

# Power-Efficient Trajectory Adjustment and Temporal Routing for Multi-UAV Networks

Y.-W. Peter Hong<sup>1</sup>, *Senior Member, IEEE*, Ray-Hsiang Cheng,  
Yu-Cheng Hsiao, and Jang-Ping Sheu<sup>2</sup>, *Fellow, IEEE*

**Abstract**—This work proposes power-efficient trajectory adjustment and temporal routing algorithms for a network of fixed-wing unmanned aerial vehicles (UAV) deployed to gather data from underlying sensors in the field. To stay afloat, each UAV follows a circular trajectory above its responsible service area with a radius that can be adjusted to reduce its propulsion energy consumption. The data gathered by the UAVs are sent to the data-gathering node using multihop transmissions over other circulating UAVs. First, given the multihop transmission paths from all UAVs to the data-gathering node, power-efficient flight-radius adjustment strategies are proposed based on the total power minimization and lifetime maximization criteria while maintaining connectivity over the multihop paths. Then, by establishing the relationship between routing in UAV networks and that in general temporal graphs, we propose a power-efficient (PE) temporal path algorithm based on the minimization of the total pair-wise flight power consumption among consecutive UAVs on the path. Finally, we propose an iterative procedure to refine the initial phases of the UAVs' circular trajectories to further improve the power efficiency. Computer simulations are provided to demonstrate the effectiveness of the proposed schemes in terms of both total power minimization and lifetime maximization.

**Index Terms**—Unmanned aerial vehicle, routing, energy conservation, mobile communication, wireless sensor network.

## I. INTRODUCTION

UNMANNED aerial vehicles (UAVs) have been widely adopted in both civilian and military applications, such as real-time surveillance [1], wildfire monitoring [2], search and rescue operations [3], remote sensing [4], disaster recovery [5], [6], etc. Their application scenarios continue to expand as UAVs become more powerful in terms of sensing, communication, and computation capabilities. More recently, the use of UAVs in wireless communications has also attracted much

attention due to the UAVs' deployment flexibility and ability to provide better transmission channels through increased line-of-sight (LoS) probabilities. In cellular applications, UAVs have been adopted as wireless relays or mobile base-stations to achieve temporary coverage or data-offloading, even in hostile areas [7], [8]. In wireless sensor network (WSN) applications, UAVs have also been used as mobile sink nodes to efficiently gather information from ground sensors [9], [10]. For more complex or large-scale tasks, multitasking and cooperation among multiple UAVs are often necessary to accomplish the application goals. In these scenarios, it is important to enable efficient air-to-air direct or multihop transmissions, which can be challenging in highly dynamic UAV networks. This work focuses specifically on the use of multiple UAVs for data-gathering and feedback communications in WSNs and Internet-of-Things.

Most works in the literature on the use of UAVs for data-gathering in WSNs consider either the single-UAV path planning (or trajectory design) problem [9]–[12], where the goal is to determine the most efficient flight path for a single UAV to traverse the entire sensor field, or the multi-UAV placement problem [13]–[17], where UAVs work collaboratively to monitor the sensor field. Specifically, for the single-UAV path planning problem, [9] proposed a joint UAV trajectory design and sensors' cluster head selection scheme that takes into consideration the sensors' energy consumption, bit error rate, and the UAV's flight time. Reference [10] proposed the joint optimization of the UAV's trajectory and sensors' wakeup schedules to minimize the maximum energy consumption of all sensors subject to constraints on the reliability of the data collection. Reference [11] considered the problem of age-optimal path planning and proposed two age-optimal trajectories to achieve the task. Reference [12] determined the optimal UAV's flight speed and sensors' transmit powers to minimize the UAV's total flight time from start to finish on a simple linear network. In multi-UAV networks, UAVs are often deployed at fixed positions above their responsible service areas, and the gathered information is transmitted through multiple hops to the data-gathering node. In terms of UAV placement, [13] examined the optimal placement of multiple UAVs for data collection subject to constraints on the sensor coverage and the existence of connected paths from all UAVs to the data-gathering node. Reference [14] proposed a dynamic UAV placement policy that maximizes the value of the acquired sensor information in each time slot subject to constraints on the connectivity between UAVs and the

Manuscript received May 4, 2020; revised June 26, 2020; accepted July 29, 2020. Date of publication August 4, 2020; date of current version November 20, 2020. This work was supported in part by the Ministry of Science and Technology, Taiwan, under Grant MOST 107-2634-F-007-005, Grant 108-2634-F-007-004, and Grant 109-2634-F-007-015. This article was presented in part at the IEEE International Conference on Communications, May 2019. The associate editor coordinating the review of this article and approving it for publication was E. Ayanoglu. (*Corresponding author: Y.-W. Peter Hong.*)

Y.-W. Peter Hong is with the Institute of Communications Engineering, National Tsing Hua University, Hsinchu 30013, Taiwan, and also with the MOST Joint Research Center for AI Technology and AII Vista Healthcare, Taipei 10617, Taiwan (e-mail: ywhong@ee.nthu.edu.tw).

Ray-Hsiang Cheng and Jang-Ping Sheu are with the Department of Computer Science, National Tsing Hua University, Hsinchu 30013, Taiwan (e-mail: qazwsx31243@gmail.com; sheujp@cs.nthu.edu.tw).

Yu-Cheng Hsiao is with the Department of Electrical Engineering, National Tsing Hua University, Hsinchu 30013, Taiwan (e-mail: h411235@gmail.com). Digital Object Identifier 10.1109/TGCN.2020.3014101

reliability of the sensors' communication links. Reference [15] examined the joint optimization of the UAVs' trajectories and the sensors' wakeup schedules by taking into consideration the tradeoff between the transmission energy of ground sensors and the UAVs' propulsion energy. Furthermore, by treating UAVs as pure relays between a source and a destination, [16] determined the UAVs' placement by maximizing the end-to-end signal-to-noise ratio for both multihop and dual-hop relaying scenarios, and [17] proposed to jointly determine the UAVs' trajectories and transmission powers to maximize the end-to-end throughput of a predetermined multihop route. Notice that, in [16] and [17], UAVs are treated as pure relays and are not responsible for data-gathering from a field of ground sensors.

Determining optimal multihop routes in UAV networks can be a challenging task due to the UAVs' high mobility and intermittent connectivity. Several routing protocols, e.g., [18]–[24], were proposed in the literature to cope with these challenges. Among the class of proactive routing protocols, [18] proposed an extension of the well-known optimized link state (OLSR) protocol that takes into consideration the relative speed between UAVs. A speed-aware predictive-OLSR protocol was proposed, where the expected transmission count (ETX) metric used for multi-point relay selection is weighted by the relative speed of the communicating UAVs. Similarly, [19] proposed a mobility and load-aware OLSR algorithm that takes into account the packet load at each UAV to establish more stable routes with less congestion, and [20] further considered the queuing delay as a routing metric to balance traffic load and reduce end-to-end delay. Moreover, [21] proposed an improved B.A.T.M.A.N. protocol by leveraging a prediction of future trajectories of the UAVs in the routing protocol to avoid unexpected route breaks and packet loss. Among the class of reactive routing protocols, where routes are found on-demand whenever packets need to be sent, [22] proposed a reactive-greedy-reactive (RGR) routing protocol that combines the use of ad-hoc on-demand distance vector and greedy geographic forwarding protocols. Reference [23] proposed a modified-RGR that selects UAVs with high link reliability to forward the packets. [24] proposed a robust and reliable predictive (RARP) routing protocol using a hybrid unicasting and geocasting scheme. This scheme utilizes directional transmission and dynamic angle adjustment to reduce the route setup time and extend the average path lifetime. Recent surveys of UAV routing algorithms can be found in [25] and [26].

