' high mobility with a solid communication link and an edge server. Owing to the dynamic topology implied by UAVs, cooperative communication requires the assessment of many crucial performance metrics (Shakeri et al., 2019). As this field continues to progress, a suitable system architecture must evolve.

In recent literature, UAV–MEC architectures have been modeled with three basic components: a system layer, a UAV layer, and an IoT or mobile-device layer, as demonstrated in Fig. 6. In the following subsections, we provide a brief overview of each.

5.1. System layer

In the first layer, we discuss the system layer which aims to establish a connection with the edge server. In this layer, there exist one or multiple base stations that are responsible for connecting with the edge server. The principle working strategy is that the base station primarily receives the offloaded task from the UAV and transfers it to the edge server. When the execution is finished at the edge server, the result is sent back from the edge server to the BS and served to the user. The system layer also may collect additional network information. In Ref. (Yi Liu et al., 2020c), the authors considered a system operator that collects essential network information such as the computational capability of each UAV, the channel quality, and various job demands. Doing so enables the overall offloading performance of the network to be more efficient.

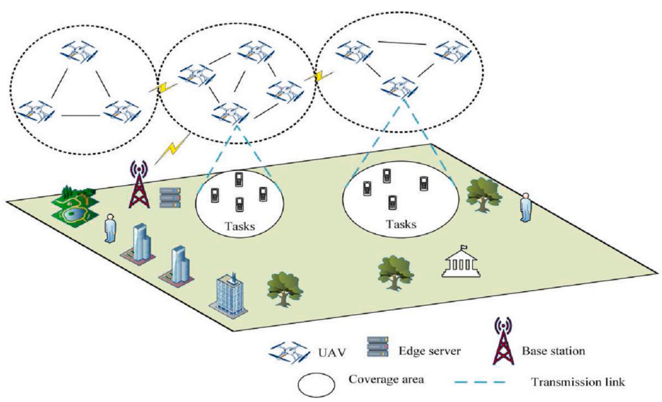

Fig. 6. Illustration of a UAV-enabled computation offloading scenario.

5.2. UAV layer

In this layer, the UAVs hover over the mobile or IoT devices to provide various computational support to the mobile devices. Typically, the UAV can act as a MEC server where heavy computational tasks will be offloaded to the UAV. UAVs are equipped with onboard computational facilities with which they perform a computational task. Due to the limited battery lifetime and computational capacity, the system should only offload those tasks which can be computed by the UAV; otherwise, the system performance will deteriorate as it will cause higher latency. Furthermore, another most important thing is to maintain strong connectivity to get connected with the network. Some existing works focused on UAV cooperation (Yi Liu et al., 2020d) in this fashion. UAVs can either hover at a fixed height, or they can adjust altitudes to support other mobile devices. In either case, effective trajectory mapping is necessary.

5.3. IoT/mobile device layer

IoT and mobile devices are responsible for meeting user demands, whether doing so entails augmented reality, social networking sites, or video streaming. In all cases, vast numbers of data are produced and relayed. When the number of devices in the network increases, task performance can quickly generate a system overload. Hence, we realize the need for MEC servers. Computation can occur locally at such devices if the task is not overwhelming because many devices have high processing capacities. However, with smart devices, if we consider a real-time location-based tracking scenario in a vehicular environment, high-mobility requirements are added. Hence, any IoT or mobile system designed to support such a scenario can quickly become overwhelmed by the local throughput.

6. Computation offloading algorithms

Next, we shed light on existing offloading algorithms. Following the best practice (Kitchenham, 2004) of a literature review and utilizing Google Scholar and IEEE Explore search engines, we found more than 500 articles which deal with task offloading techniques in edge computing domain. Next, we filtered around 350 scientific publications based on the relevance of the scope of this study because we are dealing with only UAV-aided offloading techniques in this study. Then, the rest of the articles were filtered based on the number of citations, downloads, novelty, and quality of the study. Repetitive or similar algorithms were omitted so that we can provide a taxonomy which is up to date with the current literature. The offloading algorithms selected for this study, which were published between 2017 and 2021, can provide major trends and frequently used techniques that significantly enhances the offloading performance in UAV-MEC networks. We classify them in terms of the operational mode (Mach and Becvar, 2017), (Shakarami et al., 2020b). These modes include binary offloading, partial offloading, and load relaying as shown in Fig. 7. We study each algorithm in terms of design objective, performance, special features, characteristics, and offloading mechanism. We also discuss the advantages and disadvantages of each scheme.

6.1. Binary offloading approaches

Tasks that are executed locally at the edge device or offloaded for execution at the UAV are defined by binary offloading. Because the offloading decision is either TRUE or FALSE, the decision is binary. The decision is made based on crucial performance metrics of task execution, offloading delay, transmission time, energy efficiency, and total cost. The scenario is complicated when the network is dense with a high number of IoT devices or when the task size is large or mobile (e.g., at the UAV layer).

Here, we categorize the existing binary offloading algorithms in

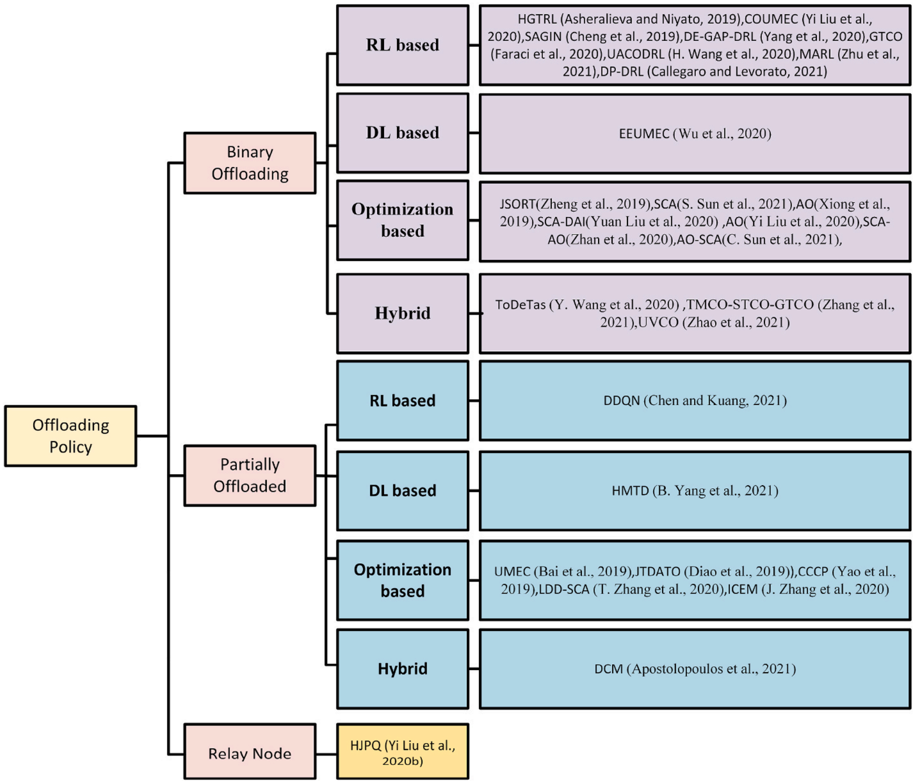

Fig. 7. Classification of offloading algorithms in UAV-enabled MEC systems.

terms of RL, deep learning (DL), optimization, and hybrid algorithms.

1) RL-based Offloading

RL-based algorithms have been used in various fields of wireless communication, owing to the uncertain behavior of the communications environment. Because the network entities must act on those uncertain behaviors, RL is a near-perfect solution; it enables agents to take random actions to reach an optimal policy, especially in complex environments (Luong et al., 2019). Existing studies show that RL algorithms can deal with massive amount of offloading request generated by mobile devices autonomically. In such a scenario, offloading decision is a very complex task because it involves several metrics such as availability of resource, resource demand, and current network status (Alam et al., 2019).

a) Hierarchical Game-Theoretic RL (HGTRL)

A hierarchical and cooperative coalition formation-based offloading algorithm was proposed in Ref (Asheralieva and Niyato, 2019). wherein the authors placed UAVs as players in a game scenario acting in such a way that the overall payoffs were maximized. They presented a hierarchical structure in which the first level resulted in a coalition that could act cooperatively. However, to demonstrate the noncooperative manner of individual stations owned by different service providers, the second level comprised identical subgames. The authors showed that their algorithm reached an optimal state wherein the formation was stable, and the BSs followed a combined strategy to produce an offloading strategy. The authors proposed a game-theory-based RL approach that could find an optimal strategy for BSs via a Markov decision process. They demonstrated that BSs and UAVs can adapt mixed strategies, even when the players are not aware of the actions taken by other BSs. To evaluate performance, they compared their proposed strategy to other benchmark methods in terms of the numbers of users and payoffs achieved when reaching an optimal state.

- *Advantages*: Considering that the BSs can be owned and operated by different vendors, the BS is independent; thus, each player learns the missing information and acts accordingly. The UAV and the BS act cooperatively to offload tasks.
- *Limitations*: The proposed strategy consists of a two-level algorithm for coalition formation and action, which is very complex. Furthermore, the static allocation of computing power is unrealistic in most scenarios.

b) Collaborative Offloading in UAV-enabled MEC (COUMEC)

A novel algorithm for offloading computational tasks to UAVs was presented in Ref (Liu et al., 2020c). which focuses on the cooperative behavior of UAVs, enabling them to offload tasks to other UAVs. The authors showed that UAVs can collaboratively work with others to maximize total network utility and to equalize and reduce computational and communication costs. To achieve this, they focused on limited computational resources and proposed a Markov decision process combined with a deep RL (DRL) to optimize target parameters, assuming that the UAV could also perform as a computational server alongside an edge server. This research showed that the system does not need to send all information to the central operator. This allows the implementation of such systems in areas where there is an immense task load generated from IoT devices.

- *Advantages:* The solution considers the cooperative behavior of UAVs, which helps offload computational tasks from others while mitigating their limited lifetimes and computational power.
- *Limitations:* The noncooperative method cannot provide maximum utility without the cooperation of other UAVs. Therefore, in a dense network having a high number of mobile devices, this system would not work well because the other UAVs might execute tasks at the same time.

c) Space Air-Ground Integrated Network (SAGIN)

A SAGIN network was demonstrated in Ref (Cheng et al., 2019). for offloading computational tasks, where UAVs provide additional computational support as edge servers. The authors considered a remote area with IoT devices on the ground performing heavy computationally intensive tasks, such as surveillance and monitoring. Because the area lacked cellular communications, the SAGIN network was equipped with complete caching, edge computing, and network provisioning capabilities. Their work helped to determine the allocation of resources alongside the scheduling of offloading tasks in a dynamic network. They further investigated the Markov decision process to better understand uncertain system dynamics. Utilizing the Markov decision process has a significant advantage in deciding upon uncertain system entities. To better handle the dynamicity of the network, the authors suggested an on-the-fly approach for DRL. They utilized common policy-gradient methods to act in the complex action space, and for fast convergence to the optimal, an actor-critic technique was adopted. This study revealed that such a system could generate optimal performance by jointly allocating resources in a virtual machine as well as via task assignment. The system minimized the total cost better than other benchmark methods.

- *Advantages*: The system learns the dynamic environment using a hybrid deep neural network, a policy-gradient approach, and an actor-critic network.
- *Limitations*: Computational offloading at local devices was not analyzed. Neither was multi-UAV performance; only a single UAV can execute the task, which may cause latency in real implementations.

d) Differential Evolution (DE) and General Assignment Problem (DE-GAP-DRL) based on DRL

A DRL-based hybrid load-balancing strategy was presented in Ref (Yang et al., 2020). wherein the authors investigated the mitigation of limited computational resources by introducing UAVs as MEC nodes to improve computational capabilities. The authors suggested a DE-based UAV deployment algorithm that optimized task execution time. Initially, they considered that all UAVs were at randomly assigned positions. Then, they allocated the maximum load to each depending on its location. To tackle the intercommunication burdens among UAVs and IoT devices, they utilized a GAP, and an approximation-based algorithm was presented to determine the connections between IoT nodes and the UAV. To manage incoming tasks, they proposed a DRL algorithm for assigning tasks to the UAV. They demonstrated that it could adapt to dynamic system environments while controlling the effective allocation of network resources, enabling more effective handling of recently arriving tasks. The decision to either offload at a UAV or process the task locally created a binary offloading problem.

- *Advantages*: A technique was proposed to balance the load of the tasks between IoT devices and UAVs, significantly improving system performance. Combining the resource allocation problem and the offloading decision significantly improved overall performance.
- *Limitations:* This study assumed that all devices had the same energy budget, task-input data size, maximum processing power, and maximum transmit power, which is unrealistic.

e) Game-theory Computational Offloading (GTCO)

A novel cooperative technique for offloading tasks to other UAVs to tackle power consumption, total delay, and job loss was presented in Ref. (Faraci et al., 2020). Here, the authors considered a situation wherein there was a significant amount of latency caused by ground-generated data taking up a long time to reach the UAV. In the proposed system, the UAVs were equipped with a computational facility similar to an edge server. Owing to the excessive number of computational resources onboard and the limited power of the UAV, the authors introduced a system controller that could decide whether to turn the central controller on or off. Furthermore, the controller could offload tasks from the overloaded UAV to an underloaded one by leveraging RL to reach an optimal policy under uncertain and dynamic conditions in large and complex action space. The authors presented three offloading approaches using greedy and traversal-based algorithms as evaluation benchmarks. Their method minimized the total system cost for computing and reached the Nash equilibrium.

To calculate the total system power consumption, the authors initially considered consumption with no active processor, denoted as $\wp_S$, consumption by each processor, denoted as $\wp_{\mu P}$, and the total number of processors, denoted as $b(n)$. The instantaneous power consumption, $\wp_{Proc}\ (n)$, was then denoted as

$$\wp_{Proc}\ (n) = \wp_S + b(n)\cdot\wp_{\mu P.} \tag{1}$$

The authors considered a central controller for efficient resource management, wherein the resource manager could release, allocate, and coordinate all the requests by the UAV of the system, resulting in better load balancing.

- *Advantages*: The proposed system is flexible under varying configuration parameters. Furthermore, the allocation of resources via the central controller enables load balancing.
- *Limitations*: This study did not consider the transmission interference between the UAV and the edge server, which has a significant impact on system behavior.

f) UAV-assisted Computation Offloading Based on DRL (UACODRL)

The authors of Ref (H. Wang et al., 2020). attempted to reach an optimal offloading state by concentrating on total system costs. They considered a system having multiple wireless users and UAVs having limited capacities. Optimally, wireless users can offload tasks to UAVs. The authors considered that the UAVs could be recharged wirelessly using solar panels. Under this scenario, the authors formulated an optimization problem, wherein the focus was on minimizing the total cost of offloading by combinedly considering energy consumption, bandwidth costs, and total delay. Considering the large action space, the authors used the K-means algorithm to classify several types of computational tasks, resulting in the reduction of the dimensionality of the action space because a large action space can slow down the learning rate.

Subsequently, they proposed an optimized deep-Q-network (DQN) for DRL-based offloading, called double DQN (DDQN), which minimizes the total cost by solving the overestimation problem (Van Hasselt et al., 2016). The author compared the performance of the proposed method in terms of task arrival probability, cumulative reward delay cost, and the rate of arrival of renewable resources. They then compared their proposed method to four other benchmark techniques, wherein the location of offloading was considered. The proposed UACODRL technique significantly outperformed the other benchmarks in terms of average cumulative reward. Although the algorithm performed significantly well, with an increase in users, performance was negatively affected because it also increased the power consumption. Hence, in a large and dense network, system performance will suffer.

