electronics

MDPI

Article

Cache-Enabled Data Rate Maximization for Solar-Powered UAV Communication Systems

Pham Duy Thanh [1], Tran Nhut Khai Hoan [2], Hoang Thi Huong Giang [1] and Insoo Koo [1,*]

[1] School of Electrical Engineering, University of Ulsan, Ulsan 680-749, Korea; duythanhtdtk12@gmail.com (P.D.T.); huonggiangtdt@gmail.com (H.T.H.G.)

[2] Department of Electronics and Telecommunication Engineering, College of Engineering Technology, Can Tho University, Can Tho 900000, Vietnam; tnkhoan@ctu.edu.vn

* Correspondence: iskoo@ulsan.ac.kr; Tel.: +82-52-259-1249

Received: 26 September 2020; Accepted: 17 November 2020; Published: 20 November 2020

Abstract: Currently, deploying fixed terrestrial infrastructures is not cost-effective in temporary circumstances, such as natural disasters, hotspots, and so on. Thus, we consider a system of caching-based UAV-assisted communications between multiple ground users (GUs) and a local station (LS). Specifically, a UAV is exploited to cache data from the LS and then serve GUs' requests to handle the issue of unavailable or damaged links from the LS to the GUs. The UAV can harvest solar energy for its operation. We investigate joint cache scheduling and power allocation schemes by using the non-orthogonal multiple access (NOMA) technique to maximize the long-term downlink rate. Two scenarios for the network are taken into account. In the first, the harvested energy distribution of the GUs is assumed to be known, and we propose a partially observable Markov decision process framework such that the UAV can allocate optimal transmission power for each GU based on proper content caching over each flight period. In the second scenario where the UAV does not know the environment's dynamics in advance, an actor-critic-based scheme is proposed to achieve a solution by learning with a dynamic environment. Afterwards, the simulation results verify the effectiveness of the proposed methods, compared to baseline approaches.

Keywords: unmanned aerial vehicle; content caching; non-orthogonal multiple access; power allocation; reinforcement learning; energy harvesting

1. Introduction

Lately, wireless communication has been evolving not only for high throughput, but also for ultra-reliability, efficient energy consumption, and to support highly diversified applications with heterogeneous requirements for quality of service (QoS) [1]. To this end, extensive research efforts have mainly been devoted to fixed terrestrial infrastructures such as ground base stations (BSs), access points, and relays, which generally restrict their capability to cost-effectively meet the ever-increasing multifarious traffic demand. In order to address this problem, there is a great deal of growing interest in providing wireless connectivity from the sky under various airborne platforms, such as unmanned aerial vehicles (UAVs) [2], balloons [3], and helikites [4]. Currently, UAV's classification has a broad diversity. It was classified according to the basis of weight, altitude, and range, wings and rotors, and their application [5]. By leveraging low-altitude UAVs (i.e., less than about one kilometer above the ground [5]), the wireless communication system can provide swift deployment and high flexibility in mobility [2]. UAVs can be used as a flying base station to handle short-term, erratic traffic demand in hotspots, such as a concerts and sports events, or through data offloading for congestion mitigation [6,7]. In other words, the UAV can provide additional aid as either a stand-alone aerial BS [4,8], or it can serve as a part of a heterogeneous network in a multi-tier

airborne cellular network [6]. It is conceivable that thanks to UAV-enabled wireless communication, QoS can be enhanced in terms of better channel conditions, guaranteeing line-of-sight (LOS) links, faster deployment, and better maneuverability. Moreover, it can be applied in many practical scenarios, such as public safety communications [9], Internet of Things (IoT) communications [10], and massive machine-type communications [11]. For natural disaster scenarios, the authors in [12] investigated the UAV-enabled method to search for survivors after disasters, while the deployment tool for the emergency networks was proposed by using the UAVs in a realistic disaster scenario in [13]. With the above apparent advantages, it is foreseen that UAV-enabled communications will promisingly play a more important role in future wireless communication systems.

1.1. Related Works

In recent years, the reputation of non-orthogonal multiple access (NOMA) has risen intensively as a promising solution to critical issues in next-generation wireless systems [14]. By allowing multiple devices to operate with the same frequency, time, or code resources, the NOMA technique has exhibited improved spectral efficiency and balanced and fair access, compared to orthogonal multiple access (OMA) approaches [15,16]. It should be noted that the NOMA method is typically based on superposition coding (SC) at the transmitters and successive interference cancellation (SIC) at the receivers. Many research efforts have paid attention to combinations of NOMA and UAV-enabled wireless communications technologies [17–20]. In [17], the authors proposed multiple access mode selection (NOMA/OMA) based on conditions for a better outage probability in a UAV-enabled downlink wireless network. Sohail et al. [18] investigated a power allocation approach to maximize the sum rate of the network by reducing the energy expense of the UAV, and numerical results were obtained by simulating various deployment environments, such as rural, urban, and dense-urban. In [19], an efficient joint placement and power allocation method for a single UAV providing data services as a base station for multiple ground users was investigated to improve the total data rate of the UAV-assisted NOMA network. Besides, the authors in [20] introduced a comprehensive framework for UAV base station cooperation in UAV-assisted NOMA networks, where the UAV and BS cooperate with each other to serve ground users simultaneously. As a result, by jointly relying on the UAV trajectory and NOMA precoding optimization, the proposed scheme demonstrated large throughput in the system. In [21], the authors proposed the UAV-aided scheme to guarantee secure transmission for the ground receivers. In the following, we discuss solutions among research efforts dealing with issues of UAV communication systems.

1.1.1. UAV Communications Using Data Caching

During the past few decades, the driving forces behind traffic development have shifted from connection-centric communication demand (e.g., text messages and smart phones) to content-centric communication demand (e.g., popular music or video streaming). Although small base stations are densely employed to accommodate the ever-increasing traffic demand, a heavy traffic burden is still imposed on the backhaul links. One potential solution is to properly cache popular content at the network edge (i.e., UAVs, D2D devices, or relays) to serve the same requests of users without duplicate transmissions via the backhaul links. More explicitly, in UAV-assisted edge caching, UAVs generally cache the content via a limited wireless backhaul link and then distribute it to ground users. In this regard, many contributions to caching in UAV-assisted communication systems have been made [22–26]. In [22], UAVs were dispatched to store enhancement layer segments of video beforehand and then provided the transmissions to users who requested the videos. Chen et al. [23] proposed appropriate content caching during off-peak times in a cloud radio access network, which is based on the user's behavior prediction. In [24], an effective liquid state machine learning approach was investigated to predict the content-request distribution of users in Long-Term Evolution-Unlicensed (LTE-U) UAV transmissions systems while having only limited information on the network states. The proposed algorithm also enables UAVs to optimally allocate the bandwidth over licensed and

unlicensed bands and to satisfy the queue stability requirements of each user. In addition, local content caching at the ground users (GUs) was also studied [25,26]. In particular, the authors in [25] proposed a solution to deal with the endurance issue at the UAV in which the ground users cooperatively cache files delivered by the UAV. Then, the files are retrieved either from its local cache or neighbor cache. Meanwhile, a joint UAV-trajectory and time-scheduling approach was proposed in [26] to maximize the secrecy rate in UAV-relaying systems.

1.1.2. UAV Communications Using Energy Harvesting

From the standpoint of wireless communication, UAV-enabled communication system operations are quite energy consuming owing to the support of the UAV's propulsion in the air, the communications with users, and application-based purposes. Therefore, UAVs usually have very limited endurance due to energy constraints. To address this issue, several methods have been introduced to alleviate UAVs energy consumption by, for example, reducing the UAV's weight [27] and planning energy-efficient UAV flight paths [28,29]. The authors in [28] investigated a path planning algorithm that minimizes energy consumption while satisfying coverage and resolution. Meanwhile, an efficient approach was proposed to maximize the UAV's energy efficiency under the constraints on the trajectory [29]. However, the energy supply for the UAVs is still basically unsustainable due to the limited battery capacity. Thus, the fundamental UAV endurance problem remains unresolved. Currently, energy harvesting of radio frequency (RF) signals has attracted increasing interest in UAV communications for the purpose of prolonging network lifetime or maximizing system throughput [30–32]. In [30], the authors designed a power splitting-based relaying scheme for a UAV capable of energy harvesting (EH) and information forwarding to optimize the network throughput. Similarly, in order to save energy in the UAV's battery and to enhance flight endurance, Yang et al. [31] applied EH technology to the UAV for collecting RF energy from the ground base station, and an outage probability analysis in terms of urban environment parameters was derived. Apart from RF-powered UAV communication systems, other energy resources for UAV's operation have also been applied by using solar power [32–35] or laser power [36]. In [32], the UAV was equipped with solar panels to convert the harvested solar energy to electrical energy with the aim of enabling long endurance flights. The authors in [33] proposed the optimal trajectory planning to maximize the harvested solar energy. However, the design did not consider the impact of harvested solar energy for communications. Sun et al. [34] investigated the resource allocation for multi-channel solar-powered UAV systems to maximize the throughput. Nevertheless, their constant aerodynamic power consumption model may not be applied to the realistic scenarios since the aerodynamic power consumption significantly depends on the UAV's flight velocity. Moreover, to satisfy the QoS requirements of the users, the authors in [35] proposed joint trajectory and resource allocation algorithms to maximize the system sum throughput. However, the designs for UAV communication systems in [34,35] may lead to the high mass and size overheads of the solar-powered UAVs. Furthermore, using the OMA technique can result in a low spectrum efficiency of the massive access scenarios.

1.2. Motivations and Contributions

To the best of our knowledge, in most existing works on EH-powered wireless communication systems, information about the energy harvesting arrival is assumed to be known. Thus, it is not always available for designing appropriate solutions to the real wireless UAV communication issues. Moreover, the consumption models for the propulsion energy of the UAVs are quite sophisticated and critically depend on many factors such as the UAV's trajectory, velocity, and acceleration [29,34,35]. This can increase the complexity of developing contemporary schemes. On the other hand, establishing the schemes using only harvested energy for both wireless communications and flight operations might not optimize EH-powered UAV communication performance if the UAV carries a high energy consumption aerial base station or if the harvested energy arrival rate is small [34–37]. By combining

all these issues, devising a method to optimize the service performance of the EH-powered UAV to multiple ground users is still a very challenging task, especially in unexpected circumstances such as temporary disaster areas or complex terrains.

Motivated by the above analysis, in this paper, we propose two joint caching and power allocation schemes for solar-powered, UAV-enabled NOMA communication systems under two scenarios. In the first scenario, the system has the prior knowledge of the harvested energy distribution of the UAV. On the other hand, in the second scenario, we consider the case that the system does not know the harvested energy distribution of the UAV. The GUs require the number of data items stored in the local station. Nevertheless, there are no available direct links between the local station and the GUs due to unexpected or emergency circumstances such as natural disasters, obstacles, and long-distance transmissions. The deployment of terrestrial infrastructure can be infeasible and challenging owing to sophisticated environments, as well as high operational costs. Thus, the UAV is employed to cache part of the content from the local station and deliver data to the GUs. In this work, the UAV can harvest solar energy from the ambient environment. However, the solar panel equipped on the UAV cannot sufficiently provide long-term operation due to its large mass, high mobility energy, and communication energy. To address this problem, the battery is fully recharged at the local station (LS) by the grid power whenever the UAV returns to the station.

There are two portions in the battery: mobility capacity used for flight operation and transmission capacity used for data transmissions. Mobility capacity representing the space needed for flight energy occupies a large portion of the battery. Therefore, the remaining space required for data transmissions (i.e., transmission capacity) in the battery is significantly limited. The amount of initial energy for data transmissions in the battery is not enough for providing the higher data rate to the GUs in the long term. It is supposed that the UAV always harvests the energy during its flight. Hence, during the serving time of each round, the UAV can leverage harvested solar energy to transmit data to the GUs. The mobility energy is assumed to be preserved enough in each round; thus, the harvested energy used for data transmission has a higher priority during the serving time. This means the harvested energy is used for replenishing the transmission capacity before it is used to charge the mobility capacity during the serving time. Besides, the battery is always recharged by the harvested energy during the non-serving time to reduce the grid power consumption required for charging and additional charging time when the UAV is at the LS. In other words, the harvested energy is stored in the on-board battery, which can be used not only for providing data transmission services to GUs during the serving time (i.e., the duration time that the UAV flies around the circular trajectory), but also for recharging the battery for its flight operation during the non-serving time (the time when the UAV approaches the LS and the time when the UAV goes to the serving area). Therefore, it is worth applying solar harvesting to the UAV-based communication system.