Notice that most of the above works consider the use of rotary-wing UAVs that can choose to either cruise or hover above fixed positions with less constraints on their movement. This provides more flexibility in the UAV path planning and placement problems, and also reduces the challenge due to frequent topology changes in routing problems. However, the lifetime of rotary-wing UAVs is typically short and cannot maintain usage over a long time period. Hence, we consider in this work the use of fixed-wing UAVs that typically consume less power and carry larger batteries. However, for fixed-wing UAVs to stay afloat, they must maintain constant movement with strict speed and maneuvering constraints, causing the network topology to vary rapidly over time. The continuous

change in network topology makes routing over such networks especially challenging. Existing works on UAV routing, as mentioned above, rely on mobility prediction that is difficult to perform accurately in such a highly dynamic environment.

The main objective of this work is to propose power-efficient trajectory adjustment and temporal routing algorithms for a network of fixed-wing UAVs used for data-gathering in WSNs. The UAVs are assumed to follow circular flight trajectories to maintain coverage over their responsible service areas. The constant movement of UAVs results in a so-called temporal graph [27] where the connectivity among UAVs exist only at certain time instants. A transmission path over the temporal graph can be represented by a causal sequence of temporal edges (i.e., edges that exist only at particular time instants) from the source to the destination. Given predetermined transmission paths between the UAVs and the data-gathering node, we first propose two flight-radius adjustment schemes based on the total power minimization and lifetime maximization criteria, respectively, subject to constraints on the existence of the predetermined paths. The total power minimization problem is convex and can be solved by using off-the-shelf solvers such as CVX [28], whereas the lifetime maximization problem is solved using a successive convex approximation (SCA) approach [29]. Then, by treating the minimum pairwise power consumption as the cost of each edge, we propose a power-efficient (PE) temporal path algorithm that is able to determine the optimal path between each UAV and the data-gathering node by minimizing the accumulated cost of edges within the path. Notice that edges in a temporal path must satisfy a causality constraint and, thus, cannot be solved by conventional shortest path algorithms. Finally, we propose an iterative procedure to refine the phases of the UAVs' circular trajectories as well as their flight-radii and temporal paths to further improve the power efficiency. Numerical simulations are provided to demonstrate the effectiveness of our proposed trajectory adjustment strategies in terms of total power consumption and network lifetime, respectively.

The design challenges of fixed-wing UAVs have also been examined recently in different applications. For example, [30] determined the optimal flight trajectory for a fixed-wing UAV to maximize the energy efficiency of communication between the UAV and a single ground terminal, taking into consideration the UAV's propulsion energy consumption. The optimal flight speed and radius were determined for a practical circular trajectory around the ground terminal. Reference [31] further examined the tradeoff between the transmission energy of the ground terminal and the propulsion energy of the UAV. Moreover, [32] considered the use of a circulating fixed-wing UAV as the relay between two ground terminals. A variable rate decode-and-forward protocol was proposed to enhance the end-to-end information rate. [33] considered the use of circulating UAVs to cover ground users and proposed a dynamic UAV placement and radius adjustment algorithm to increase the probability of end-to-end link connections. However, the temporal routing and the impact of trajectory control on the energy consumption were not discussed.

The remainder of this article is organized as follows. In Section II, we describe the system model and establish its

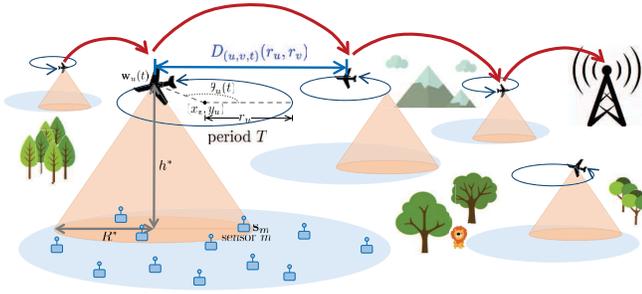


Fig. 1. Illustration of a multihop UAV network for monitoring and data-gathering.

relation with temporal graphs. In Section III, two flight-radius adjustment algorithms are proposed, namely, the total power minimization and the lifetime maximization algorithms. In Sections IV and V, the PE temporal path and the iterative phase readjustment algorithms are proposed to further reduce the power consumption. Finally, we demonstrate the effectiveness of the proposed schemes in Section VI, and conclude in Section VII.

## II. SYSTEM MODEL

Let us consider a multi-UAV network consisting of  $N$  UAVs that are deployed to gather data from  $M$  ground sensors, denoted by the set  $\mathcal{S} \triangleq \{1, 2, \dots, M\}$ . Each UAV is assumed to follow a circular flight-trajectory to maintain coverage over a subset of sensors, as illustrated in Fig. 1. Different from rotary-wing UAVs that can be deployed at fixed positions, fixed-wing UAVs must maintain constant movement in order to stay afloat. The circular trajectory allows UAVs to maintain the required movement while covering their responsible service areas. The time required for each UAV to complete one flight cycle around the circular trajectory (i.e., the time required to gather one round of sensor data) is assumed to be  $T$ , which depends on the data generation frequency of the underlying sensors. Moreover, we assume that all UAVs fly at the same altitude  $h^*$ , which is chosen to achieve maximum ground coverage under the air-to-ground channel model in [34]. The maximum radius of the ground coverage is denoted by  $R^*$ , which can be derived following the procedures in [35].

Let  $r_u$  be the radius of the circular trajectory of UAV  $u$  and let  $\mathbf{c}_u \triangleq (x_u, y_u)$  be the horizontal coordinates of the trajectory center. In this case, the location of UAV  $u$  at time  $t$  can be represented by

$$\mathbf{w}_u(t) \triangleq (x_u + r_u \cos \theta_u(t), y_u + r_u \sin \theta_u(t)) \quad (1)$$

where  $\theta_u(t)$  is the phase of UAV  $u$ 's position at time  $t$ . By letting  $\theta_{u,0}$  be the initial angular offset at time  $t = 0$ , we have

$$\theta_u(t) = \theta_{u,0} + \frac{2\pi t}{T} \quad (2)$$

if the UAV is flying in the counter-clockwise direction and

$$\theta_u(t) = \theta_{u,0} - \frac{2\pi t}{T} \quad (3)$$

if the UAV is flying in the clockwise direction.

