

Article

Optimizing Trajectory and Dynamic Data Offloading Using a UAV Access Platform

Amanjot Kaur ^{1,2,*}, Shashi Shekhar Jha ¹, Jiong Jin ²  and Hadi Ghaderi ³

¹ Department of Computer Science and Engineering, Indian Institute of Technology Ropar, Rupnagar 140001, Punjab, India

² School of Science, Computing and Engineering Technologies, Swinburne University of Technology, Melbourne, VIC 3122, Australia

³ School of Business, Law and Entrepreneurship, Swinburne University of Technology, Melbourne, VIC 3122, Australia

* Correspondence: amanjot.kaur@iitrpr.ac.in

Abstract: The use of unmanned aerial vehicles (UAV) as an integrated sensing and communication platform is emerging for surveillance and tracking applications, especially in large infrastructure-deficient environments. In this study, we develop a multi-UAV system to collect data dynamically in a resource-constrained context. The proposed approach consists of an access platform called Access UAV (*A_UAV*) that stochastically coordinates the data collection from the Inspection-UAVs (*I_UAVs*) equipped with a visual sensor to relay the same to the cloud. Our approach jointly considers the trajectory optimization of *A_UAV* and the stability of the data queues at each UAV. In particular, the Distance and Access Latency Aware Trajectory (DLAT) optimization for *A_UAVs* is developed, which generates a fair access schedule for *I_UAVs*. Moreover, a Lyapunov-based online optimization ensures the system stability of the average queue backlogs for dynamic data collection while minimizing total system energy. Coordination between *I_UAV* and *A_UAV* is achieved through a message-based mechanism. The simulation results validate the performance of our proposed approach against several baselines under different parameter settings.

Keywords: trajectory planning; data collection; unmanned aerial vehicles (UAVs); access schedule; multi-UAV system; optimization



Citation: Kaur, A.; Jha, S.S.; Jin, J.; Ghaderi, H. Optimizing Trajectory and Dynamic Data Offloading Using a UAV Access Platform. *IoT* **2022**, *3*, 473–492. <https://doi.org/10.3390/iot3040025>

Academic Editor: Hyun-Ho Choi

Received: 18 October 2022

Accepted: 17 November 2022

Published: 24 November 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Intelligent solutions utilising unmanned aerial vehicles (UAV) are emerging in various domains such as wireless sensing [1], payload delivery [2], precision agriculture [3] and search and rescue operations [4]. Moreover, with the current trend of automation, sensing, and information exchange within Industry 4.0 environments, UAV-based applications are also finding their place in construction, mining, agriculture and logistics industries, especially for resource tracking and operational monitoring using aerial imagery. UAV-based solutions are particularly beneficial in large infrastructure-deficient environments as they offer ease of deployment, quick access to ground-truth data, and higher reachability and coverage [5,6]. Furthermore, autonomous or semi-autonomous UAV-based solutions could facilitate many industry-specific audits such as progress monitoring, resource and safety inspections, environmental hazards, and many more. Although integrating a multi-UAV based visual sensing and monitoring system has many benefits, developing such a system is challenging. A few of those challenges include (1) monetary budget constraints which limit the number of deployed UAVs, (2) limited battery of UAVs restricting the observation span, (3) limited on-board processing capabilities of UAVs which necessitates online offloading of data, and (4) limited connectivity for data gathering and offloading tasks in infrastructure-deficient environments that requires efficient trajectory planning. The computationally intensive nature of the data collection task and limited on-board computation

of UAVs impede the deployment of such solutions to collect and/or process data within large infrastructure-deficient sites, such as tracking the progress of complex construction projects. Hence, efficient multi-UAV mechanisms to collect and offload data from the field or points of interests (PoIs) to the backend system (cloud) are required to leverage the advantages of multi-UAV systems further. To address the above-mentioned challenges, a feasible solution could be the adoption of a heterogeneous multi-UAV framework.

UAVs can also be considered as integrated sensing and communication platforms for providing mobile edge computing services [7–9]. The edge-based UAV platform could be flexible and cost-effective for computational and data-offloading tasks in infrastructure-deficient environments. Furthermore, with the optimal trajectory design of an edge-based access platform, the overall coverage of the PoIs can be improved. Thus an Access UAV (*A_UAV*) platform can be optimized for data collection, basic data processing, and data offloading to the cloud within infrastructure-deficient environments. However, if the PoIs are arbitrarily distributed in 3D space such as the monitoring points in under-construction projects, the maneuverability of the UAVs becomes challenging. UAVs that can be more agile by design are therefore better suited to reach arbitrary locations quickly and safely. Furthermore, such agile UAVs can carry different sensors such as multi-spectral cameras, near-infrared cameras, etc. to collect the requisite data from the locations. Hence, instead of overloading the *A_UAV* with data collection and offloading tasks, it would be prudent to segregate the UAV functions to optimize them for specific tasks. In this paper, we introduce the Inspection UAVs (*I_UAVs*) that are morphologically different from the *A_UAV*, as they are smaller and more flexible to maneuver at lower heights. The *I_UAV* act as a mobile visual sensing platform to collect visual data from different locations in the environment. *I_UAVs* can be considered mobile sensors scattered throughout the environment. The *A_UAV* thus need to locate these dynamically moving *I_UAVs* in order to collect data from them.

Another problem is the buffer overflow of UAVs in data collection and offloading tasks. The limited on-board processing capability and shared bandwidth to transfer data leading to the overall system's instability. In addition, the varying data traffic and continuous movement of UAVs make it challenging to stabilize the system in a deterministic manner. Researchers have used online Lyapunov optimization [10] to address such system instabilities. Lyapunov optimization considers the stability of a system with time-varying data and optimizes the time averages of the system utility and queue backlogs.

In this study, we address the challenges of deploying a heterogeneous multi-UAV system comprising a single UAV access platform called the Access UAV (*A_UAV*) and multiple Inspection UAVs (*I_UAVs*) for dynamic data collection in infrastructure-deficient environments. We propose a Distance and Access Latency Aware Trajectory (DLAT) of the *A_UAV* by considering the fair access schedules of the dynamically moving *I_UAVs*. In addition, to address system instabilities due to queue backlogs, the proposed solution employs a Lyapunov-based online optimization approach. The overall problem has been formulated as minimizing the total energy consumption of the system. Furthermore, as the set of *I_UAVs* and *A_UAV* operate independently without any central entity, their coordination is ensured through a message-based estimation mechanism.

The contributions of this research are fourfold, which are summarised as follows:

- An optimization model has been proposed for optimal trajectory planning of *A_UAV* to fairly access the dynamically moving *I_UAVs* in different time slots. The optimization model focuses on minimizing the distance travelled and generating a fair access schedule of *I_UAVs*.
- From an applications point of view, this research focuses on trajectory optimization of the *A_UAV* to consider the limited battery constraint and the distance from the location of battery replenishment unit at the site.
- Further, a model based on the Lyapunov based online optimization framework is proposed to minimize the energy consumption and the queue backlog of *A_UAV* and *I_UAVs* with limited storage capacity.

- Finally, extensive numerical experiments are conducted to evaluate the efficiency and performance of the proposed framework against multiple baselines.

The rest of the paper is organised as follows: Section 2 summarises the related literature on edge based UAV applications. Section 3 presents the proposed multi-UAV framework and the system model. The overall system objective is discussed in Section 4. Sections 5 and 6 discuss the access latency aware trajectory optimization and Lyapunov based system stability, respectively. Sections 7 and 8 discuss the experiments and results. Finally, Section 9 concludes the paper.

2. Related Work

Various studies have emphasized that trajectory planning of UAVs is an integral component of the UAV-based inspection and monitoring applications [11,12]. In [13], the authors presented the reconstruction of a 3D model and highlighted the importance of UAV trajectories for computer vision techniques to reconstruct the 3D structure accurately. In [14], the authors discussed how MEC can be divided into different architectures based on the role of UAVs, which could be users, computing entities, or data relay entities. The UAV-enabled MEC system is commonly employed in different scenarios to improve user experience and service availability or to increase the system's efficiency. The trajectory optimization of UAVs is an integral part of such MEC systems as it affects the energy consumption of the system and the service schedule of static or dynamic sensors. UAVs could be deployed to relay data further or provide partial computing to improve the system overall quality of service (QoS). In [15], multiple UAVs were deployed for data relay task from mobile devices to the BS. The overall objective was to minimize the energy consumption of mobile devices by jointly optimizing the task scheduling and UAV trajectories in resource-constrained environment. Using a different approach, Ref. [16] proposed a single UAV-mounted cloudlet to serve a set of mobile users. The overall framework minimizes the energy consumption of mobile users, while optimizing the trajectory of the UAV-mounted cloudlet. The work of Xu et al. [17] also considered the multi-UAV based computing framework to minimize the latency of mobile device data relay task either by on-board computing or relaying to BS. In [18], a hierarchical multi-coalition UAV MEC network was discussed where the resource-constrained UAVs could offload task to other UAVs with high computational resources to improve the overall system efficiency. However, the authors did not consider the queue optimization, dynamic access of UAVs and challenges of an infrastructure-deficient environment as modelled in our work. In [19], authors focused on minimizing the weighted sum of energy consumption of UAV enabled MEC system. They performed joint optimization of computation resource scheduling, bandwidth allocation to user equipment (UEs), and trajectory optimization of UAV-based edge servers with static ground sensors. The advantage of using multiple UAVs in the MEC system is further studied in the work of Diao et al. [20], where the effects of joint optimization of trajectories of multiple UAVs to improve the system metrics were considered. However, the dynamic evolution of the data queues of the UAV-based MEC system could alleviate the problem of queue stability and data offloading.