Instead of using conventional orthogonal multiple access (OMA) (e.g., TDMA, FDMA, CDMA), which causes low spectrum efficiency, the NOMA technique is applied to enhance the data rate of the UAV system in which the UAV can simultaneously transmit data to the GUs. In this paper, there are three phases of the UAV's operation: (1) performing the caching update process and then approaching the serving area, (2) flying along the circular trajectory while doing the communication process, and (3) returning to the LS for re-caching the files and recharging the battery, as shown in Figure 1a. The caching update process is implemented at the local station in which the UAV pre-caches part of the content from the local station and replenishes the battery for the next round. Then, it approaches its serving area to start flying along the predefined circular trajectory where the GUs can be served. Next, the communication process of the UAV will be executed in which the UAV can transmit data based on the content requests of the GUs during the UAV's flight following the predefined circular trajectory. After finishing a circular trajectory flight period, the communication process will temporarily be terminated, and the UAV needs to go back to the LS for re-caching the content and battery recharging. These processes will repeat until the UAV satisfies the GU's requests. In this paper, using solar harvesting for the UAV will help relieve the burden of grid power-based

energy consumption. Furthermore, finding the proper solution for the solar-powered UAV to provide the energy-efficient communications is still a challenging task under the limited energy harvesting technology. This can make the solar-powered UAV system more applicable to the real wireless system scenarios. In a nutshell, the main contributions can be summarized as follows.

- Firstly, we study a model of a cache-enabled downlink UAV communication network. Ground users request data items stored in the library of a local station, but direct links are not available. Thus, the solar-powered UAV is employed to cache content from the local station and then approach distant users to execute data transmissions using NOMA technology. However, the UAV is equipped with both limited battery capacity and cache capacity. Therefore, we aim to efficiently allocate the harvested energy to the GUs for the long-term operation.
- Secondly, we formulate the problem of the sum data rate maximization as the framework of a partially observable Markov decision process (POMDP). An iteration-based dynamic programming approach is proposed to obtain the optimal policy for the UAV in order to maximize the system data rate under the assumption that the UAV has prior environment information. With this method, the UAV can efficiently cache the content from the local station at the beginning of each flight period and allocate an appropriate portion of transmission power for the GUs throughout every time slot under energy and cache constraints.
- Thirdly, we present another approach using an actor-critic-based reinforcement learning algorithm to deal with the problem in the scenario where the UAV does not have information on environment dynamics in advance. With the actor-critic-based method, the UAV can interact with the environment and gradually learn the optimal policy as time goes on, based on trial-and-error without prior environment knowledge.
- Lastly, extensive numerical results are provided to validate the proposed algorithm's performance through various network parameters. We show that, with joint caching and power allocation, the two proposed schemes are superior to the benchmark schemes where the UAV greedily utilizes transmission power without long-term considerations.

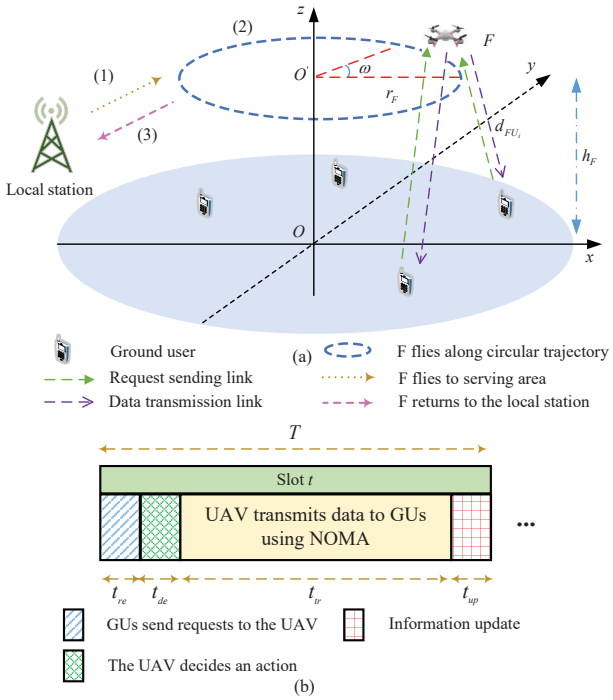

Figure 1. (**a**) The considered network with one UAV (unmanned aerial vehicles) and multiple ground users (GUs). (**b**) The time-frame structure.

The remainder of this paper is organized as follows. The model for the EH-powered UAV downlink communication system is presented in Section 2. Next, we describe the proposed POMDP-based joint cache scheduling and power allocation scheme in Section 3, and the proposed actor-critic-based learning framework is presented in Section 4. The discussions on the simulation results are elaborated in Section 5. Finally, we conclude this work in Section 6.

2. System Model

We consider a caching-based UAV-enabled downlink wireless transmission system adopting non-orthogonal multiple access and content caching technologies where a UAV, F, is employed as a mobile base station to serve a group of I ground users, denoted by $\mathbb{I} = \{1, 2, ..., I\}$. We assume GUs do not have direct links to the local station (LS) where all content that the GUs requests is stored. This kind of network scenario can be a practical instance in suburban environments where the deployment of communication infrastructures is still restricted or in urban environments where damage of the infrastructures may happen due to natural disasters. Thus, the remote users may not get services from a local station. For that reason, the UAV is dispatched to obtain cached contents from the LS, and it then flies along a predefined trajectory to transmit the requested data to GUs. In the existing works, given the user distribution in the network, many effective methods have been used to optimize the UAV's placement and UAV's trajectory [19,20]. Besides, the approach to maximize the coverage area of the UAVs was well studied in [8]. Therefore, in this paper, we do not optimize the flight trajectory and coverage region of the UAV in the network. Instead, we aim to maximize the long-term throughput based on the predefined trajectory. For that reason, we assume that the circular trajectory of the UAV's flight is known based on the locations of the GUs, and the coverage region of F is large enough to guarantee the connection to all GUs by following the predefined circular trajectory with a reasonable radius and altitude. This means that the GUs are still in the UAV's coverage during the UAV's circular flight such that the GUs always get the data delivery from F. It is noteworthy that the system still can be well applied for other flight trajectories of the UAV. Our main goal is to allocate the appropriate transmission power to the GUs and schedule data caching of the UAV under a predefined trajectory to obtain the maximal long-term data rate of the system.

Each data transmission is executed in every time slot t, and meanwhile, each caching action is executed at the beginning of a flight period, which is determined as a round in which the UAV flies to the serving area and then flies along its predefined circular trajectory and returns to the LS. However, due to a limited cache capacity, it can only periodically cache part of the content from the LS at the beginning of every flight period. The GUs are assumed to have a fixed power supply, whereas the UAV has a limited-capacity battery. Hence, UAV F is equipped with an energy harvester to scavenge solar energy from the ambient environment to replenish its battery. We assume the UAV works in an ideal environment without any environment factors (e.g., wind). Suppose that the UAV continuously flies at a constant velocity, v_F, in a circular trajectory with radius r_F, at altitude h_F, and the UAV position repeats every T_F (seconds). Thus, the flight length for the circular trajectory is defined as $T_F = \frac{2\pi r_F}{v_F}$, and the number of time slots discretized in each circular trajectory length is determined as $N_F = \frac{T_F}{T}$, where T is the time slot duration. Note that the UAV's location is assumed to be unchanged during each time slot when T is chosen sufficiently small in the system [35].

Without loss of generality, we consider three-dimensional (3D) Cartesian coordinates (x, y, z) where $(x, y, 0)$ represents the ground plane and z is the altitude. The location of GU_i is denoted as $\mathbf{p}_i = (x_i, y_i, 0), i \in \mathbb{I}$. In fact, when disasters occur, the network infrastructure may be corrupted. However, the GUs can still position their location easily thanks to a GPS decoder, which is integrated into most mobile devices currently. Thus, the GUs can report their locations to the UAV such that the UAV can calculate the flight trajectory to serve the GUs' requests. For the devices without GPS, the UAV can still estimate the GUs locations based on the received signal strength indicator (RSSI), which is well studied in the literature [38,39]. Furthermore, when the locations of the users are known, determining the flight trajectory of the UAV was proposed in [20]. In this paper, we do not focus on

an approach to obtain the GUs' locations and the UAV's trajectory. Instead, we mainly focus on the power allocation with data caching at the UAV to maximize the long-term data rate of the system. Therefore, it is assumed that the GUs' locations and the UAV's trajectory are known in advance. Herein, we establish the formulation for the circular trajectory of the UAV in the serving area, which is defined as the region where the GUs are located. The 3D setup of the considered network consisting of the LS, the UAV, and multiple GUs is illustrated in Figure 1a. Point O', located at $\mathbf{p}_{O'} = (0, 0, h_F)$, is the center of the circular trajectory with radius r_F, in which F flies. Let ω denote the angle of the circle of F's location with respect to the x-axis. The location of F at time slot t can be determined as $\mathbf{p}_F(t) = (x_F(t), y_F(t), z_F(t)) = (r_F \cos\omega(t), r_F \sin\omega(t), h_F)$. The time frame structure of the system is illustrated in Figure 1b. The time frame is divided into four phases: GUs' requests (t_{re}), UAV's decision (t_{de}), data transmission (t_{tr}), and information update (t_{up}). At the start of a time slot, the GUs will send data item requests to F. Then, a decision will be determined at F by allocating the transmission power to the GUs based on the current state of the system. Subsequently, data transmission will be conducted according to the assigned power portions for the GUs in the data transmission phase. Finally, the system will update its state at the end of the time slot.

2.1. Channel and Transmission Models

According to the above network setup, the time-dependent distance between F and GU_i can be calculated as:

$$d_{FU_i}(t) = \|\mathbf{p}_F(t) - \mathbf{p}_i\|, \tag{1}$$

where $\|.\|$ denotes the Euclidean norm operation. In practice, the air-to-ground wireless channels from the UAV to GUs are normally dominated by LOS links, where the quality of the channel only depends on communication distance [40]. Moreover, UAV-assisted information dissemination is more necessary in rural regions than in urban regions [2]. In rural regions, building density is very low, and thus, the probability of non-line-of-sight links is also low. Therefore, in this paper, wireless channels from F to the GUs are assumed to follow a free-space path loss model. As a consequence, channel power gain from F to GU_i at time slot t can be expressed as [41]:

$$h_{FU_i}(t) = \beta_0 d_{FU_i}^{-\alpha}(t) = \frac{\beta_0}{\|\mathbf{p}_F(t) - \mathbf{p}_i\|^{\alpha}}, \tag{2}$$

where β_0 represents the channel power gain at the reference distance, $d_0 = 1$ m, which depends on the carrier frequency, antenna gain, etc.; and α is the path loss exponent. Suppose that F has access to the flight control and location information of the GUs for power allocation. Besides, it is worth noting that the channel gain between F and the GUs varies over period T_F due to the movement of F. Given the location of F at time slot t, the channels of the GUs are sorted in F to apply NOMA.

Typically, a NOMA scheme enables a base station to serve multiple users simultaneously over the same frequency band. The power portions for users are assigned in an inversely proportional manner based on their channel conditions, in which the low channel gain user requires a higher allocated transmission power, and vice versa. We assume that each GU's channel gain is placed in an ascending manner in time slot t.

According to the downlink NOMA principle, UAV F will transmit a combined signal, $s_F(t)$, to all GUs with the assigned power portions in time slot t. Specifically, with the content requests of the GUs in time slot t, the transmitted signal by UAV F can be written as:

$$s_F(t) = \sum_{i=1}^{I} \sqrt{\lambda_i(t) P_F(t)} s_i(t), \tag{3}$$

where $s_i(t)$ is the normalized information for GU_i in time slot t with $\mathbb{E}\left[|s_i|^2\right] = 1$; $P_F(t) = \frac{e^{tr}(t)}{t_{tr}}$ represents the total transmission power that F uses to transmit data to the GUs, in which $e^{tr}(t)$ is the amount of transmission energy used by F in the time slot; and $\lambda_i(t)$ denotes the power portion allocated for GU_i in time slot t (s.t. $\sum_{i=1}^{I} \lambda_i = 1$). The received signal at GU_i in time slot t can be given by:

$$y_{U_i}(t) = \sqrt{h_{FU_i}(t)} \sum_{i=1}^{I} \sqrt{\lambda_i(t) P_F(t)} s_i(t) + n_i, \tag{4}$$

where n_i is the zero-mean additive Gaussian noise with variance σ^2 at GU_i. Let us denote the descending order vector of power portions, as $\mathbf{o}(t) = [o(1), o(2), ..., o(I)] \,|o(n) \in \mathbb{I}$. The GU with the highest power portion (with index $o(1)$), treats all signals of other GUs as interference and directly decodes its own information without using SIC. Nevertheless, other GUs need to employ the SIC process where they first decode signals that are stronger (i.e., the GUs with a higher assigned portion) than their own desired signals. Then, those signals will be subtracted from the received signal, and this process will continue until the GUs' own signals are decoded. In other words, each GU will decode its own information by treating other GUs' signals (with smaller power portions) as interference. As explained above, assume that all the signals of $\mathrm{GU}_{o(l)}$, for $l < n$, have been perfectly decoded by $\mathrm{GU}_{o(n)}$. Thus, the signal-to-interference-plus-noise ratio (SINR) at $\mathrm{GU}_{o(n)}$ for decoding its own information is given as:

$$\gamma_{\mathrm{GU}_{o(n)}}(t) = \frac{\lambda_{o(n)}(t) P_F(t) h_{FU_{o(n)}}(t)}{h_{FU_{o(n)}}(t) \sum_{j=n+1}^{I} \lambda_{o(j)}(t) P_F(t) + \sigma^2}. \tag{5}$$