- *Advantages*: The proposed scheme can be applied to heterogeneous wireless networks, and the UAVs can be charged wirelessly using a renewable energy supply.
- *Limitations*: The moving trajectory of the UAV is fixed at certain points, which is unrealistic. Considering dynamic environmental conditions, there can be obstacles at various heights, causing the assumption of a fixed trajectory to become invalid.

g) Multi-agent RL (MARL)-based Offloading

The MARL approach was proposed in Ref (Zhu et al., 2021). where two RL agents performed two different tasks individually in a UAV-mounted edge-computing architecture. The novelty of this study lay in its consideration of the interdependency of the task and its dynamicity. At each time slot, when a task arrives at the queue, the two agents are responsible for deciding the target device for execution and the amount of bandwidth needed. The main goal is to minimize the average response time by determining an optimal policy that incorporates task assignment and bandwidth allocation. To define the total task response time, t_i^k, the following equation was derived:

$$t_i^k = q_i^k + c_i^k + p_i^k, k \in \mathscr{K}, i \in \{1, \ldots, d_k\}, \tag{2}$$

where q_i^k is the queueing time, c_i^k is the communication delay, p_i^k is the processing delay, and d_k is the number of tasks.

To achieve their objective, the authors used a Markov decision process with corresponding states, actions, and rewards. To accelerate the learning process, the authors introduced a novel advantage evaluation function for mitigating the credit assignment prominent in MARL scenarios. Because the proposed method was actor-critic based, the authors used the actors as agents and a critic to evaluate the Q value. The proposed technique converged to the optimal state, and the response time was significantly reduced for complicated missions. Because the response time is a crucial issue for UAV edge-computing scenarios, this study addressed it by considering the shortest task response time.

- *Advantages*: For critical missions, the task response time is crucial. Furthermore, the interdependency of the task is another aspect considered.
- *Limitations*: The behavior of UAVs is noncooperative; thus, in a large network, when there is a huge task load, UAVs cannot offload tasks to other free UAVs that produce latency.

h) Dynamic Programming and DRL-based Offloading

In urban areas, UAVs are used primarily for tracking and monitoring city areas. From the perspective of object detection in surveillance and monitoring applications, the UAV's next action depends on the results obtained from object detection. Therefore, delays in task execution are crucial performance determinants (Callegaro and Levorato, 2021). To address this, a system that considers network load parameters was established to minimize the sum of energy expense and delay. The network comprises a UAV, an edge server, and an access point that connects the UAV to the server.

The authors formulated the optimization problem by utilizing a semi-Markov process to minimize total costs. The authors defined the Markov process states as primary, termination, data transmission, and queuing types. The binary decision was made by the UAV, referring to either local computing or offloading. Performance was evaluated using parameters that depend on server load and wireless channel quality. Simulations showed that the agents who were aware of system dynamics had advantages over agents lacking this information because they could not make decisions based on channel situation. Furthermore, the trajectory of the UAV is crucial, and the authors illuminated the average delays over the trajectory concerning system load. The study was concluded by demonstrating a good delay reduction.

- *Advantages*: System performance was analyzed from three perspectives. Furthermore, the strategy is adaptive, meaning that, depending on the server load, the system can act accordingly.
- *Limitations*: The system does not consider the UAV mobility model. In monitoring and tracking systems, the UAV will monitor certain areas wherein obstacles are unpredictable. These considerations make the study much more complex and require further attention.

2) *DL-based Offloading: Energy-efficient UAV-enabled* MEC *offloading (EEUMEC)*

DL applications are widely used for object tracking and face detection applications. In UAV networks, the use of DL models is increasing at a fast pace. The ability to learn patterns and to extract precise features motivates the incorporation of DL models in wireless networks. In this section, we review DL-based computation offloading approaches in a UAV–MEC network. To handle the challenges of predicting the optimal positions of ground vehicles and UAVs in real-time, hybrid DL algorithms have been investigated in recent studies to obtain optimal offloading approaches.

Energy efficiency assurance is crucial to UAV networks. In Ref (Wu et al., 2020). the authors developed a system model consisting of a UAV, a MEC server, and IoT devices. They sought to place the UAV at a position where the offloading could occur smoothly. For this, the authors proposed a position optimization algorithm that takes the coordinates of the IoT devices as input and outputs the coordinates, altitude, and antenna power of the UAV. A long short-term memory (LSTM)-based task prediction algorithm was proposed, wherein the task set was fed into the network, and the algorithm predicted the task size so that the UAV could quickly analyze the required computation and offloading techniques before offloading. To maximize energy efficiency and determine the offloading strategy, an objective function was introduced, wherein a branch-and-bound-based algorithm was proposed. The performance was compared to a random strategy of task offloading and incurred a lower energy debt, showing that task prediction using LSTM has a significant impact.

- *Advantages*: Position estimation of a UAV can occur dynamically, which is a crucial issue for data/task offloading.
- *Limitations*: The task data size is predicted using LSTM, which does not often deliver great precision. This can result in even greater energy consumption.

3) Optimization-based Offloading

Optimization techniques have been widely used for task-offloading in UAV-mounted edge-computing technologies. As the name suggests, optimization algorithms focus on single- or multi-objective optimizations to enhance network performance. In addition to optimizing the total energy consumption, offloading via a secure communication channel as well as computational and communicational resource allocation strategies have been jointly considered in recent studies that use convex optimization techniques (Gu et al., 2021). Furthermore, depending on the specific task requirement, communication channel, and UAV capability, successive convex approximation (SCA) optimization methods have been effective in facilitating decision-making regarding whether a UAV should serve as a MEC server or as a relay node (Z. Z. Liu et al., 2021).

a) *Joint Stochastic Offloading, Resource Allocation, and Trajectory Scheduling (JSORT)*

A novel joint optimization algorithm consisting of offloading, resource allocation, and UAV trajectory schedule was presented in Ref. (Zheng et al., 2019), wherein the authors used a constraint-based method to demonstrate how the total consumption of energy in UAVs and smart mobile devices can be minimized in a MEC system in which a UAV hovers and acts as an edge server. The system can either execute the task locally on the user's smart mobile device, or the task can be executed at the UAV via offloading. The authors assumed that the task size can vary. For this scenario, the authors utilized the Lyapunov-based optimization algorithm for analyzing the task queue for formulating the problem. The problem was further divided into three sub-problems. For combined offloading and resource allocation, the authors used the alternating direction method of multipliers, and for trajectory scheduling, they produced a convex solver method. Finally, their joint optimization algorithm performed the combined optimization of all-three parameters. They evaluated their proposed strategy in terms of the average queue length per time slot. Simulation results indicate that the proposed method saved energy by tuning the system parameters.

- *Advantages*: This system can deal with unpredictable environments, allowing it to be widely deployed.
- *Limitations*: The system does not consider the interference between data communications, and it only concentrates on the energy consumption minimization of smart mobile devices.

b) *SCA-based Offloading*

Deploying UAVs in a 3-dimensional (3D) manner to perform computational tasks is a challenge that was addressed in Ref. (S. Sun et al., 2021). The authors considered that the UAVs performed computational tasks offloaded from the user equipment via wireless channels that do not overlap, and there was no interference among them. The LoS components provided the communication channel between the UAV and the user equipment as a Rician fading equation:

$$g_{i,j} = \sqrt{\frac{k_{i,j}}{k_{i,j}+1}}g + \sqrt{\frac{1}{k_{i,j}+1}}\widetilde{g}, \tag{3}$$

where g refers to the LoS component, $\widetilde{g}$ refers to the Rayleigh fading component, and $k_{i,j}$ corresponds to the Rician factor, referring to the ratio of the power in the LoS component to the fading component.

Under this scenario, the authors optimized the horizontal and vertical positions of the UAV and reduced mission completion time using SCA. In their simulation, the authors evaluated their system against various other UAV platforms and user devices. The proposed method effectively distributed tasks between UAVs by adjusting their 3D deployment strategy. Furthermore, with an increase in the number of user devices, the task completion time showed an upward sloping curve. However, increasing the number of UAVs decreased task completion time significantly. The system provided significant improvements over horizontal position optimization, vertical position optimization, and traditional LoS link-based optimization methods.

- *Advantages*: This solution optimizes all three spatial axes for the deployment of UAVs and introduces multi-UAV deployment.
- *Limitations*: It is assumed that all user equipment has constant transmission power, which is unrealistic.

c) *Alternative Optimization (AO)-based Offloading*

To address UAV total energy consumption, the authors of (Xiong et al., 2019) jointly optimized data bit transmission, offloading decision-making, and aircraft trajectory. In their model, the UAV follows a trajectory while evaluating tasks from IoT/mobile devices and performs them based on the offloading decision and device vicinity. To minimize total energy consumption, the authors assumed that the designed trajectory was a closed loop that could conserve energy.

Because this problem is a non-convex and mixed integer, an AO algorithm was proposed considering the number of IoT mobile devices and the UAV trajectory. As the UAV approaches the vicinity of the mobile devices, the devices offload more tasks to conserve energy. The performance was compared with two benchmark methods wherein they considered all tasks being executed locally and being offloaded. The proposed method achieved the least energy consumption.

- *Advantages*: This system considers the sequential execution of tasks so that the UAV is not required to travel excessively, resulting in lower energy consumption.
- *Limitations*: Only a single UAV-based offloading scheme was considered, and introducing multiple UAVs required a more complex trajectory design.

d) *SCA and Decomposition and Iteration (DAI)-based Offloading*

Considering the limited battery life of the UAV and the required cooperative behaviors, the authors of Ref (Yuan Liu et al., 202e). considered a UAV-enabled offloading scenario that managed offloading task size, processor cycles per second, transmission power, and UAV trajectory. As a non-convex problem, the authors proposed an SCA approach alongside a DAI-based algorithm. In the system model, the authors considered that the UAV was equipped with an edge server and an energy transmitter with which the UAV could recharge smart devices. The UAV was assumed to fly at a fixed height in a Cartesian coordinate system. They mathematically proved the correctness of their proposed method and provided simulation results, revealing that with the increase in idle smart devices, the time to reach the optimum increases, as it introduces more variables to the optimization function. Both algorithms achieved similar minimal energy performance, satisfying the original goal of the proposed strategy.

- *Advantages*: For a large number of data, the SCA-based approach converges faster, whereas, for lower energy consumption, the AI-based algorithm is a better approach.
- *Limitations*: The UAV height was only 10 m, which is far lower than the real-life scenarios.

e) *AO for Offloading*

Performance metrics related to service quality (e.g., offloading service time, UAV trajectory, and offloading decisions) were considered in Ref. (Yi Liu et al., 2020d). The authors proposed a profit-based method with which users were charged for UAV computations. They used an iterative optimization algorithm consisting of three decision stages: offloading, the time elapsed, and UAV trajectory. They assumed that in addition to the onboard battery of the UAV, it could be charged via microwave LoS. The tasks were offloaded using a binary strategy to the UAV via a wireless channel using an offloading factor of either zero or one, where zero denotes that the task is computed locally, and one refers to the UAV performing the task as the edge server. Because the system was designed with a profit-centric goal, the authors formulated the problem as the maximization of total service utility, which consisted of computational and communication utilities. The simulation results demonstrate several interesting outcomes. With an increase in the user's transmission power, the total service utility increased. The proposed method was compared to three benchmark techniques, and the proposed method achieved the maximum service utility.

- *Advantages:* The limited battery lifetime of UAVs and microwave recharging were considered. Moreover, with the increase in user transmission power, the service utility of the proposed method showed an upward slope, owing to the increased amount of harvested energy.
- *Limitations*: The algorithm has a high computational complexity, owing to consecutive iterations. Furthermore, the system model comprised a single UAV. When the number of users increases, it increases the total overhead, which can degrade performance.

f) *SCA and AO-based Offloading (SCA-AO)*

In Ref. (Zhan et al., 2020), the authors addressed two crucial design issues with UAV-mounted applications: total energy consumption and mission completion time. The authors demonstrated how managing these issues could minimize task completion time and total energy consumption. In their model, open space was considered, wherein a fixed-wing UAV performs computationally intensive tasks, including monitoring and surveillance. The system was well-scalable in a multi-UAV scenario, as in Ref (Zhu et al., 2021). They formulated the problem as subproblems of offloading decision, UAV speed, and resource allocation. Because joint optimization considerations create mixed-integer and non-convex situations, the authors used a path discretization technique to overcome the resultant difficulties.

The authors first proposed an SCA-based algorithm to optimize the design aspects. Based on the results, an AO algorithm was proposed to obtain a better optimal solution than previous methods. Finally, for completion-time minimization, another AO algorithm was suggested. Simulation results demonstrated that the proposed trajectory optimization enabled the UAV to fly nearer to the mobile devices for the maximum amount of time so that devices can offload tasks requiring larger data consumption requirements. Additionally, for energy use, the results demonstrated that the proposed method achieved a near-optimal solution compared with the lower-bound threshold.

- *Advantages*: The Pareto-optimal solution balances the trade-off between completion time and energy consumption. The proposed strategy achieves a near-optimal solution for minimizing energy consumption.
- *Limitations*: All devices were assumed to have identical input data sizes, which is unrealistic. Various tasks can have different task sizes based on task characteristics.

g) *AO- and SCA-based Offloading*

The authors in Ref (C. Sun et al., 2021). presented a UAV-enabled edge-computing framework that used the SCA algorithm to effectively optimize trajectory and schedule for offloading and CPU cycles at the UAV edge server. The authors considered a single UAV serving smart terminals (STs) as an additional computational server. The UAV flew along a linear trajectory, and the edge server of the UAV was accessed by multiple STs in a single time slot. To meet UAV mobility restrictions, the authors considered several constraints in terms of speed, location, and acceleration. To minimize the energy consumption while optimizing the total amount of offloading, the offloading duration and linear trajectory were leveraged. To optimize computing and offloading scheduling, three algorithms were proposed based on alternating optimization and SCA-based algorithms. From the mostly offline results, an online approach was proposed that used the results obtained from the three approaches. The system was evaluated under a UAV flying a linear trajectory on a horizontal plane. As the task size decreased, the distance between the UAV and the ST increased. Furthermore, the offloading demand profile was found to impact the UAV trajectory.

- *Advantages*: This system extends to multiple UAV scenarios, including fixed and rotary wings. UAVs travel along with designated 1D and 2D trajectories based on task demand.
- *Limitations*: A 3D trajectory that included UAV height was not considered. To enable UAV deployment where the geographic information is complex, this needs to be factored in.

4) Hybrid Offloading

Although optimization techniques have been widely used to model the offloading problem and meet other system requirements, these techniques have higher computational complexities, owing to the iterative computational procedure needed to reach convergence. These results tend to fall short in terms of maximizing system performance in dynamic and fast-changing environments, owing to the dependency on the number of user devices and their respective positions. In such cases, hybrid approaches have performed well (Jiang et al., 2020).

a) *Two-layer Deployment and Task-scheduling (ToDeTaS)-based Offloading*

UAV deployment plays a crucial role in determining network capability and scalability in terms of mobile users. The authors of Ref (Y. Wang et al., 2020). presented a novel strategy for minimizing energy consumption by considering the position and location of the UAV and by applying task scheduling. Their algorithm consists of two layers: a DE algorithm that effectively deploys the UAV and a 0–1 mixed integer programming algorithm. The authors considered that to effectively execute a task, the UAV must be in the coverage area of the mobile device. The task can then be executed either on the mobile device or the UAV.