The authors in [21] addressed the stability issues with a Lyapunov-based joint resource optimization of bandwidth usage, processing power consumption, and transmission power. Zhang et al. [22] presented a complex system within a dynamic environment that involves joint optimization of the computation resources of the multiple mobile users, UAV-BS, and trajectory optimization. The authors in [23] discussed a UAV-assisted mobile edge computing framework that jointly addressed energy minimization, trajectory optimization, CPU frequency and offloading schedule. In [24], author considered the completion time of the task along with the energy minimization and trajectory optimization of a UAV. One significant difference between our work and those reviewed in the literature is the estimation of the location of dynamic sensors (i.e., I_{UAV}). This problem brings another challenge of coordination among I_{UAV} s and A_{UAV} in the absence of ubiquitous connectivity with a limited battery. The proposed solution in this study attempts to solve both problems.

The literature also discusses the network scheduling problem along with trajectory scheduling for UAV-based MEC. In [25], authors developed a hierarchical MEC system considering online optimization of computational resources and reinforcement learning based trajectory optimization of multiple UAV-BSs for collecting data from a set of static sensors. In [26], a sense and send transmission protocol was proposed using multiple UAVs in a cellular network using an iterative trajectory sensing and scheduling algorithm. However, this approach does not consider distributed and multi-layer interaction of UAVs to collect and offload data with limited connectivity. In [27], the authors employed reinforcement learning for sensing and sending information using a decentralized setup for multiple UAVs, however, their work did not consider the multi-layer UAV network with limited connectivity. As apparent from the literature, resource scheduling in multi-UAV based solutions is a challenging task, particularly in an infrastructure-deficient environment with limited connectivity. Therefore, the dynamic deployment of mobile UAVs either to collect data or relay data to the cloud could mitigate the issues of progress tracking and job monitoring in industrial settings and aid in the performance of project deliveries. This paper proposes a solution for end-to-end data offloading in large infrastructure-deficient environments using a hierarchical multi-UAV system.

3. System Model

This section presents the key components of the proposed multi-UAV framework. The system consists of heterogeneous UAVs, including a set of N Inspection UAVs (I_UAVs) and a single UAV Access Platform (A_UAV). I_UAVs are smaller in size and more agile. They collect visual data from a set of k Point of Interests (PoIs) denoted as l_i . Because the framework considers infrastructure-less environments, limited Access Points (APs) available for cloud connectivity. Further, I_UAVs possess a limited connectivity range, making it difficult to transfer data directly to the cloud. A_UAV , which is larger in size and possesses higher computational capabilities, coordinates with the I_UAVs to collect data. We assume that the A_UAV always maintains a constant height, thus its trajectory lies in a horizontal plane. Figure 1 shows a high level overview of the system under consideration with I_UAVs tasked to collect data from the PoIs, whereas the A_UAV collects data from the dynamically moving I_UAVs and relay it to the cloud. The notations used in this study are given in Table 1.

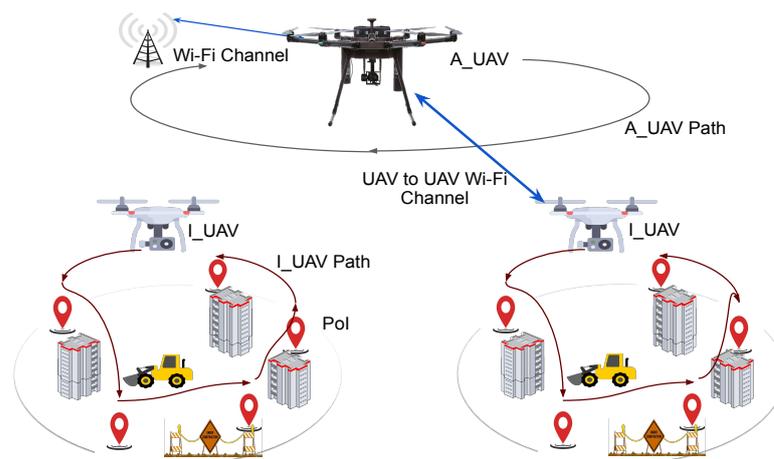


Figure 1. System Setup.

Table 1. List of Notations.

Parameters	Description
T	Set of time slots
N	Set of I_UAV s deployed to collect images
$Q_i(t)$	The queue of the i^{th} I_UAV in time slot t
$S_i(t)$	The position of the I_UAV in time slot t
$S_{access}(t)$	The position of A_UAV in time slot t
$L(t)$	The queue length of A_UAV server in time slot t
$d_i^{off}(t)$	The amount of data offloaded to A_UAV by the I_UAV in time slot t
$A_i(t)$	The amount of data bits arrived at i^{th} I_UAV in time slot t
$d_{access}^{off}(t)$	The amount of data offloaded by the A_UAV in time slot t
$p_i(t)$	Transmission power of i^{th} I_UAV in time slot t
$P_{access}(t)$	Transmission power of A_UAV in time slot t
τ	The duration of time slot
$v(t)$	The velocity of A_UAV
e_0, e_1	Environmental constants
$E_i^{comm}(t)$	Transmission energy of I_UAV
$E_{access}^{transition}(t)$	Transition energy of the A_UAV in time slot t
$E_{access}^{comm}(t)$	Transmission energy of A_UAV in time slot t
V	The tradeoff parameter between transition and transmission energy
g_0	Path loss constant
ϕ	Path loss exponent
ζ	Channel power gain
l_i	i^{th} tuple with location and data information
ψ_i	3D coordinates of the PoI
N_0	Power spectral density of noise
θ	Elevation angle of A_UAV
$R_i(t)$	Access latency of the i^{th} I_UAV in the time-slot t
$x_i(t)$	Binary variable for the selection of I_UAV s
σ	standard deviation of normal distribution used for data generation at each PoI
ϱ	the maximum distance between two consecutive PoIs in the trajectory of I_UAV s
μ	mean value of normal distribution used for data generation at each PoI
Q_{max}	max buffer of I_UAV
τ_{search}	search time to find the exact location of I_UAV s
τ_{comm}	time allotted for data transmission
τ_{trans}	time taken to transit from one location to another

3.1. Communication Channel

The communication between I_UAV and A_UAV ($A2A$ channel) has a limited range and capacity. This work assumes that the achievable data transmission rate of the i^{th} I_UAV in a given time slot as $d_i^{off}(t)$. The communication channel between I_UAV s and A_UAV involves both line-of-sight (LoS) and non-line-of-sight (NLoS) links as PoIs can be distributed vertically and longitudinally. Furthermore, the shadowing effect is also considered

due to obstructions caused by buildings and other structures in the surroundings [28,29]. The path loss of a link is given as follows:

$$L_\alpha = L_\alpha(r_0) + 10\phi \log\left(\frac{r'}{r_0}\right) + X_\sigma \tag{1}$$

where X_σ is a shadowing factor that is indirectly proportional to the altitude of the PoI, $\alpha \in \{LoS, NLoS\}$ and ϕ is the path loss exponent. The probability of LoS link, (P_{LoS}), depends on the angle of elevation and environmental constraints (e_0 and e_1) as given in Equation (2):

$$P_{LoS} = \frac{1}{1 + e_0 \cdot \exp(-e_1[\theta - e_0])} \tag{2}$$

The average path-loss is calculated as:

$$L = P_{LoS} \cdot L_{LoS} + (1 - P_{LoS}) \cdot L_{NLoS} \tag{3}$$

In this work, we have assumed Wi-Fi technology without a fixed access point for emergency or infrastructure deficient scenarios [30]. The network of I_UAVs and A_UAV provides connectivity to send collected data from PoIs to the cloud.

3.2. Data Gathering Process

Each PoI (l_j) is a tuple ($\langle d_j, O_j \rangle$), where d_j specifies the amount of data (e.g., images) to be collected and O_j denotes the 3D coordinates of the PoI. The sequence of PoIs to be visited is provided to the I_UAVs and the same is also shared with the A_UAV . During the traversal along the sequence of PoIs, if the buffer of any of the I_UAVs overflows then that I_UAV waits at the same PoI until its data is offloaded.