Consequently, the achievable rate at $\mathrm{GU}_{o(n)}$ in (b/s/Hz) to decode its own information in time slot t can be calculated as:

$$R_{\mathrm{GU}_{o(n)}}(t) = \frac{t_{tr}}{T} \log_2 \left(1 + \gamma_{\mathrm{GU}_{o(n)}}(t)\right). \tag{6}$$

Additionally, the SINR at $\mathrm{GU}_{o(n')}$ to decode the information of $\mathrm{GU}_{o(n)}$, for $n < n'$, can be expressed as:

$$\gamma^{o(n')}_{\mathrm{GU}_{o(n)}}(t) = \frac{\lambda_{o(n)}(t) P_F(t) h_{FU_{o(n')}}(t)}{h_{FU_{o(n')}}(t) \sum_{j=n+1}^{I} \lambda_{o(j)}(t) P_F(t) + \sigma^2}, \tag{7}$$

Similarly, the achievable rate at $\mathrm{GU}_{o(n')}$ in (b/s/Hz) to decode the information of $\mathrm{GU}_{o(n)}$ for $n < n'$ in time slot t can be calculated as:

$$R^{o(n')}_{\mathrm{GU}_{o(n)}}(t) = \frac{t_{tr}}{T} \log_2 \left(1 + \gamma^{o(n')}_{\mathrm{GU}_{o(n)}}(t)\right). \tag{8}$$

Finally, the sum rate of the system in time slot t can be expressed as follows:

$$R(t) = \sum_{i=1}^{I} R_{\mathrm{GU}_i}(t) = \sum_{n=1}^{I} R_{\mathrm{GU}_{o(n)}}(t), \tag{9}$$

where $R_{\mathrm{GU}_i}(t)$ represents the achievable rate at GU_i in time slot t subject to $o(n) = i \in \mathbb{I}$.

More specifically, for a better understanding, let us take an example with $I = 2$: if $h_{FU_1}(t) > h_{FU_2}(t)$, then $\lambda_1(t) < \lambda_2(t)$ and $\mathbf{o}(t) = [2, 1]$. At GU_1, by using SIC, it first decodes $s_2(t)$ and then cancels it out from (4) to decode its own signal, $s_1(t)$. Meanwhile, at GU_2, $s_2(t)$ is directly decoded without performing SIC. As a result, the achievable data rates at GU_1 and GU_2 can be respectively calculated by:

$$R_{\mathrm{GU}_1}(t) = \frac{t_{tr}}{T}\log_2\left(1 + \frac{\lambda_1(t)P_F h_{FU_1}(t)}{\sigma^2}\right) \tag{10}$$

and:

$$R_{\mathrm{GU}_2}(t) = \frac{t_{tr}}{T}\log_2\left(1 + \frac{\lambda_2(t)P_F h_{FU_2}(t)}{\lambda_1(t)P_F h_{FU_2}(t) + \sigma^2}\right). \tag{11}$$

Eventually, the sum rate of the system in time slot t can be given as follows:

$$R(t) = R_{\mathrm{GU}_1}(t) + R_{\mathrm{GU}_2}(t). \tag{12}$$

2.2. Data Request Behavior of the Ground Users

In this paper, library $\mathbb{K}$ in the LS contains K different finite data items for the requests of GUs. Data items are essentially an abstraction of application data, which might range from database records, web pages, ftp files, etc. We consider the content requests of the GUs to be unrelated to each other. Let us assume that the probability that each GU accesses the same data item in the two consecutive time slots is pretty high, but accesses to the other data item are smaller. That is realistic since the users tend to frequently access the same data source of their interest for a long duration. Thus, we model the request of each GU as a discrete-time Markov chain where the state transition probability of GU_i for two adjacent time slots is illustrated in Figure 2a. $P_{mm,i}$ and $P_{\widetilde{m}\widetilde{m},i}$ (where $P_{mm,i} = P_{\widetilde{m}\widetilde{m},i}\,|\widetilde{m} \in \mathbb{K}\backslash\{m\}$) represent the probabilities that GU_i requests the same data item, m, or another data item, $\widetilde{m}$, respectively, in two adjacent time slots. $P_{m\widetilde{m},i}$ and $P_{\widetilde{m}m,i}$ (where $P_{m\widetilde{m},i} = P_{\widetilde{m}m,i}$) are the probabilities that GU_i requests different items in two adjacent time slots. It is assumed that if the request of GU_i in time slot t is item m, then the probability that GU_i requests item $\widetilde{m}$ in time slot $t+1$ can be computed as:

$$P_{m\widetilde{m},i} = \frac{1 - P_{mm,i}}{K - 1}, \tag{13}$$

where K is the total number of data items in library $\mathbb{K}$. It is worth noting that when GU_i requests an item that is not among the cached data items in the UAV, it cannot receive that requested data from the UAV, and thus, no transmission power will be allocated for GU_i in this time slot, i.e., $\lambda_i(t) = 0$.

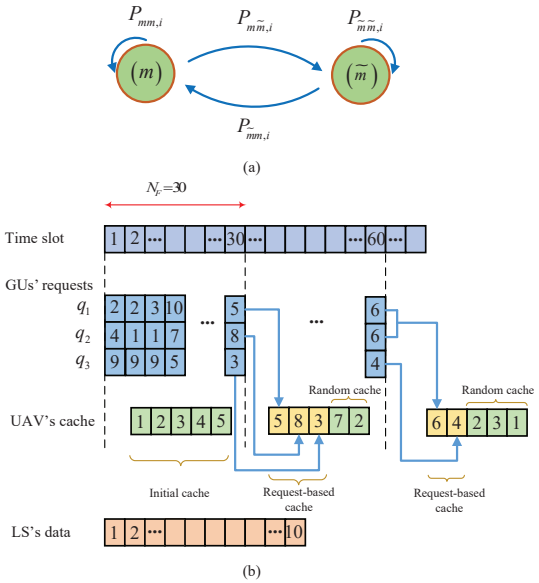

Figure 2. (**a**) The request model of GU_i. (**b**) An example of caching and serving procedures by the UAV, where $N_F = 30$, $C_F = 5$, $I = 3$, and $K = 10$.

2.3. Content Caching Model of UAV

This paper adopts a traditional caching technique for UAV F for serving the requests of the GUs in the network. Since the number of data items that can be cached by F is restricted to a caching capacity, C_F, the UAV needs to cache new data items from $\mathbb{K}$ after each flight period j to replace the old cached items. With periodical caching, performance can be enhanced according to the GUs' requests. In this paper, the non-serving time that includes the duration for the UAV to cache the items, approach the serving area, and return to the LS is approximately unchanged and will not affect the data rate maximization during the serving time. Therefore, the non-serving time can be ignored in the paper, and the term flight period can be referred to as the circular trajectory period of the UAV henceforth. Let $\mathbf{c}_j = \left[c_{j,1}, c_{j,2}, ..., c_{j,C_F}\right]$ denote the cache content vector of UAV F in period j. Based on the data request behavior of the GUs, the cache content vector, $\mathbf{c}_j$, where the data items are cached in period j is divided into two parts: the request-based cache vector, $\mathbf{c}_j^{req}$, and the random cache vector, $\mathbf{c}_j^{ran}$, and can be expressed as $\mathbf{c}_j = \left[\mathbf{c}_j^{req}, \mathbf{c}_j^{ran}\right]$, s.t. $\left|\mathbf{c}_j\right| = C_F$. The former consists of the items cached based on the latest requests of the GUs, while the latter is determined by randomly caching items from the library, except for the items in the request-based cache. In particular, at the start of new flight period j, F will cache the same data items based on the latest items requested by the GUs (i.e., the items requested at the last time slot of previous period $j-1$), and the rest of the space in $\mathbf{c}_j$ is fulfilled by randomly selecting another items from library $\mathbb{K}$ in the LS, such that each item cached in $\mathbf{c}_j$ is unique in the current period. The reason for this caching model is because the probability that GU_i requests the same item is assumed to be much greater than that of GUs requesting a different item between two adjacent time slots, i.e., $P_{mm,i} \gg P_{m\tilde{m},i}$, as presented in the previous subsection.

We use $\mathbf{q}(t) = [q_1(t), q_2(t), ..., q_I(t)]$ to denote the item request vector of the GUs, where $q_i(t) \in \{1, 2, ..., K\}$ represents the item request of GU_i at the start of time slot t, and meanwhile, N_F denotes the total number of time slots in each circular trajectory period. If the GUs request data items different from each other in the last time slot of period j, i.e., $q_i\left(jN_F\right) \neq q_{\tilde{i}}\left(jN_F\right)$, the request-based cache vector, $\mathbf{c}_{j+1}^{req}$, and the random cache vector, $\mathbf{c}_{j+1}^{ran}$, for the next period, $j+1$, can be respectively determined as follows:

$$c_{j+1,i}^{req} = q_i\left(jN_F\right) | i \in \{1, 2, ..., I\}, \tag{14}$$

and:

$$\begin{array}{l} c_{j+1,i}^{ran} | i \in \{1, 2, ..., C_F - I\} \text{ is randomly cached} \\ \text{in } \mathbb{K} \backslash \left\{c_{j+1,1}^{req}, c_{j+1,2}^{req}, ..., c_{j+1,I}^{req}\right\}, \end{array} \tag{15}$$

where $c_{j+1,i}^{req} \in \{1, 2, ..., K\}$ is the cached item i_{th} of $\mathbf{c}_{j+1}^{req}$. It is worth noting that if there are similar requested items among the GUs' requests in the last time slot of period j, then UAV F will only cache these same items one time in $\mathbf{c}_{j+1}^{req}$ for use in the next period, $j+1$, to save cache space in $\mathbf{c}_{j+1}$.

An example of the caching process by UAV F can be illustrated in Figure 2b with $N_F = 30$, $C_F = 5$, $I = 3$, and $K = 10$. In time slot $t = 30$, which belongs to period $j = 1$, the requests of the GUs are $q_1(30) = 5$, $q_2(30) = 8$, and $q_3(30) = 3$, and then, $\mathbf{c}_2^{req} = [5, 8, 3]$ and $\mathbf{c}_2^{ran} = [7, 2]$. In time slot $t = 60$ with $j = 2$, the requests of $\mathrm{GU}_1 = \mathrm{GU}_2 = 6$ are duplicates, and the request of $\mathrm{GU}_3 = 4$; thus, $\mathbf{c}_3^{req} = [6, 4]$ and $\mathbf{c}_3^{ran} = [2, 3, 1]$.

2.4. Energy Harvesting Model of the UAV

In this paper, UAV F is assumed to have a limited-capacity battery, E^{Bat}, and it is equipped with an energy harvesting circuit to harvest solar energy for its operation. UAV F can simultaneously harvest solar energy and perform other operations such as forward movement, climbing up and down, and data transmissions. In this work, we aim at efficiently using the harvested solar energy in the UAV in order to allocate proper transmission power to the GUs during the serving duration. Since the amount of

flight energy consumed for a round trip of the UAV can be approximately estimated, for simplicity, the energy portion for the mobility of the UAV is not shown in the formulation. Thus, we only consider the battery capacity portion required for the data transmission (i.e., transmission capacity), and it is also denoted as E^{Bat} for our simplified formulation purposes. If E^{Bat} is full during the serving time (i.e., the maximum value of the transmission capacity portion is achieved), the rest of the amount of harvested energy will be stored in the mobility capacity portion that is used for the UAV's flight. Herein, the amount of energy harvested by F in time slot t, denoted as $E^h(t)$, is finite, where $E^h(t) \in \left\{E_1^h, E_2^h, ..., E_\xi^h\right\}; 0 \leq E_z^h < E^{Bat}$, and $z \in \{1, 2, ..., \xi\}$ and is assumed to follow a Poisson distribution [42]. The authors in [42] carried out empirical measurements for the modeling of a solar-powered wireless sensor node in time-slotted operation and showed that the stored energy characteristics depend on many factors such as the time slot duration, light intensity, power level, and the deployment environment. As a result, the Poisson distribution model achieved a near fit for the collected measurements. The probability distribution of the energy harvested by F can be given by:

$$P^h(z) = \Pr\left[E^h(t) = E_z^h\right] = \frac{\left(E^{h,avg}\right)^z \exp(-E^{h,avg})}{z!}, \tag{16}$$

where $E^{h,avg}$ represents the mean energy harvested by F. For tractability in the simulation, the amount of harvested energy can be approximated, and the maximum harvested energy can be determined according to network parameters such that the cumulative distribution function is close enough to one.