The two-layer optimization technique was evaluated under several performance metrics (e.g., success percentage and completed missions). The proposed technique was evaluated against another hybrid method using DE with variable neighborhood decent (VND) instead of particle swarm optimization. For the evaluation of the proposed method, the authors considered a two-step DE-VND (ToDE-VND). The simulation results showed that the results obtained from the upper layer improved method accuracy, whereas with the ToDE-VND method, the two layers remained independent. Thus, a precisely scheduled deployment strategy could not be provided. Simulation results showed that the proposed technique, ToDeTaS, outperformed other techniques when tested with 10 instances of 1000 mobile devices.

- *Advantages*: The correlation between task scheduling and UAV deployment was investigated. The two-layer method outperforms other traditional offloading techniques when tested with a large number of devices.
- *Limitations*: The greedy algorithm sometimes falls short of finding the globally optimal solution because it does not consider all data.

b) *Traversal Method, Sequential Training, and GTCO (TMCO-STCO-GTCO)*

To address time latency and energy consumption, the authors of Ref (K. Zhang et al., 2021). presented a tradeoff-based approach wherein they emphasized constraints for resource competition that affected offloading decisions. In the system model, the authors considered the inclusion of one UAV with onboard capacity for processing computational services and one BS-based MEC server. Owing to the limited capacity of mobile devices, the problem was formulated as an offloading optimization problem with the cost assigned to total energy consumption and time latency. To achieve this, a game-theory-based algorithm was proposed in which every particular mobile device was considered to be a player in the game. They intensively studied the convergence of the proposed method and compared its performance to two other approaches to demonstrate the correctness of the proposed method. Simulation results showed that the proposed strategy is very scalable as it outperforms the other baseline methods when simulating 60 mobile devices. Regarding convergence, the proposed game theory-based approach has a faster convergence rate when simulated with an increased number of mobile devices.

- *Advantages*: The proposed scheme is scalable in terms of the number of mobile users, owing to its significant performance.
- *Limitations*: Because the computational complexity is O(n), where n is the number of mobile devices, the system complexity is much higher when the number of devices is larger. It is also assumed that the UAV flies at a fixed altitude, which is unrealistic in most cases.

c) *UAV-Assisted Vehicular Computation Offloading (UVCO)*

The authors of Ref (Zhao et al., 2021). considered a vehicular environment wherein the tasks were very time-sensitive, requiring extremely fast responses, owing to the dynamic environment. This created a heavy computational burden that needed to be offloaded to MEC servers. The authors utilized software-defined networking (SDN)-enabled the vehicular offloading strategy to minimize the total cost of optimization. They established a game-theory-based approach that makes decisions during task execution. Considering the dynamic vehicular environment, the study considered vehicle mobility, making the study even more complicated. Incorporating SDN enables a guarantee of information transfer by obtaining global information and transmitting the required information to each UAV via the controller. Vehicle users generate tasks in a single time slot. However, because the UAV has limited battery capacity, the total computational cost is calculated considering the execution time of the vehicle. For game formulation, the authors deployed a central controller that collected information so that all players would know the status of other players, facilitating decision-making.

The users of vehicles requiring task offloading are players in the game, and the payoff relies on processing time and energy consumption.

Furthermore, the authors provided a decentralized offloading mechanism comprising two activities: connectivity judgment between the vehicle and the server and decision-making based on the judgment. The strategy relies on UAV connectivity, and the algorithm converges when there are no more tasks to execute. For simulation, the authors considered three benchmark strategies for comparison: local computation, MEC-server offloading, and UAV-based MEC-server offloading. Simulation results showed that the proposed method significantly outperformed the benchmarks by achieving a good balance between the cost of execution time and energy consumption. The UVCO algorithm reduced the total system cost by up to 61%.

- *Advantages*: Considering both centralized and distributed modes, this method enables the system to be more dynamic and scalable compared with existing methods.
- *Limitations*: All vehicles generating the same size of computation task are considered, which is unrealistic.

5) Partial Offloading Approach

Partial offloading partitions tasks into multiple subtasks for execution at edge servers or UAVs. This happens when task execution is initiated by the UAV, and during task execution, the UAV is unable to complete a task. This process is more complicated than binary offloading because the task must be partitioned, which should be determined before offloading. Furthermore, post-execution, both results must be concatenated and sent back to the IoT device. Some studies have focused on the cooperative behavior of UAVs for this purpose. Hence, the offloading efficiency is higher when the offloading delay is minimized by offloading to the nearest UAV instead of an edge server.

For binary offloading, we categorize the existing partial offloading algorithms into RL-, DL-, optimization-based and hybrid methods.

a) RL-based Offloading and DDQN-based Offloading (DDQN)

Preventing eavesdropping and data transmission interception between UAVs and MECs is critical. The authors of Ref (Chen and Kuang, 2021). presented a method that considered this scenario. To mitigate the latency caused by the limited computational capacity of mobile devices, the authors aimed to reduce total energy consumption by allocating resources and ratios. To achieve this, the study proposed a DQN, which falls short of dynamically optimizing multiple performance metrics. The proposed method converged at ~8000 iterations. To evaluate the effectiveness, they compared their method with two others: a local computing scheme and an offloading method based on total weighted cost, MEC processing capability, weight factoring, and total users. Simulations showed that the proposed method achieved a much lower cost when the number of users was set to five. It also demonstrated adaptivity when determining tradeoffs between communications and computational resources while reducing the eavesdropping threat.

- *Advantages*: This method considers the inclusion of UAVs as an eavesdropper, and the system makes adaptive decisions to avoid eavesdropping.
- *Limitations*: To calculate the data transmission rate in the secrecy channel, the information of the channel state must be known, which is difficult to obtain in reality.

b) DL-Based Offloading: Hierarchical Machine-learning Task Distribution (HMTD)

To leverage the computing facilities provided by the MEC server, an optimization approach for DL-based target-tracking UAVs, which is widely used in urban areas, was proposed. To perform this type of task effectively using the limited computing resources and battery lifetime, offloading a portion of the task to a MEC server is required. For this reason, the authors in Ref (B. Yang et al., 2021). proposed a hierarchical architecture in which one portion of DL execution is executed at the UAV, and another portion is executed at the edge server. The study emphasized several crucial performance metrics, such as interference error rate, input data quality, and transmission bandwidth. To meet objectives, the authors considered a MEC system with multiple UAVs to track either a person or a vehicle. While executing the task, the lower level of the DL model was executed to save bandwidth for transmission. Subsequently, the next portion of the model was executed at the MEC to enhance the inference error rate. Offloading can either be binary or partial. The constraint under consideration fulfills both types.

This study also emphasized the availability of the wireless channel because its unavailability makes it impossible for the task to be offloaded. The offloading decision depends on image quality. For example, if it is good, object tracking is executed at the UAV, otherwise, it is executed at the MEC node. The proposed method was evaluated in terms of total cost and interference error rate according to the offloading ratio and total number of UAVs. Simulation results demonstrated that with total offloading, the tasks were executed at the MEC server, which supports the correctness of the proposed method.

- *Advantages*: The performance of this method was evaluated in terms of poor data quality and interference rate; thus, it can be deployed even in complex scenarios where interference is high.
- *Limitations:* The UAV is assumed to orbit in a horizontal trajectory at a fixed altitude and that the channel condition does not change within each offloading procedure. This is unrealistic.

c) Optimization-based Offloading: Offloading for UAV-MEC System (UMEC)

Maintaining secure communications is crucial for UAV-enabled MEC. The authors of Ref (Bai et al., 2019). proposed a physical-layer offloading technique in which a UAV performs a computational task to be offloaded to an edge node. The authors considered two eavesdroppers at a fixed location and another at a random position. Transmissions between the edge node and the UAV could be intercepted by the eavesdropper. The problem was convex as it allows the problem to be formulated as cases of active and passive eavesdroppers, and three offloading decisions were considered: zero, partial, and full.

The proposed methodology was evaluated under several conditions to better understand system security and how the UAV's height affected the offloading decision concerning the total energy consumption. The simulation results showed that offloading depends on the size of the task, such that larger tasks are offloaded. With an increase in UAV altitude, the maximum task size was reduced, and energy consumption increased. Additionally, for identical locations of eavesdroppers, performance was hampered because of the reduction of the jamming power at the eavesdropper.

- *Advantages*: Secrecy performance is considered concerning other performance metrics, such as altitude, eavesdropping location, and task size. The system can successfully offload tasks in the presence of an eavesdropper at a certain location.
- *Limitations*: UAV mobility and limited energy are not considered. UAVs must return to the charging station.

d) *Joint Task Data Allocation and Trajectory Optimization (JTDATO)*

The authors of Ref (Diao et al., 2019). considered the allocation of data and trajectory optimization intending to minimize the total energy consumption in a hovering UAV-enabled MEC system to provide computational services to ground mobile terminals (MTs). Specifically, the authors considered different energy consumption rates of specific MTs when individual devices consumed more computational resources, and most of the computing occurred at the devices instead of the flying edge server. In the system model, the authors considered that all tasks were independent and could be divided into several portions to be offloaded.

To effectively optimize both the UAV trajectory and the allocation of bits, the authors divided the problem into two subproblems and formulated them as convex optimization problems. The authors proposed the JTDATO algorithm, which solves the individual subproblems while jointly optimizing both the trajectory and data allocation. To evaluate system performance, the authors presented an optimized trajectory alongside a data transfer and maximum energy consumption scheme. The proposed method-maintained speed while traversing trajectory when the distance between the UAV and the MTs was minimal. Therefore, more MTs can offload tasks to edge servers, thus conserving energy. To further investigate the performance of the proposed method, the authors also considered the random deployment of MTs. In this case, the energy consumption was much lower than baseline methods.

- *Advantages*: The method addresses fairness among multiple MTs to minimize the total energy consumption. Despite being an iterative approach, the algorithm converges faster.
- *Limitations*: The mobility of MTs can change the energy consumption scenario assumed in this study. High mobility adds more dynamicity to the system.

e) *Concave Convex Procedure (CCCP)*

Trajectory design is one of the most important aspects of the UAV-MEC network, as UAVs must cover a given distance to provide computational services to ground users. The authors of Ref (Yao et al., 2019). proposed a system in which they considered the joint design of the UAV trajectory alongside the offloading problem. In the system model, they considered that a UAV hovers over ground users and that the computational capability of the UAV is more powerful than that of the ground-user devices. Each user can offload any part of a task to the UAV and execute the rest locally, presenting a partial offloading scheme.

The authors aimed to minimize the total delay among users by considering the offloading task ratio. Thus, scheduling and user-variable metrics were used, and the estimated trajectory along which the UAV was covered. Several constraints, such as communication quality and user energy consumption, were also considered. The authors suggested a penalty dual decomposition method-based optimization algorithm. In the inner loop, to optimize the variables considered, the authors used the CCCP algorithm, and for the outer loop, augmented Lagrange multipliers were updated, and the penalty method was introduced. In addition to reducing the complexity, an L0-norm algorithm was proposed to reduce delays amongst users in a single period.

- *Advantages*: The proposed method can obtain reduced delays with lower complexity. Maintaining an optimal delay with communication constraints is a real challenge that is addressed in this method.
- *Limitations*: The data size of the computing results is much smaller, which might change in real-life scenarios. The method does not consider the runtime cost.

f) *Lagrange Duality and Successive Convex Approximation (SCA)-based Offloading (LDD-SCA)*

To deal with latency-critical tasks, UAVs can play a significant role. A Lagrange duality method-based technique was proposed in Ref (T. Zhang et al., 2020). where the authors investigated three possible offloading strategies that can be decided for each terminal device: local computing, partial offloading, and UAV as a relay service. The authors focused on reducing energy consumption by optimizing scheduling, UAV planned trajectories, and power allocations. Because the joint optimization of several performance metrics makes the problem non-convex, the authors divided the original problem into sub-problems. The height of the UAV was fixed, and the collision was avoided; thus, the UAV traversed from a given starting point to an endpoint.

Two special trajectory cases were considered: fixed and optimized. In the numerical results, the authors proposed several benchmarking techniques: straight trajectory design, relay-only, and lacking UAV cooperation. The results showed that approximately 15 iterations were required to reach optimal convergence, indicating that the proposed technique is highly efficient. The energy consumed by the communications and computations were measured, demonstrating that cooperation among UAVs assures lower energy consumption in terms of the joint design of computational and communications resources and optimized trajectories. Additionally, when the task requirements increased from the terminal devices, the performance did not degrade, which shows the correctness of the joint optimization strategy.

- *Advantages*: The UAV can fly near terminal devices that have tasks of higher priority to avoid path loss. This results in faster execution of the overall task flow, as a lower prioritized task can be executed later. This enables the system to be maintained.
- *Limitations*: Local computing is not considered when comparing this study with other benchmark methods. Owing to the various types of offloading techniques, local computing must be considered when evaluating performance.

g) *Iterative Computation Efficiency Maximization (ICEM)*

For partial offloading, the calculation of the bit allocation is often complicated as it requires the maintenance of several other crucial performance metrics, such as computational capacity. In Ref. (J. Zhang et al., 2020), the authors demonstrated a partial computation offloading-based UAV-enabled MEC network wherein they focused on user association, computing resource allocation, and task scheduling. They investigated a multi-UAV-mounted MEC framework in which the UAV has the onboard computational capacity and flies from a start point to an endpoint for partial offloading using smart mobile devices. The channel between the UAV and the ground devices was frequency-division multiple access. For offloading, it was assumed that each device can connect with a maximum of one UAV. Additionally, the constraints ensured that the fewest computational bits were used at each ground device. To this end, they redesigned their model using the Dinkelbach-based method to transform the problem into a parametric programming problem.

In the proposed method, the subproblems were solved individually to obtain the optimal processing cycle, transmission power, and offloading decision. For the simulations, they assumed that five smart mobile devices were randomly distributed. UAVs were assumed to fly at a fixed altitude of 100 m. For the evaluation of the proposed method, the authors proposed benchmark methods. One included a fixed trajectory, assuming all tasks were offloaded to the edge server. Another included offloading with stable channel capacity, and another had computation bit maximization as the only goal. Extensive simulation results demonstrated that the proposed technique outperformed other benchmark techniques by maintaining the performance metrics in terms of computational services.

- *Advantages*: With an ample amount of computation and communication resources, this technique can perform offloaded tasks effectively while ensuring the quality of experience for the users.
- *Limitations*: The iterative optimization algorithm has a double-loop structure, which is why the algorithm has high computational complexity.

6) Hybrid Offloading: Distributed Algorithm for Convergence to Pure Nash Equilibrium (DCM)

A novel game-theory-based offloading decision framework was proposed in Ref (Apostolopoulos et al., 2021). wherein the authors established a UAV-enabled MEC server that considered the latency and computational capacity of users. Here, user satisfaction was particularly emphasized regarding how their risk-aware behaviors impact offloading decisions. The proposed system assumes that some users will offload portions of the task to the MEC server. To address potential UAV-enabled MEC server failure, the authors adopted a risk-uncertainty-based prospect theory behavioral model that determines user behaviors based on uncertainty. In this context, the authors proved the existence of a pure Nash equilibrium, and a distributed algorithm was proposed. For the computational task, the authors considered that they were of the same density as those generated by the users. Each cycle can partition the tasks into several portions and offload them to either the ground MEC server or the UAV-MEC server. Each user tries to maximize their satisfaction utility, which is extracted from their overview of the computation on the local machine as well as the MEC server aboard the UAV.