In order to gather and offload data, the A_UAV communicates with a single I_UAV in a time slot. Let us denote the data gathered by each of the I_UAVs in a time slot t by $A_i(t)$. Let $Q_i(t)$ be the queue of the i^{th} I_UAV and $d_i^{off}(t)$ denotes the amount of data offloaded to the A_UAV by the i^{th} I_UAV in time slot t . The recursive equation to update the $Q_i(t)$ is as follows:

$$Q_i(t + 1) = \max\{Q_i(t) - d_i^{off}(t), 0\} + A_i(t) \tag{4}$$

Let $L(t)$ be the queue of the A_UAV where A_UAV accepts the data from the selected I_UAV in the time slot t . The following equation updates $L(t)$ recursively:

$$L(t + 1) = \max\{L(t) + d_i^{off}(t) - d_{access}^{off}(t), 0\} \tag{5}$$

where $d_{access}^{off}(t)$ is the amount of data offloaded to the cloud by the A_UAV in time slot t . Figure 2 shows the different functions performed by an A_UAV in a single time-slot. The decision function takes negligible time to decide on the next I_UAV for data gathering, followed by the transition function where A_UAV takes τ_{trans} time to move near the next possible location to connect with the chosen I_UAV . The search function (τ_{search}) estimates the location of the selected I_UAV based on the queue and position estimation algorithm given in Algorithm 1. The bound on the maximum time required to estimate the position of I_UAVs is discussed in Section 5.1. Finally, the data transmission function establishes the successful communication with the I_UAV (if it is not shadowed). The sequence of the functions mentioned above is repeated for every time slot. The next section describes the objective of the system and formulates it as an optimization problem.

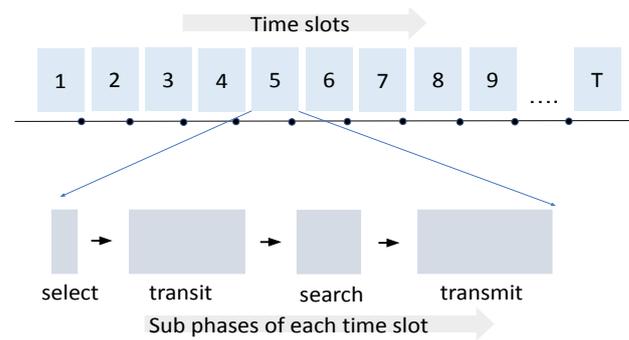


Figure 2. Division of a timeslot with different functions of A_{UAV} .

Algorithm 1 Estimated position and queue length of I_{UAV}

Initialization:

```

time_elapsed ← cur_timeslot-last_access_timeslot
li = last_accessed_position of I_UAVs
ψmin = li
ψmax = li
Qi = last_accessed_buffer I_UAVs
curr_location_data = data left at last_accessed_position of I_UAVs
Modes = Min or Max
if Modes == Min then D = Dmin
if Modes == Max then D = Dmax
for all I_UAVs do
  for all Modes do
    j ← 0
    while j ≤ time_elapsed do
      if Qi ≤ Qmax then
        if curr_location_data is not collected then
          li = last_accessed_position
          Qi = last_accessed_buffer+data_at_current_loc
          j ← j + 1
        else
          i ← i + 1
          li = next_position
          Qi = last_accessed_buffer+Dmin or Dmax
          j ← j + 1
          if Mode == Min then
            Qi,min = Qi
            ψmin = ψmin ∪ li
          else
            Qi,max = Qi
            ψmax = ψmax ∪ li
        else
          li = last_accessed_position
          Qi = last_accessed_buffer
          break
    return Qi,max, ψmax, Qi,min, ψmin

```

4. System Objective

In the proposed framework, the offloading of data happens at two stages—(1) from I_{UAV} to A_{UAV} and (2) from A_{UAV} to the cloud. Our main focus is to achieve end-to-

end data offloading to the cloud by minimizing the total energy consumption of the whole system (E_{sys}) given as:

$$E_{sys}(t) = E_{access}^{trans}(t) + E_{access}^{comm}(t) + E_{access}^{hover}(t) + \left(\sum_{i=1}^N (E_i^{comm}(t)) \right) \quad (6)$$

where $E_{access}^{trans}(t)$ is the transition energy of the A_UAV , $E_{access}^{comm}(t)$ is the transmission of the A_UAV , $E_{access}^{hover}(t)$ is the hovering energy of A_UAV and $E_i^{comm}(t)$ is the transmission energy of the i^{th} I_UAV . The following subsections discusses the details of calculating each component of energy consumption in Equation (6).

4.1. Transition Energy of A_UAV

The transition energy of A_UAV refers to the energy consumed when moving from one location to another [11,16,31] which is given as:

$$E_{access}^{trans} = \kappa \cdot ||vel(t)||^2 \tau_{trans} \quad (7)$$

where κ is a constant that depends on the total mass of the A_UAV , $vel(t)$ is the velocity of A_UAV and τ_{trans} is the time taken to transit from one location to another.

4.2. Transmission Energy of A_UAV

A_UAV offloads data to the cloud via a wireless channel [32]. The transmission energy consumed to transmit the data to the cloud is given as:

$$E_{access}^{comm}(t) = \left(2^{\frac{d_{access}^{off}(t)}{W \cdot \tau}} - 1 \right) \cdot \frac{N_0 W}{\zeta} \cdot \tau_{comm} \quad (8)$$

where τ_{comm} is the time allotted for data transmission. Other parameters such as $d_{access}^{off}(t)$, W , τ , N_0 , ζ are defined in the Table 1.

4.3. Hovering Energy of A_UAV

A_UAV hovers above the PoI to collect the data. The hovering energy consumed to collect the data is given as:

$$E_{access}^{hover}(t) = P_{hover} \cdot \tau_{hover} \quad (9)$$

where, P_{hover} is the power consumed while hovering per unit time and τ_{hover} is the time for hovering.

4.4. Transmission Energy of I_UAVs

The energy consumed for offloading the $d_i^{off}(t)$ data bits at time slot t from the selected I_UAV to the A_UAV using the Air to Air channel of bandwidth W Hz is given similarly to Equation (8) as:

$$E_i^{comm}(t) = \left(2^{\frac{d_i^{off}(t)}{W \cdot \tau}} - 1 \right) \cdot \frac{N_0 W}{\zeta} \cdot \tau \quad (10)$$

The wireless (Air to Air) channel power gain (ζ) from I_UAV to A_UAV can be given as:

$$\zeta = g_0 \cdot \left(\frac{r_0}{r'} \right)^\phi \quad (11)$$

where g_0 is the path loss constant, r_0 is the reference distance, r' is the distance between the UAVs, ϕ is the path loss exponent and τ is the time.

Given the system's energy consumption, our goal is to find the optimal settings to minimize the expected cumulative energy across the time horizon. The decision variables in every time slot t that affect the total system's energy are given by the set $\pi(t) = \{p_i(t), P_{access}(t), S_{access}(t)\}$ corresponding to the transmission energies of the I_UAVs &

A_UAV , and the transition energy of A_UAV , respectively. Moreover, the channel information for the data offloading task is not deterministic and varies in the environment, hence the amount of data arrived at the A_UAV becomes stochastic which depends on the channel characteristics and the position of the selected I_UAV . Further, this framework does not consider the energy consumed for the movement of I_UAVs as the PoIs are predefined and the I_UAVs follow a predetermined trajectory consuming constant energy. The overall optimization model for the stable system performance is formulated as:

$$\mathbf{P1} \quad \min_{\pi(t)} \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=1}^T \mathbb{E}[E_{sys}(t)] \quad (12)$$

s.t.

$$p_i(t) \leq p^{max}, \quad \forall i, \forall t \quad (13)$$

$$P_{access}(t) \leq P^{max}, \quad \forall t \quad (14)$$

$$\|S_{access}(t) - S_i(t)\| \leq v_{max} \tau, \quad \forall i, \forall t \quad (15)$$

$$\sum_{i=1}^N \frac{R_i(t)(1 - x_i(t))}{N} \leq R_{max}, \quad \forall i, \forall t \quad (16)$$

$$d_i^{off}(t) \leq Q_i(t), \quad \forall i, \forall t \quad (17)$$

$$d_i^{off}(t) \leq W \tau_{comm} \log_2 \left(1 + \frac{\zeta p^{max}(t)}{N_o W} \right), \quad \forall i, \forall t \quad (18)$$

$$d_{access}^{off}(t) \leq W \tau_{comm} \log_2 \left(1 + \frac{\zeta P^{max}(t)}{N_o W} \right), \quad \forall t \quad (19)$$

$$\lim_{T \rightarrow \infty} \frac{\mathbb{E}[Q_i(t)]}{T} = 0, \quad \forall i, \forall t \quad (20)$$

$$\lim_{T \rightarrow \infty} \frac{\mathbb{E}[L(t)]}{T} = 0, \quad \forall t \quad (21)$$

Constraints (13) and (14) define the maximum transmission power of I_UAVs and A_UAV , respectively. Constraint (15) limits the maximum transition energy of A_UAV for every transition and Constraint (16) limits the time that has elapsed since the last access of i^{th} I_UAV to be less than R_{max} . Additionally (17)–(19) bound the number of transmitted bits. Constraints (20) and (21) establish the rate stability of all the system queues (I_UAVs and A_UAV). Next, the model to optimize the trajectory of the A_UAV with respect to the trajectories of I_UAVs is discussed.