2.5. Sum Rate Maximization Formulation

In this paper, we aim to optimize the transmission power allocated to the GUs and the content caching by UAV F such that the sum cumulative data rate of ground users can be maximized in a long-term operation. Thus, the problem formulation can be expressed as follows:

$$\begin{aligned} &\max_{\lambda_i(t), P_F(t), \mathbf{c}_j} \left(\sum_{k=t}^{\infty} \sum_{i=1}^{I} R_{\mathrm{GU}_i}(k)\right) \\ &s.t. \quad \sum_{i=1}^{I} \lambda_i(t) = 1, && \text{(a)} \\ &\quad 0 \leq P_F(t) \leq P_F^{\max}, && \text{(b)} \\ &\quad c_{j,i} \neq c_{j,\bar{i}} \,|i \in \{1, 2, ..., C_F\}, && \text{(c)} \end{aligned} \tag{17}$$

where $\mathbf{c}_j$ is the cache content vector of UAV F in flight period j; $P_F^{\max}$ represents the upper bound of the transmission power that F can use to transmit data to the GUs. Constraint (a) specifies that the UAV totally assigns its transmission power, $P_F(t)$, to GUs that request items from the UAV's cache in time slot t. Constraint (b) guarantees that the total transmission power for GUs in each time slot is no greater than the maximum transmission power that the UAV can use without causing it to be inactive owing to an energy shortage. Finally, Constraint (c) ensures that every cached item is unique in the cache content vector for period j, where $c_{j,i}$ represents the i_{th} item of cache content vector $\mathbf{c}_j$.

It is worth noting that although maximizing the energy utilization in the current time slot can optimize the temporal data rate of the system, it may cause inactivity upon data transmission in the subsequent time slots due to an energy shortage in F. Consequently, it can significantly degrade the long-term sum rate of the network. Furthermore, dynamic data requests of the GUs will also affect the performance of the system, since the caching constraint on F is taken into account. Therefore, according to the system state, finding an optimal policy for joint cache scheduling and power allocation in F to obtain the maximum long-term sum rate of the system is the main goal of this study.

3. Proposed Solution Using the POMDP Framework

In this section, we propose a joint optimal cache scheduling and power allocation scheme using a POMDP framework for F over the long run, based on prior information for the harvested energy distribution and the request model for the GUs. To be more specific, after receiving the requests by the GUs, F will allocate the optimal transmission power for each GU in order to obtain the maximized long-term sum data rate for the system. The problem of sum data rate maximization is first formulated as the framework of a partially observable Markov decision process where the effect of the decision in the current time slot on the subsequent time slots is taken into account [43]. Subsequently, the optimal policy can be obtained by adopting the approach of value iteration-based dynamic programming [44].

3.1. Markov Decision Process

The Markov decision process (MDP) is generally defined as a tuple $\langle \mathbb{S}, \mathbb{A}, \mathbb{P}, \varphi \rangle$, where $\mathbb{S}$, $\mathbb{A}$, and $\mathbb{P}$ are the state space, action space, and state transition probability space, respectively; $\varphi : \mathbb{S} \times \mathbb{A} \mapsto \mathbb{R}$ represents the reward function. We define the system state as $s(t) = \left(e^{rm}(t), \omega(t), \boldsymbol{\theta}(t), t_{in}(t), \mathbf{c}_j\right) \in \mathbb{S}$, where $e^{rm}(t)$ is the remaining energy in F; $0 \leq \omega(t) \leq 2\pi$ is the angle of the circle for F's location with respect to the x-axis; $\boldsymbol{\theta}(t) = [\theta_1(t), \theta_2(t), ..., \theta_I(t)]$ is the belief vector, with $\theta_i(t)$ as the belief (probability) that the requested content of GU_i will be in the current cache content vector, $\mathbf{c}_j$, in time slot t; $t_{in}(t) \in \{1, 2, ..., N_F\}$ is the index of time slot t in terms of flight period j. Note that $\mathbf{c}_j$ will only be updated based on the requests of the GUs at the end of time slot t when $t_{in}(t) = N_F$ in each flight period, and meanwhile, $s(t)$ is always updated based on the selected action by F and the amount of harvested energy at the end of each time slot. The set of actions can be denoted as $\mathbb{A} = \left\{\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_{|\mathbb{A}|}\right\}$, where $\mathbf{a}_v = \left[e_v^{tr}, \lambda_{1,v}, \lambda_{2,v}, ..., \lambda_{I,v}\right] | v \in \{1, 2, ..., |\mathbb{A}|\}$ is the action v in $\mathbb{A}$; where e_v^{tr} ($0 \leq e_{\min}^{tr} \leq e_v^{tr} \leq e_{\max}^{tr}$) is the transmission energy in UAV F, and $0 \leq \lambda_{i,v} \leq 1$ is the power portion assigned for GU_i. The notations $e_{\min}^{tr}$ and $e_{\max}^{tr}$ represent the minimum and maximum transmission energy in the UAV. We further define the reward for the system as the sum data rate of the network. Thus, given state $s(t)$ and action $\mathbf{a}(t)$, the corresponding reward, denoted by $R\left(s(t), \mathbf{a}(t)\right)$, is computed by using Equation (9).

The operation of UAV F can be expressed as follows. At a given time instant t, F employs action $\mathbf{a}(t)$ based on the system state and the content requests of the GUs, and then, the reward for the system, $R\left(s(t), \mathbf{a}(t)\right)$, will be achieved at the end of the time slot. Action $\mathbf{a}(t)$ causes the system to transit from state $s(t)$ to a new state, $s(t+1)$. Thus, the state of the system will be updated for the next operation when the data transmission in time slot t is finished.

In this paper, we aim to find the optimal transmission power allocation policy based on the cache scheduling discussed in Section 2.3 for UAV F in each slot t in order to maximize the accumulated reward from the time slot to the time horizon. In addition, transmission power is determined by using transmission energy e^{tr} and transmission data duration t_{tr}, i.e., $P_F(t) = \frac{e^{tr}(t)}{t_{tr}}$. Therefore, according to the above MDP formulation, Equation (17) can be rewritten as follows:

$$\mathbf{a}_{opt}(t) = \underset{\mathbf{a}(t) \in \mathbb{A}}{\arg\max} \left\{ \sum_{k=t}^{\infty} \beta^{k-t} R\left(s(k), \mathbf{a}(k)\right) | s(t) \right\}, \tag{18}$$

where $0 \leq \beta \leq 1$ is the discount factor, which indicates the effect of the current action on the future rewards. According to the dynamic item requests of the GUs, the observation is defined as the probable case that shows whether the item requests of the GUs are in cached items of F in a given time slot and will be discussed in the next subsection.

3.2. Observation Description

This section introduces possible observations, the respective rewards, and the ways to update the system state for the next time slot according to the selected action of a given time slot. Let us consider a network with two GUs ($I = 2$) connecting to UAV F to acquire data according to their requests. At the given state, $s(t)$, the requests of the GUs are $q_1(t)$ and $q_2(t)$, and the UAV executes action $\mathbf{a}(t)$. It is obvious to note that for all possible observations, the angle of UAV F in the next time slot is updated as $\omega(t+1) = \omega_{next}(t)$, where $\omega_{next}(t)$ denotes the next angle of the UAV in its predefined circular flight trajectory. In the following, we present a way to update other information regarding the remaining energy, the belief vector, the transition probability, the time slot index, and the cache content vector in each observation for this particular circumstance. These can be respectively described as follows.

3.2.1. Observation 1 (O_1)

The requests of both GU_1 and GU_2 are in the cached items in $\mathbf{c}_j$ of UAV F. The probability that the event happens can be calculated as:

$$\Pr[O_1] = \theta_1(t)\theta_2(t). \tag{19}$$

The reward can be obtained as follows:

$$R(s(t), \mathbf{a}(t) | O_1) = \sum_{i=1}^{2} R_{\mathrm{GU}_i} = R_{\mathrm{GU}_1} + R_{\mathrm{GU}_2}, \tag{20}$$

where R_{GU_i} can be obtained by using Equation (6). The belief vector can be updated as follows:

$$\boldsymbol{\theta}(t+1) = [P_{mm,1} + \tau_1, P_{mm,2} + \tau_2], \tag{21}$$

where $\tau_i = \left(\frac{1-P_{mm,i}}{K-1}\right)(C_F - 1) \,|\, i \in \{1, 2, ..., I\}$. Next, the remaining energy in F for the next time slot is:

$$e^{rm}(t+1) = \begin{cases} \min\left(e^{rm}(t) - e^{tr}(t) + E^h(t), E^{Bat}\right) & \text{if } t_{in}(t) < N_F \\ E^{Bat} & \text{otherwise} \end{cases} \tag{22}$$

with transition probability:

$$\Pr[e^{rm}(t+1) | e^{rm}(t)] = \begin{cases} \Pr\left[E^h(t) = E^h_z\right] & \text{if } t_{in}(t) < N_F \\ 1 & \text{otherwise} \end{cases}, \tag{23}$$

where $\Pr\left[E^h(t) = E^h_z\right]$ can be calculated as in Equation (16). To explain Equations (22) and (23) with the case of $t_{in}(t) = N_F$, the remaining energy (i.e., the energy in transmission capacity) at F is always full because UAV F finishes one circular trajectory and returns to the LS for recharging its battery. The index of the next time slot in terms of flight period j can be updated as:

$$t_{in}(t+1) = \begin{cases} t_{in}(t) + 1 & \text{if } t_{in}(t) < N_F \\ 1 & \text{otherwise} \end{cases}. \tag{24}$$

Finally, the cache content vector can be updated by:

$$\mathbf{c}_j = \begin{cases} \mathbf{c}_j & \text{if } t_{in}(t) < N_F \\ \left[\mathbf{c}^{req}_{j+1}, \mathbf{c}^{ran}_{j+1}\right] & \text{otherwise} \end{cases}, \tag{25}$$

where $\mathbf{c}_{j+1}^{req}$ and $\mathbf{c}_{j+1}^{ran}$ can be determined with Equations (14) and (15), respectively. It is important to note that the UAV will only update the cache content vector when it is in the last time slot of period j.

3.2.2. Observation 2 (O_2)

The request of GU_1 is in the cached items in $\mathbf{c}_j$, but that of GU_2 is not in $\mathbf{c}_j$ of UAV F. The probability that the event happens can be calculated as:

$$\Pr[O_2] = \theta_1(t)\ (1-\theta_2(t))\,. \tag{26}$$

The reward can be obtained as follows:

$$R(s(t), \mathbf{a}(t)\,|O_2\,) = R_{\mathrm{GU}_1}, \tag{27}$$

where R_{GU_1} can be calculated with Equation (6). The belief vector can be updated as follows:

$$\boldsymbol{\theta}(t+1) = \begin{cases} \left[P_{mm,1}+\tau_1, \left(\frac{1-P_{mm,2}}{K-1}\right) C_F\right] & \text{if } t_{in}(t) < N_F \\ [P_{mm,1}+\tau_1, P_{mm,2}+\tau_2] & \text{otherwise} \end{cases}, \tag{28}$$

where τ_i is calculated in a way similar to Equation (21). The remaining energy, the transition probability, the index of the time slot, and the cache content vector can be updated with Equations (22)–(25), respectively.

3.2.3. Observation 3 (O_3)

The request of GU_1 is not in the cached items in $\mathbf{c}_j$, but that of GU_2 is in $\mathbf{c}_j$. The probability that the event occurs can be calculated as:

$$\Pr[O_3] = (1-\theta_1(t))\,\theta_2(t)\,. \tag{29}$$

The reward can be obtained as follows:

$$R(s(t), \mathbf{a}(t)\,|O_3\,) = R_{\mathrm{GU}_2}, \tag{30}$$

where R_{GU_2} can be computed with Equation (6). The belief vector can be updated as follows:

$$\boldsymbol{\theta}(t+1) = \begin{cases} \left[\left(\frac{1-P_{mm,1}}{K-1}\right) C_F, P_{mm,2}+\tau_2\ \right] & \text{if } t_{in}(t) < N_F \\ [P_{mm,1}+\tau_1, P_{mm,2}+\tau_2\ ] & \text{otherwise} \end{cases}, \tag{31}$$

where τ_i is calculated as it is in Equation (21). The remaining energy, the transition probability, the index of the time slot, and the cache content vector can be updated with Equations (22)–(25), respectively.