The optimal offloading decision was decided based on the maximization of satisfaction utility. Furthermore, a negative value is implied when there is any failure of the MEC server regarding computational capacity. In this study, a distributed algorithm for converging to the Nash equilibrium was proposed with complexity analysis. Regarding the equilibrium state, the authors declared that the worst case would be to compute the task locally because it would consume a significant amount of time, owing to the limited capability of the mobile user. For the simulation, the authors considered 200 users and 10 MEC servers with three UAV-enabled servers. They compared the proposed method to five other benchmarking methods having varying characteristics. The simulation results demonstrated that the proposed method achieved lower overhead and served users by maintaining a satisfactory environment, which demonstrated its superiority.

- *Advantages:* User resource uncertainty is captured. Furthermore, an overhead reference point of local processing is managed in consideration of overall user satisfaction.
- *Limitations:* UAV mobility is not considered. Furthermore, a homogeneous computational task was considered, which is not feasible in the current scenario because of the growing number of applications that are very much data-hungry and have dynamic task sizes.

7) Load Relaying: Heuristic Joint Power and Quality of Experience (HJPQ)

After receiving a task from the edge server, the UAV can either execute it locally or transfer it to any nearby edge server. However, because the UAV has scarce resources, it can relay the task to another edge server. Maintaining efficient connectivity remains a challenge. With UAVs, multiple users can maintain connectivity without any direct communication requirements (Liu et al., 2020d). In some areas, such as remote areas, hills, mountains, or areas with obstacles affecting the quality of service, it is almost impossible to maintain direct communication; thus, UAVs can be very handy in such a scenario by working as a relay node.

Owing to the high mobility of UAV networks and the limited energy available to mobile users, optimizing both transmission capacity and UAV position is required for UAV–MEC systems. Owing to the limited battery life of a UAV, fast charging is a crucial design aspect. The authors of Ref (Q. Wang et al., 2021). proposed a UAV-enabled relay network that maximized the energy efficiency of the system by considering the offloading duration of the user equipment, the wireless transmission channel between the user equipment and UAV, and the deployment of the UAV.

In this study, cooperative UAVs were considered to act as a relay node for ground users, enabling them to offload tasks to the cloud. Because the UAV only relays, the decision variable is binary. In a single time slot, one UAV can only communicate with one user using an integrated antenna. To jointly optimize metrics, the authors searched for a global optimization solution by utilizing an iterative search approach. For the search method, the authors utilized a traditional genetic algorithm.

To evaluate the performance of the HJPQ method, a deep deterministic policy-gradient (DDPG) approach was used as a benchmark. For simulation, 150 UAVs provided offloading relay services in a 3D space. They showed that the proposed method converged after approximately 320 iterations, whereas DDPG required 400 episodes to converge. The user equipment was divided into two groups: higher and lower priorities. The results showed that the randomly assigned flights did not consider the delay constraints imposed by the priorities. Furthermore, the energy efficiency increased significantly with both techniques. However, when the number of UEs reached 200, DDPG outperformed HJPQ by a significant margin, owing to the delay constraints. Overall, the HJPQ algorithm had the upper hand as it had lower computational complexity.

- *Advantages:* The proposed method has very low complexity, and UAVs play a cooperative role among them for relaying offloading tasks for ground users.
- *Limitations:* The height of the user equipment and the antenna height were neglected in the proposed study.

7. Comparison of offloading algorithms

7.1. Performance comparison criteria

In this section, we compare existing offloading algorithms qualitatively in terms of major characteristics and novelty of the offloading algorithms. To get a holistic idea of the key characteristics, we provide three tabular comparisons.

Design approach and main idea: Table 4 represents all 28 approaches that were reviewed in this paper, including the role of the UAV, offloading decision, and the summary of the work. To design a UAV-based offloading framework for MEC, the decision regarding to offloading is the most crucial because it involves the design of the computational model which can affect the offloading performance.

Operational characteristics: Table 5 discusses the innovative features prevalent in all 28 offloading algorithms. Specifically, we concentrate on the task size, consideration of resource allocation, UAV cooperative behavior, the number of users and UAVs, and the control and mobility of UAVs. Because computational offloading is strongly correlated with allocation of resource, most studies consider jointly optimizing these two metrics. Furthermore, to maintain communication while offloading the task, the control and mobility of UAVs are very critical. Although most studies consider a centralized controller, the distributed control of UAVs is also being popular due to the large and complex network environment.

Evaluated performance metrics, objective, and application scenario: In Table 6, we present an in-depth performance-centric comparison along with outstanding feature and novelty to evaluate each offloading algorithm. We present the evaluated performance metrics of all the algorithms along with the optimization objective to obtain notion such as which metrics are still under studied and needs further research. Most approaches focus on the total system cost, delay minimization, energy consumption minimization, and optimal trajectory design under challenging conditions. However, performance metrics such as service provisioning, successfully completed task rate, computation rate, and queuing delay are not well-studied in the presented approaches. Moreover, most do not present a detailed effective mobility plan, which is another crucial design necessity. Furthermore, joint consideration of both computation and communication resources is overlooked in most offloading algorithms, which can deteriorate offloading performance. To design an effective and efficient offloading scheme, additional study is needed to ensure reliable offloading performance in UAV–MEC scenarios.

7.2. Summary and insights

From the comparison table, minimizing energy consumption and total delay is the most significant metric, which is not surprising. Offloaded tasks deteriorate the battery storage of the UAV, and the inability

Table 4
Comparison of existing offloading algorithms in terms of design approach and main idea.

Protocol	UAV usage	Offloading type	Main idea and features
HGTRL (Asheralieva and Niyato, 2019)	Edge server	Binary	Defined a way for developing a coalition as well as maximizing payoffs.
COUMEC (Liu et al., 2020d)	Edge server	Binary	A collaborative computation offloading scheme for the proposed UEC network while considering the limited computation capabilities of UAVs
SAG-IOT (Cheng et al., 2019)	Edge server	Binary, cloud processing	Handle the multidimensional Space air-ground integrated network resources and learn the dynamic network conditions.
DE-GAP-DRL (Yang et al., 2020)	Edge server	Binary	Proposed location-based load balancing for UAV and an approximation algorithm for dynamical allocation of resources.
GTCO (Faraci et al., 2020)	Edge server	Binary	Balanced the task load by controlling the UAVs computing unity with a system controller.
UACODRL (H. Wang et al., 2020)	Edge server, load relay	Binary	Investigated to find the near-optimal offloading scheme to minimize the total cost.
MARL (Zhu et al., 2021)	Edge server	Binary	Demonstrated the problem of providing the shortest mission response time in the UAVs-assisted edge computing scenario.
DP-DRL (Callegaro and Levorato, 2021)	Edge server	Binary	Explore an urban environment for UAV operations and use a simulated environment to minimize the energy consumption while satisfying a delay constraint.
EEUMEC (Wu et al., 2020)	Edge server	Binary	Optimized the transmission performance of the UAV during the information transaction
JSORT (Zheng et al., 2019)	Edge server	Binary	Jointly optimized the offloading scheme, user resource allocation, and trajectory optimization.
SCA (S. Sun et al., 2021)	Edge server	Binary	Focused on optimizing the deployment of the UAV to find out task completion time.
AO (Xiong et al., 2019)	Edge server	Binary	Demonstrated energy minimization by focusing on bit allocation, trajectory, and the offloading decision.
SCA-DAI (Yuan Liu et al., 2020e)	Edge server	Binary	Jointly optimized the size of offloading task, CPU cycle per second, the transmission power, and the UAV trajectory.
AA (Liu et al., 2020d)	Edge server	Binary	Jointly optimized offloading service time, UAV trajectory, and the offloading decision and an incentive-based mechanism where the users are charged for the service.
SCA-AO (Zhan et al., 2020)	Edge server	Binary	Considered the mission completion time and the energy optimization technique
AO-SCA (C. Sun et al., 2021)	Edge server	Binary	Minimized the energy consumption optimizing the total amount of offloading, offloading duration, and the linear trajectory.
ToDeTaS (Y. Wang et al., 2020)	Edge server	Binary	Proposed a two-step approach where the first layer UAV is deployed into the system and in the next layer, they move their focus onto optimizing the scheduling of task
TMCO-STCO-GTCO (K. Zhang et al., 2021)	Edge server	Binary	Presented a tradeoff-based approach where they emphasize constraints concentrating on the competition in terms of resource and factors involving offloading decision
UVCO (Zhao et al., 2021)	Edge server	Binary	Utilized Software-defined networking (SDN) enabled vehicular offloading strategy so that the total cost for optimizing the task is minimized
DDQN (Chen and Kuang, 2021)	Edge server, load relay	Partial	Utilized Double Deep Q Network (DQN) for optimizing the total cost because using a DDQN solves the overestimation problem which is dominant in DQN
HMTD (B. Yang et al., 2021)	Edge server	Partial	Emphasized several crucial performance metrics such as interference error rate as well as the input data quality and the transmission bandwidth.
UMEC (Bai et al., 2019)	Edge server	Partial	Consider eavesdropping scenario with two eavesdroppers one at a fixed location, another at a random position
JTDATO (Diao et al., 2019)	Edge server	Partial	The proposed method solves the individual subproblems to jointly optimize both the trajectory and the data allocation
CCCP (Yao et al., 2019)	Edge server	Partial	Proposed such a system where they considered joint design of UAV trajectory alongside offloading perspective
LDD-SCA (T. Zhang et al., 2020)	Edge server, Load relaying	Partial	The proposed method focused on reducing the energy consumption by combinedly optimizing the scheduling, UAV planned trajectory, and the allocation of the power
ICEM (J. Zhang et al., 2020)	Edge server	Partial	Focused on user association, computing resource allocation, and scheduling of tasks.
DCP (Apostolopoulos et al., 2021)	Edge server	Partial	Authors adopt a risk-uncertainty based behavioral model named Prospect theory which determines the user behavior model when there is uncertainty
HJPQ (Q. Wang et al., 2021)	Load relay	Load relaying	A relay network by considering the offloading duration of the user equipment, the wireless transmission channel

(continued on next page)

Table 4 (*continued*)

Protocol	UAV usage	Offloading type	Main idea and features
			between the user equipment and the UAV, and the deployment of UAV

to successfully execute tasks and return the results impacts the overall system. Consideration of successful task completion, task arrival, task deadline can be crucial to enhance the overall task completion performance.

Additionally, in a multi-UAV network, because the UAVs have dynamic topologies owing to their high mobility, ensuring reliable, secure, and effective communications amongst nodes is a major challenge.

Table 5
Comparison of existing offloading algorithms in terms of operational characteristics.

Protocol	Task size	Resource allocation	UAV behavior	Control	UAV height	UAV speed	Number of UAVs	Number of users	Mobility	Bandwidth
HGTRL (Asheralieva and Niyato, 2019)	Random	Available	Cooperative	Decentralized	10 m	10 m/s	Multiple (10)	10	Yes	5–20 MHz
COUMEC (Yi Liu et al., 2020c)	Random	Available	Cooperative	Centralized, Distributed	–	–	Multiple (10)	1000–5000	–	–
SAG-IOT (Cheng et al., 2019)	Random (5–15 MB)	–	Non-cooperative	–	10 m	–	Single	–	Yes	–
DE-GAP-DRL (Yang et al., 2020)	Fixed	Available	Non-cooperative	–	100 m	50 m/s	Multiple	20	Yes	1 MHz
GTCO (Faraci et al., 2020)	7 MB	Available	Cooperative	Centralized	–	–	Multiple (4)	–	–	975 KHz–6.2 MHz
UACODRL (H. Wang et al., 2020)	–	–	Non-cooperative	–	80 m	–	Multiple (3)	30	–	10 MHz
MARL (Zhu et al., 2021)	–	–	Non-cooperative	–	70 m	–	Multiple (3–9)	–	–	10 MHz
DP-DRL (Callegaro and Levorato, 2021)	–	–	Non-cooperative	–	15 m	–	Single	–	–	–
EEUMEC (Wu et al., 2020)	Deterministic	–	Non-cooperative	–	200 m	2.8 m/s	Single	–	Deterministic	400 MHz
JSORT (Zheng et al., 2019)	Random	Available	Non-cooperative	Distributed	10 m	10 m/s	Single	4	Yes	10 MHz
SCA (S. Sun et al., 2021)	–	–	Non-cooperative	–	40–80 m	–	Multiple	10–80	–	10 MHz
AO (Xiong et al., 2019)	5–15 MB	Available	Non-cooperative	–	10 m	20 m/s	Single	3	Yes	40 MHz
SCA-DAI (Yuan Liu et al., 2020e)	–	–	Non-cooperative	Distributed	10 m	20 m/s	Single	6	Yes	40 MHz
AA (Liu et al., 2020c)	–	–	Cooperative	–	50 m	30 m/s	Multiple (5)	5	–	5 MHz
SCA-AO (Zhan et al., 2020)	100 MB	Available	–	–	100 m	3 m/s	Single	5	–	1 MHz
AO-SCA (C. Sun et al., 2021)	2.5–10 MB	–	Non-cooperative	–	100 m	3–50 m/s	Single	6	Yes	1 MHz
ToDeTaS (Y. Wang et al., 2020)	10–1000 KB	Available	Non-cooperative	–	100 m	–	Multiple (100)	100–1000	–	100 MHz
TMCO-STCO-GTCO (K. Zhang et al., 2021)	10–30 MB	–	Non-cooperative	–	100 m	–	Single	60	–	4–40 MHz
UVCO (Zhao et al., 2021)	–	–	–	Centralized, Decentralized	300 m	20 m/s	Single	30	Yes	20 MHz
DDQN (Chen and Kuang, 2021)	–	–	–	–	80 m	–	Multiple (3)	30	–	10 MHz
HMTD (B. Yang et al., 2021)	780 KB	Not Available	Non-cooperative	–	100 m	20 m/s	Multiple (2)	–	Yes	10 MHz
UMEC (Bai et al., 2019)	400 KB	Available	Co-operative	Centralized	200–400 m	–	Single	–	Yes	50 MHz
JTDATO (Diao et al., 2019)	–	–	Non-cooperative	–	50 m	3 m/s-50 m/s	Single	–	Yes	1 MHz
CCCP (Yao et al., 2019)	$1 \times 10^7 \sim 5 \times 10^7$ b	–	Non-cooperative	–	100 m	50 m/s	Single	8	–	1 MHz
LDD-SCA (T. Zhang et al., 2020)	0.4 Mb	Yes	Non-cooperative	–	20 m	20 m/s	Single	3	Yes	10 MHz
ICEM (J. Zhang et al., 2020)	–	Yes	Non-cooperative	–	100 m	50 m/s	Multiple	5	–	20 MHz
DCP (Apostolopoulos et al., 2021)	1000–5000 b	–	Non-cooperative	–	–	–	Multiple (3)	200	Yes	5 MHz
HJPQ (Q. Wang et al., 2021)	1–10 Mb	–	Cooperative	–	500–2000 m	–	150	200	–	–

Note: "–" means that the information is not specified in the corresponding literature.

Table 6
Comparison of Existing Offloading Algorithms in terms of Performance Metrics, Optimization Objective, and Application Domain.