5. Distance and Latency Aware Trajectory (DLAT) Optimization

Flexible and dynamic trajectory planning of A_UAV is crucial to applications where terrestrial communication infrastructure is missing. As already mentioned, the position of I_UAVs changes in every time-slot since they move through different PoIs to collect data. The A_UAV 's trajectory needs to be planned so that it can connect and access an I_UAV in a time-slot before the I_UAV 's queue overflows. Whenever an I_UAV 's queue gets full, it does not move to its next designated PoI. Instead, it sojourns at the same PoI until it can offload its data to the A_UAV and free up the queue space. In order to choose one of the I_UAVs to gather data, the A_UAV would require the real-time information about the queues of all I_UAVs in each time-slot. This information is not available *a priori* due to the dynamic nature of the system queues. We use a message passing based approach for estimating the queues of I_UAVs to make a selection. Further, the trajectory of the A_UAV must be optimized to consume minimal energy.

The trajectory optimization model of A_UAV optimizes the trade-off between the transition energy of A_UAV and the access latencies of all I_UAVs . In addition, this access latency based data offloading generates an access fair schedule for the I_UAVs to offload their data to the A_UAV . The access latency ($R_i(t)$) of the i^{th} I_UAV in the time-slot t is

the difference between the time of its last access by the A_UAV and the current time-slot. The distance and latency aware trajectory optimization of A_UAV is formulated as:

$$P2 \min_{S_{access}(t)} \sum_{t=1}^T \sum_{i=1}^N \|S_{access}(t+1) - S_{access}(t)\|^2 - V p_i(t) \tag{22}$$

s.t.

$$\|S_{access}(t) - S_i(t)\| \leq v_{max} \tau, \quad \forall i, \forall t \tag{23}$$

$$\sum_{i=1}^N \frac{R_i(t)(1 - x_i(t))}{N} \leq R_{max}, \quad \forall i, \forall t \tag{24}$$

$$\sum_{i=1}^N (x_i(t) \cdot Q_i(t)) \geq 0, \quad \forall i, \forall t \tag{25}$$

$$\sum_{i=1}^N x_i(t) = 1, \quad \forall i, \forall t \tag{26}$$

$$p_i(t) \leq p^{max}, \quad \forall i, \forall t \tag{27}$$

$$x_i(t) \in \{0, 1\}, \quad \forall i, \forall t \tag{28}$$

where the first constraint (23) signifies that the distance travelled within a time-slot is limited by the maximum velocity. Constraint (24) limits the time that has elapsed since the last access of i^{th} I_UAV to be less than R_{max} . The constraint in (25) selects the I_UAV which has data to offload whereas (26) enforces the selection of only one of the I_UAV s in a time-slot. The selected I_UAV transmission power should be bounded as given in (27).

5.1. Estimating Position and Queue Length

The exact position and queue length of I_UAV s is not known to the A_UAV *a priori*. The A_UAV maintains the last access statistics of each I_UAV using status messages. The track of status messages received over time helps in computing the position (l_i) and queue length ($Q_i(t)$) of I_UAV s in a time-slot. The status message comprises of the remaining queue size at the time of access and the data to be collected at the current PoI. Moreover, the pre-computed trajectory of each I_UAV provides the set of PoIs to be visited by each I_UAV . Algorithm 1 describes the procedure to estimate the queue length of each I_UAV in every time-slot.

5.2. Estimation of Search Time Bound

A_UAV estimates the location of I_UAV s in each time slot using the last access statistics. It could search the set of candidate locations to locate the precise location of selected I_UAV s, which contributes to the search time. The bound on the search time depends on the data generation rate and the maximum buffer of I_UAV s as derived below.

Lemma 1. *The search time τ_{search} to locate the exact location of I_UAV with max buffer size Q_{max} is given as:*

$$\tau_{search}(|\psi|) \leq \frac{1}{3} \frac{\sigma \cdot Q_{max} \cdot \rho}{v_{max}(\mu^2 - \sigma^2)} \tag{29}$$

where ρ is the maximum distance between two consecutive PoIs in the possible set of locations to be searched and $|\psi|$ is the number of candidate locations for I_UAV and data generation process at each PoI follows the normal distribution $D \sim \mathcal{N}(\mu, \sigma^2)$

Proof. The time taken to find the location of I_UAV depends on the travel distance to cover the candidate PoI locations as given in Equation (30).

$$\tau_{search}(|\psi|) \geq |\psi| \cdot \frac{\rho}{v_{max}} \tag{30}$$

By generality,

$$|\psi_{min}| \geq |\psi_{max}| \tag{31}$$

where $\psi_{min} = \{l_i, \dots, l_{i,min}\}$ is the set of locations visited when each location has minimum data D_{min} to be collected whereas $\psi_{max} = \{l_i, \dots, l_{i,max}\}$ is the set of locations when maximum data D_{max} is present at each location. As the memory of each I_UAV is bounded by Q_{max} , it covers less number of locations for ψ_{max} as shown in Equation (31). Similarly, the data collected in both the scenarios will be same as the maximum memory size is fixed. The candidate locations are defined as the locations starting at $l_{i,max}$ and ending at $l_{i,min}$. Intuitively, the number of candidate locations $|\psi| = |\psi_{min}| - |\psi_{max}|$.

$$\begin{aligned} |\psi_{min}| \cdot D_{min} &= |\psi_{max}| \cdot D_{max} \\ |\psi_{min}| &= |\psi_{max}| \cdot \frac{D_{max}}{D_{min}} \\ |\psi_{min}| - |\psi_{max}| &= \frac{(D_{max} - D_{min})}{D_{min}} \cdot |\psi_{max}| \\ |\psi_{min}| - |\psi_{max}| &= \frac{(D_{max} - D_{min})}{D_{min}} \cdot \frac{Q_{max}}{D_{max}} \end{aligned} \tag{32}$$

From the above derivation, the locations in search trajectory are influenced by data rate and maximum limit of memory size for I_UAV s. The upper and lower limit of normally distributed data is given as $D_{max} = \mu + \sigma$ and $D_{min} = \mu - \sigma$ respectively. Thus Equation (30) can be written as

$$\tau_{search}(|\psi|) \leq \frac{1}{3} \frac{\sigma \cdot Q_{max} \cdot \rho}{v_{max}(\mu^2 - \sigma^2)} \tag{33}$$

□

To calculate the upper bound for Equation (33), ρ is the distance between consecutive PoIs which could be calculated from the pre-calculated trajectory of I_UAV s based on shortest path.

6. Energy Aware Data Offloading

The model presented in **P1** in Section 4 is a stochastic optimization problem as the arrival of data in the system queue is stochastic. Using the online Lyapunov optimization algorithm, we solve the stochastic optimization in **P1** and jointly stabilize all queues by finding the optimal policy to access each I_UAV in each time-slot. The quadratic Lyapunov function, as given in Equation (36) associates a scalar measure to the queues of the system. Further, the stability of the system is maintained by a guaranteed mean rate stability of the evolving queues as given in Equations (34) and (35).

$$\lim_{T \rightarrow \infty} \frac{\mathbb{E}[Q_i(t)]}{T} = 0, \forall i \tag{34}$$

$$\lim_{T \rightarrow \infty} \frac{\mathbb{E}[L(t)]}{T} = 0 \tag{35}$$

$$Z(v(t)) = \frac{1}{2} \left[\sum_{i=1}^N Q_i(t)^2 + L(t)^2 \right] \tag{36}$$

where $v(t) = [\{Q_i(t)\}_{i=1}^N, L(t)]$ consists of all system queues at a time t and $Z(\cdot)$ is quadratic Lyapunov function of system queues. The **Lyapunov drift** corresponding to the above function is given as:

$$\Delta Z(v(t)) = \mathbb{E}[(z(v(t+1)) - z(v(t)))] \tag{37}$$

The Lyapunov drift plus a penalty function is minimized to stabilize the queue backlog of the system is given as:

$$\Delta DP(t) = \Delta Z(v(t)) + V \cdot \mathbb{E}[E_{sys}(t)] \tag{38}$$

where V is a positive constant that controls the trade-off between Lyapunov drift and the expected system energy. A high value of parameter V signifies more weight on minimizing the energy of the system at the cost of high queue backlog. Therefore, V acts as a trade-off parameter between system energy and queue backlog. An upper bound on $\Delta Z(v(t))$ can be derived as follows, (for details see [10])

$$\Delta Z(v(t)) \leq \mathbb{E}\left[-\sum_{i=1}^N Q_i(t) \cdot d_i^{off}(t) + \mathbb{E}[L(t) \cdot (-d_{access}^{off}(t))]\right] + C \tag{39}$$

where C is a deterministic constant.