3.2.4. Observation 4 (O_4)

The requests of both GU_1 and GU_2 are not in the cached items in $\mathbf{c}_j$ of UAV F. The UAV will stay silent; and hence, there is no reward in this case, i.e., $R(s(t), \mathbf{a}(t)\,|O_4\,) = 0$. The probability that the event occurs can be calculated as:

$$\Pr[O_4] = (1-\theta_1(t))\,(1-\theta_2(t))\,. \tag{32}$$

The belief vector can be updated as follows:

$$\boldsymbol{\theta}(t+1) = \begin{cases} \left[\left(\frac{1-P_{mm,1}}{K-1}\right) C_F, \left(\frac{1-P_{mm,2}}{K-1}\right) C_F\right] & \text{if } t_{in}(t) < N_F \\ \left[P_{mm,1} + \tau_1, P_{mm,2} + \tau_2\right] & \text{otherwise} \end{cases}. \tag{33}$$

The remaining energy in F for the next time slot is:

$$e^{rm}(t+1) = \begin{cases} \min\left(e^{rm}(t) + E^h(t), E^{Bat}\right) & \text{if } t_{in}(t) < N_F \\ E^{Bat} & \text{otherwise} \end{cases} \tag{34}$$

with the transition probability being the same as Equation (23). Similarly, the index of the time slot and the cache content vector can be updated with Equations (24) and (25), respectively.

3.3. Value Iteration-Based Dynamic Programming Solution

According to the POMDP principle, the value function is defined as the maximum value of the cumulative discounted system reward that starts from the current time slot to the infinite time horizon, and it is used to select the optimal action for the UAV. Thus, given a state $s(t)$, the value function can be given as follows:

$$V_{s(t)} = \max_{\mathbf{a}(t)\in\mathbb{A}} \left\{ \begin{array}{c} \sum_{k=t}^{\infty} \beta^{k-t} \times \sum_{O_m} \Pr\left[O_m\right] \\ \times \sum_{e^{rm}(k+1)} \Pr\left[e^{rm}(k+1) \,|e^{rm}(k), O_m\right] \\ \times R\left(s(k), \mathbf{a}(k)\right) |s(t) \end{array} \right\}, \tag{35}$$

where $\Pr\left[O_m\right]$ represents the probability that the observation O_m occurs; $\Pr\left[e^{rm}(k+1) \,|e^{rm}(k), O_m\right]$ is the probability that the remaining energy of the UAV will transfer from $e^{rm}(k)$ to $e^{rm}(k+1)$ with corresponding observation O_m; $R\left(s(k), \mathbf{a}(k)\right)$ indicates the reward of the system when it takes the action $\mathbf{a}(k)$ at the state $s(t)$.

The value function in Equation (35) can be obtained by using value iteration-based dynamic programming [44]. Owing to the dynamic item requests of the GUs and the harvested energy, the expected reward for the possible actions in the current time slot will be considered in each time slot. Accordingly, the optimal decision of the UAV in time slot t can be obtained as follows:

$$\mathbf{a}_{opt}(t) = \arg\max_{\mathbf{a}(t)\in\mathbb{A}} \left\{ \underbrace{\begin{array}{l} R^{im}\left(s(t), \mathbf{a}(t)\right) \\ + \sum_{t+1} \Pr\left[e^{rm}(t+1) \,|e^{rm}(t)\right] V_{s(t+1)} \end{array}}_{\text{expected reward of action } \mathbf{a}(t) \text{ at state } s(t)} \right\}, \tag{36}$$

where $R^{im}\left(s(t), \mathbf{a}(\mathbf{t})\right)$ is the expected immediate reward for the system based on action $\mathbf{a}(t)$, which can be obtained by Equation (9). The term $\sum_{t+1} \Pr\left[e^{rm}(t+1) \,|e^{rm}(t)\right] V_{s(t+1)}$ is the expected future reward from action $\mathbf{a}(t)$ in time slot $t+1$, where $V_{s(t+1)}$ can be achieved by solving the problem in Equation (35). For the above setup, the MDP problem in Equation (18) can be transferred to Equation (36), and the optimal policy for long-term data rate maximization can be obtained by using the POMDP framework. The flowchart of the proposed POMDP-based approach is given Figure 3. For further details, the procedure of the slot-by-slot operation of the system when using this scheme is presented in Algorithm 1.

Algorithm 1 Operation of the UAV when using the proposed POMDP-based scheme to obtain the maximum long-term data rate in N time slots.

Input: d_{FU_i}, K, σ^2, T, T_{re}, T_{de}, T_{up}, C_F, E^{Bat}, $E^{h,avg}$, $e^{tr}_{\min}$, $e^{tr}_{\max}$, P_{mm}, $P_{m\tilde{m}}$, h_F, r_F, T_F, α, β_0, β.
Output: Optimal action $\mathbf{a}_{opt}(t)$.
Define $\mathbb{S}$, $\mathbb{A}$, and $\mathbb{P}$.
Apply iteration-based dynamic programming to obtain the value function for every possible state of $\mathbb{S}$ in Equation (35).
for $t = t_0$ // Start from time slot $t = t_0$
    Define the current system state, $s(t)$.
    Receive the requests of GUs, $\mathbf{q}(t)$.
    if no request by GUs is in $\mathbf{c}_j$
        Stay silent.
    else
        Calculate $R^{im}(s(t), \mathbf{a}(t))$ using Equation (9).
        Calculate the corresponding expected future reward of each action $\mathbf{a}(t)$ using Equations (16) and (35).
        Determine $\mathbf{a}_{opt}(t)$ using Equation (36).
        if Action is "stay silent" (i.e., $e^{tr}(t) = 0$)
            Stay silent.
        else
            if only one GU's request is in $\mathbf{c}_j$
                Transmit data to that GU.
            else
                Transmit data to GUs by using NOMA.
            end if
            Obtain the immediate reward for the system.
        end if
    end if
    Update $\mathbf{c}_j$ when $t_{in}(t) = N_F$ with Equations (14) and (15).
    Update system state $s(t+1)$.
end for// The number of considered time slots N.

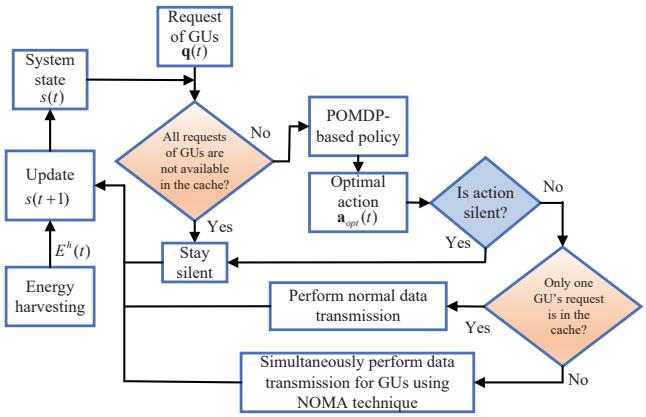

Figure 3. The flowchart of the proposed partially observable Markov decision process (POMDP)-based scheme. TD, temporal difference.

4. Proposed Solution Using the Actor-Critic Learning Framework

In the previous section, we elaborated on the POMDP-based solution to the joint cache scheduling and power allocation problem of the UAV-enabled communication system where prior knowledge of the energy harvesting distribution of the GUs is assumed to be known. Nevertheless, it is hard to identify an evolution model in some network scenarios due to the complexity of network dynamics; hence, acquiring prior information regarding the arrival of harvested energy of a user may be impractical in some circumstances. Moreover, the value iteration programming technique requires

a large number of formulations and much computational overhead. For this reason, we formulate and propose a kind of model-free reinforcement learning (namely, an actor-critic-based method) to deal with the MDP problem assuming there is no prior information on the energy harvesting distribution. Although applying the actor-critic learning approach may lead the system to a locally optimal policy [45], it helps the system learn information about the dynamic wireless environment by interacting directly with the environment to generate a policy without having information on essential network models a priori. Hence, this model-free learning approach can benefit from less formulation and fewer computational effort, compared to the POMDP-based algorithm.

Subsequently, we present the classic actor-critic learning-based scheme to obtain the solution to the MDP problem described in the previous section.

4.1. Actor-Critic Framework Formulation

Generally, the actor-critic framework is composed of three main components: an actor, a critic, and the environment. The actor is responsible for taking an action according to a policy; meanwhile, the critic evaluates the quality of the action and adjusts the policy through temporal difference (TD) [46]. The generalized actor-critic framework is illustrated in Figure 4.

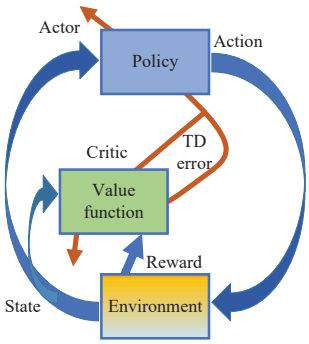

Figure 4. The schematic of the classic actor-critic learning framework.

The value function for the actor-critic-based framework in this paper is the total discounted reward from the current time slot, and it can be modified according to policy Ω during the training phase, which can be obtained as follows [47]:

$$V_s = R\left(s, \Omega\left(s\right)\right) + \beta \sum_{s' \in \mathbb{S}} \Pr\left[s' \left| s, \Omega\left(s\right)\right.\right] V_{s'}, \tag{37}$$

where $\Pr\left[s' \left| s, \Omega\left(s\right)\right.\right]$ represents the transition probability that the system will transfer to state s' after taking an action based on policy $\Omega\left(s\right)$ in state s. Similar to the POMDP-based scheme, in this paper, the actor-critic framework is in charge of determining the optimal policy, $\Omega^*(s)$, and thus, the problem in Equation (18) can be rewritten as:

$$\Omega^*(s) = \arg\max_{\mathbf{a} \in \mathbb{A}} \left\{ R\left(s, \mathbf{a}\right) + \beta \sum_{s' \in \mathbb{S}} \Pr\left[s' \left| s, \mathbf{a}\right.\right] V_{s'} \right\}. \tag{38}$$

In time slot t, the UAV selects and then executes an action, $\mathbf{a}(t)$, based on the current state, $s(t)$, and the current policy, Ω, which is determined by applying a Gibbs soft-max function [47] as follows:

$$\Omega\left(\mathbf{a}(t) \left| s(t)\right.\right) = \Pr\left[\mathbf{a}(t) \in \mathbb{A} \left| s(t)\right.\right] = \frac{e^{\Theta(\mathbf{a}(t)|s(t))}}{\sum\limits_{\mathbf{a} \in \mathbb{A}} e^{\Theta(\mathbf{a}|s(t))}}, \tag{39}$$

where $\Theta\left(\mathbf{a}(t) \left| s(t)\right.\right)$ is the tendency of the UAV to select action $\mathbf{a}(t)$ when the system is in state $s(t)$. Note that this parameter can be adjusted over time such that the UAV can select the best action for

each state when the training phase finishes. After the action is executed, the system will transit to a new state, $s(t+1)$, with transition probability:

$$\Pr\left[s' \in \mathbb{S} \,|s(t), \mathbf{a}(t)\right] = \begin{cases} 1 & \text{if } s' = s(t+1) \\ 0 & \text{otherwise} \end{cases} \tag{40}$$

and the corresponding immediate reward, $R\left(s(t), \mathbf{a}(t)\right)$, will be obtained as expressed in Equation (9). By applying Equation (40) to Equation (38), it obviously implies that the actor-critic-based scheme does not need to have information on the energy arrival distribution in advance, since it actually explores the next state, $s(t+1)$, at the end of time slot t after performing action $\mathbf{a}(t)$. As a result, at the end of the time slot, the critic component will evaluate the quality of the action performed by the UAV by using the TD error. In other words, determining the value function's difference from current state $s(t)$ at the end of each time step will help the UAV gradually find the maximum value function that maps state $s(t)$ to optimal action $\mathbf{a}_{opt}(t)$. Consequently, the TD error in time slot t, which is referred to as the difference between the left and right sides of the Bellman equation [47], is computed as follows:

$$\Delta(t) = R\left(s(t), \mathbf{a}(t)\right) + \beta V_{s(t+1)} - V_{s(t)}. \tag{41}$$

Then, the value function for state $s(t)$ will be updated by:

$$V_{s(t)} = V_{s(t)} + \alpha_c \Delta(t), \tag{42}$$

where α_c denotes the critic step size. Furthermore, the actor component will modify the policy according to the tendency as:

$$\Theta\left(\mathbf{a}(t)\,|s(t)\,\right) = \Theta\left(\mathbf{a}(t)\,|s(t)\,\right) + \alpha_a \Delta(t), \tag{43}$$

where α_a represents the actor step size. According to Equations (42) and (43), the training stage will be terminated as convergence occurs, and the convergence rate will significantly depend on the values of both α_c and α_a. Therefore, the optimal value of these parameters can be adjusted by following empirical designs on various applications.