Literature	Evaluated performance metrics	Optimization objective	Outstanding features	Application scenario	Evaluation tool
HGTRL (Asheralieva and Niyato, 2019)	Total offloading delay Energy consumption	Maximizes the long-term payoff (inversely proportional to delay and energy cost)	Joint coalition formation and learning approach	Mobile device	OPNET
COUMEC (Liu et al., 2020c)	Service drop rate Network utility	Maximizes the total utility by deciding optimal offloading and resource allocation policies.	Jointly optimized offloading decision and computation resource allocation	Mobile device	–
SAG-IOT (Cheng et al., 2019)	Average total delay Total cost	Minimizes the total system cost in terms of delay, energy consumption of IoT user, edge, and server usage cost	Joint optimization of edge server resource allocation and task scheduling	IoT user	–
DE-GAP-DRL (Yang et al., 2020)	Balancing task load Average transmission cost	Minimizes the average slowdown of tasks in UAVs	Load balance guaranteeing the coverage constraint satisfying quality of service of IoT nodes	Ground IoT devices	–
GTCO (Faraci et al., 2020)	Mean delay Loss probability Power consumption gain	Maximizes an objective function defined in terms of power consumption, delay, and loss probability by offloaded and non-offloaded jobs	An extension of a fifth-generation network slice with MEC-UAVs	Ground devices	–
UACODRL (H. Wang et al., 2020)	Cumulative reward (total cost)	Minimizes the weighted sum of the delay, energy consumption, and bandwidth cost	Considers stochastic task and arrival rate with time-varying channel state	Wireless users	–
MARL (Zhu et al., 2021)	Average mission response time Queueing, communication, and processing time of the missions	Minimizes the mission response time	Considers inter-dependency of the task, dynamic state of the network and limited energy constraint of UAVs	Surveillance	–
DP-DRL (Callegaro and Levorato, 2021)	Offloading and local computing probability Average delay	Minimizes the weighted sum of delay and energy expense	Practical where heuristics are used for controlling task offloading	Building inspection	–
EEUMEC (Wu et al., 2020)	Energy consumption Delay	Optimizes network energy efficiency	Jointly optimized UAV position and predicted data size of the task	IoT device	–
JSORT (Zheng et al., 2019)	Average energy consumption Task queue length	Minimizes average weighted energy consumption of mobile devices and UAVs	Considers computation offloading, resource allocation, and trajectory scheduling in problem formulation.	Smart mobile device	–
SCA (S. Sun et al., 2021)	Task completion time	Minimizes the completion time of all the offloaded tasks	Considers optimizing the UAV deployment.	User device	–
AO (Xiong et al., 2019)	Number of bits Total energy consumption UAV trajectory	Minimizes the overall energy consumption	Jointly optimized offloading decision making, bit allocation, and UAV trajectory	IoT device	MATLAB
SCA-DAI (Yuan Liu et al., 2020e)	Computation time Computing energy of UAV Bit allocation.	Minimizes the total required energy of UAV	Decomposition and iteration approach to reduce complexity	Sensor device	–
AA (Yi Liu et al., 2020d)	Service utility Offloading decision making UAV trajectory	Maximizes the service utility	Considers profit-based MEC service provider	User device	–
SCA-AO (Zhan et al., 2020)	UAV energy consumption Completion time Offloading time fraction UAV CPU frequency	Jointly minimizes the energy consumption and completion time of UAVs	A tradeoff between energy and completion time	IoT device	MATLAB
AO-SCA (C. Sun et al., 2021)	Task arrival Deadline UAV energy consumption	Minimizes the UAV energy consumption and ensure the deadline of tasks	Provides both online and offloading algorithm	Smart terminals	MATLAB
ToDeTaS (Y. Wang et al., 2020)	Average number of completed task Successfully completed task rate	Minimizes the system energy consumption	A two-layer optimization method for jointly optimizing UAV deployment and task scheduling	Mobile user	MATLAB
TMCO-STCO-GTCO (K. Zhang et al., 2021)	Average system total cost	Minimizes a weighted cost of energy consumption and time latency for all mobile devices	Best possible tradeoff between energy consumption and time latency	Mobile device	MATLAB
UVCO (Zhao et al., 2021)	Average system cost with change in offloaded data size and the number of CPU cycles	Optimizes the execution time of the vehicular computation task and system energy consumption to ensure the quality of service	Considers the dynamic nature of vehicles	Vehicles	–
DDQN (Chen and Kuang, 2021)	Total weighted cost Latency Energy consumption	Minimizes the weighted cost of latency and energy consumption	Confidential data transmission in presence of UAV attacker	IoT device	Python
HMTD (B. Yang et al., 2021)	Total weighted sum cost Inference error	Minimizes the weighted sum cost with the inference error constraint	Considers real-life constraints such as varying image quality and resource sharing among UAVs	Target tracking	–
UMEC (Bai et al., 2019)	Energy consumption Computation load	Minimizes the energy consumption of UAVs	Considers both active and passive eavesdroppers	Surveillance	MATLAB
JTDATO (Diao et al., 2019)	UAV Trajectory Distance between UAV and mobile terminals Energy consumption among Mobile terminals	Minimizes the energy consumption of all mobile terminals	Jointly optimize task allocation and trajectory optimization	Mobile terminals	MATLAB
CCCP (Yao et al., 2019)		Minimizes the total transmission energy consumption of all users	The ground user can choose the most beneficial server.	User device	MATLAB

(*continued on next page*)

Table 6 (*continued*)

Literature	Evaluated performance metrics	Optimization objective	Outstanding features	Application scenario	Evaluation tool
	UAV trajectory Energy consumption of UAV Transmission power of all users				
LDD-SCA (T. Zhang et al., 2020)	Energy consumption Average computation bits UAV trajectory	Minimizes the total energy consumption.	Jointly considers communication, computation, and flight energy	Terminal device	MATLAB
ICEM (J. Zhang et al., 2020)	Optimal trajectory of UAV Flying velocity Computation efficiency of system Energy consumption.	Maximizes the computation efficiency	Considers user association, computation, communication, resource allocation, and UAV trajectory	Smart mobile device	MATLAB
DCP (Apostolopoulos et al., 2021)	Total offloaded data, User satisfaction utility Average expected overhead	Maximizes the total utility from the decision taken and minimizes the overall time and energy overhead	Considers the risk-aware behavior of users and the shared computing resource failure.	Ground user	MATLAB
HJPQ (Q. Wang et al., 2021)	UAV moving distance Delay Energy efficiency	Minimizes the total offloading delay and overall energy consumption by UAV and user equipment.	Multiple-input and multiple-output (MIMO) is utilized to enable independent and parallel data transmission.	User device	MATLAB, Python

Note: "–" means that the information is not specified in the corresponding literature.

Hence, most studies focused on either military or civilian applications in densely populated areas for joint metrics, offloading delays, effective UAV deployments, trajectories, mission completion times, and resource allocations to enhance overall system performance. For example, for crowd surveillance, video capturing, video processing, face recognition, and face matching, tremendous overhead is generated in a single UAV–MEC system. In such cases, collaborative methodologies may provide potential solutions because they enable cooperative task execution.

However, there are several challenges associated with collaborative UAV deployments. Communication links can drop between UAVs, owing to battery drainage, malfunctions, and terrain. Overlapping areas must be controlled to maximize coverage. Most studies utilized a traditional optimization method in which the overall network information is obtained by the UAV and utilized to make an offloading decision. A central controller was also assumed in some studies to collect the required network information, channel information, etc. In contrast, in coalitional UAV systems, the decision is made by each member by interacting with others, enabling them to collaboratively decide upon an optimization plan in a distributed fashion (R. R. Chen et al., 2021).

Depending on other metrics, game-theoretic algorithms can be used effectively because they model individual interactions among coalitions and solve the required optimization problems, unlike conventional optimization techniques. Additionally, RL and DL algorithms have been widely used for task offloading problems to address dynamic channel conditions and UAV navigations in unknown environments. RL enables the learning of unknown network conditions and makes decisions via multiple random actions. However, some RL algorithms (e.g., DQN) have a high convergence rate, owing to their longer training time, which depends on the network architecture.

Furthermore, DL algorithms having a higher number of computational updates and longer training times cause significant energy losses in a UAV network. Considering these facts, hybrid offloading algorithms have shown superior performance using this approach. In Ref. (Y. Wang et al., 2020), the authors proposed a similar two-layer offloading algorithm in which they dealt with UAV deployment in the first layer using a DE algorithm. Unlike other DL techniques, in the second layer, task scheduling was optimized. In summary, while designing an offloading algorithm, it is necessary to consider network conditions in which the UAVs will operate and the tasks to be executed. Therefore, the prevailing issues in the offloading scenario and the network conditions will be given proper attention so that a scalable and effective offloading algorithm can be designed.

Most studies consider tasks generated from IoT devices or mobile users considering the data-intensive task requirement. However, the task generated from a vehicular network in surveillance applications poses a different challenge because UAVs need an efficient mobility plan to support the extremely dynamic nature. Furthermore, the consideration of topology formation depending on the dynamic nature of the environment is overlooked by most studies. Thus, further research is needed to design robust offloading algorithms for such complex application scenarios.

8. Lessons learned and open issues

Considering the immense significance of offloading in a UAV-enabled MEC environment, it is essential to motivate further research in this field, as there are still key challenges to overcome.

8.1. Lessons learned

UAV-mounted MEC systems have become a reality because of the immensely available potential provided by UAVs in fields of healthcare, smart cities, military operations, and civil applications. Extensive research is being conducted to ensure end-to-end connectivity.

1) Selecting Appropriate Application Scenarios

Owing to the variance of wide application scenarios, choosing the most appropriate and suitable offloading algorithm is key to enhancing network performance and addressing challenges.

Scenarios where the number of users is very large (such as overpopulated city areas and augmented reality-based games), deep learning approaches, or optimization-based algorithms will incur higher overhead on the UAV because most of the algorithms are based on iterative approach and have longer computation and training time. Furthermore, a single UAV system will not be able to process an excessive amount of data offloaded from user devices because of the limited processing power and battery lifetime. In such cases, cooperative UAV-enabled hybrid algorithms may be a good choice, because a multi-agent system enables only combinations of relay nodes and MEC servers. However, it also allows a better choice of offloading algorithms to avoid excessive delays.

In military operations and disaster areas where surveillance and monitoring are the primary tasks, and the number of users and their mobility is not determining factors, alongside multi-UAV systems, single UAV-enabled RLs and optimization-based approaches can be chosen as the desired offloading policies, because, in such cases, the UAVs are not challenged by high computational tasks, unlike in urban areas. Furthermore, RL enables agents to effectively capture and learn dynamic network conditions, such as those that we have observed in our study.

2) *Selection of UAV Role*

The role of the UAV plays a major role in the UAV–MEC system. We have observed that the UAV can act as either a user, a MEC server, or a relay node. The prerequisites of each role vary. For example, to deploy UAVs collaboratively, some may only need to act as relay nodes instead of performing offloading tasks. In such cases, UAVs having high transmission power must be selected as relay nodes because failing to relay the task will result in degraded system performance.

3) Selecting decision-making process

In UAV-enabled edge computing, the most crucial metric is the decision making which can be taken either in a centralized manner or in a decentralized fashion. In a centralized manner, there is a central controller which is responsible for taking all the decisions in a centralized way. However, in this technique, the controller may face network congestion due to the unequal transmission power of UAVs and the varying wireless transmission characteristics. The decentralized technique can be a viable solution because each UAV uses the information available from other UAVs and, thus, UAVs decide whether to offload or not. Communication and computational aspects must be jointly considered for dealing with such a scenario (Chen et al., 2019).

4) *Providing Choice of Services*

Task characteristics play an important role in UAV–MEC networks. UAVs are equipped with finite battery capacities and small payloads. The selection of an inappropriate service may result in node failure and disruption in the communication channel. For example, face recognition and video processing tasks require strong computational and battery power, which may not be readily available. However, simple calculations do not consume as much energy. Thus, it is equally important to carefully choose target services by keeping the UAV and task constraints in mind.

5) *Tradeoff between Energy Consumption and Offloading Delays*

To provide seamless connectivity, it may not always be possible to minimize energy efficiency and offloading delay. In such cases, finding a tradeoff may be optimal (Zhan et al., 2020) (Ke et al., 2021). For example, in delay-sensitive application scenarios, energy consumption can be tolerated at a certain level so that the computational result can be offloaded, and the results can be downloaded with the minimum delay possible.

6) *Ensuring Task Completion*

Successful task execution is the most important characteristic and fundamental optimization objective of a UAV-aided wireless network (Zhan et al., 2020). For example, in a multi-UAV system, owing to various critical conditions (e.g., UAV battery failures, accidents, and bad-weather conditions), one UAV may fail to complete its assigned task or collaborate with other UAVs. Hence, the formation of the topology may be lost, and the entire system may fail. Thus, it must be ensured that the topology can be formed so that the task will be completed while maintaining fairness. The authors in (Y. Xu et al., 2021a) consider minimizing the task completion time by jointly optimizing the size of the time slot, device scheduling, resource allocation, and UAV trajectory in a multi-UAV system, where UAVs are assisting terminal devices to successfully complete the task.

8.2. *Open issues*

Regardless of the vast number of privileges provided by the UAV–MEC server, task offloading faces several crucial challenges that are yet to be addressed. In this section, open issues and research challenges related to future research directions are presented from the offloading perspective.

1) Cooperative Control

Cooperative behaviors in a UAV network are of great significance, owing to the highly dynamic task requirements of mobile users and IoT devices. Owing to the variances in the characteristics of the environment (e.g., IoT devices, mobile users, and vehicular environments), task attributes, computational and communication requirements, task priorities, task sizes, computational capacities, and communication capacities change dynamically. Hence, any UAV with limited computational capacity may fail to complete an assigned task, which can be addressed by designing a collaborative UAV network where the overloaded UAVs can be avoided (Sacco et al., 2021a). In such networks, UAVs can offload or relay tasks to other UAVs or BSs having the required computing resources. Furthermore, owing to the limited battery lifetime of UAVs, it is crucial to building such networks. Various cooperation techniques for enhancing offloading performance in a UAV–MEC network have been studied (Hu et al., 2020; Joo et al., 2021; Seid et al., 2021a). However, with the growth of IoT devices, task sizes have increased with constraints in terms of delay and risk. This must be considered when designing advanced UAV–MEC architectures. In addition, in a collaborative UAV network, the UAV deployment and resource allocation are tightly coupled (He et al., 2021). An eavesdropper has various characteristics that generate interference in the transmission channel. Cooperation between two edge nodes for collaborative offloading schemes and communication media provides additional future pursuits in this area (Q. Wang et al., 2020) (Sacco et al., 2021b).

2) Data Security and Privacy

The insurance of secure transmissions poses a significant challenge to UAV networks. When offloading data, eavesdroppers may exist who can operate in several modes. Additionally, the position of an attacker can change dynamically. UAVs operate mostly in remote and sensitive areas where the risk of eavesdropping is potentially high (Seid et al., 2021a). These are challenges that result from this fact. Recent studies have focused on designing UAV–MEC systems to fight against potential eavesdroppers for maintaining the secrecy and confidentiality of the offloaded data (Gu et al., 2021) (Garg et al., 2018) (Y. Y. Zhou et al., 2020). However, intelligent attackers also advance fast. Federated-learning-based offloading models have been proposed to mitigate the need to share confidential data by enabling the learning of a common model in a distributed manner at the network edge. Jamming-resistant techniques that optimize UAV reception have also been studied (Wang et al., 2019).

3) Task-flow Based on Priority and Interdependency

In a dynamic environment, IoT devices can perform various tasks. Some may require immediate results, whereas others do not. Hence, the system must decide which tasks are executed in order. To achieve this, an effective task priority model must be developed, which is often overlooked in most of the existing task offloading studies (J. Xu et al., 2021). Hence, the system must decide which tasks are executed in order. Additionally, there may be some interdependencies between tasks, because random executions may result in inefficient throughput. The efficient grouping of adjacent tasks sharing common resources has been investigated in recent studies (Han and Shi, 2020) (Liu et al., 2020a).