In practice, the radius and the center of each UAV's circular trajectory may be constrained by the positions of the sensors that it covers. For example, suppose that UAV  $u$  is assigned to gather information from a subset of sensors  $\mathcal{S}_u \subset \mathcal{S}$ . Under ground coverage radius  $R^*$ , UAV  $u$  is able to successfully receive data from sensor  $m$  at time  $t$  if the ground sensor's horizontal coordinates  $\mathbf{s}_m \triangleq (s_{m1}, s_{m2})$  satisfy  $\|\mathbf{s}_m - \mathbf{w}_u(t)\| \leq R^*$ . Hence, we say that sensor  $m$  is covered by UAV  $u$  if there exists a time  $t$  within each flight cycle that allows the UAV to successfully receive data from sensor  $m$ , that is, if  $\min_{t \in [0, T]} \|\mathbf{s}_m - \mathbf{w}_u(t)\| \leq R^*$ . Hence, to cover all sensors in subset  $\mathcal{S}_u$ , the flight radius  $r_u$  and trajectory center  $\mathbf{c}_u$  must satisfy  $r_u - R^* \leq \|\mathbf{s}_m - \mathbf{c}_u\| \leq r_u + R^*$ , for all  $m \in \mathcal{S}_u$ . That is, given  $\mathbf{c}_u$ , the flight radius  $r_u$  is confined to the interval

$$\begin{aligned} \bar{r}_{u,\min} &\triangleq \max \left\{ \max_{m \in \mathcal{S}_u} \|\mathbf{s}_m - \mathbf{c}_u\| - R^*, \bar{r}_{\text{phy}} \right\} \leq r_u \\ &\leq \min_{m \in \mathcal{S}_u} \|\mathbf{s}_m - \mathbf{c}_u\| + R^* \triangleq \bar{r}_{u,\max}, \end{aligned} \quad (4)$$

where  $\bar{r}_{\text{phy}}$  is the smallest circulating radius that is physically allowed by a fixed-wing UAV. In this work, we focus on the optimization of the flight-radius and multihop transmission paths while assuming that the UAVs' trajectory centers (i.e.,  $\mathbf{c}_u$ , for all  $u$ ) and the sensors' association (i.e., the subsets  $\mathcal{S}_u$ , for all  $u$ ) are given, e.g., following a standard  $k$ -means clustering approach. Moreover, we assume that the transmission time required for each sensor to upload its data to its associated UAV is negligible. That is, each sensor's transmission can be completed instantaneously upon entering the coverage of its associated UAV. We argue that this assumption is reasonable in sensor network applications where the data generated by each sensor within a flight cycle is usually small. We understand that, in practice, the transmission of each sensor may occupy a finite duration and, thus, must be carefully scheduled in a time-division multiple access (TDMA) fashion. However, this requires further study that is beyond the scope of the current work.

Suppose that the data gathered by the UAVs are forwarded periodically every  $T$  to the data-gathering node through multihop paths. We say that a connection from UAV  $u$  to UAV  $v$  exists at time  $t$  if UAV  $v$  is within the transmission radius of UAV  $u$  (i.e.,  $\delta_u$ ) at this time, that is, if

$$D_{(u,v,t)}(r_u, r_v) \leq \delta_u, \quad (5)$$

where

$$\begin{aligned} D_{(u,v,t)}(r_u, r_v) &\triangleq \left\| (x_u + r_u \cos \theta_u(t), y_u + r_u \sin \theta_u(t)) \right. \\ &\quad \left. - (x_v + r_v \cos \theta_v(t), y_v + r_v \sin \theta_v(t)) \right\| \end{aligned} \quad (6)$$

is the distance between UAVs  $u$  and  $v$  at time  $t$ . This connection may not be bidirectional since the transmission radius may vary for different UAVs. In practice, the transmission between UAVs must occur over a finite slot duration  $\tau$ . However, to simplify our discussions, we first assume that  $\tau$  is negligible and, thus, the transmission can occur instantaneously at the connection time  $t$ . Then, based on our proposed solution, the connectivity over a finite slot duration  $\tau$  in practice can be

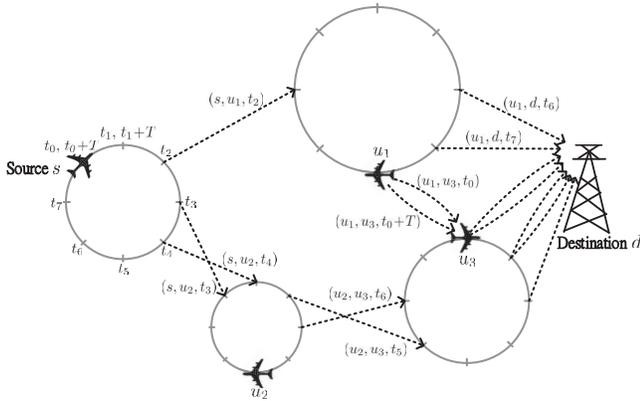


Fig. 2. Example of a temporal graph generated by the UAV network.

guaranteed by a simple fine-tuning of the flight radius, as to be discussed in Section III. Moreover, due to UAVs' mobility, the connections between UAVs may exist only intermittently within each cycle, resulting in a so-called *temporal graph* [27], i.e., a graph with edges labeled by their occurrence times.

Specifically, let  $\mathcal{G} = (\mathcal{V}, \mathcal{E}_{[t_0, t_0+\lambda]})$  be a directed temporal graph with  $\mathcal{V}$  being the set of vertices (or UAVs) and  $\mathcal{E}_{[t_0, t_0+\lambda]}$  being the set of temporal edges that exist during time interval  $[t_0, t_0+\lambda]$ . An edge  $e \in \mathcal{E}_{[t_0, t_0+\lambda]}$  is represented by a tuple  $(u, v, t)$ , where  $u, v \in \mathcal{V}$  and  $t \in [t_0, t_0+\lambda]$  is the time instant at which UAV  $u$  can transmit to UAV  $v$ . In fact, an edge  $e = (u, v, t)$  exists only if  $D_{(u,v,t)}(r_u, r_v) \leq \delta_u$  (i.e., if UAV  $v$  is in the transmission range of UAV  $u$  at time  $t$ ) and  $t \in [t_0, t_0+\lambda]$ . Here,  $t_0$  can be viewed as the time for which the data is to be transmitted and  $\lambda$  can be viewed as the maximum tolerable delay. We consider a time-slotted system, where time is represented by integer multiples of the slot duration  $\tau$ . In the temporal graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E}_{[t_0, t_0+\lambda]})$ , we can define a temporal path from  $s$  to  $d$  (denoted by  $\text{Path}_{s \rightarrow d}$ ) as an ordered sequence of edges  $\{(u_1 = s, v_1, t_1), (u_2, v_2, t_2), \dots, (u_k, v_k = d, t_k)\}$ , where  $v_i = u_{i+1}$  and  $t_i < t_{i+1}$  for  $i = 1, \dots, k-1$ , and  $(u_i, v_i, t_i) \in \mathcal{E}_{[t_0, t_0+\lambda]}$ , for all  $i$ . Notice that the tail of each temporal edge must be the head of the next edge in the sequence, and that temporal ordering must be satisfied to ensure causality of the packet forwarding operation. In the multi-UAV network under consideration, all UAVs need to forward their local information to the data-gathering node. Hence, a path  $\text{Path}_{s \rightarrow d}$  from the source UAV  $s$  to the destination  $d$  (i.e., the data-gathering node) should exist for all  $s \in \mathcal{V}$ . An example of the temporal graph is illustrated in Fig. 2. Here,  $\{(s, u_1, t_2), (u_1, d, t_6)\}$  and  $\{(s, u_2, t_3), (u_2, u_3, t_5), (u_3, d, t_0+T)\}$  are both temporal paths from  $s$  to  $d$ , whereas  $\{(s, u_1, t_2), (u_1, u_3, t_0), (u_3, d, t_1)\}$  and  $\{(s, u_2, t_3), (u_2, u_3, t_5), (u_1, d, t_6)\}$  are not since, in the former case, the temporal edges  $(u_1, u_3, t_0)$  and  $(u_3, d, t_1)$  occur before the existence of  $(s, u_1, t_2)$  (i.e., causality is violated) and, in the latter case, the tail of edge  $(u_2, u_3, t_5)$  is not equal to the head of edge  $(u_1, d, t_6)$  (i.e., the two consecutive edges are not connected).