4.2. Actor-Critic Training Description

The details of the training process for the proposed actor-critic-based scheme, presented in Algorithm 2, can be summarily expressed as follows. At the start of time slot t, the UAV will execute action $\mathbf{a}(t)$ based on current state $s(t)$ and the item requests of the GUs, $\mathbf{q}(t)$. The UAV has to stay silent when none of requests of GUs are in the content cached in the UAV, or it will transmit the corresponding data to the GUs when at least one GU's request is in $\mathbf{c}_j$. The corresponding immediate reward, $R\left(s(t), \mathbf{a}(t)\right)$, and the information of the next state, $s(t+1)$, will be gained based on the observations presented in Section 3.2. The UAV then modifies its parameters, such as $\Delta(t)$, $V_{s(t)}$, $\Theta\left(\mathbf{a}(t)\,|s(t)\,\right)$, and $\Omega\left(\mathbf{a}(t)\,|s(t)\,\right)$, at the end of each time slot. In addition, it is worth noting that the UAV will only re-cache the LS items into $\mathbf{c}_j$ when it finishes a flight period. Unlike the proposed POMDP-based scheme, where the optimal policy is obtained based on an offline formulation that requires energy harvesting distribution information, the proposed actor-critic-based scheme determines the policy from a practical learning process, and thus, it can converge to the locally optimal policy [45]. In other words, by applying the actor-critic solution, we do not need to know the energy harvesting distribution in advance for the transition probability calculation in order to achieve the optimal policy, as in the POMDP-based solution. As a result, it can make this scheme more practical in various network scenarios where no prior knowledge regarding the environment dynamics is known.

Algorithm 2 The detailed training process of the UAV using the proposed actor-critic-based scheme.

Input: d_{FU_i}, K, σ^2, T, T_{re}, T_{de}, T_{up}, C_F, E^{Bat}, $e^{tr}_{\min}$, $e^{tr}_{\max}$, P_{mm}, $P_{m\tilde{m}}$, h_F, r_F, T_F, α, β_0, β.
Output: Optimal policy $\Omega^*(s)$.
Define $\mathbb{S}$ and $\mathbb{A}$.
Define the total number of time slots for training, N_t.
Initialize $\Theta(\mathbf{a}|s)$, V_s, and $\Omega(s)$ where $\mathbf{a}(t) \in \mathbb{A}$, $s \in \mathbb{S}$.
repeat
    Define the current system state, $s(t)$.
    Receive the requests of GUs, $\mathbf{q}(t)$.
    if no request by the GUs is in $\mathbf{c}_j$
        Stay silent.
    else
        Choose an action $\mathbf{a}(t) \in \mathbb{A}$ according to $\Omega(s(t))$.
        if Action is "stay silent" (i.e., $e^{tr}(t) = 0$)
            Stay silent.
        else
            if only one GU's request is in $\mathbf{c}_j$
                Transmit data to that GU.
            else
                Transmit data to GUs by using NOMA.
            end if
            Obtain the immediate reward for the system.
        end if
    end if
    Calculate TD error, $\Delta(t)$, using Equation (41).
    Adjust value function, $V_{s(t)}$, using Equation (42).
    Update tendency, $\Theta(\mathbf{a}(t)|s(t))$, using Equation (43).
    Update the policy, $\Omega(\mathbf{a}(t)|s(t))$, using Equation (39).
    Update $\mathbf{c}_j$ when $t_{in}(t) = N_F$ with Equations (14) and (15).
    Update system state, $s(t+1)$.
until // the training converges or $t = N_t$.

For the comparison of complexity between the two proposed methods, the main computational difference between the two approaches is that the POMDP-based scheme needs to find the value function for the state-space through an offline approach. This leads to higher computational complexity when using Algorithm 1. Specifically, the complexity for each iteration in the POMDP scheme can be computed as $O\left(|\mathbb{A}|\,|\mathbb{S}|^2\left|\mathbb{O}^{obs}\right|\,|\mathbb{P}|\right)$, where $\left|\mathbb{O}^{obs}\right|$ is the number of possible observations. Let us define that the computational complexity for the UAV in each state during the training in Algorithm 2 is $O(1)$. Thus, the total complexity of Algorithm 2 depends on the system state and action spaces and can be calculated as $O(|\mathbb{A}|\,|\mathbb{S}|)$. Furthermore, the convergence rate of the actor-critic scheme is considerably dependent on the actor and critic step sizes. As a consequence, these values should be carefully chosen according to other system parameters. We further provide the summary of the most used symbols and notations in Table 1 to make the paper more readable.

Table 1. Summary of symbols and notations.

Description	Symbol	Description	Symbol
Number of training time slots	N_t	Channel power gain from the UAV to GU_i at time t	$h_{FU_i}(t)$
Number of data items	K	Distance from the UAV to GU_i at time t	$d_{FU_i}(t)$
Time slot duration	T	Location of GU_i	$\mathbf{p}_i$
Request sending time	t_{re}	Center point of the circular trajectory	$\mathbf{p}_{O'}$
Action decision time	t_{de}	Location of the UAV at time t	$\mathbf{p}_F(t)$
Updating time	t_{up}	Angle of the circle of the UAV's location with respect to the x-axis	ω
Caching capacity	C_F	Transmitted signal by the UAV at time t	$s_F(t)$
Battery capacity	E^{Bat}	Received signal at GU_i at time t	$y_{U_i}(t)$
Minimum transmission energy	$e^{tr}_{\min}$	Data rate at GU_i at time t	$R_{GU_i}(t)$
Maximum transmission energy	$e^{tr}_{\max}$	Sum rate of the system at time t	$R(t)$
Mean harvested energy	$E^{h,avg}$	Cache content vector	$\mathbf{c}_j$
Transition probability: from item m to item m	P_{mm}	Request-based cache vector	$\mathbf{c}^{req}_j$
Transition probability: from item m to others	$P_{m\tilde{m}}$	Random cache vector	$\mathbf{c}^{ran}_j$
Altitude of the UAV	h_F	Item request of GU_i at time t	$q_i(t)$
Flight radius of the UAV	r_F	Total transmission power used by the UAV at time t	$P_F(t)$
Flight period	T_F	Noise variance	σ^2
Actor step size	α_a	Channel power gain at the reference distance	β_0
Critic step size	α_c	Path loss exponent	α
Discount factor	β	Belief vector at time t	$\boldsymbol{\theta}(t)$

5. Simulation Results

In this section, we present the numerical simulation results regarding the performance of the two proposed schemes and those of other benchmark schemes based on the Myopicmethod [48]: a Myopic-NOMA scheme, a Myopic-NOMA-RCscheme, and a Myopic-OMA scheme. The term "Myopic" represents the solution in which the optimal decision is made only for the current time slot without considering the future evolution. In the Myopic-NOMA scheme, the UAV always transmits data with optimal transmission power to the GUs by using NOMA whenever more than two GUs' requests are in the cached content of the UAV. Similarly, in the Myopic-NOMA-RC scheme, the UAV randomly caches items from the LS and always transmits data to the GUs with the optimal transmission power by using NOMA. Lastly, in the Myopic-OMA scheme, OMA data transmission is always used with the optimal transmission power. In particular, with this scheme, the data transmission phase is divided into I_{oma} equal sub-slots, where $I_{oma}(t)$ is the number of involved GUs for the data transmissions in time slot t, and the UAV will transmit the corresponding data to each GU through each sub-slot. Therefore, the sum data rate of the Myopic-OMA scheme in time slot t can be calculated with $R^{OMA}(t) = \sum_{i=1}^{I_{oma}(t)} \frac{t_{tr}}{I_{oma}(t)T} \log_2\left(1 + \frac{\lambda_i P_F(t) h_{FU_i}(t)}{\sigma^2}\right)$. Nevertheless, these benchmark schemes only consider the current time slot for maximizing the sum rate. Thus, their optimal policy is made by using the maximum level of transmission energy available in the battery in the current time slot, which can lower system performance in a long operation owing to an energy shortage for data transmissions in subsequent time slots. Meanwhile, the proposed schemes consider not only the current reward, but also the future reward, which was thoroughly presented in Sections 3 and 4. Thus, in the following, we can verify the effectiveness of the two proposed schemes under changes in network parameters. Table 2 shows the parameter setup, and the network topology with $I = 3$ is illustrated in Figure 5. Unless otherwise stated, the transmission energy in the UAV is divided into five equal levels ranging from $0 \leq LV1 \leq LV2 \leq ... \leq LV5 \leq E^{Bat}$, and there are eight levels in the UAV's battery, from zero to E^{Bat}. The span of power portion λ is 0.025. In this paper, the simulation results were achieved by averaging $N = 2 \times 10^5$ time slots. Besides, the harvested energy was stochastically generated in each slot by a Poisson distribution with the mean value of harvested energy $E^{h,avg} = 75$ µJ. During the serving time, there might be no energy for data transmissions by the UAV, which is referred to as energy shortage. In that case, it has to stay silent and wait for upcoming harvested energy in subsequent time slots to transmit data to the GUs.

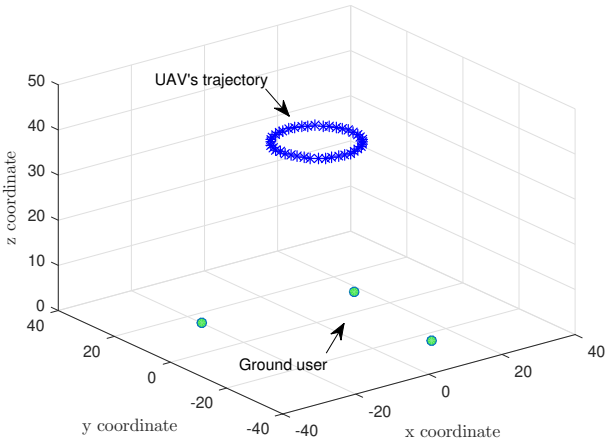

Figure 5. The network topology.

Table 2. Simulation parameters.

Parameter	Notation	Value
Number of training time slots	N_t	2×10^5
Number of data items	K	300 items
Time slot duration	T	200 ms
Request sending time	t_{re}	1 ms
Action decision time	t_{de}	1 ms
Updating time	t_{up}	1 ms
Caching capacity	C_F	120 items
Battery capacity	E^{Bat}	300 µJ
Minimum transmission energy	$e^{tr}_{\min}$	50 µJ
Maximum transmission energy	$e^{tr}_{\max}$	250 µJ
Mean harvested energy	$E^{h,avg}$	75 µJ
Transition probability: from item m to item m	P_{mm}	0.8
Transition probability: from item m to others	$P_{m\tilde{m}}$	0.2
Altitude of the UAV	h_F	40 m
Flight radius of the UAV	r_F	10 m
Flight period	T_F	8 s
Actor step size	α_a	0.1
Critic step size	α_c	0.1
Path loss exponent	α	3
Channel power gain at the reference distance	β_0	-40 dB
Noise variance	σ^2	-120 dBm
Discount factor	β	0.95

We first examine the convergence rate of the actor-critic-based scheme during the training process under various values of λ_c and λ_a for the mean value of harvested energy, $E^{h,avg} = 75$ µJ, based on the achievable sum rate calculated every 1000 time steps, as shown in Figure 6. Besides, the optimal value line is plotted according to the policy obtained by the POMDP-based approach. It is noted that the convergence condition of the algorithm is defined as the convergence condition of the sum data rate. That means that during the training process, the sum data rate is averaged after every batch of 1000 training time slots, and then, the difference between two adjacent updates, Δ_c, is calculated. In the simulation, we set the convergence condition for the algorithm at $|\Delta_c| < 7 \times 10^{-3}$. It is observed from Figure 6 that the sum rate of the system after each iteration of 1000 slots sharply increases in the first 100,000 time slots and then gradually converges to a locally optimal policy that depends on the values of λ_c and λ_a. Therefore, in the simulation, we repeated the training process a number of times and then selected the policy with the proper actor and critic step size values that provide the maximum average rate. In particular, with step sizes greater than 0.1, the proposed scheme provides faster convergence; however, it leads to a lower data rate after 200,000 time slots of training. We can

also see that if we keep decreasing the step size values to less than 0.1, the algorithm might converge to a worse policy due to overfitting. Besides, it is obvious that with the network parameters in this paper, the proposed scheme with critic and actor step sizes $\alpha_c = \alpha_a = 0.1$ provides better performance, in which the data rate mostly converges to the optimal value, given by the POMDP-based scheme, after 200,000 time slots of training. Therefore, we chose actor-critic step size values at $\alpha_c = \alpha_a = 0.1$ for the rest of the simulations.

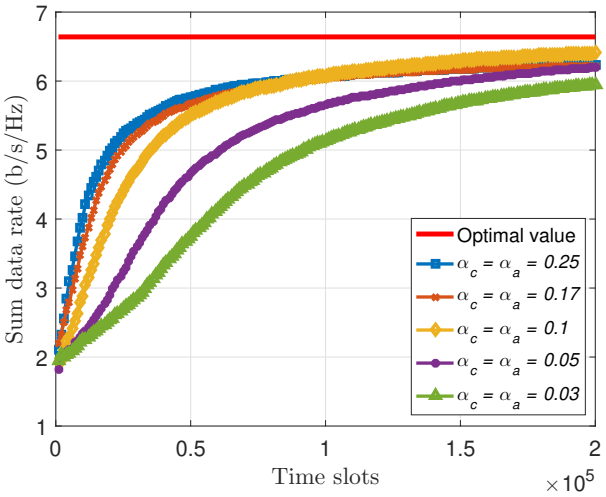

Figure 6. The convergence of the proposed actor-critic-based algorithm according to the mean value of harvested energy.