As a result, the upper bound of the drift plus penalty function becomes

$$\Delta DP(t) \leq C - \mathbb{E}\left[\sum_{i=1}^N Q_i(t) \cdot d_i^{off}(t)\right] - \mathbb{E}[L(t) \cdot d_{access}^{off}(t)] + V \cdot \mathbb{E}[E_{system}(t)|v(t)] \tag{40}$$

Hence, the original formulation **P1** can be reduced to **P3** which bounds the system's drift to keep it stable.

$$\mathbf{P3} \quad \min_{p_i(t), P_{access}(t)} \mathbb{E}\left[-\sum_{i=1}^N Q_i(t) d_i^{off}(t)\right] - \mathbb{E}[L(t) d_{access}^{off}(t)] + V \cdot \mathbb{E}[E_{sys}(t)] \tag{41}$$

s.t.

$$p_i(t) \leq p^{max}, \quad \forall i, \forall t \tag{42}$$

$$d_i^{off}(t) \leq Q_i(t), \quad \forall i, \forall t \tag{43}$$

$$d_i^{off}(t) \leq W \tau_{comm} \log_2\left(1 + \frac{\zeta p^{max}(t)}{N_o W}\right), \forall i, \forall t \tag{44}$$

$$P_{access}(t) \leq P^{max}, \quad \forall t \tag{45}$$

$$d_{access}^{off}(t) \leq W \tau_{comm} \log_2\left(1 + \frac{\zeta P^{max}(t)}{N_o W}\right), \forall t \tag{46}$$

As can be observed, the constraints in **P3** is a subset of the constraints in **P1**. To further simplify the solution of the optimization formulation, we reformulate **P3** as two separate sub-problems provided the positions of A_UAV and I_UAV are fixed in a given time slot t . The Lyapunov based online optimization is optimal for a stochastic system to derive the overall optimal solution [33].

6.1. Optimization of Transmission Energies of I_UAVs

The first sub-problem deals with the optimization of parameters related to the I_UAV . The variables $S_{access}(t)$, i.e., the position of A_UAV and the offloaded data $d_i^{off}(t)$ of the selected i^{th} I_UAV are coupled in a particular time-slot. The fixed position of A_UAV decouples these variables. In the optimization model **P 3.1**, the transmission energy is optimized for a single time-slot (t) given the position of A_UAV :

$$\mathbf{P 3.1} \quad \min_{p_i(t)} -\sum_{i=1}^N Q_i(t) \cdot d_i^{off}(t) + V \cdot \tau_{comm} \cdot \sum_{i=1}^N p_i(t) \tag{47}$$

s.t.

$$p_i(t) \leq p^{max}, \quad \forall i \tag{48}$$

$$d_i^{off}(t) \leq Q_i(t), \quad \forall i \tag{49}$$

$$d_i^{off}(t) \leq W\tau_{comm} \log_2(1 + \frac{\zeta p^{max}(t)}{N_0W}), \forall i \tag{50}$$

The objective function in **P 3.1** is a convex function. The first & second constraints are linear and the third constraint is upper bounded by a concave function. As a result, the stationary point of the objective function is found to be: $p_i^*(t) = \min\{\max\{(\frac{Q_i(t)W}{V} - \frac{N_0W}{\zeta}), 0\}, p^{max}\}$.

6.2. Optimization of Transmission Energy of A_UAV

The second sub-problem deals with the optimization of the A_UAV parameters for the amount of data offloaded to the cloud at time t . The updated optimization model is given as:

$$\mathbf{P\ 3.2} \min_{P_{access}(t)} -L(t) \cdot d_{access}^{off}(t) + V \cdot \tau_{comm} \cdot P_{access}(t) \tag{51}$$

s.t.

$$P_{access}(t) \leq P^{max} \tag{52}$$

$$d_{access}^{off}(t) \leq L(t) \tag{53}$$

$$d_{access}^{off}(t) \leq W\tau_{comm} \log_2(1 + \frac{\zeta P^{max}(t)}{N_0W}) \tag{54}$$

The model **P 3.2** has a convex optimization objective subject to convex constraints to solve for the optimal transmission power of the A_UAV. The stationary point of the optimization model is $P_{access}(t) = \min\{\max\{(\frac{L(t)W}{V} - \frac{N_0W}{\zeta}), 0\}, P^{max}\}$.

Thus, the derived stationary points of the optimization model using the Lyapunov optimization framework are calculated in every time-step to optimize the A_UAV trajectory and data-offloading tasks. The overall proposed solution approach is presented in Algorithm 2. Next, we discuss the experimentation setup for evaluating the proposed solution.

Algorithm 2 Proposed Solution Approach

Initialize: Trajectories of all I_UAV_i and list of PoIs l_i .

Time: $t = 0$

while $t \leq T$ **do**

1. Compute and offload $d_{access}^{off}(t)$ as using P 3.2

2. Update $L(t)$

3. Using Algorithm 1 estimate the $\{Q_i(t)\}_{i=1}^N$ and $\{S_i(t)\}_{i=1}^N$

4. Select the i^{th} I_UAV to collect data using P2

5. Compute $d_i^{off}(t)$ for i^{th} I_UAV using P 3.1 to offload data to A_UAV

6. Update $Q_i(t)$

7. $t = t + 1$

7. Experimentation

In this section, we present the simulation setup to validate the efficacy of our proposed Distance and Latency Aware Trajectory Optimization with Lyapunov based energy-aware data offloading followed by results and discussions. The simulation parameters are listed in Table 2.

Table 2. List of Simulation Parameters.

Parameters	Values
Channel Bandwidth	1 MHz
κ	1
Noise Power for I_UAV	10^{-13}
Noise Power for A_UAV	10^{-20}
The path-loss constant g_0	10^{-4}
The path loss exponent θ	2 to 4
Memory capacity of I_UAV (Mem_{max})	10^5 bits

We have considered a 600×600 square meter area with PoIs spread along the region in disjoint clusters and at heights ranging from 70 to 80 m above the ground. All experiments are conducted for at least 30 times and the average of results are plotted. We sample 150 PoI locations uniformly randomly in three disjoint clusters. From a practical point of view of a multi-UAV system, we consider a system of three I_UAV s with one A_UAV in all the simulation experiments. Each I_UAV is assigned to a cluster where I_UAV s randomly choose a starting location within the cluster. The sequence of PoIs to be visited by each I_UAV is generated using the shortest path algorithm. Before proceeding to the next PoI, an I_UAV collects all the data ($A_i(t)$) from that PoI. In the data collection process, an I_UAV may sojourn at the same PoI across multiple time-slots until all the data ($A_i(t)$) of PoI is collected. For each PoI, the amount of data to be collected is modeled as a Gaussian distribution with a mean of 150 Kb and variance of 50 Kb. The A_UAV gets partial information about the data generated at each location so it could not accurately estimate the location of I_UAV in the next time slot; as a result, it has to search for candidate locations to access the selected I_UAV s as discussed in Section 5.1. The trade-off parameter V ranges from 10 to 10^{10} . The length of each time slot (τ) is 25 s divided into different sub slots as shown in Figure 2. The selection of I_UAV is assumed to take negligible time whereas transition may take up to 20 s. The search and transmit function takes total of 5 s. The maximum transmission power for A_UAV and I_UAV are 5 W and 2 W, respectively [25]. The other simulation parameters are listed in Table 2.

In order to validate the performance of our proposed approach, we compared our proposed approach with a set of baseline approaches on two broad categories of optimization parameters viz. Trajectory planning and Data offloading. We consider the following baseline approaches:

- Distance Aware Trajectory planning (DAT): In this approach, the A_UAV selects to access an I_UAV based on the shortest distance from the current location in each time slot.
- Round Robin based Trajectory planning (RR): In this approach, the A_UAV accesses I_UAV s in sequential order in each time slot.
- Maximum Transmission Power (MAX) data offloading: In each time slot, the A_UAV and the I_UAV operate at the maximum transmission power to offload data.

The proposed approaches are as follows:

- Distance and Latency Aware Trajectory Optimization (DLAT): In this approach, the A_UAV selects the I_UAV based on the minimum distance, with maximum bits to offload and access latency constraint as given in the trajectory optimization problem.
- Hybrid Approach for Trajectory Scheduling (HDLAT): In this approach, the A_UAV selects the I_UAV based on the minimum distance and access latency constraint as given in the trajectory optimization problem up to a certain threshold of battery, i.e., 75% of the total battery. Beyond the threshold, the scheduling algorithm switches to the DAT strategy (proposed approach).
- Lyapunov Optimization for data offloading: In each time slot, the A_UAV and the I_UAV calculate the optimal value of transmission energy using the Lyapunov Optimization.