In the data-gathering scenario under consideration, each UAV completes a flight cycle and, thus, generates data to transmit every  $T$ . In this case, we can assume that  $t_0 = kT$ ,

for some integer  $k$ . The period  $T$  depends on how fast the sensor readings vary over time and needs to be gathered again. By allowing UAVs to follow a periodic flight pattern above its sensing area, the temporal paths will reappear periodically and can be used to transmit the data that is collected from the sensors every integer multiple of  $T$ . This avoids the need to compute new temporal paths every cycle and, thus, reduces the computational complexity.

### III. POWER-EFFICIENT FLIGHT-RADIUS ADJUSTMENT

In this section, the optimal flight-radius of the UAVs' circular trajectories are determined based on the total power minimization and the lifetime maximization objectives, respectively. Here, we focus only on the flight (or propulsion) power consumption of the UAVs since the transmission power is typically several orders of magnitude smaller. Moreover, we assume that the temporal paths from all UAVs to the destination are given and focus on improving power efficiency through radius adjustment. An efficient algorithm to determine the optimal temporal path from each UAV to the destination will be introduced in Section IV.

Specifically, by [30], we know that the flight power consumption of a circulating UAV, say UAV  $u$ , depends nonlinearly on the velocity  $v_u$  and radius  $r_u$  of the circular trajectory, and can be described by

$$P(v_u, r_u) = \left( c_1 + \frac{c_2}{g^2 r_u^2} \right) v_u^3 + \frac{c_2}{v_u}. \quad (7)$$

Here,  $g$  is the gravitational acceleration with nominal value  $9.8 \text{ m/s}^2$  whereas  $c_1$  and  $c_2$  are parameters related to the weight of the aircraft, wing area, and air density, etc. Typical values of  $c_1$  and  $c_2$  are given by  $c_1 = 9.26 \cdot 10^{-4}$  and  $c_2 = 2250$  [30]. Notice that the first cubic term in (7) (i.e., the term with coefficient  $c_1$ ) models the power required for the UAV to overcome the air resistance force caused by the aircraft's skin friction, form drag, etc when moving forward. The second cubic term is due to the acceleration required in the direction normal to the velocity vector to maintain a circular flight pattern with constant velocity. The third term, which is inversely proportional to the velocity, models the power required for redirecting air to generate the lift used to compensate the aircraft's weight. We can see that, when the velocity is zero, the power required for the UAV to stay afloat goes to infinity, which is consistent with the fact that fixed-wing UAVs must maintain constant motion in order to stay afloat. Further details of the flight power consumption model can be found in [30]. Since the time required to complete one flight cycle is  $T$ , the velocity of UAV  $u$  under flight radius  $r_u$  can be computed as  $v_u = \frac{2\pi r_u}{T}$ . By substituting  $v_u$  into (7), we have

$$\bar{P}(r_u) \triangleq P\left(\frac{2\pi r_u}{T}, r_u\right) \quad (8)$$

$$= \left( c_1 + \frac{c_2}{g^2 r_u^2} \right) \left( \frac{2\pi r_u}{T} \right)^3 + \frac{c_2 T}{2\pi r_u} \quad (9)$$

$$= \frac{c_1 8\pi^3}{T^3} r_u^3 + \frac{c_2 8\pi^3}{g^2 T^3} r_u + \frac{c_2 T}{2\pi} \frac{1}{r_u}. \quad (10)$$

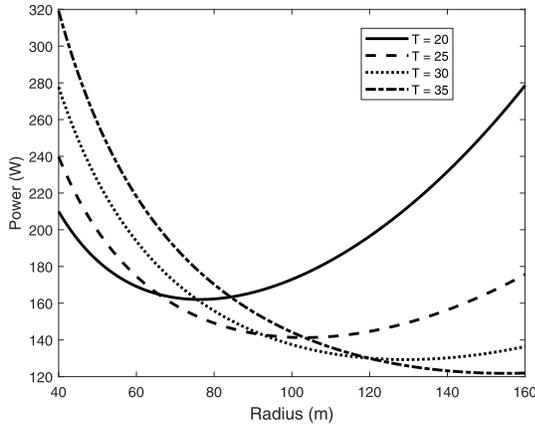


Fig. 3. Relation between the power consumption and the radius of the circular trajectory for  $T = 20, 25, 30$ , and  $35$ .

Notice that the function  $\bar{P}(r_u)$  is convex with respect to  $r_u$ , for  $r_u > 0$ , but does not increase monotonically with  $r_u$ .

In Fig. 3, we plot the function with respect to  $r_u$  for different values of  $T$ . We can see that, in all cases, it is not always power-efficient to adopt the smallest possible flight radius. In fact, following the model in (7), when the radius is small, a larger power is required to support the acceleration in the direction normal to the velocity vector and also to stay afloat at lower velocity. However, as  $T$  increases, the impact due to air resistance and perpendicular acceleration decreases whereas the power required to stay afloat increases significantly, causing the minimum-power radius to increase. In the following, we propose two radius adjustment strategies for the multi-UAV network, namely, the total power minimization strategy and the lifetime maximization strategy.

#### A. Strategy I: Total Power Minimization

In the total power minimization strategy, we propose to find the flight radii of all UAVs, i.e.,  $\{r_s\}_{s \in \mathcal{V}}$ , that minimize the total power consumption subject to constraints given by the need to preserve the paths from all UAVs to the data-gathering node. Suppose that  $\text{Path}_{s \rightarrow d}$  is the temporal path from UAV  $s$  to destination  $d$  (i.e., the data-gathering node). To preserve the connectivity of all edges on the temporal path, the radii of UAVs on the temporal path must satisfy  $D_{(u,v,t)}(r_u, r_v) \leq \delta_u$ , for all  $(u, v, t) \in \text{Path}_{s \rightarrow d}$ . The paths originating from different UAVs generate different sets of constraints. In this case, the total power minimization problem can be formulated as

$$\min_{\{r_s\}_{s \in \mathcal{V}}} \sum_{s \in \mathcal{V}} \bar{P}(r_s) \quad (11a)$$

$$\text{subject to } D_{(u,v,t)}(r_u, r_v)^2 \leq \delta_u^2, \quad \forall (u, v, t) \in \cup_{s \in \mathcal{V}} \text{Path}_{s \rightarrow d} \quad (11b)$$

$$\bar{r}_{s,\min} \leq r_s \leq \bar{r}_{s,\max}, \quad \forall s \in \mathcal{V}. \quad (11c)$$

Notice that  $D_{(u,v,t)}(r_u, r_v)^2$  is a quadratic function of  $(r_u, r_v)$ . Hence, the optimization problem given above is convex and can be solved using off-the-shelf software, such as CVX [28].

It is worthwhile to note that, by satisfying the constraints in (11b), connections are guaranteed only at the exact time

instants associated with the edges  $(u, v, t) \in \cup_{s \in \mathcal{V}} \text{Path}_{s \rightarrow d}$ . In practice, the transmission between UAVs must occur over a finite slot duration  $\tau$ . In this case, it is necessary to increase the radii  $\{r_s\}_{s \in \mathcal{V}}$  proportionally until the connectivity of all edges  $(u, v, t) \in \cup_{s \in \mathcal{V}} \text{Path}_{s \rightarrow d}$  can be maintained over duration  $\tau$ . This will be considered in our experiments.

#### B. Strategy II: Lifetime Maximization

In the lifetime maximization strategy, we determine the optimal radius adjustment policy with the goal of maximizing the lifetime of the multi-UAV network. Here, lifetime is defined as the time for which all UAVs remain active (i.e., the time until one of the UAV's battery become depleted). This is often dominated by a bottleneck UAV that is most restrictive in terms of its connectivity constraints or limited in terms of battery energy resources.