Figure 7 shows the sum rate according to the mean value of harvested energy in the UAV. It can be seen that the throughput of the system increases when the mean value of the harvested energy goes up. That is because the UAV can harvest more energy from the environment; thus, a number of higher power transmissions can be used for data transmissions during its flight period. We can see that the system rates of the proposed schemes dominate the conventional schemes in which the actor-critic-based method can be approximately as good as the POMDP-based method, and the two proposed schemes can provide a system data rate 10% higher than the Myopic approaches. Next, we compare the energy efficiency of the schemes with respect to mean value of harvested energy in Figure 8. In this study, we aim to efficiently utilize the solar harvested energy of the UAV in the long-term operation. When the transmission capacity is full during the serving time, the rest of harvested energy can also be stored for the mobility capacity portion to support the UAV's flight. Moreover, the overflow energy of the battery is considered as the wasted energy consumption of the system. For that reason, in the simulation, the energy consumption is calculated as the total harvested energy during the UAV's operation. All schemes with each mean value of harvested energy, in Figure 8, have the same total amount of energy consumption in $N = 2 \times 10^5$ time slots. In the paper, energy efficiency is defined as the sum data rate over the total harvested energy during the UAV's operation. As a consequence, the curves in Figure 8 can be interpreted as the sum-rate according to energy consumption.

In order to explore the behavior in terms of transmission power by the UAV, in Figure 9, we plot the statistics of the actions in the POMDP scheme, the actor-critic scheme, the Myopic-NOMA scheme, and the Myopic-OMA scheme over 200,000 time slots. The notation $TM - LVx$ represents the transmission mode with a level of LVx where $LVx \in \{LV1, LV2, ..., LV5\}$ is the level of transmission energy. We can see in Figure 9 that the Myopic-NOMA scheme and the Myopic-OMA scheme tend to choose the highest transmission power for the purpose of maximizing the instant reward. Obviously, the statistics of selected actions in these myopic schemes are similar, but the achievable reward of the NOMA scheme is higher than that of the OMA scheme owing to the effective utilization of the NOMA technique. However, due to the limitation on harvested energy, using too much energy in a time slot

may cause the energy shortage, in which the UAV has to stay silent for many future time slots. This will lower the data rate of the system. On the other hand, simultaneously assigning an appropriate amount of transmission energy can give the UAV more chances to stay active and transmit data to the GUs under the environment dynamics, such that a maximum long-term data rate can be guaranteed.

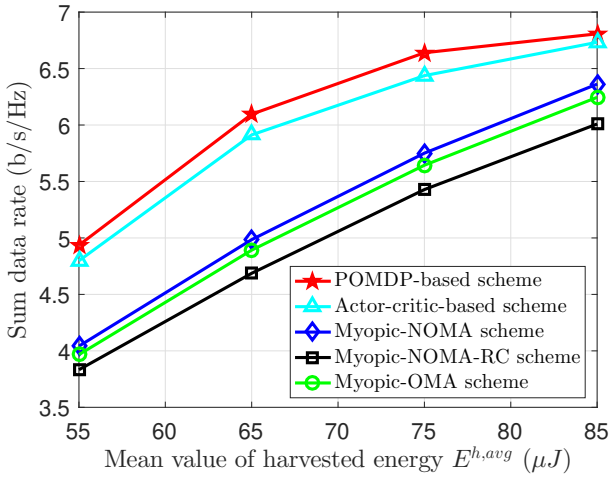

Figure 7. The sum data rate according to the mean value of harvested energy.

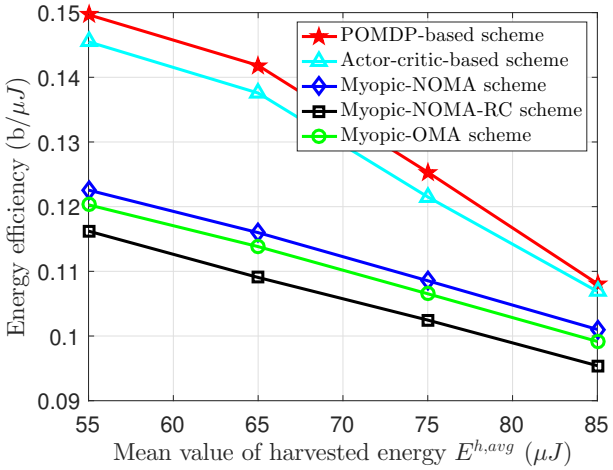

Figure 8. The energy efficiency according to the mean value of harvested energy.

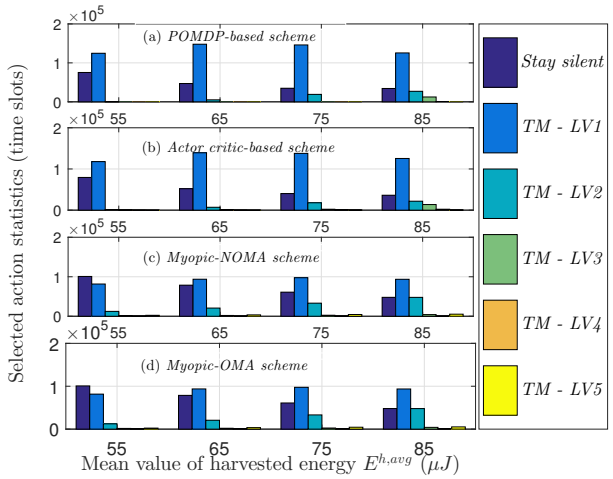

Figure 9. Statistics for the selected actions of the proposed schemes according to the mean value of harvested energy.

In Figure 10, we plot the sum data rate according to different values of caching capacity. The curves show that the system performance is enhanced if the UAV has a higher caching capacity. Obviously, with a larger value of C_F, the UAV can store more items from the LS, and then, the probability that the GUs' requests are in the cached content of the UAV will increase, which leads to the higher data transmission rate. On the other hand, we can see that the higher P_{mm} also brings higher performance of the system. The reason is that the GUs will more frequently request their own items of interest during the time slots.

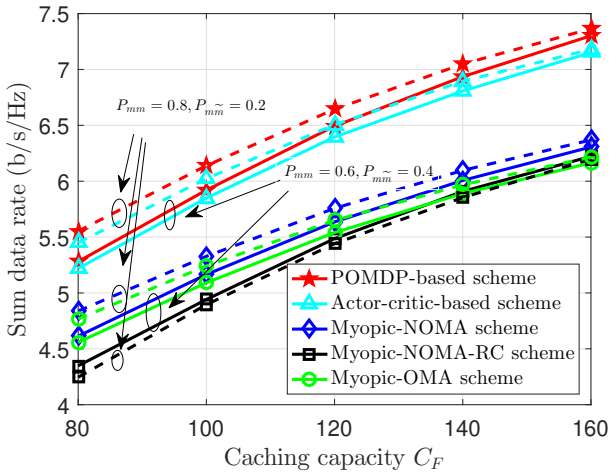

Figure 10. The sum data rate with respect to caching capacity.

Figures 11 and 12, respectively, show the impact of noise variance at the GUs and the effect of the altitude of the UAV on the system reward. We can see that system performance notably declined as the noise power at the ground users (as well as the altitude of the UAV) grew. In order to explain this, noise power will lower the throughput for each GU's data recipient, and meanwhile, a farther distance between F and the GUs will increase path loss during data transmissions.

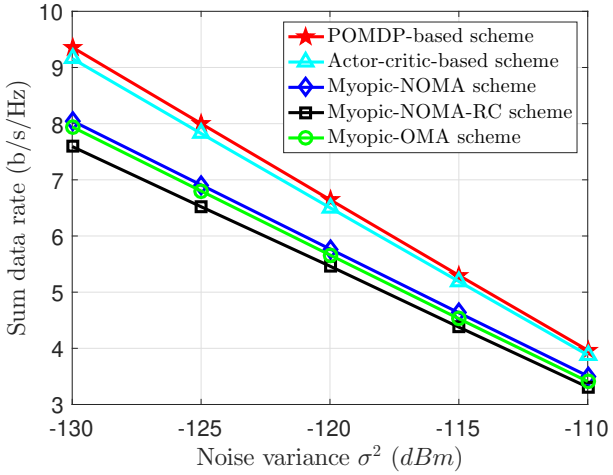

Figure 11. The sum data rate under different values of noise variance.

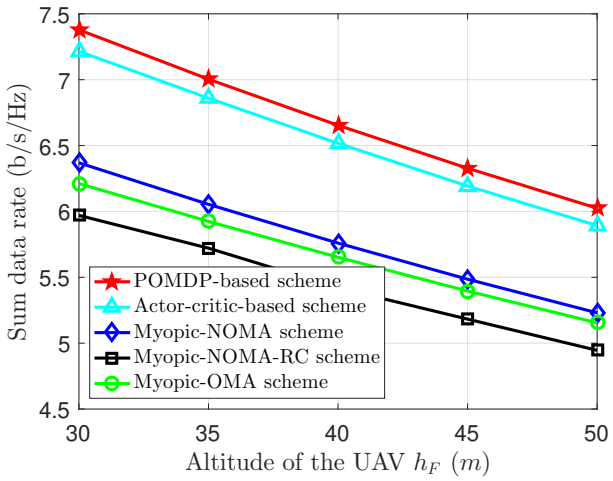

Figure 12. The sum data rate versus various values of the altitude of the UAV.

Finally, we further investigated the joint effect of both the number of items, K, in the library, and caching capacity C_F in the UAV on the system data rate. Figure 13 indicates that the system reward will increase with an increment in the ratio of C_F over K. For example, if the number of items is $K = 300$, the data rate of the system will go up when increasing caching capacity C_F. Furthermore, the results of the POMDP-based and actor-critic schemes are superior to the Myopic-NOMA scheme. The reason is that the proposed POMDP scheme exploits prior information on the harvested energy distribution and on the request model of the GUs, and then, it calculates the possible situations and corresponding probabilities. The actor-critic method can explore the information from interacting directly with the environment, and it then learns the optimal policy through trial and error. Consequently, the next state of the system can be predicted, and the UAV can efficiently allocate transmission power for the GUs based on NOMA and caching technologies under the long-term operation considerations. On the other hand, the presented numerical results validate the effectiveness of the proposed approaches through various network parameters in this paper.

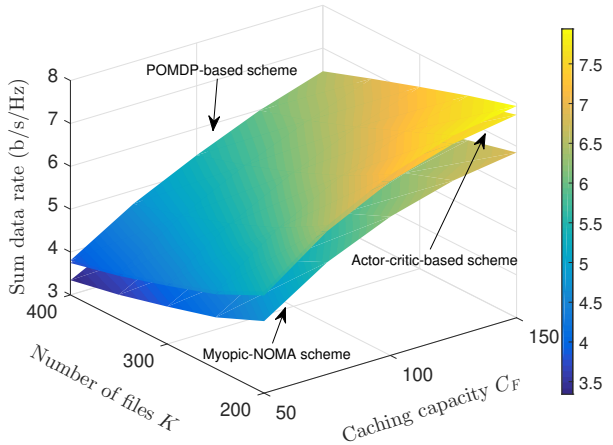

Figure 13. The sum data rate according to different values of K and C_F.

6. Conclusions

In this paper, we investigated non-orthogonal multiple access with data caching for UAV-enabled downlink transmissions under constraints on energy and the caching capacity in the solar-powered UAV. The two innovative approaches, based on POMDP and the actor-critic frameworks, were proposed for a joint cache scheduling and resource allocation issue to maximize the long-term data rate of the system in cases with and without prior information of the energy arrival distribution. The optimal policy can be obtained by using the two proposed schemes, such that the UAV can efficiently use harvested solar

energy to transmit data to a group of ground users that need a service fulfilling their item requests. Eventually, the numerical results via MATLAB simulations verified the superiority of the proposed schemes, compared to baseline alternatives in which the context under long-term data rate maximization is not taken into account under diverse network conditions. The shortcoming of this work is that the high formulation complexity and computational complexity may be considerably imposed on multi-UAV systems, where the coverage region for data communications is extended by deploying multiple UAVs to meet surging data transmission demands. In this regard, a deep reinforcement learning framework can be one of the promising solutions to the optimization issues in large state and space UAV systems for 5G and beyond 5G, which is considered in our future research directions.

Author Contributions: Conceptualization, P.D.T., T.N.K.H. and I.K.; formal analysis, P.D.T.; methodology, P.D.T., T.N.K.H. and H.T.H.G.; validation, P.D.T., H.T.H.G. and I.K.; writing–original draft preparation, P.D.T.; writing–review and editing, H.T.H.G. and I.K.; supervision, T.N.K.H. and I.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Research Foundation of Korea (NRF) grant through the Korean Government (MSIT) under Grant NRF-2018R1A2B6001714.