Additionally, considering delay-sensitive tasks that require immediate execution is another crucial aspect of task flows (J. Hu et al., 2019).

4) UAV-enabled Wireless Power Transfer

Owing to the limited battery lifetime of UAVs, wireless power transfer has a remarkable impact on offloading performance. In a single UAV network, if there is no additional power supply, the UAV cannot provide long service. Hence, it will require frequent recharging. Thus, wireless recharging is desirable. Some studies examined joint offload-charging schemes (J. Wang et al., 2020). However, the allocation of computational and communication resources, trajectories, topologies, and scheduling are important issues that cannot be ignored.

5) Trajectory Planning

The mobility of UAVs, ground users, and IoT devices is challenging to the efficient management of offloading tasks. UAVs must traverse along a certain trajectory so that ground users can offload tasks with minimum energy consumption. The greater the distance between the UAV and the mobile devices, the greater the energy consumption. Furthermore, transmission delays affect trajectory design. Although several DRL and convex optimization studies have been provided, 2D trajectories and 3D flight paths remain challenging, owing to high UAV mobility (J. Zhang et al., 2020), (Diao et al., 2020; Feng et al., 2019; Ji et al., 2021; Lu et al., 2021; Ning et al., 2021a; Qin et al., 2021; Y. Xu et al., 2021b). Additionally, the height of the UAV can vary; hence, the trajectory can change dynamically. Furthermore, in a multi-UAV network, avoiding collisions remains a challenging problem that requires special attention. Furthermore, areas where the density of mobile devices is high and each has a delay-sensitive task to offload to the UAV comprises another crucial scenario that requires attention (Guo and Liu, 2020). Some studies have shown that path planning for UAVs has a significant impact on achieving higher throughput, compared with conventional strategies (Liu et al., 2020b). With the increased number of users and multiple MEC servers, the computational cost exponentially grows, which can be addressed by jointly considering the trajectory control and offloading ratio (L. Zhang et al., 2021).

6) Blockchain-enabled Offloading

Blockchain technologies provide a promising model for secure networks. Owing to the ubiquitous characteristics of UAVs, crucial challenges, such as their interconnectivity, their secure communications and management, and their data storage must be carefully addressed. Blockchain technology leverages distributed resources to maintain the security of shared data (Alladi et al., 2020). However, there is a lack of research that investigates how a network can remain secure under malicious attacks. Furthermore, with the growing number of devices and the growth of 5G and the future 6G, data sizes are also exponentially changing. With traditional security mechanisms, it is nearly impossible to combat intelligent attacks. Recent studies have shown that incorporating blockchain in a UAV-enabled offloading system significantly improves network performance and enhances stability and security (Alladi et al., 2020). Incorporating an authentication process that validates the security certification associated with an offloading task before actually executing is a potential solution. Furthermore, encryption techniques can enforce data being transferred during offloading.

7) Resource Scheduling and Dynamic Allocation

With current advancements, applications of augmented reality and location-based services are being widely used, especially with mobile devices. Because of the data processing, mobile devices are extremely resource hungry. In an offloading scenario, competition among devices will continue to be a challenge, as each device will seek to obtain offloading services as fast as possible. Owing to this competition, an efficient resource scheduling algorithm is required. Furthermore, resource allocation must be maintained so that tasks having lower priority will not consume too many resources. Furthermore, when a resource is freed after execution, it should be immediately available for allocation to another task. This is even more challenging when multiple users and UAVs are involved and the task generation of the users is stochastic (Ning et al., 2021b). A dynamic resource allocation scheme is also required during allocation. This was not considered in most extant studies. In addition, with ambiguous knowledge on user mobility and the offloading request, designing an effective resource allocation algorithm requires considering several factors (Li et al., 2020). Thus, resource allocation during UAV–MEC processing is an interesting area for further study (Wan et al., 2020).

8) Energy Efficiency

Energy efficiency is needed to ensure that the offloading task to the edge server succeeds on time. In a UAV network, this is made more challenging by the high mobility of users and their devices. In an actual situation, the problems are exacerbated by the limited battery energy of the UAV, the ground users, and the limited bandwidth resource. Existing studies have attempted to solve the energy efficiency problem by addressing all these constraints (Lyu et al., 2021). The problems are exacerbated by the limited energy and computational capacities of the UAVs and the smart end devices. Total UAV energy comprises communications, computations, and propulsion factors (Alladi et al., 2020; Wu et al., 2017, 2019; Zeng and Zhang, 2017; Zhang et al., 2017). Thus, a potential future solution should integrate wireless power transfer, millimeter-wave communication channels, and energy beamforming, which may increase throughput, lower latency, and increase sustainable energy. Furthermore, the distance between the UAV and the charging station can be reduced to reduce charging times. Additional research is needed in this area to meet 5G and 6G wireless requirements.

9) Service Provisioning

Service provisioning refers to the requisition of various services according to the device requirement of computation offloading (Zhang and Ansari, 2021). UAVs must store various data about the tasks they must process including pre-trained machine-learning models, data, objects, and computational resources. Deciding on what services are provided and what data or objects are handled and stored efficiently is very important and consumes energy. Additionally, the decision involves several other metrics such as path planning, energy efficiency, latency, and transmission bandwidth. This requires the need for jointly optimizing the metrics in such a way so that the placement does not affect the overall network performance. Very few studies have investigated service provisioning under resource and latency constraints (Zhang and Ansari, 2020) (Qu et al., 2021).

10) *Mobility Pattern for Ground Users*

Mobility control is a crucial challenge to UAV-enabled MEC systems. Extant studies only considered such mobility mechanisms based on the time that offloading occurs. Certain nodes in the trajectory are assumed to be fixed during offloading, which can work well only when the mobility pattern shows regularity. However, this does not work well in scenarios where mobile users have irregular mobility patterns, which result from the allocation of dynamic resources. Ground users must search for resources that are idle during offloading. Thus, users may exit the communication channel before the task is offloaded. Hence, it is necessary to reassign available offloading nodes. A possible solution is to analyze the user mobility patterns and determine which users are stable during communications. Hence, users that move out of the communication channel will have a lower stability score, which may exclude them from UAV offloading. Determining the UAV's position by clustering the ground users can be an effective technique for ensuring efficient offloading performance (Nouri et al., 2021a).

11) *Topology Formation for Better Coverage*

In a multi-UAV network, connectivity is an issue that affects system performance. Most UAV-enabled MEC systems consider remote services for wildlife and forest monitoring and disaster response. Thus, interference has not been well-studied. Furthermore, the battery life can affect the loss of connectivity. Many studies are needed to determine whether a node has any limiting factors that might hamper network connectivity. Methods based on artificial intelligence may resolve this problem, because they allow the prediction of the network scenario in advance, which mitigates the risk of losing connectivity. Furthermore, with nodes having standard networking quality, topology can be established by incorporating networking strategies with path-planning techniques. Dynamic UAV clustering-based offloading techniques have also been studied (L. Hu et al., 2019).

12) Duplex Technique

The result of the offloaded task should be transferred to the source device after the task is offloaded to the onboard edge server. Owing to the difference in the characteristics of both tasks, they cannot occur simultaneously using half-duplex techniques. Full-duplex modes can instead solve this issue by enabling offloading the computational task and transmitting the results. Furthermore, handling the interference of both communication types is crucial to further study.

13) UAV and BS assisted hybrid MEC

In most existing studies, terrestrial MEC servers is assumed to perform a heavy computational task which may fail sometimes due to disaster or similar scenario. UAV and BS-assisted hybrid MEC servers can be a potential solution to this problem. However, depending on the size of the computation task, UAV may not be able to finish the whole task due to the limited computational capability and battery energy. To handle such a scenario, a hybrid UAV-and-BS-enabled MEC system was proposed in (Dai et al., 2021), where one base station and multiple UAVs are deployed for facilitating the MEC services between UAVs and the edge servers. The authors consider the binary offloading mode, the number of UAVs, the transmit power of devices, and a dynamic trajectory plan for minimizing the energy consumption of the MEC server.

9. Conclusion

In this study, we performed a comprehensive survey of computation offloading in a UAV-enabled MEC environment. We classified the extant offloading algorithms in terms of their three offloading approaches: binary, partial, and load relaying. Then, we comparatively analyzed them with regard to their innovative features, strengths, and weaknesses. To facilitate their computational capacities in areas where the infrastructure is quite challenging, a UAV-enabled MEC server will be highly advantageous and convenient, and it should provide seamless connectivity in military and disaster-prone areas. From the comparisons provided in this study, offloading schemes can now be better selected. We also presented crucial lessons learned and presented future research ideas. UAV power and energy efficiency are crucial issues. Thus, designing an offloading scheme that addresses dynamic network conditions and user mobility is needed. We believe that this study will be helpful for future designs and implementations of efficient offloading schemes for UAV–MEC systems.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors thank the editor and anonymous referees for their comments, which helped improve the quality of this survey. This research was supported in part by the National Research Foundation of Korea (NRF) grant, funded by the Korean government (MSIT) (No. 2019R1F1A1060501). All correspondence should be addressed to Sangman Moh (smmoh@chosun.ac.kr).

References

Abrar, M., Ajmal, U., Almohaimeed, Z.M., Gui, X., Akram, R., Masroor, R., 2021. Energy efficient UAV-enabled mobile edge computing for IoT devices: a review. IEEE Access 9, 127779–127798. https://doi.org/10.1109/ACCESS.2021.3112104.

Alam, M.G.R., Hassan, M.M., Uddin, M.Zi, Almogren, A., Fortino, G., 2019. Autonomic computation offloading in mobile edge for IoT applications. Future Generat. Comput. Syst. 90, 149–157. https://doi.org/10.1016/j.future.2018.07.050.

Alladi, T., Chamola, V., Sahu, N., Guizani, M., 2020. Applications of blockchain in unmanned aerial vehicles: a review. Veh. Commun. 23 https://doi.org/10.1016/j.vehcom.2020.100249.

Alwis, C. De, Kalla, A., Pham, Q.-V., Kumar, P., Dev, K., Hwang, W.-J., Liyanage, M., 2021. Survey on 6G frontiers: trends, applications, requirements, technologies and future research. IEEE Open J. Commun. Soc. 2, 836–886. https://doi.org/10.1109/ojcoms.2021.3071496.

Apostolopoulos, P.A., Fragkos, G., Tsiropoulou, E.E., Papavassiliou, S., 2021. Data offloading in UAV-assisted multi-access edge computing systems under resource uncertainty. IEEE Trans. Mobile Comput. 1233, 1–16. https://doi.org/10.1109/TMC.2021.3069911.

Arafat, M.Y., Habib, M.A., Moh, S., 2020. Routing protocols for UAV-aided wireless sensor networks. Appl. Sci. 10, 1–23. https://doi.org/10.3390/APP10124077.

Arafat, M.Y., Moh, S., 2021a. Bio-inspired approaches for energy-efficient localization and clustering in uav networks for monitoring wildfires in remote areas. IEEE Access 9, 18649–18669. https://doi.org/10.1109/ACCESS.2021.3053605.

Arafat, M.Y., Moh, S., 2021b. A Q-learning-based topology-aware routing protocol for flying ad hoc networks. IEEE Internet Things J. 4662 https://doi.org/10.1109/jiot.2021.3089759, 1–1.

Arafat, M.Y., Moh, S., 2019a. A survey on cluster-based routing protocols for unmanned aerial vehicle networks. IEEE Access 7, 498–516. https://doi.org/10.1109/ACCESS.2018.2885539.

Arafat, M.Y., Moh, S., 2019b. Routing protocols for unmanned aerial vehicle networks: a survey. IEEE Access 7, 99694–99720. https://doi.org/10.1109/ACCESS.2019.2930813.

Arafat, M.Y., Moh, S., 2019c. Localization and clustering based on swarm intelligence in UAV networks for emergency communications. IEEE Internet Things J. 6, 8958–8976. https://doi.org/10.1109/JIOT.2019.2925567.

Arafat, M.Y., Moh, S., 2018. Location-aided delay tolerant routing protocol in UAV networks for post-disaster operation. IEEE Access 6, 59891–59906. https://doi.org/10.1109/ACCESS.2018.2875739.

Arafat, M.Y., Poudel, S., Moh, S., 2021. Medium access control protocols for flying ad hoc networks: a review. IEEE Sensor. J. 21, 4097–4121. https://doi.org/10.1109/JSEN.2020.3034600.

Asheralieva, A., Niyato, D., 2020. Distributed dynamic resource management and pricing in the IoT systems with blockchain-as-a-service and UAV-enabled mobile edge computing. IEEE Internet Things J. 7 https://doi.org/10.1109/JIOT.2019.2961958, 1974–1993.

Asheralieva, A., Niyato, D., 2019. Hierarchical game-theoretic and reinforcement learning framework for computational offloading in UAV-enabled mobile edge computing networks with multiple service providers. IEEE Internet Things J. 6, 8753–8769. https://doi.org/10.1109/JIOT.2019.2923702.

Bai, T., Wang, J., Ren, Y., Hanzo, L., 2019. Energy-efficient computation offloading for secure uav-edge-computing systems. IEEE Trans. Veh. Technol. 68, 6074–6087. https://doi.org/10.1109/TVT.2019.2912227.

Barbarossa, S., Sardellitti, S., Di Lorenzo, P., 2014. Communicating while computing: distributed mobile cloud computing over 5G heterogeneous networks. IEEE Signal Process. Mag. 31, 45–55. https://doi.org/10.1109/MSP.2014.2334709.

Bhattacharya, A., De, P., 2017. A survey of adaptation techniques in computation offloading. J. Netw. Comput. Appl. 78, 97–115. https://doi.org/10.1016/j.jnca.2016.10.023.

Bonomi, F., Milito, R., Zhu, J., Addepalli, S., 2012. Fog computing and its role in the internet of things. MCC'12 - Proc. 1st ACM Mob. Cloud Comput. Work. 13–15 https://doi.org/10.1145/2342509.2342513.

Callegaro, D., Levorato, M., 2021. Optimal edge computing for infrastructure-assisted UAV systems. IEEE Trans. Veh. Technol. 70, 1782–1792. https://doi.org/10.1109/TVT.2021.3051378.

Chen, B., Zhou, H., Yao, J., Guan, H., 2019. When edge computing meets aerial vehicles: an energy-efficient edge cloud architecture for intelligent multi-UAV. IEEE Trans. Serv. Comput. 1374 https://doi.org/10.1109/TSC.2019.2962469.

Chen, J., Wu, Q., Xu, Y., Qi, N., Fang, T., Jia, L., Dong, C., 2021. A multi-leader multi-follower stackelberg game for coalition-based UAV MEC networks. IEEE Wirel. Commun. Lett. 2337 https://doi.org/10.1109/lwc.2021.3100113, 1–1.

Chen, L., Kuang, X., 2021. Intelligent mobile edge computing networks for internet of things. IEEE Access 9, 95665–95674. https://doi.org/10.1109/ACCESS.2021.3093886.

Chen, R., Cui, L., Wang, M., Zhang, Y., Yao, K., Yang, Y., Yao, C., 2021. Joint computation offloading, channel access and scheduling optimization in UAV swarms: a game-theoretic learning approach. IEEE Open J. Comput. Soc. https://doi.org/10.1109/ojcs.2021.3100870, 1–1.

Cheng, X., Lyu, F., Quan, W., Zhou, C., He, H., Shi, W., Shen, X., 2019. Space/aerial-assisted computing offloading for IoT applications: a learning-based approach. IEEE J. Sel. Area. Commun. 37, 1117–1129. https://doi.org/10.1109/JSAC.2019.2906789.