Specifically, by letting  $E_{\text{batt},u}$  be the battery energy at UAV  $u$ , the lifetime of UAV  $u$  can be computed as  $E_{\text{batt},u}/\bar{P}(r_u)$ , and the lifetime maximization problem can be formulated as

$$\max_{\{r_s\}_{s \in \mathcal{V}}} \min_{s \in \mathcal{V}} \frac{E_{\text{batt},s}}{\bar{P}(r_s)} \quad (12a)$$

$$\text{subject to (11 b), (11c)}. \quad (12b)$$

By introducing the auxiliary variable  $\eta$ , this problem can be formulated equivalently as

$$\max_{\{r_s\}_{s \in \mathcal{V}}, \eta} \eta \quad (13a)$$

$$\text{subject to (11 b), (11c)} \quad (13b)$$

$$\frac{E_{\text{batt},s}}{\bar{P}(r_s)} \geq \eta, \quad \forall s \in \mathcal{V}. \quad (13c)$$

Here,  $\eta$  can be viewed as the lifetime of the UAV network (i.e., the shortest lifetime among all UAVs). Notice that the constraint in (13c) can be written equivalently as

$$\frac{1}{\eta} \geq \frac{\bar{P}(r_s)}{E_{\text{batt},s}}. \quad (14)$$

This constraint is non-convex, making the optimization problem difficult to solve in general. Here, we adopt a successive convex approximation (SCA) approach [29], which yields efficient solutions to this problem in an iterative manner.

Suppose that  $\eta^{(\ell)}$  is the solution of  $\eta$  obtained in the  $\ell$ -th iteration of the SCA algorithm. Given  $\eta^{(\ell)}$ , we can lower-bound the left-hand-side of (14) with its first-order approximation about the point  $\eta = \eta^{(\ell)}$ , which yields

$$\frac{1}{\eta} \geq \frac{1}{\eta^{(\ell)}} - \frac{1}{(\eta^{(\ell)})^2} (\eta - \eta^{(\ell)}). \quad (15)$$

Then, in iteration  $\ell + 1$ , we solve the approximate convex optimization problem

$$\max_{\{r_s\}_{s \in \mathcal{V}}, \eta} \eta \quad (16a)$$

$$\text{subject to (11 b), (11c)} \quad (16b)$$

$$\frac{1}{\eta^{(\ell)}} - \frac{1}{(\eta^{(\ell)})^2} (\eta - \eta^{(\ell)}) \geq \frac{\bar{P}(r_s)}{E_{\text{batt},s}}, \quad \forall s \in \mathcal{V}. \quad (16c)$$

where the left-hand-side of the constraint in (14) is replaced with its lower bound in (15). By replacing  $1/\eta$  with its lower bound, the constraint becomes tighter and, thus, the resulting solution must also fall within the feasible set of the original problem. Notice that this problem is convex and can be solved using off-the-shelf solvers such as CVX [28]. The iterative procedure continues until convergence. The convergence to a Karush-Kuhn-Tucker (KKT) point is guaranteed as shown in [29, Th. 1].

Recall that, due to the UAVs' periodic flight patterns, the flight radii (and also the temporal paths to be described in the next section) need not be updated dynamically since the temporal paths reappear periodically in each flight cycle. Therefore, the solutions can be computed offline in a centralized manner before the UAVs are dispatched. However, it would also be interesting to explore distributed implementations of the algorithm to enable dynamic adjustment in fast varying environments (e.g., the case where the underlying sensors may be mobile). Distributed optimization algorithms, such as the consensus alternating direction method of multipliers (consensus-ADMM) algorithm proposed in [36], [37], can be adopted to achieve this task, but require further studies that are beyond the scope of this work. In the above, we determined the optimal flight-radius for the UAVs' circular trajectories while preserving predetermined temporal paths. The problem remains as to how the optimal temporal paths from each UAV to the destination can be found, which is the subject of the following section.

#### IV. POWER-EFFICIENT TEMPORAL PATH ALGORITHM

In this section, we propose a power-efficient (PE) temporal path algorithm that can be used to identify a path from each source UAV  $s$  to the destination  $d$  that enables more flexibility for power reduction through the radius adjustment algorithm proposed in the previous section. This is done by finding a path that yields the minimum accumulated edge-wise power consumption as described below. Notice that a temporal path must satisfy a sequential temporal order and, thus, conventional routing algorithms cannot be directly applied here.

Specifically, let us define the cost of edge  $(u, v, t)$  as

$$\rho(u, v, t) \triangleq \min_{\substack{r_u, r_v: D_{(u, v, t)}(r_u, r_v) \leq \delta_u, \\ R_{u, \min} \leq r_u \leq R_{u, \max}, R_{v, \min} \leq r_v \leq R_{v, \max}}} [\bar{P}(r_u) + \bar{P}(r_v)], \quad (17)$$

for  $v \neq d$ . Here,  $\rho(u, v, t)$  can be viewed as the minimum sum power required to maintain connectivity between UAVs  $u$  and  $v$  at time  $t$  (i.e., the minimum edge-wise power consumption of  $(u, v, t)$ ). For  $v = d$ , the cost is given by  $\rho(u, d, t) \triangleq \min_{\substack{r_u: D_{(u, d, t)}(r_u, 0) \leq \delta_u, \\ R_{u, \min} \leq r_u \leq R_{u, \max}}} \bar{P}(r_u)$ . The optimization problem that must be solved to obtain the cost is convex and, thus, can be solved using off-the-shelf solvers, such as CVX [28]. By the above definition, the smaller the cost, the more power efficient the UAVs associated with the edge can be. Consequently, the cost of path  $\text{Path}_{s \rightarrow d} = \{(v_1 = s, v_2, t_1), (v_2, v_3, t_2), \dots, (v_k, v_{k+1} = d, t_k)\}$  is defined as the sum of the cost over all edges on the path,

#### Algorithm 1 Power-Efficient (PE) Temporal Path Algorithm

**Input:** A temporal graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , source vertex  $s$ , time interval  $[t_0, t_0 + \lambda]$ .

**Output:** The minimum total cost from source  $s$  to destination  $d$  within  $[t_0, t_0 + \lambda]$ , and the corresponding path  $\text{Path}_{s \rightarrow d}$ .