Experiments were conducted by taking a combination of one of the approaches from both the categories: (1) DAT + MAX, (2) DLAT + MAX, (3) RR + MAX, (4) HDLAT + MAX, (5) DAT + Lyapunov, (6) RR + Lyapunov (7) DLAT + Lyapunov (proposed approach) and (8) HDLAT + Lyapunov.

8. Results and Discussion

In this section, we discuss the comparative performances of our proposed approaches with baseline approaches.

8.1. Transmission Power and Average Buffer Size

Figure 3a depicts the effect of value of parameter V with respect to the transmission power of the A_UAV . It is evident from Figure 3a that all combinations with max power consumption for data offloading always consume the maximum energy, making the average transmission power consumption the same across different values of V . For baseline and proposed approach with Lyapunov based data offloading, a drop in the energy consumption can be observed for $\log(V)$ values beyond 7, because large V forces the system to consume less energy, consequently less data is transmitted to A_UAV .

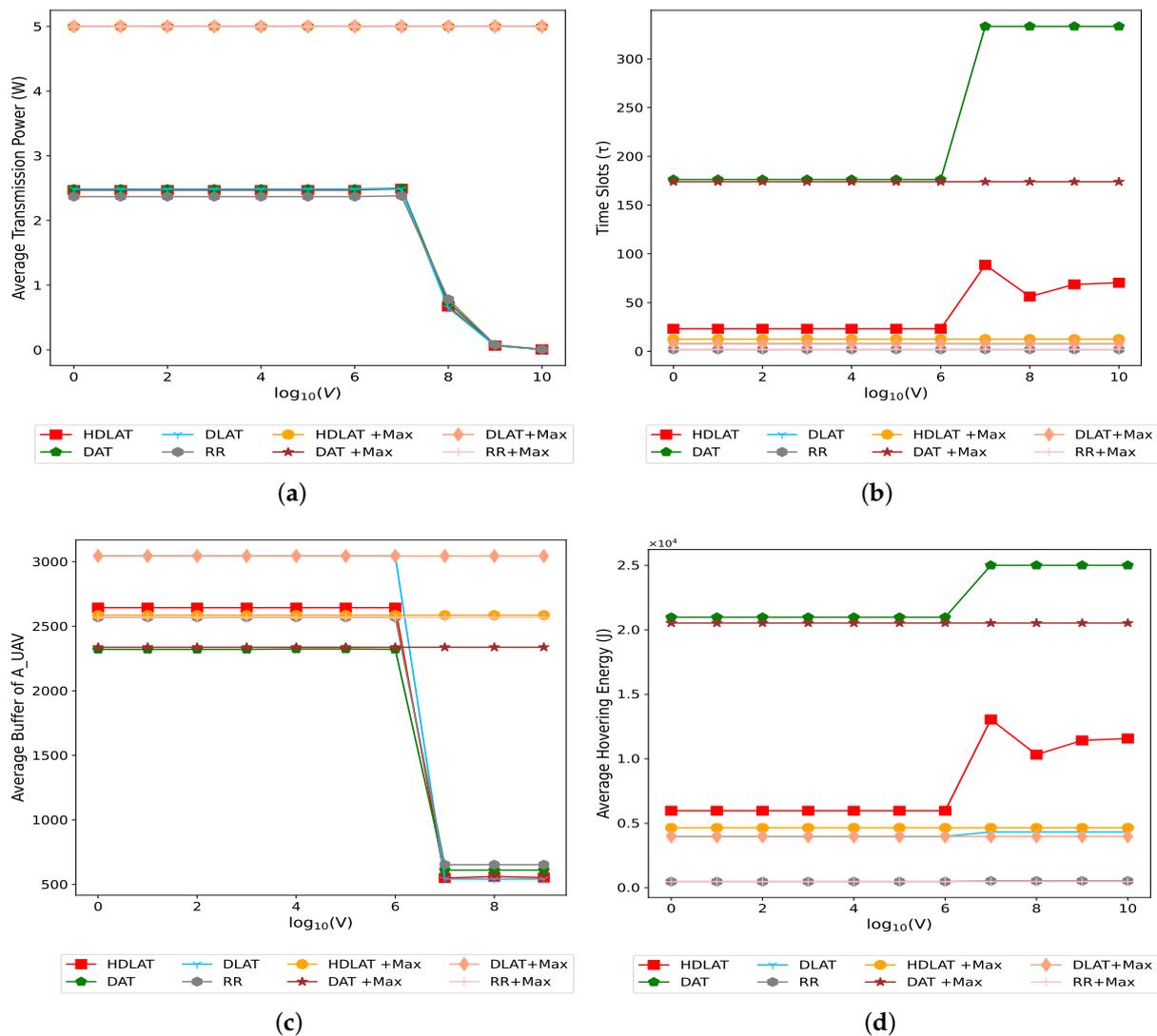


Figure 3. Cont.

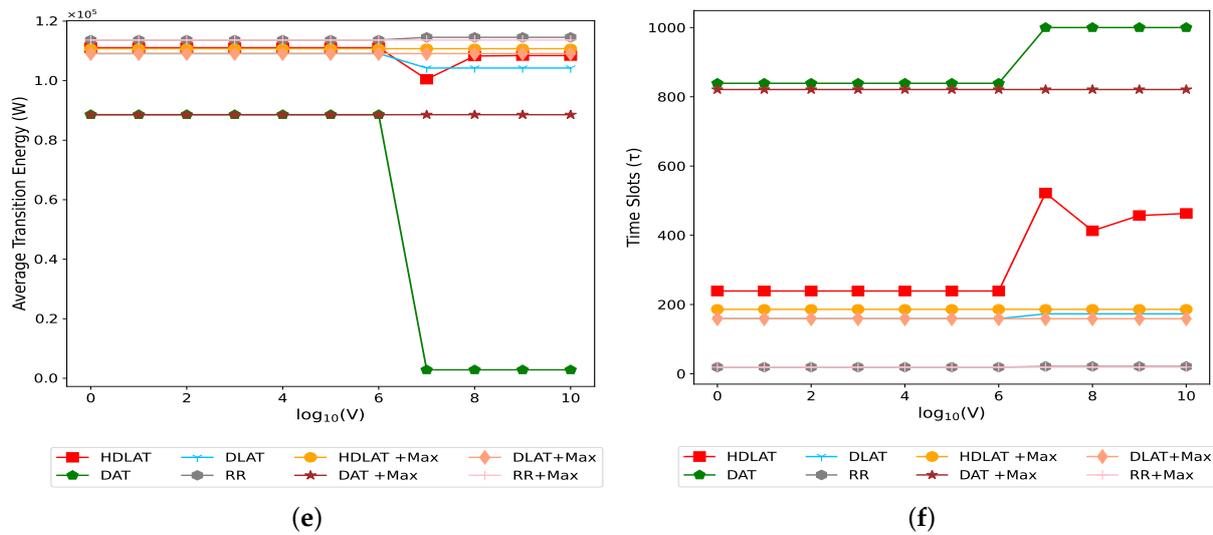


Figure 3. Various Results of *A_UAV*. (a) Analysis of Average Transmission Power of *A_UAV*. (b) Analysis of Average Access Latency of *A_UAV*. (c) Analysis of Average Buffer of *A_UAV*. (d) Analysis of Hovering Energy of *A_UAV*. (e) Analysis of Transition Energy of *A_UAV*. (f) Analysis of Total Flight Time of *A_UAV*.

It can be observed in Figure 4a that the average buffer length of *I_UAVs* increases in line with the rise of V after hitting the inflection point. The value of V between 6 and 7 could maintain the queue buffer and consume less transmission energy for *I_UAVs*. Similarly, *A_UAV* has fewer data transmitted from the *I_UAVs* for higher values of V, which would decrease the total data collection or transmission further by *A_UAV* as shown in Figure 3c.

8.2. Hovering Energy of *A_UAV*

The plot in Figure 3d depicts the impact of trade off parameter V on the hovering energy of *A_UAV*. The total hovering energy starts increasing after the inflection point because *A_UAV* takes more time slots to collect the same amount of data from *I_UAVs*. As a result, the energy consumption of *A_UAV* significantly increases for DAT and HDLAT, whereas it remains constant for DLAT (total flying time is less). Similarly, Figure 3e shows the evolution of total transition energy with V. It is interesting to observe that for DAT baseline *A_UAV* stays in the field for a longer time. As a result, transition energy is higher compared to DLAT and HDLAT. However, the *A_UAV* transition energy starts decreasing after the inflexion point. Similarly, DAT and HDLAT consume less battery in every time-slot as both save on transition energy by selecting the nearest *I_UAV*. This allows *A_UAV* to stay longer in the field, which is illustrated in Figure 3f. The baseline DAT remains for a more extended time, whereas RR stays for the least number of time slots before running out of battery. The proposed approaches of DLAT and HDLAT lie between the extreme baselines for the different analyses conducted. This shows that the proposed approach has an optimised trade-off between energy saving along with the end-to-end data offloading from multiple *I_UAVs*.