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

References

1. Osseiran, A.; Boccardi, F.; Braun, V.; Kusume, K.; Marsch, P.; Maternia, M.; Queseth, O.; Schellmann, M.; Schotten, H.; Taoka, H.; et al. Scenarios for 5G mobile and wireless communications: The vision of the METIS project. *IEEE Commun. Mag.* **2014**, *52*, 6–35. [CrossRef]
2. Zeng, Y.; Zhang, R.; Lim, T.J. Wireless communications with unmanned aerial vehicles: Opportunities and challenges. *IEEE Commun. Mag.* **2016**, *54*, 36–42. [CrossRef]
3. Project Loon. Available online: https://x.company/loon/ (accessed on 15 July 2017).
4. Chandrasekharan, S.; Gomez, K.; Al-Hourani, A.; Kandeepan, S.; Rasheed, T.; Goratti, L.; Reynaud, L.; Grace, D.; Bucaille, I.; Wirth, T.; et al. Designing and implementing future aerial communication networks. *IEEE Commun. Mag.* **2016**, *54*, 26–34. [CrossRef]
5. Chamola, V.; Kotesh, P.; Agarwal, A.; Naren; Gupta, N.; Guizani, M. A Comprehensive Review of Unmanned Aerial Vehicle Attacks and Neutralization Techniques. *Ad Hoc Netw.* **2020**, 102324. [CrossRef]
6. Bor-Yaliniz, I.; Yanikomeroglu, H. The new frontier in RAN heterogeneity: Multi-tier drone-cells. *IEEE Commun. Mag.* **2016**, *54*, 48–55. [CrossRef]
7. Lyu, J.; Zeng, Y.; Zhang, R. UAV-aided offloading for cellular hotspot. *IEEE Trans. Wirel. Commun.* **2018**, *6*, 3988–4001. [CrossRef]
8. Mozaffari, M.; Saad, W.; Bennis, M.; Debbah, M. Efficient deployment of multiple unmanned aerial vehicles for optimal wireless coverage. *IEEE Commun. Lett.* **2016**, *20*, 1647–1650. [CrossRef]
9. Merwaday, A.; Guvenc, I. UAV assisted heterogeneous networks for public safety communications. In Proceedings of the 2015 IEEE Wireless Communications and Networking Conference Workshops (WCNCW), New Orleans, LA, USA, 9–12 March 2015; pp. 329–334.
10. Mozaffari, M.; Saad, W.; Bennis, M.; Debbah, M. Mobile Internet of Things: Can UAVs provide an energy-efficient mobile architecture? In Proceedings of the 2016 IEEE Global Communications Conference (GLOBECOM), Washington, DC, USA, 4–8 December 2016; pp. 1–6.
11. Soorki, M.N.; Mozaffari, M.; Saad, W.; Manshaei, M.H.; Saidi, H. Resource allocation for machine-to-machine communications with unmanned aerial vehicles. In Proceedings of the 2016 IEEE Globecom Workshops (GC Wkshps), Washington, DC, USA, 4–8 December 2016; pp. 1–6.
12. Alsaeedy, A.A.R.; Chong, E.K.P. 5G and UAVs for Mission-Critical Communications: Swift Network Recovery for Search-and-Rescue Operations. *Mob. Netw. Appl.* **2020**, *25*, 2063–2081. [CrossRef]
13. Deruyck, M.; Wyckmans, J.; Joseph, W.; Martens, L. Designing UAV-aided emergency networks for large-scale disaster scenarios. *J. Wirel. Commun. Netw.* **2018**, *2018*, 79. [CrossRef]
14. Saito, Y.; Kishiyama, Y.; Benjebbour, A.; Nakamura, T.; Li, A.; Higuchi, K. Non-Orthogonal Multiple Access (NOMA) for Cellular Future Radio Access. In Proceedings of the 2013 IEEE 77th Vehicular Technology Conference (VTC Spring), Dresden, Germany, 2–5 June 2013; pp. 1–5.

15. Chen, Z.; Ding, Z.; Dai, X.; Zhang, R. An optimization perspective of the superiority of noma compared to conventional OMA. *IEEE Trans. Signal Process.* **2017**, *65*, 5191–5202. [CrossRef]
16. Xu, P.; Cumanan, K. Optimal power allocation scheme for non-orthogonal multiple access with α-fairness. *IEEE J. Sel. Areas Commun.* **2017**, *35*, 2357–2369. [CrossRef]
17. Sharma, P.K.; Kim, D.I. UAV-enabled downlink wireless system with non-orthogonal multiple access. In Proceedings of the 2017 IEEE Globecom Workshops (GC Wkshps), Singapore, 4–8 December 2017; pp. 1–6.
18. Sohail, M.F.; Leow, C.Y.; Won, S. Non-Orthogonal Multiple Access for Unmanned Aerial Vehicle Assisted Communication. *IEEE Access* **2018**, *6*, 22716–22727. [CrossRef]
19. Liu, X.; Wang, J.; Zhao, N.; Chen, Y.; Zhang, S.; Ding, Z.; Yu, F.R. Placement and power allocation for NOMA-UAV networks. *IEEE Wirel. Commun. Lett.* **2019**. [CrossRef]
20. Zhao, N.; Pang, X.; Li, Z.; Chen, Y.; Li, F.; Ding, Z.; Alouini, M.-S. Joint trajectory and precoding optimization for UAVassisted NOMA networks. *IEEE Trans. Commun.* **2019**. [CrossRef]
21. Wang, W.; Tang, J.; Zhao, N.; Liu, X.; Zhang, X.Y.; Chen, Y.; Qian, Y. Joint Precoding Optimization for Secure SWIPT in UAV-Aided NOMA Networks. *IEEE Trans. Commun.* **2020**, *68*, 5028–5040. [CrossRef]
22. Zhao, N.; Cheng, F.; Yu, F.R.; Tang, J.; Chen, Y.; Gui, G.; Sari, H. Caching UAV Assisted Secure Transmission in Hyper-Dense Networks Based on Interference Alignment. *IEEE Trans. Commun.* **2018**, *66*, 2281–2294. [CrossRef]
23. Chen, M.; Mozaffari, M.; Saad, W.; Yin, C.; Debbah, M.; Hong, C.S. Caching in the sky: Proactive deployment of cache-enabled unmanned aerial vehicles for optimized quality-of experience. *IEEE J. Sel. Areas Commun.* **2017**, *35*, 1046–1061. [CrossRef]
24. Chen, M.; Saad, W.; Yin, C. Liquid State Machine Learning for Resource and Cache Management in LTE-U Unmanned Aerial Vehicle (UAV) Networks. *IEEE Trans. Wirel. Commun.* **2019**, *18*, 1504–1517. [CrossRef]
25. Xu, X.; Zeng, Y.; Guan, Y.L.; Zhang, R. Overcoming Endurance Issue: UAV-Enabled Communications With Proactive Caching. *IEEE J. Sel. Areas Commun.* **2018**, *36*, 231–1244. [CrossRef]
26. Cheng, F.; Gui, G.; Zhao, N.; Chen, Y.; Tang, J.; Sari, H. UAV-Relaying-Assisted Secure Transmission With Caching. *IEEE Trans. Commun.* **2019**, *67*, 3140–3153. [CrossRef]
27. Cui, X.M.; Wang, W.J.; Fang, Z.P. Present situation and some problems analysis of small-size unmanned air vehicles. *Flight Dyn.* **2005**, *23*, 1.
28. Franco, C.D.; Buttazzo, G. Energy-aware coverage path planing of UAVs. In Proceedings of the 2015 IEEE International Conference on Autonomous Robot Systems and Competitions, Vila Real, Portugal, 8–10 April 2015.
29. Zeng, Y.; Zhang, R. Energy-efficient UAV communication with trajectory optimization. *IEEE Trans. Wirel. Commun.* **2017**, *16*, 3747–3760. [CrossRef]
30. Hua, M.; Li, C.; Huang, Y.; Yang, L. Throughput maximization for UAV-enabled wireless power transfer in relaying system. In Proceedings of the 2017 9th International Conference on Wireless Communications and Signal Processing (WCSP), Nanjing, China, 11–13 October 2017; pp. 1–5.
31. Yang, L.; Chen, J.; Hasna, M.O.; Yang, H. Outage Performance of UAV-Assisted Relaying Systems With RF Energy Harvesting. *IEEE Commun. Lett.* **2018**, *22*, 2471–2474. [CrossRef]
32. Oettershagen, P.; Melzer, A.; Mantel, T.; Rudin, K.; Stastny, T.; Wawrzacz, B.; Hinzmann, T.; Alexis, K.; Siegwart, R. Perpetual flight with a small solar-powered UAV: Flight results, performance analysis and model validation. In Proceedings of the 2016 IEEE Aerospace Conference, Big Sky, MT, USA, 5–12 March 2016; pp. 1–8.
33. Lee, J.-S.; Yu, K.-H. Optimal path planning of solar-powered UAV using gravitational potential energy. *IEEE Trans. Aerosp. Electron. Syst.* **2017**, *53*, 1442–1451. [CrossRef]
34. Sun, Y.; Ng, D.W.K.; Xu, D.; Dai, L.; Schober, R. Resource allocation for solar powered UAV communication systems. In Proceedings of the 2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Kalamata, Greece, 25–28 June 2018; pp. 1–5.
35. Sun, Y.; Xu, D.; Ng, D.W.K.; Dai, L.; Schober, R. Optimal 3D-Trajectory Design and Resource Allocation for Solar-Powered UAV Communication Systems. *IEEE Trans. Commun.* **2019**, *67*, 4281–4298. [CrossRef]
36. Ouyang, J.; Che, Y.; Xu, J.; Wu, K. Throughput Maximization for Laser-Powered UAV Wireless Communication Systems. In Proceedings of the 2018 IEEE International Conference on Communications Workshops (ICC Workshops), Kansas City, MO, USA, 20–24 May 2018; pp. 1–6.

37. Wu, H.; Tao, X.; Zhang, N.; Shen, X. Cooperative UAV Cluster-Assisted Terrestrial Cellular Networks for Ubiquitous Coverage. *IEEE J. Sel. Areas Commun.* **2018**, *36*, 2045–2058. [CrossRef]
38. Liang, J.; Liang, Q. RF Emitter Location Using a Network of Small Unmanned Aerial Vehicles (SUAVs). In Proceedings of the 2011 IEEE International Conference on Communications (ICC), Kyoto, Japan, 5–9 June 2011; pp. 1–6. [CrossRef]
39. Whiting, S. Radio-Frequency Transmitter Geolocation Using Non-Ideal Received Signal Strength Indicators (2018). All Graduate Theses and Dissertations. 7038. Available online: https://digitalcommons.usu.edu/etd/7038 (accessed on 28 April 2018).
40. Zeng, Y.; Zhang, R.; Lim, T.J. Throughput Maximization for UAV-Enabled Mobile Relaying Systems. *IEEE Trans. Commun.* **2016**, *64*, 4983–4996 . [CrossRef]
41. Hu, D.; Zhang, Q.; Li, Q.; Qin, J. Joint Position, Decoding Order, and Power Allocation Optimization in UAV-Based NOMA Downlink Communications. *IEEE Syst. J.* **2020**, *14*, 2949–2960. [CrossRef]
42. Lee, P.; Eu, Z.A.; Han, M.; Tan, H.-P. Empirical modeling of a solar-powered energy harvesting wireless sensor node for time-slotted operation. In Proceedings of the 2011 IEEE Wireless Communications and Networking Conference, Cancun, Quintana Roo, Mexico, 28–31 March 2011; pp. 179–184.
43. Cao, X.-R.; Guo, X. Partially Observable Markov Decision Processes With Reward Information: Basic Ideas and Models. *IEEE Trans. Autom. Control* **2007**, *52*, 677–681. [CrossRef]
44. Bertsekas, D.P. *Dynamic Programming and Optimal Control*, 2nd ed.; Athena Scientific: Nashua, NH, USA, 2000.
45. Konda, V.R.; Tsitsiklis, J.N. Actor-critic algorithms. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2000.
46. Wang, W.; Kwasinski, A.; Niyato, D.; Han, Z. A Survey on Applications of Model-Free Strategy Learning in Cognitive Wireless Networks. *IEEE Commun. Surv. Tutor.* **2016**, *18*, 1717–1757. [CrossRef]
47. Sutton, R.S.; Barto, A.G. *Reinforcement Learning: An Introduction*; MIT Press: Cambridge, MA, USA, 1998.
48. Challita, U.; Saad, W. Network Formation in the Sky: Unmanned Aerial Vehicles for Multi-Hop Wireless Backhauling. In Proceedings of the GLOBECOM 2017-2017 IEEE Global Communications Conference, Singapore, 4–8 December 2017; pp. 1–6. [CrossRef]