Cisco, T., Internet, A., 2020. Cisco: 2020 CISO benchmark report. Comput. Fraud Secur. 2020, 4. https://doi.org/10.1016/s1361-3723(20)30026-9.

Dai, B., Niu, J., Ren, T., Hu, Z., Atiquzzaman, M., 2021. Towards energy-efficient scheduling of UAV and base station hybrid enabled mobile edge computing. IEEE Trans. Veh. Technol. 9545, 1–16. https://doi.org/10.1109/TVT.2021.3129214.

de Souza, A.B., Rego, P.A.L., Carneiro, T., Rodrigues, J.D.C., Filho, P.P.R., de Souza, J.N., Chamola, V., de Albuquerque, V.H.C., Sikdar, B., 2020. Computation offloading for vehicular environments: a survey. IEEE Access 8, 198214–198243. https://doi.org/10.1109/ACCESS.2020.3033828.

Deng, X., Li, J., Guan, P., Zhang, L., 2021. Energy-efficient UAV-aided target tracking systems based on edge computing. IEEE Internet Things J. 4662, 1–8. https://doi.org/10.1109/JIOT.2021.3091216.

Diao, X., Wang, M., Zheng, J., Cai, Y., 2020. Fairness-aware offloading and trajectory optimization for multi-UAV enabled multi-access edge computing. IEEE Access 8, 124359–124370. https://doi.org/10.1109/ACCESS.2020.3006112.

Diao, X., Yang, W., Yang, L., Cai, Y., 2021. UAV-Relaying-Assisted multi-access edge computing with multi-antenna base station: offloading and scheduling optimization. IEEE Trans. Veh. Technol. 9545 https://doi.org/10.1109/tvt.2021.3101298, 1–1.

Diao, X., Zheng, J., Cai, Y., Wu, Y., Anpalagan, A., 2019. Fair data allocation and trajectory optimization for UAV-assisted mobile edge computing. IEEE Commun. Lett. 23, 2357–2361. https://doi.org/10.1109/LCOMM.2019.2943461.

Faraci, G., Grasso, C., Schembra, G., 2020. Design of a 5G network slice extension with MEC UAVs managed with reinforcement learning. IEEE J. Sel. Area. Commun. 38, 2356–2371. https://doi.org/10.1109/JSAC.2020.3000416.

Feng, G., Wang, C., Li, B., Lv, Haibin, Zhuang, X., Lv, Hongwu, Wang, H., Hu, X., 2019. UAV-assisted wireless relay networks for mobile offloading and trajectory optimization. Peer-to-Peer Netw. Appl 12, 1820–1834. https://doi.org/10.1007/s12083-019-00793-5.

Garg, S., Singh, A., Batra, S., Kumar, N., Yang, L.T., 2018. UAV-empowered edge computing environment for cyber-threat detection in smart vehicles. IEEE Netw 32, 42–51. https://doi.org/10.1109/MNET.2018.1700286.

Gu, X., Zhang, G., Wang, M., Duan, W., Wen, M., Ho, P.-H., 2021. UAV-aided energy efficient edge computing networks: security offloading optimization. IEEE Internet Things J. 4662 https://doi.org/10.1109/jiot.2021.3103391, 1–1.

Guo, H., Liu, J., 2020. UAV-enhanced intelligent offloading for internet of things at the edge. IEEE Trans. Ind. Inf. 16, 2737–2746. https://doi.org/10.1109/TII.2019.2954944.

Gupta, L., Jain, R., Vaszkun, G., 2016. Survey of important issues in UAV communication networks. IEEE Commun. Surv. Tutorials 18, 1123–1152. https://doi.org/10.1109/COMST.2015.2495297.

Habib, A., Arafat, Y., Moh, S., 2018. Routing protocols based on reinforcement learning for wireless sensor networks: a comparative study. J. Adv. Res. Dyn. Control Syst. 427–435.

Han, D., Shi, T., 2020. Secrecy capacity maximization for a UAV-assisted MEC system. China Commun 17, 64–81. https://doi.org/10.23919/JCC.2020.10.005.

He, X., Jin, R., Dai, H., 2021. Multi-hop task offloading with on-the-fly computation for multi-UAV remote edge computing. IEEE Trans. Commun. 6778, 1–13. https://doi.org/10.1109/TCOMM.2021.3129902.

Hu, J., Jiang, M., Zhang, Q., Li, Q., Qin, J., 2019. Joint optimization of UAV position, time slot allocation, and computation task partition in multiuser aerial mobile-edge computing systems. IEEE Trans. Veh. Technol. 68, 7231–7235. https://doi.org/10.1109/TVT.2019.2915836.

Hu, L., Tian, Y., Yang, J., Taleb, T., Xiang, L., Hao, Y., 2019. Ready player one: UAV-Clustering-Based multi-task offloading for vehicular VR/AR gaming. IEEE Netw 33, 42–48. https://doi.org/10.1109/MNET.2019.1800357.

Hu, X., Wong, K.K., Yang, K., Zheng, Z., 2019. UAV-assisted relaying and edge computing: scheduling and trajectory optimization. IEEE Trans. Wireless Commun. 18, 4738–4752. https://doi.org/10.1109/TWC.2019.2928539.

Hu, X., Wong, K.K., Zhang, Y., 2020. Wireless-powered edge computing with cooperative UAV: task, time scheduling and trajectory design. IEEE Trans. Wireless Commun. 19, 8083–8098. https://doi.org/10.1109/TWC.2020.3019097.

Hu, Z., Zeng, F., Xiao, Z., Fu, B., Jiang, H., Chen, H., 2021. Computation efficiency maximization and QoE-provisioning in UAV-enabled MEC communication systems. IEEE Trans. Netw. Sci. Eng. 8, 1630–1645. https://doi.org/10.1109/TNSE.2021.3068123.

Ji, J., Zhu, K., Yi, C., Niyato, D., 2021. Energy consumption minimization in UAV-assisted mobile-edge computing systems: joint resource allocation and trajectory design. IEEE Internet Things J. 8, 8570–8584. https://doi.org/10.1109/JIOT.2020.3046788.

Jiang, C., Cheng, X., Gao, H., Zhou, X., Wan, J., 2019. Toward computation offloading in edge computing: a survey. IEEE Access 7, 131543–131558. https://doi.org/10.1109/ACCESS.2019.2938660.

Jiang, F., Wang, K., Dong, L., Pan, C., Xu, W., Yang, K., 2020. Deep-learning-based joint resource scheduling algorithms for hybrid MEC networks. IEEE Internet Things J. 7, 6252–6265. https://doi.org/10.1109/JIOT.2019.2954503.

Joo, S., Kang, H., Kang, J., 2021. CoSMoS: cooperative sky-ground mobile edge computing system. IEEE Trans. Veh. Technol. XX https://doi.org/10.1109/tvt.2021.3094584, 1–1.

Ke, H., Wang, H., Sun, W., Sun, H., 2021. Adaptive computation offloading policy for multi-access edge computing in heterogeneous wireless networks. IEEE Trans. Netw. Serv. Manag. XX, 1–18. https://doi.org/10.1109/TNSM.2021.3118696.

Kitchenham, B., 2004. Procedures for performing systematic reviews, version 1.0. Empir. Software Eng. 33, 1–26.

Li, M., Cheng, N., Gao, J., Wang, Y., Zhao, L., Shen, X., 2020. Energy-efficient UAV-assisted mobile edge computing: resource allocation and trajectory optimization. IEEE Trans. Veh. Technol. 69, 3424–3438. https://doi.org/10.1109/TVT.2020.2968343.

Liao, Z., Ma, Y., Huang, J., Wang, Jianxin, Member, S., Wang, Jin, Member, S., 2021. Hotspot : a UAV-assisted dynamic mobility-aware offloading for mobile-edge computing in 3. D Space 8, 10940–10952.

Lin, H., Zeadally, S., Chen, Z., Labiod, H., Wang, L., 2020. A survey on computation offloading modeling for edge computing. J. Netw. Comput. Appl. 169, 102781. https://doi.org/10.1016/j.jnca.2020.102781.

Liu, Baichuan, Zhang, W., Chen, W., Huang, H., Guo, S., 2020a. Online computation offloading and traffic routing for uav swarms in edge-cloud computing. IEEE Trans. Veh. Technol. 69, 8777–8791. https://doi.org/10.1109/TVT.2020.2994541.

Liu, Qian, Shi, L., Sun, L., Li, J., Ding, M., Shu, F.S., 2020b. Path planning for UAV-mounted mobile edge computing with deep reinforcement learning. IEEE Trans. Veh. Technol. 69, 5723–5728. https://doi.org/10.1109/TVT.2020.2982508.

Liu, R., Liu, A., Qu, Z., Xiong, N.N., 2021. An UAV-enabled intelligent connected transportation system with 6G communications for internet of vehicles. IEEE Trans. Intell. Transport. Syst. 1–15. https://doi.org/10.1109/TITS.2021.3122567.

Liu, Yi, Qiu, M., Hu, J., Yu, H., 2020c. Incentive UAV-enabled mobile edge computing based on microwave power transmission. IEEE Access 8, 28584–28593. https://doi.org/10.1109/ACCESS.2020.2971962.

Liu, Yi, Xie, S., Zhang, Y., 2020d. Cooperative offloading and resource management for UAV-enabled mobile edge computing in power IoT system. IEEE Trans. Veh. Technol. 69, 12229–12239. https://doi.org/10.1109/TVT.2020.3016840.

Liu, Yuan, Xiong, K., Ni, Q., Fan, P., Letaief, K. Ben, 2020e. UAV-assisted wireless powered cooperative mobile edge computing: joint offloading, CPU control, and trajectory optimization. IEEE Internet Things J. 7, 2777–2790. https://doi.org/10.1109/JIOT.2019.2958975.

Liu, Z., Tan, X., Wen, M., Wang, S., Liang, C., 2021. An energy-efficient selection mechanism of relay and edge computing in UAV-assisted cellular networks. IEEE Trans. Green Commun. Netw. 2400, 1–11. https://doi.org/10.1109/TGCN.2021.3086084.

Lu, W., Ding, Y., Gao, Y., Hu, S., Wu, Y., Zhao, N., Gong, Y., 2021. Resource and trajectory optimization for secure communications in dual-UAV-MEC systems. IEEE Trans. Ind. Inf. 3203 https://doi.org/10.1109/tii.2021.3087726, 1–1.

Luo, Y., Ding, W., Zhang, B., 2021. Optimization of task scheduling and dynamic service strategy for multi-UAV-enabled mobile-edge computing system. IEEE Trans. Cogn. Commun. Netw. 7, 970–984. https://doi.org/10.1109/TCCN.2021.3051947.

Luong, N.C., Hoang, D.T., Gong, S., Niyato, D., Wang, P., Liang, Y.C., Kim, D.I., 2019. Applications of deep reinforcement learning in communications and networking: a survey. IEEE Commun. Surv. Tutorials 21, 3133–3174. https://doi.org/10.1109/COMST.2019.2916583.

Lyu, L., Zeng, F., Xiao, Z., Zhang, C., Jiang, H., Havyarimana, V., 2021. Computation bits maximization in UAV-enabled mobile edge computing system. IEEE Internet Things J. XX, 1–12. https://doi.org/10.1109/JIOT.2021.3123429.

Mach, P., Becvar, Z., 2017. Mobile edge computing: a survey on architecture and computation offloading. IEEE Commun. Surv. Tutorials 19, 1628–1656. https://doi.org/10.1109/COMST.2017.2682318.

Motlagh, N.H., Bagaa, M., Taleb, T., 2017. UAV-based IoT platform: a crowd surveillance use case. IEEE Commun. Mag. 55, 128–134. https://doi.org/10.1109/MCOM.2017.1600587CM.

Ning, Z., Dong, P., Wen, M., Wang, X., Guo, L., Kwok, R.Y.K., Poor, H.V., 2021a. 5G-Enabled UAV-to-community offloading: joint trajectory design and task scheduling. IEEE J. Sel. Area. Commun. 39, 3306–3320. https://doi.org/10.1109/JSAC.2021.3088663.

Ning, Z., Yang, Y., Wang, X., Guo, L., Gao, X., Guo, S., Wang, G., 2021b. Dynamic computation offloading and server deployment for UAV-enabled multi-access edge computing. IEEE Trans. Mobile Comput. 1233 https://doi.org/10.1109/tmc.2021.3129785, 1–1.

Nouri, N., Abouei, J., Sepasian, A.R., Jaseemuddin, M., Anpalagan, A., Plataniotis, K.N., 2021a. 3D multi-UAV placement and resource allocation for energy-efficient IoT communication. IEEE Internet Things J. 4662, 1–18. https://doi.org/10.1109/JIOT.2021.3091166.

Nouri, N., Fazel, F., Abouei, J., Plataniotis, K., 2021b. Multi-UAV placement and user association in uplink MIMO ultra-dense wireless networks. IEEE Trans. Mobile Comput. 1233, 1–18. https://doi.org/10.1109/TMC.2021.3108960.

Poudel, S., Moh, S., 2021. Hybrid path planning for efficient data collection in uav-aided wsns for emergency applications. Sensors 21. https://doi.org/10.3390/s21082839.

Poudel, S., Moh, S., 2019. Medium access control protocols for unmanned aerial vehicle-aided wireless sensor networks: a survey. IEEE Access 7, 65728–65744. https://doi.org/10.1109/ACCESS.2019.2917948.

Qin, X., Song, Z., Hao, Y., Sun, X., 2021. Joint resource allocation and trajectory optimization for multi-UAV-assisted multi-access mobile edge computing. IEEE Wirel. Commun. Lett. 10, 1400–1404. https://doi.org/10.1109/LWC.2021.3068793.

Qu, Y., Dai, H., Wang, H., Dong, C., Wu, F., Guo, S., Wu, Q., 2021. Service provisioning for UAV-enabled mobile edge computing. IEEE J. Sel. Area. Commun. 8716, 1–18. https://doi.org/10.1109/JSAC.2021.3088660.

Ren, T., Niu, J., Dai, B., Liu, X., Hu, Z., Xu, M., Guizani, M., 2021. Enabling efficient scheduling in large-scale UAV-assisted mobile edge computing via hierarchical reinforcement learning. IEEE Internet Things J. 4662, 1–15. https://doi.org/10.1109/JIOT.2021.3071531.

Rodrigues, T.K., Suto, K., Nishiyama, H., Liu, J., Kato, N., 2020. Machine learning meets computation and communication control in evolving edge and cloud: challenges and future perspective. IEEE Commun. Surv. Tutorials 22, 38–67. https://doi.org/10.1109/COMST.2019.2943405.

Sacco, A., Esposito, F., Marchetto, G., 2021a. Resource inference for sustainable and responsive task offloading in challenged edge networks. IEEE Trans. Green Commun. Netw. 5, 1114–1127. https://doi.org/10.1109/TGCN.2021.3091812.

Sacco, A., Esposito, F., Marchetto, G., Montuschi, P., 2021b. Sustainable task offloading in UAV networks via multi-agent reinforcement learning. IEEE Trans. Veh. Technol. 70, 5003–5015. https://doi.org/10.1109/TVT.2021.3074304.

Seid, A.M., Boateng, G.O., Anokye, S., Kwantwi, T., Sun, G., Liu, G., 2021a. Collaborative computation offloading and resource allocation in multi-UAV assisted IoT networks: a deep reinforcement learning approach. IEEE Internet Things J. 8, 12203–12218. https://doi.org/10.1109/JIOT.2021.3063188.