- 1: Initialize the lists  $\{\mathcal{L}_u\}_{u \in \mathcal{V}}$  such that  $\mathcal{L}_s = \{(s, 0, t_0)\}$  and  $\mathcal{L}_u = \emptyset$ , for all  $u \in \mathcal{V} \setminus \{s\}$ . (An element in  $\mathcal{L}_u$  is  $(\text{Pre}[u], \text{AccCost}[u], \text{Time}[u])$ , where  $\text{AccCost}[u]$  is the cost accumulated on the subpath from  $s$  to  $u$ ,  $\text{Time}[u]$  is the time that the path arrives at  $u$  and  $\text{Pre}[u]$  is the node preceding  $u$  on the path.)
- 2: **for** each incoming edge  $e = (u, v, t)$  in temporal order **do**
- 3:   **if**  $\mathcal{L}_u$  is not empty **then**
- 4:     Find  $(\text{Pre}'[u], \text{AccCost}'[u], \text{Time}'[u]) \in \mathcal{L}_u$  such that  $\text{Time}'[u] = \max\{\text{Time}[u] : (\text{Pre}[u], \text{AccCost}[u], \text{Time}[u]) \in \mathcal{L}_u, \text{Time}[u] \leq t\}$ ;
- 5:      $\text{AccCost}[v] \leftarrow \text{AccCost}'[u] + \rho(u, v, t)$ ;
- 6:      $\text{Time}[v] \leftarrow t$ ;
- 7:     Insert  $(u, \text{AccCost}[v], \text{Time}[v])$  into  $\mathcal{L}_v$ ;
- 8:     Remove dominated elements in  $\mathcal{L}_v$ ;
- 9:   **end if**
- 10: **end for**
- 11:  $a \leftarrow d$ ;
- 12:  $t \leftarrow t_0 + \lambda$ ;
- 13: **while**  $a \neq s$  **do**
- 14:   Find  $(\text{Pre}'[a], \text{AccCost}'[a], \text{Time}'[a]) \in \mathcal{L}_a$  such that  $\text{Time}'[a] = \max\{\text{Time}[a] : (\text{Pre}[a], \text{AccCost}[a], \text{Time}[a]) \in \mathcal{L}_a, \text{Time}[a] \leq t\}$ ;
- 15:   Prepend edge  $(\text{Pre}'[a], a, \text{Time}'[a])$  to  $\text{Path}_{s \rightarrow d}$ ;
- 16:    $a \leftarrow \text{Pre}'[a]$ ;
- 17:    $t \leftarrow \text{Time}'[a]$ ;
- 18: **end while**
- 19: **return**  $\text{Path}_{s \rightarrow d}$ ;

i.e.,  $\sum_{(u', v', t') \in \text{Path}_{s \rightarrow d}} \rho(u', v', t')(r_{u'}, r_{v'})$ . The proposed PE temporal path algorithm aims to identify the path with the minimum accumulated cost. Notice that, by treating  $\rho(u, v, t)$  as a measure of distance between nodes  $u$  and  $v$  at time  $t$ , finding the PE temporal path is equivalent to finding the shortest distance path in a temporal graph. However, different from conventional routing problems, the "shortest" temporal path problem cannot be solved by Dijkstra-type algorithms since the prefix subpath to a "shortest" temporal path may not be a "shortest" temporal path to an earlier node on the path [27]. In Algorithm 1, we modify the shortest path-distance algorithm proposed in [27] using the new distance measure mentioned above to solve the problem.

Specifically, in the PE temporal path algorithm described in Algorithm 1, each UAV, say UAV  $u$ , records a list  $\mathcal{L}_u$  consisting of the incoming time of potential subpaths leading up to the current UAV and their respective accumulated costs up to that point. In particular, an element in the list  $\mathcal{L}_u$  can be represented by the triplet  $(\text{Pre}[u], \text{AccCost}[u], \text{Time}[u])$ , where  $\text{Pre}[u]$  is the UAV preceding  $u$  on the subpath from  $s$  to  $u$ ,  $\text{AccCost}[u]$  is the accumulated cost of this subpath, and  $\text{Time}[u]$  is the time that the subpath arrives at  $u$ . The lists  $\{\mathcal{L}_u\}_{u \in \mathcal{V}}$  are constructed by evaluating all edges one-by-one in temporal order. In particular, when evaluating the edge  $e = (u, v, t)$ , a new subpath to UAV  $v$  may be explored and, thus, may lead to an update of the list  $\mathcal{L}_v$ . To do so, we first select an element in  $\mathcal{L}_u$  that corresponds to a

subpath with the latest arrival time at UAV  $u$ , i.e., the element  $(\text{Pre}'[u], \text{AccCost}'[u], \text{Time}'[u]) \in \mathcal{L}_u$  where  $\text{Time}'[u] \geq \text{Time}[u]$ , for all  $(\text{Pre}[u], \text{AccCost}[u], \text{Time}[u]) \in \mathcal{L}_u$  with  $\text{Time}[u] \leq t$ . Notice that, by design of the temporal path algorithm (as will be clear later), the path from  $s$  to  $u$  with the latest arrival time corresponds to the path with the minimum accumulated cost due to the removal of the so-called dominated subpaths in each step. Therefore, by appending edge  $e = (u, v, t)$  to this subpath, we obtain a new subpath from  $s$  to  $v$  with the least accumulated cost  $\text{AccCost}[v] = \text{AccCost}'[u] + \rho_{(u,v,t)}$ , computed up to this point. The arrival time of this subpath at UAV  $v$  is given by  $\text{Time}[v] = t$  and the UAV preceding  $v$  on the subpath is  $\text{Pre}[v] = u$ . The element  $(\text{Pre}[v], \text{AccCost}[v], \text{Time}[v])$  is then inserted into the list  $\mathcal{L}_v$ . Notice that, if there exists two elements  $(\text{Pre}_1[v], \text{AccCost}_1[v], \text{Time}_1[v])$  and  $(\text{Pre}_2[v], \text{AccCost}_2[v], \text{Time}_2[v])$  in  $\mathcal{L}_v$  such that  $\text{AccCost}_1[v] < \text{AccCost}_2[v]$  and  $\text{Time}_1[v] \leq \text{Time}_2[v]$ , then it is no longer necessary to store the element  $(\text{Pre}_2[v], \text{AccCost}_2[v], \text{Time}_2[v])$  in  $\mathcal{L}_v$  since the accumulated cost of any path that contains the subpath corresponding to  $(\text{Pre}_2[v], \text{AccCost}_2[v], \text{Time}_2[v])$  can be reduced by substituting the subpath with that corresponding to  $(\text{Pre}_1[v], \text{AccCost}_1[v], \text{Time}_1[v])$ . In this case, we say that  $(\text{Pre}_2[v], \text{AccCost}_2[v], \text{Time}_2[v])$  is dominated by  $(\text{Pre}_1[v], \text{AccCost}_1[v], \text{Time}_1[v])$  and, thus, can be removed from  $\mathcal{L}_v$ . After scanning through all temporal edges, the optimal temporal path can be found by backtracking from the destination  $d$  to the source UAV  $s$ . That is, starting from  $d$ , we search for the element in  $\mathcal{L}_d$  with the smallest accumulated cost (i.e.,  $\text{AccCost}[d]$ ), and prepend the preceding UAV (i.e.,  $\text{Pre}[d]$ ) associated with the element to the path. We repeat the process similarly for the preceding UAV and continue doing so until  $s$  is included into the path. The above procedures are further described in the example below.