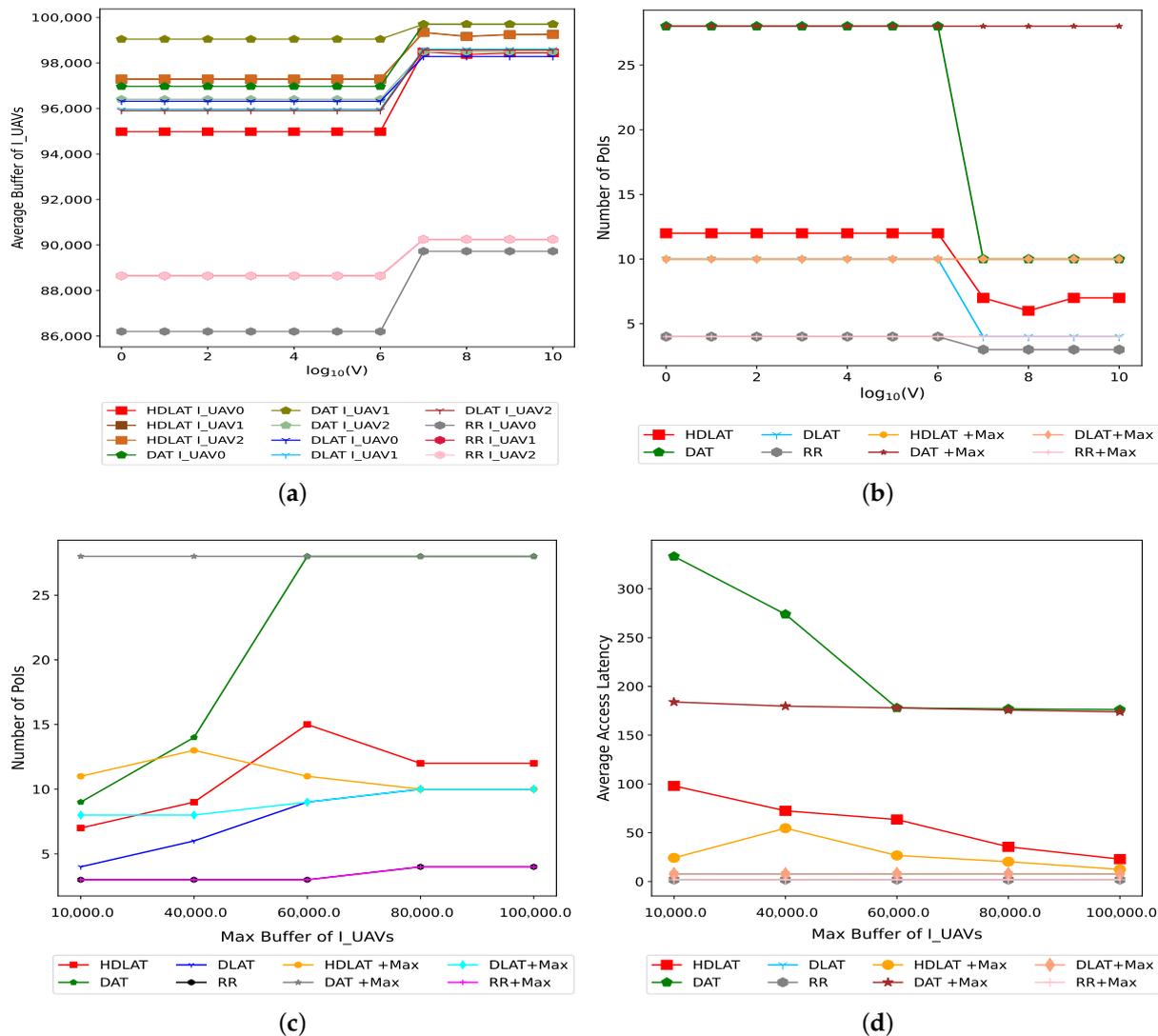


Figure 4. Various Results of the System. (a) Analysis of Average Buffer of I_{UAV} . (b) Analysis of Total Number of Pools Covered. (c) Analysis of Total Number of Pools Covered and Max Buffer of I_{UAV} s. (d) Analysis of Average Access Latency and Max Buffer of I_{UAV} s.

8.3. I_{UAV} s Access Latency

Figure 3b shows the effect of trade off parameter V on the average access latency of the system. The trajectory of the A_{UAV} affects the access sequence and waiting times of the I_{UAV} s to offload their data to A_{UAV} . This can be observed in the proposed DLAT + Lyapunov, which has an upper bound on access latency throughout the system. Similarly, RR based baseline approach has an access latency of 2 time-slots whereas for both DAT and DAT + MAX baseline approaches, the average access latency is higher. The average access latency for DAT baseline approach becomes worse with increasing V . The same remains stable for DLAT in both scenarios. HDLAT, as per expectation, remains between the DLAT and DAT approaches. It could be related to the fact that an increase of V causes the A_{UAV} to spend more time slots to collect the data from I_{UAV} s. An increase in flying time of A_{UAV} is influenced by a decrease in transition and transmission power, which increases the average access latency of DAT and HDLAT approaches. In contrast, it remains constant for DLAT and RR because of latency constraints. Our proposed approaches lie within the extreme baselines and maintain the average access latency by saving on transmission energy and transition energy in HDLAT by switching from DLAT to DAT after 75 percent of the battery is consumed. The Average energy consumption of A_{UAV}

includes transmission, transition, and hovering energy consumption. Similarly, the tradeoff between the average access latency and the average energy consumption can be observed as the average access latency of the system reduces, the average energy consumption increases. By the definition and from Figure 3a,b,d,e, this can be observed that the RR baseline has the least average access latency as well as the highest energy consumption whereas, DAT has the highest average access latency and the least energy consumption. From Figure 3b, this can be observed that for HDLAT the average access latency of A_UAV is reduced by approximately 70% as compared to the greedy approach (DAT) and remains constant for DLAT. The RR baseline has the least average access latency, but the gap between HDLAT and RR is much lesser as compared to the gap between HDLAT and DAT.

8.4. Coverage of PoIs

The coverage of PoIs by the A_UAV can be defined as the number of PoI locations whose data has been offloaded to A_UAV by the I_UAV s. Figure 4b shows the effect of trade-off parameter V value on the number of PoIs covered in the system. It can be observed that A_UAV can serve more PoIs for both DLAT and HDLAT approaches than the RR baseline approach. The DAT-baseline approach serves relatively more PoIs than DLAT and HDLAT by saving on transition and transmission energy, but not maintaining low access latencies of I_UAV s. In Figure 4c, the effect of increasing the buffer size of I_UAV s on the PoIs is shown. It can be observed, the number of PoIs served increases with an increase in buffer size. In our proposed approach, the performance of DLAT and HDLAT is a tradeoff between two extreme baselines. DAT baseline approach covers more locations but at the cost of access latency, as shown in Figure 4d. In a similar manner to average access latency, the tradeoff between the PoIs coverage and the average energy consumption is also evident from Figures 3e and 4b. The approach with higher average energy consumption also has a reduced PoIs coverage. RR has the highest energy consumption and covers the least number of PoIs, whereas DAT is having the least energy consumption but covers the maximum number of PoIs. In the optimal processing zone of $\log(V)$ between 6 and 7, HDLAT covers more than double the number of PoIs as compared to RR while consuming less energy.

9. Conclusions

In this paper, we have introduced an online scheduling approach to collect data in resource-constrained and infrastructure-deficient environments. The paper presents a heterogeneous multi-UAV based framework for end-to-end data collection and offloading using a distance and latency-aware trajectory optimization. The Lyapunov optimization approach is used to ensure the system's stability in terms of expected system queue backlogs by breaking the system optimization problem into manageable sub-problems. The simulation results show that the access latency of our proposed (DLAT & HDLAT) approaches perform better than other baseline approaches. Moreover, the system parameter V analysis has shown a trade-off between the queue stability and the system utility. The paper also presents a detailed analysis of different forms of energy consumption of A_UAV . In the future, we plan to optimize the trajectories of I_UAV s jointly with that of the A_UAV . The energy optimization of I_UAV s could be analyzed while designing the optimal trajectory of both the I_UAV s and A_UAV s. Further, the effect of introducing more than one A_UAV could be studied in the overall optimization approach. Along with this, the use of multi-agent reinforcement learning could be explored to solve the problem by introducing the coordination among the A_UAV s and I_UAV s.