Seid, A.M., Boateng, G.O., Lu, J., Mareri, B., Jiang, W., Sun, G., 2021b. Multi-agent DRL for task offloading and resource allocation in multi-UAV enabled IoT edge network. IEEE Trans. Netw. Serv. Manag. 18, 4531–4547. https://doi.org/10.1109/TNSM.2021.3096673.

Shakarami, A., Ghobaei-Arani, M., Shahidinejad, A., 2020a. A survey on the computation offloading approaches in mobile edge computing: a machine learning-based perspective. Comput. Network. 182, 107496. https://doi.org/10.1016/j.comnet.2020.107496.

Shakarami, A., Ghobaei-Arani, M., Shahidinejad, A., 2020b. A survey on the computation offloading approaches in mobile edge/cloud computing environment: a stochastic-based perspective, computer networks. J. Grid Comput. https://doi.org/10.1016/j.comnet.2020.107496.

Shakarami, A., Shahidinejad, A., Ghobaei-Arani, M., 2020c. A review on the computation offloading approaches in mobile edge computing: a game-theoretic perspective. Software Pract. Ex. 50, 1719–1759. https://doi.org/10.1002/spe.2839.

Shakeri, R., Al-Garadi, M.A., Badawy, A., Mohamed, A., Khattab, T., Al-Ali, A.K., Harras, K.A., Guizani, M., 2019. Design challenges of multi-UAV systems in cyber-physical applications: a comprehensive survey and future directions. IEEE Commun. Surv. Tutorials 21, 3340–3385. https://doi.org/10.1109/COMST.2019.2924143.

Siriwardhana, Y., De Alwis, C., Gur, G., Ylianttila, M., Liyanage, M., 2020. The fight against the COVID-19 pandemic with 5G technologies. IEEE Eng. Manag. Rev. 48, 72–84. https://doi.org/10.1109/EMR.2020.3017451.

Sun, C., Ni, W., Wang, X., 2021. Joint computation offloading and trajectory planning for UAV-assisted edge computing. IEEE Trans. Wireless Commun. 1276 https://doi.org/10.1109/TWC.2021.3067163.

Sun, S., Zhang, G., Mei, H., Wang, K., Yang, K., 2021. Optimizing multi-UAV deployment in 3-D space to minimize task completion time in UAV-enabled mobile edge computing systems. IEEE Commun. Lett. 25, 579–583. https://doi.org/10.1109/LCOMM.2020.3029144.

Tun, Y.K., Park, Y.M., Tran, N.H., Saad, W., Pandey, S.R., Hong, C.S., 2020. Energy-efficient resource management in UAV-assisted mobile edge computing. IEEE Commun. Lett. 25, 249–253. https://doi.org/10.1109/lcomm.2020.3026033.

Van Hasselt, H., Guez, A., Silver, D., 2016. Deep reinforcement learning with double Q-Learning. 30th AAAI Conf. Artif. Intell. AAAI 2016, 2094–2100.

Wan, S., Lu, J., Fan, P., Letaief, K.B., 2020. Toward big data processing in IoT: path planning and resource management of UAV base stations in mobile-edge computing system. IEEE Internet Things J. 7, 5995–6009. https://doi.org/10.1109/JIOT.2019.2954825.

Wang, H., Ke, H., Sun, W., 2020. Unmanned-aerial-vehicle-assisted computation offloading for mobile edge computing based on deep reinforcement learning. IEEE Access 8, 180784–180798. https://doi.org/10.1109/ACCESS.2020.3028553.

Wang, J., Jin, C., Tang, Q., Xiong, N., Srivastava, G., 2020. Intelligent ubiquitous network accessibility for wireless-powered MEC in UAV-assisted B5G. IEEE Trans. Netw. Sci. Eng. 4697 https://doi.org/10.1109/tnse.2020.3029048, 1–1.

Wang, L., Wang, K., Pan, C., Xu, W., Aslam, N., Hanzo, L., 2021a. Multi-agent deep reinforcement learning-based trajectory planning for multi-UAV assisted mobile edge computing. IEEE Trans. Cogn. Commun. Netw. 7, 73–84. https://doi.org/10.1109/TCCN.2020.3027695.

Wang, L., Wang, K., Pan, C., Xu, W., Aslam, N., Nallanathan, A., 2021b. Deep reinforcement learning based dynamic trajectory control for UAV-assisted mobile edge computing. IEEE Trans. Mobile Comput. 1233, 1–15. https://doi.org/10.1109/TMC.2021.3059691.

Wang, Q., Gao, A., Hu, Y., 2021. Joint power and QoE optimization scheme for multi-UAV assisted offloading in mobile computing. IEEE Access 9, 21206–21217. https://doi.org/10.1109/ACCESS.2021.3055335.

Wang, Q., Shao, S., Guo, S., Qiu, X., Wang, Z., 2020. Task allocation mechanism of power internet of things based on cooperative edge computing. IEEE Access 8, 158488–158501. https://doi.org/10.1109/ACCESS.2020.3020233.

Wang, X., Han, Y., Wang, C., Zhao, Q., Chen, X., Chen, M., 2019. In-edge AI: intelligentizing mobile edge computing, caching and communication by federated learning. IEEE Netw 33, 156–165. https://doi.org/10.1109/MNET.2019.1800286.

Wang, Y., Ru, Z.Y., Wang, K., Huang, P.Q., 2020. Joint deployment and task scheduling optimization for large-scale mobile users in multi-UAV-enabled mobile edge computing. IEEE Trans. Cybern. 50, 3984–3997. https://doi.org/10.1109/TCYB.2019.2935466.

Wu, G., Miao, Y., Zhang, Y., Barnawi, A., 2020. Energy efficient for UAV-enabled mobile edge computing networks: intelligent task prediction and offloading. Comput. Commun. 150, 556–562. https://doi.org/10.1016/j.comcom.2019.11.037.

Wu, Q., Li, G.Y., Chen, W., Ng, D.W.K., Schober, R., 2017. An overview of sustainable G reen 5G networks. IEEE Wirel. Commun. 72–80.

Wu, Q., Liu, L., Zhang, R., 2019. Fundamental trade-offs in communication and trajectory design for UAV-enabled wireless network. IEEE Wirel. Commun. 26, 36–44. https://doi.org/10.1109/MWC.2018.1800221.

Xiong, J., Guo, H., Liu, J., 2019. Task offloading in UAV-aided edge computing: bit allocation and trajectory optimization. IEEE Commun. Lett. 23, 538–541. https://doi.org/10.1109/LCOMM.2019.2891662.

Xu, D., Li, Y., Chen, X., Li, J., Hui, P., Chen, S., Crowcroft, J., 2018. A survey of opportunistic offloading. IEEE Commun. Surv. Tutorials 20, 2198–2236. https://doi.org/10.1109/COMST.2018.2808242.

Xu, J., Liu, X., Li, X., Zhang, L., Jin, J., Yang, Y., 2021. Energy aware computation management strategy for smart logistic system with MEC. IEEE Internet Things J. 4662, 1–17. https://doi.org/10.1109/JIOT.2021.3115346.

Xu, J., Ota, K., Dong, M., 2020. Big data on the fly: UAV-mounted mobile edge computing for disaster management. IEEE Trans. Netw. Sci. Eng. 7, 2620–2630. https://doi.org/10.1109/TNSE.2020.3016569.

Xu, Y., Zhang, T., Loo, J., Yang, D., Xiao, L., 2021a. Completion time minimization for UAV-assisted mobile-edge computing systems. IEEE Trans. Veh. Technol. 9545 https://doi.org/10.1109/tvt.2021.3112853, 1–1.

Xu, Y., Zhang, T., Yang, D., Liu, Y., Tao, M., 2021b. Joint resource and trajectory optimization for security in UAV-assisted MEC systems. IEEE Trans. Commun. 69, 573–588. https://doi.org/10.1109/TCOMM.2020.3025910.

Xue, J., Hu, Q., An, Y., Wang, L., 2020. Joint task offloading and resource allocation in UAV-enabled mobile edge computing. IEEE Internet Things J. 177, 77–85. https://doi.org/10.1016/j.comcom.2021.06.014.

Yang, B., Cao, X., Yuen, C., Qian, L., 2021. Offloading optimization in edge computing for deep-learning-enabled target tracking by internet of UAVs. IEEE Internet Things J. 8, 9878–9893. https://doi.org/10.1109/JIOT.2020.3016694.

Yang, L., Yao, H., Wang, J., Jiang, C., Benslimane, A., Liu, Y., 2020. Multi-UAV-enabled load-balance mobile-edge computing for IoT networks. IEEE Internet Things J. 7, 6898–6908. https://doi.org/10.1109/JIOT.2020.2971645.

Yang, Z., Chen, M., Liu, X., Liu, Y., Chen, Y., Cui, S., Poor, H.V., 2021. AI-driven UAV-NOMA-MEC in next generation wireless networks. IEEE Wirel. Commun.

Yao, Y., Diao, X., Wang, X., Zheng, J., Cai, Y., Zhou, J., 2019. Joint offloading strategy and trajectory optimization for aerial-ground cooperative mobile edge computing systems. IET Conf. Publ. 31–37. https://doi.org/10.1049/cp.2019.1145, 2019.

Yaqoob, I., Ahmed, E., Gani, A., Mokhtar, S., Imran, M., Guizani, S., 2016. Mobile ad hoc cloud: a survey. Wireless Commun. Mobile Comput. 16, 2572–2589. https://doi.org/10.1002/wcm.2709.

Yazid, Y., Zazi, I.E., Gonzalez, A.G., Oualkadi, A. El, Arioua, M., 2021. UAV-Enabled Mobile Edge-Computing for IoT Based on AI : A Comprehensive Review. drones.

Yilmaz, S.S., Ozbek, B., 2021. Multi-helper NOMA for cooperative mobile edge computing. IEEE Trans. Intell. Transport. Syst. 1–10. https://doi.org/10.1109/TITS.2021.3116421.

Zeng, Y., Zhang, R., 2017. Energy-efficient UAV communication with trajectory optimization. IEEE Trans. Wireless Commun. 16, 3747–3760. https://doi.org/10.1109/TWC.2017.2688328.

Zhan, C., Hu, H., Liu, Z., Wang, Z., Mao, S., 2021. Multi-UAV-enabled mobile-edge computing for time-constrained IoT applications. IEEE Internet Things J. 8, 15553–15567.

Zhan, C., Hu, H., Sui, X., Liu, Z., Niyato, D., 2020. Completion time and energy optimization in the UAV-enabled mobile-edge computing system. IEEE Internet Things J. 7, 7808–7822. https://doi.org/10.1109/JIOT.2020.2993260.

Zhang, J., Zhou, L., Zhou, F., Seet, B.C., Zhang, H., Cai, Z., Wei, J., 2020. Computation-efficient offloading and trajectory scheduling for multi-UAV assisted mobile edge computing. IEEE Trans. Veh. Technol. 69, 2114–2125. https://doi.org/10.1109/TVT.2019.2960103.

Zhang, K., Gui, X., Ren, D., Li, D., 2021. Energy-latency tradeoff for computation offloading in UAV-assisted multiaccess edge computing system. IEEE Internet Things J. 8, 6709–6719. https://doi.org/10.1109/JIOT.2020.2999063.

Zhang, L., Ansari, N., 2021. Optimizing the operation cost for UAV-aided mobile edge computing. IEEE Trans. Veh. Technol. 70, 6085–6093. https://doi.org/10.1109/TVT.2021.3076980.

Zhang, L., Ansari, N., 2020. Latency-aware IoT service provisioning in UAV-aided mobile-edge computing networks. IEEE Internet Things J. 7, 10573–10580. https://doi.org/10.1109/JIOT.2020.3005117.

Zhang, L., Zhang, Z.Y., Min, L., Tang, C., Zhang, H.Y., Wang, Y.H., Cai, P., 2021. Task offloading and trajectory control for UAV-assisted mobile edge computing using deep reinforcement learning. IEEE Access 9, 53708–53719. https://doi.org/10.1109/ACCESS.2021.3070908.

Zhang, Q., Chen, J., Ji, L., Feng, Z., Han, Z., Chen, Z., 2020. Response delay optimization in mobile edge computing enabled UAV swarm. IEEE Trans. Veh. Technol. 69, 3280–3295. https://doi.org/10.1109/TVT.2020.2964821.

Zhang, S., Wu, Q., Xu, S., Li, G.Y., 2017. Fundamental green tradeoffs: progresses, challenges, and impacts on 5G networks. IEEE Commun. Surv. Tutorials 19, 33–56. https://doi.org/10.1109/COMST.2016.2594120.

Zhang, T., Xu, Y., Loo, J., Yang, D., Xiao, L., 2020. Joint computation and communication design for UAV-assisted mobile edge computing in IoT. IEEE Trans. Ind. Inf. 16, 5505–5516. https://doi.org/10.1109/TII.2019.2948406.
Zhao, L., Yang, K., Tan, Z., Li, X., Sharma, S., Liu, Z., 2021. A novel cost optimization strategy for SDN-enabled UAV-assisted vehicular computation offloading. IEEE Trans. Intell. Transport. Syst. 22, 3664–3674. https://doi.org/10.1109/TITS.2020.3024186.
Zheng, X., Li, M., Tahir, M., Chen, Y., Alam, M., 2019. Stochastic computation offloading and scheduling based on mobile edge computing. IEEE Access 7, 72247–72256. https://doi.org/10.1109/ACCESS.2019.2919651.
Zhou, F., Hu, R.Q., Li, Z., Wang, Y., 2020. Mobile edge computing in unmanned aerial vehicle networks. IEEE Wirel. Commun. 27, 140–146. https://doi.org/10.1109/MWC.001.1800594.
Zhou, Y., Pan, C., Yeoh, P.L., Wang, K., Elkashlan, M., Vucetic, B., Li, Y., 2020. Secure communications for UAV-enabled mobile edge computing systems. IEEE Trans. Commun. 68, 376–388. https://doi.org/10.1109/TCOMM.2019.2947921.
Zhu, S., Gui, L., Zhao, D., Cheng, N., Zhang, Q., Lang, X., 2021. Learning-based computation offloading approaches in UAVs-assisted edge computing. IEEE Trans. Veh. Technol. 70, 928–944. https://doi.org/10.1109/TVT.2020.3048938.

S. M. ASIFUL HUDA received the B.S. degree in computer science and engineering from East West University, Bangladesh, in 2019. He is currently pursuing his M.S. degree in computer engineering at Chosun University, South Korea. From 2019 to 2020, he worked as a software engineer with Technology and Business Solution Ltd., Bangladesh. His research interest includes but is not limited to unmanned aerial vehicle networks, mobile edge computing, machine learning, and deep reinforcement learning.

SANGMAN MOH received the Ph.D. degree in computer engineering from Korea Advanced Institute of Science and Technology (KAIST), South Korea, in 2002. Since then, he is a professor at the Dept. of Computer Engineering at Chosun University, Korea. From 2006 to 2007, he was on leave at Cleveland State University, USA. Until 2002, he had been with Electronics and Telecommunications Research Institute (ETRI), Korea, where he served as a project leader after receiving the M.S. degree in computer science from Yonsei University, Korea, in 1991. His research interests include mobile computing and networking, *ad hoc* and sensor networks, cognitive radio networks, unmanned aerial vehicle networks, and mobile-edge computing. Dr. Moh is a member of the IEEE, the ACM, the IEICE, the KIISE, the IEIE, the KIPS, the KICS, the KMMS, the IEMEK, the KISM, and the KPEA.