*Example:* Let us consider the example in Fig. 2, where we have 4 UAVs (denoted by  $s, u_1, u_2$ , and  $u_3$ ), and one destination  $d$ . In this case, the edges in the temporal graph can be listed in temporal order as  $\{(u_1, u_3, t_0), (u_3, d, t_0), (u_3, d, t_1), (s, u_1, t_2), (u_3, d, t_2), (s, u_2, t_3), (s, u_2, t_4), (u_2, u_3, t_5), (u_1, d, t_6), (u_2, u_3, t_6), (u_1, d, t_7), (u_1, u_3, t_0 + T), (u_3, d, t_0 + T), (u_3, d, t_1 + T)\}$ . Note that not all possible edges are listed for ease of exposition. The lists maintained by the UAVs are initialized as  $\mathcal{L}_s = \{(s, 0, t_0)\}$  and  $\mathcal{L}_u = \emptyset$ , for all  $u \in \mathcal{V} \setminus \{s\}$ , where  $(s, 0, t_0)$  represents the trivial subpath from  $s$  to itself at time  $t_0$ . By scanning the edges in temporal order, we first arrive at the edge  $(u_1, u_3, t_0)$ . At this point, no subpaths from  $s$  to  $u_1$  have yet been explored and, thus, the list maintained by UAV  $u_1$  is empty (i.e.,  $\mathcal{L}_{u_1} = \emptyset$ ). In this case, the edge  $(u_1, u_3, t_0)$  cannot lead to a feasible subpath from  $s$  to  $u_3$  and, thus, is omitted. The same holds for edges  $(u_3, d, t_0)$  and  $(u_3, d, t_1)$ . Then, by continuing to examine the edge  $(s, u_1, t_2)$ , we are able to find the element  $(s, 0, t_0) \in \mathcal{L}_s$  and, thus, is able to update the list at UAV  $u_1$  by  $\mathcal{L}_{u_1} = \{(s, \rho_{(s,u_1,t_2)}, t_2)\}$ . In Fig. 4(a), we plot the edge corresponding to the recorded subpath in blue and the edges that are omitted in gray. Continuing in this fashion, edge  $(u_3, d, t_2)$  is omitted, since  $\mathcal{L}_{u_3} = \emptyset$ , and edges

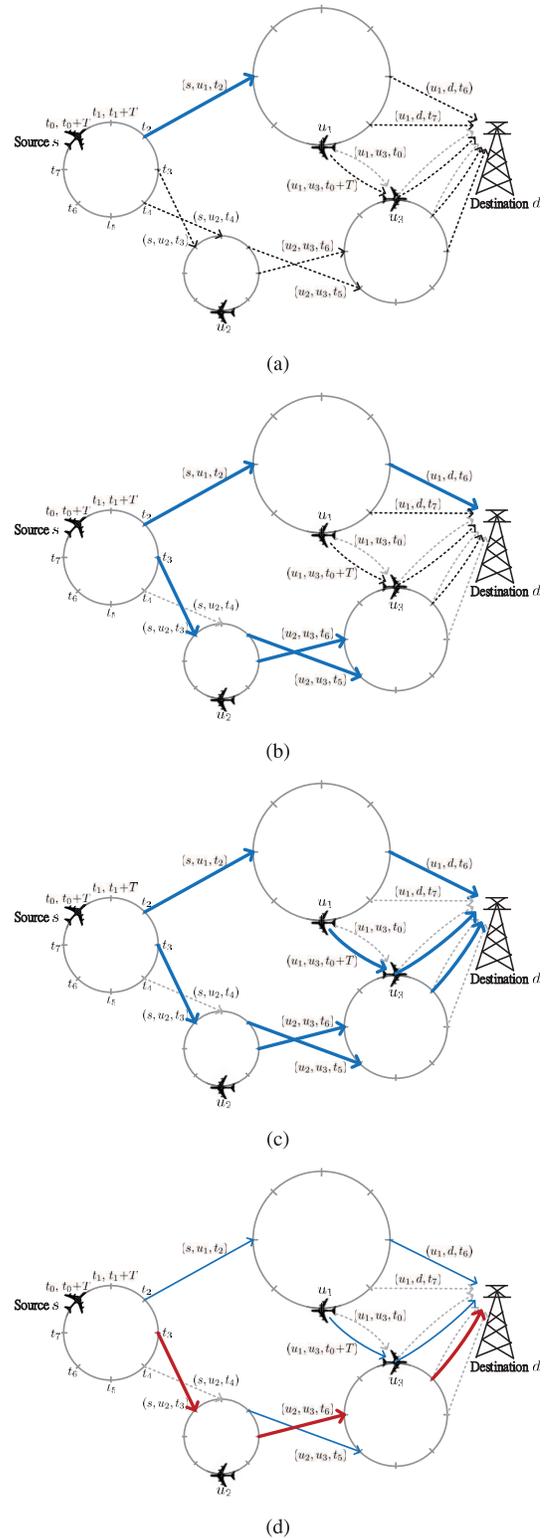


Fig. 4. Example of the power-efficient temporal path algorithm.

$(s, u_2, t_3)$  and  $(s, u_2, t_4)$  lead to an update of the list at UAV  $u_2$  as  $\mathcal{L}_{u_2} = \{(s, \rho_{(s,u_2,t_3)}, t_3), (s, \rho_{(s,u_2,t_4)}, t_4)\}$ . By assuming that  $\rho_{(s,u_2,t_3)} < \rho_{(s,u_2,t_4)}$ , the subpath associated with the element  $(s, \rho_{(s,u_2,t_4)}, t_4)$  cannot lead to a temporal path with smaller accumulated cost than  $(s, \rho_{(s,u_2,t_3)}, t_3)$  and, thus, is removed to yield  $\mathcal{L}_{u_2} = \{(s, \rho_{(s,u_2,t_3)}, t_3)\}$ . Similarly, when

examining edges  $(u_2, u_3, t_5)$ ,  $(u_1, d, t_6)$ , and  $(u_2, u_3, t_6)$ , the lists at UAV  $u_3$  and  $d$  are updated as  $\mathcal{L}_{u_3} = \{(u_2, \rho(s, u_2, t_3) + \rho(u_2, u_3, t_5), t_5), (u_2, \rho(s, u_2, t_3) + \rho(u_2, u_3, t_6), t_6)\}$  and  $\mathcal{L}_d = \{(u_1, \rho(s, u_1, t_2) + \rho(u_1, d, t_6), t_6)\}$ , respectively. Both subpaths leading up to  $u_3$  are recorded since we assume here that  $\rho(s, u_2, t_3) + \rho(u_2, u_3, t_6) < \rho(s, u_2, t_3) + \rho(u_2, u_3, t_5)$ . The recorded subpaths are illustrated in Fig. 4(b). By scanning the remaining edges, the lists at UAV  $u_3$  and the destination  $d$  are updated as  $\mathcal{L}_{u_3} = \{(u_2, \rho(s, u_2, t_3) + \rho(u_2, u_3, t_5), t_5), (u_2, \rho(s, u_2, t_3) + \rho(u_2, u_3, t_6), t_6), (u_1, \rho(s, u_1, t_2) + \rho(u_1, u_3, t_0+T), t_0+T)\}$  and  $\mathcal{L}_d = \{(u_1, \rho(s, u_1, t_2) + \rho(u_1, d, t_6), t_6), (u_3, \rho(s, u_2, t_3) + \rho(u_2, u_3, t_6) + \rho(u_3, d, t_0+T), t_0+T), (u_3, \rho(s, u_2, t_3) + \rho(u_2, u_3, t_6) + \rho(u_3, d, t_1+T), t_1+T)\}$ , as illustrated in Fig. 4(c). After scanning through all possible edges, we proceed to find the optimal temporal path from  $s$  to  $d$ . To do so, we first initialize the path as  $\text{Path}_{s \rightarrow d} = \emptyset$  and search for the element with the minimum accumulated cost in  $\mathcal{L}_d$ . This yields the element  $(u_3, \rho(s, u_2, t_3) + \rho(u_2, u_3, t_6) + \rho(u_3, d, t_1+T), t_1+T)$ , which implies that the edge  $(u_3, d, t_1+T)$  should be included into the optimal temporal path. Therefore, the desired temporal path is updated as  $\text{Path}_{s \rightarrow d} = (u_3, d, t_1+T) \cup \text{Path}_{s \rightarrow d}$ . Next, we backtrack to the previous node  $u_3$  and search for the element with the minimum cost in  $\mathcal{L}_{u_3}$  that occurs before  $t_1+T$ . The selected element is  $(u_2, \rho(s, u_2, t_3) + \rho(u_2, u_3, t_6), t_6)$  and the temporal path is updated as  $\text{Path}_{s \rightarrow d} = \{(u_2, u_3, t_6), (u_3, d, t_1+T)\}$ . Finally, by backtracking to node  $u_2$ , we arrive at the temporal path  $\text{Path}_{s \rightarrow d} = \{(s, u_2, t_3), (u_2, u_3, t_6), (u_3, d, t_1+T)\}$ , as illustrated in Fig. 4(d).