Author Contributions: Conceptualization, A.K. and S.S.J.; methodology, A.K. and S.S.J.; software, A.K.; validation, A.K. and S.S.J.; investigation, S.S.J. and J.J.; writing original draft preparation, A.K.; writing review and editing, S.S.J., J.J. and H.G.; supervision, S.S.J. and J.J.; All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not Applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mozaffari, M.; Saad, W.; Bennis, M.; Nam, Y.H.; Debbah, M. A tutorial on UAVs for wireless networks: Applications, challenges, and open problems. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 2334–2360. [[CrossRef](#)]
2. Ruggiero, F.; Lippiello, V.; Ollero, A. Aerial manipulation: A literature review. *IEEE Robot. Autom. Lett.* **2018**, *3*, 1957–1964. [[CrossRef](#)]
3. Malik, A.W.; Rahman, A.U.; Qayyum, T.; Ravana, S.D. Leveraging fog computing for sustainable smart farming using distributed simulation. *IEEE Internet Things J.* **2020**, *7*, 3300–3309. [[CrossRef](#)]
4. Alotaibi, E.T.; Alqefari, S.S.; Koubaa, A. LSAR: Multi-UAV collaboration for search and rescue missions. *IEEE Access* **2019**, *7*, 55817–55832. [[CrossRef](#)]
5. Hamledari, H.; Davari, S.; Sajedi, S.O.; Zangeneh, P.; McCabe, B.; Fischer, M. UAV Mission Planning Using Swarm Intelligence and 4D BIMs in Support of Vision-Based Construction Progress Monitoring and As-Built Modeling. In Proceedings of the Construction Research Congress, New Orleans, LA, USA, 2–4 April 2018; pp. 43–53.
6. Fu, Y.; Mei, H.; Wang, K.; Yang, K. Joint optimization of 3D trajectory and scheduling for solar-powered UAV systems. *IEEE Trans. Veh. Technol.* **2021**, *70*, 3972–3977. [[CrossRef](#)]
7. Yu, Z.; Gong, Y.; Gong, S.; Guo, Y. Joint task offloading and resource allocation in UAV-enabled mobile edge computing. *IEEE Internet Things J.* **2020**, *7*, 3147–3159. [[CrossRef](#)]
8. Asim, M.; Mashwani, W.K.; Shah, H.; Belhaouari, S.B. An evolutionary trajectory algorithm for multi-UAV-assisted MEC system. *Soft Comput.* **2022**, *26*, 7479–7492. [[CrossRef](#)]
9. Asim, M.; Mashwani, W.K.; Belhaouari, S.B.; Hassan, S. A novel genetic trajectory planning algorithm with variable population size for multi-UAV-assisted mobile edge computing system. *IEEE Access* **2021**, *9*, 125569–125579. [[CrossRef](#)]
10. Neely, M. Stochastic network optimization with application to communication and queueing systems. *Synth. Lect. Commun. Netw.* **2010**, *3*, 1–211. [[CrossRef](#)]
11. Zeng, Y.; Zhang, R. Energy-efficient UAV communication with trajectory optimization. *IEEE Trans. Wirel. Commun.* **2017**, *16*, 3747–3760. [[CrossRef](#)]
12. Wang, Y.; Chen, M.; Pan, C.; Wang, K.; Pan, Y. Joint Optimization of UAV Trajectory and Sensor Uploading Powers for UAV-assisted Data Collection in Wireless Sensor Networks. *IEEE Internet Things J.* **2021**, *9*, 11214–11226. [[CrossRef](#)]
13. Asadi Boroujeni, K.; Han, K. Perspective-based image-to-bim alignment for automated visual data collection and construction performance monitoring. In *Computing in Civil Engineering*; American Society of Civil Engineers: Reston, VA, USA, 2017; pp. 171–178.
14. Mach, P.; Becvar, Z. Mobile edge computing: A survey on architecture and computation offloading. *IEEE Commun. Surv. Tutor.* **2017**, *19*, 1628–1656. [[CrossRef](#)]
15. Dai, B.; Niu, J.; Ren, T.; Hu, Z.; Atiquzzaman, M. Towards Energy-Efficient Scheduling of UAV and Base Station Hybrid Enabled Mobile Edge Computing. *IEEE Trans. Veh. Technol.* **2022**, *71*, 915–930. [[CrossRef](#)]
16. Jeong, S.; Simeone, O.; Kang, J. Mobile Edge Computing via a UAV-Mounted Cloudlet: Optimization of Bit Allocation and Path Planning. *IEEE Trans. Veh. Technol.* **2018**, *67*, 2049–2063. [[CrossRef](#)]
17. Xu, Y.; Zhang, T.; Loo, J.; Yang, D.; Xiao, L. Completion Time Minimization for UAV-Assisted Mobile-Edge Computing Systems. *IEEE Trans. Veh. Technol.* **2021**, *70*, 12253–12259. [[CrossRef](#)]
18. Wu, Q.; Chen, J.; Xu, Y.; Qi, N.; Fang, T.; Sun, Y.; Jia, L. Joint Computation Offloading, Role and Location Selection in Hierarchical Multi-coalition UAV MEC Networks: A Stackelberg Game Learning Approach. *IEEE Internet Things J.* **2022**, *9*, 18293–18304. [[CrossRef](#)]
19. Hu, X.; Wong, K.K.; Yang, K.; Zheng, Z. UAV-assisted relaying and edge computing: Scheduling and trajectory optimization. *IEEE Trans. Wirel. Commun.* **2019**, *18*, 4738–4752. [[CrossRef](#)]
20. Diao, X.; Wang, M.; Zheng, J.; Cai, Y. Fairness-Aware Offloading and Trajectory Optimization for Multi-UAV Enabled Multi-Access Edge Computing. *IEEE Access* **2020**, *8*, 124359–124370. [[CrossRef](#)]
21. Hu, Q.; Cai, Y.; Yu, G.; Qin, Z.; Zhao, M.; Li, G.Y. Joint offloading and Trajectory design for UAV-enabled mobile edge computing systems. *IEEE Internet Things J.* **2018**, *6*, 1879–1892. [[CrossRef](#)]
22. Zhang, J.; Zhou, L.; Tang, Q.; Ngai, E.C.H.; Hu, X.; Zhao, H.; Wei, J. Stochastic computation offloading and Trajectory scheduling for UAV-assisted mobile edge computing. *IEEE Internet Things J.* **2018**, *6*, 3688–3699. [[CrossRef](#)]
23. Sun, C.; Ni, W.; Wang, X. Joint Computation Offloading and Trajectory Planning for UAV-Assisted Edge Computing. *IEEE Trans. Wirel. Commun.* **2021**, *20*, 5343–5358. [[CrossRef](#)]
24. Zhan, C.; Hu, H.; Sui, X.; Liu, Z.; Niyato, D. Completion time and energy optimization in the UAV-enabled mobile-edge computing system. *IEEE Internet Things J.* **2020**, *7*, 7808–7822. [[CrossRef](#)]
25. Wan, S.; Lu, J.; Fan, P.; Letaief, K.B. Towards Big data processing in IoT: Path Planning and Resource Management of UAV Base Stations in Mobile-Edge Computing System. *IEEE Internet Things J.* **2019**, *7*, 5995–6009. [[CrossRef](#)]

26. Zhang, S.; Zhang, H.; Di, B.; Song, L. Cellular cooperative unmanned aerial vehicle networks with sense-and-send protocol. *IEEE Internet Things J.* **2018**, *6*, 1754–1767. [[CrossRef](#)]
27. Hu, J.; Zhang, H.; Song, L. Reinforcement learning for decentralized trajectory design in cellular UAV networks with sense-and-send protocol. *IEEE Internet Things J.* **2018**, *6*, 6177–6189. [[CrossRef](#)]
28. Kurt, S.; Tavli, B. Path-Loss Modeling for Wireless Sensor Networks: A review of models and comparative evaluations. *IEEE Antennas Propag. Mag.* **2017**, *59*, 18–37. [[CrossRef](#)]
29. Al-Hourani, A.; Kandeepan, S.; Lardner, S. Optimal LAP altitude for maximum coverage. *IEEE Wirel. Commun. Lett.* **2014**, *3*, 569–572. [[CrossRef](#)]
30. Motlagh, N.H.; Taleb, T.; Arouk, O. Low-altitude unmanned aerial vehicles-based internet of things services: Comprehensive survey and future perspectives. *IEEE Internet Things J.* **2016**, *3*, 899–922. [[CrossRef](#)]
31. Di Franco, C.; Buttazzo, G. Energy-aware coverage path planning of UAVs. In Proceedings of the 2015 IEEE International Conference on Autonomous Robot Systems and Competitions, Vila Real, Portugal, 8–10 April 2015; pp. 111–117.
32. Lu, J.; Wan, S.; Chen, X.; Chen, Z.; Fan, P.; Letaief, K.B. Beyond empirical models: Pattern formation driven placement of UAV base stations. *IEEE Trans. Wirel. Commun.* **2018**, *17*, 3641–3655. [[CrossRef](#)]
33. Mao, Y.; Zhang, J.; Letaief, K.B. A Lyapunov optimization approach for green cellular networks with hybrid energy supplies. *IEEE J. Sel. Areas Commun.* **2015**, *33*, 2463–2477. [[CrossRef](#)]