It is worthwhile to note that, in the above, the PE temporal path is found by considering arbitrary initial phases for the UAVs' circular trajectories. However, with arbitrary initial phases, the UAVs' may not meet at their closest points within their flight cycle and yields less flexibility for radius adjustment. This can be improved upon through a carefully designed phase readjustment scheme as to be proposed in the following section.

## V. ITERATIVE TRAJECTORY PHASE READJUSTMENT ALGORITHM

In this section, we propose an iterative phase readjustment algorithm to enable more flexibility in the radius adjustment of Section III. The key idea is to adjust the initial phases of the UAVs' circular trajectories so that the UAVs can meet (and, thus, communicate) at the closest points of their trajectories. Once the phases are adjusted as such, the trajectory radii and the temporal paths can be updated more flexibly to further improve the power efficiency.

Specifically, suppose that the temporal paths from all UAVs to the destination have been determined. By following the paths in reverse order, we can construct a spanning tree from the destination  $d$  to all UAVs using the depth-first search algorithm [38]. Based on the spanning tree, we divide the UAVs into multiple subsets according to their levels within the tree, as illustrated in Fig. 5. Let us denote the set of UAVs in level  $\ell$  as  $\mathcal{U}_\ell \subset \mathcal{U} \triangleq \{1, \dots, N\}$  and let  $L$  be the maximum depth of the spanning tree. The parent of node  $u$  in the

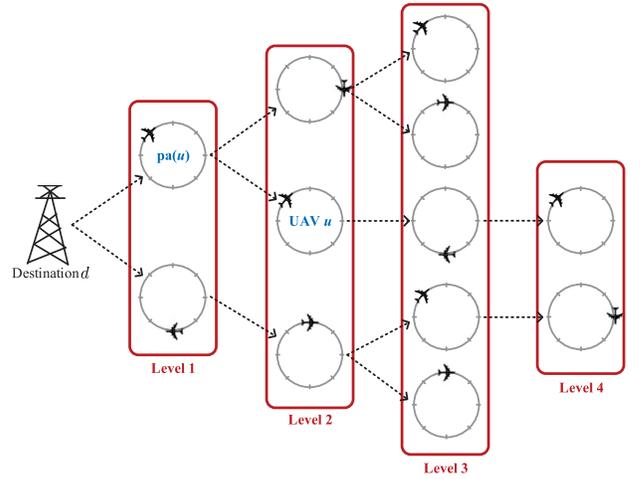


Fig. 5. Example of the spanning tree obtained by performing a depth-first search in the reverse order of the temporal graphs.

tree is denoted by  $\text{pa}(u)$ . Based on the spanning tree, the initial phases of the UAVs' circular trajectories can be adjusted level-by-level from the root node (i.e., the destination  $d$ ) to all leaf nodes in a way that allows each UAV to meet its parent-UAV at the closest points of their trajectories. In each step, the initial phases of UAVs in level  $\ell$  are matched to their respective parent-UAVs in level  $\ell - 1$  according to the following proposition.

*Proposition 1:* Let  $\theta_{u,0}$  and  $\theta_{v,0}$  be the initial phases of UAVs  $u$  and  $v$ , respectively. Suppose that the circular trajectories of the UAVs do not overlap with each other. For two UAVs to meet at the closest points of their trajectories, their initial phases must satisfy

$$\theta_{u,0} - \theta_{v,0} \pmod{2\pi} = \pi \quad (18)$$

if the UAVs fly in the same direction (either clockwise or counter-clockwise), or

$$\theta_{uv}^* + \pi - \theta_{v,0} \pmod{2\pi} = \theta_{u,0} - \theta_{uv}^* \pmod{2\pi} \quad (19)$$

if the UAVs fly in opposite directions, where  $\theta_{uv}^*$  is chosen such that  $\cos \theta_{uv}^* = \frac{x_v - x_u}{\|c_v - c_u\|}$  and  $\sin \theta_{uv}^* = \frac{y_v - y_u}{\|c_v - c_u\|}$  (i.e.,  $\theta_{uv}^*$  is the angle of the vector connecting the trajectory centers of UAVs  $u$  and  $v$ ).

*Proof:* Observe that the closest points of the circular trajectories of UAVs  $u$  and  $v$  occur at the points where the two circular trajectories intersect with the line connecting their trajectory centers, as illustrated in Fig. 6. Let  $\theta_{uv}^*$  be the angle of the vector connecting the trajectory centers of UAVs  $u$  and  $v$  such that  $\cos \theta_{uv}^* = \frac{x_v - x_u}{\|c_v - c_u\|}$  and  $\sin \theta_{uv}^* = \frac{y_v - y_u}{\|c_v - c_u\|}$ . Note that  $\theta_{vu}^* = \theta_{uv}^* + \pi$ . Based on this observation, the results can be proven separately for the two cases as follows.

- (i) First, let us consider the case where both UAVs fly in the counter-clockwise direction, as shown in Fig. 6. In this case, for UAVs  $u$  and  $v$  to meet at the closest points of their respective trajectories at time  $t$ , the initial phases of the two trajectories must satisfy

$$\theta_{u,0} + \frac{2\pi t}{T} \pmod{2\pi} = \theta_{uv}^* \pmod{2\pi} \quad (20)$$

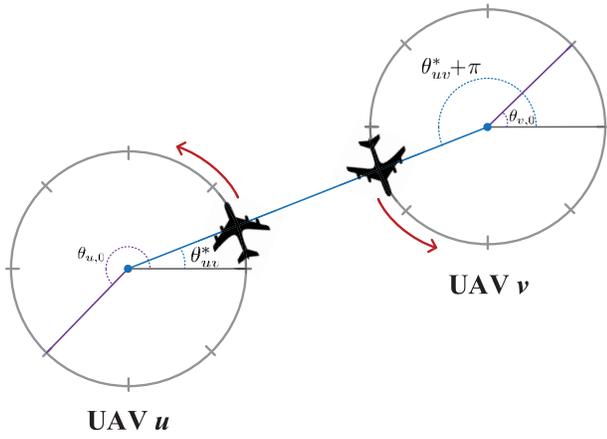


Fig. 6. Example of two UAVs meeting at their closest points.

and

$$\theta_{v,0} + \frac{2\pi t}{T} \mod 2\pi = \theta_{uv}^* + \pi \mod 2\pi \quad (21)$$

By subtracting (21) and (20), we have

$$\theta_{v,0} - \theta_{u,0} \mod 2\pi = \pi. \quad (22)$$

This implies that the phases of the two UAVs must always maintain a difference of  $\pi$  regardless of the time instant  $t$ . This condition does not depend on the actual value of  $\theta_{uv}^*$ . The case where both UAVs fly in the clockwise direction can be proved similarly.

- (ii) Secondly, let us consider the case where UAVs  $u$  and  $v$  fly in counter-clockwise and clockwise directions, respectively. In this case, for UAVs  $u$  and  $v$  to meet at the closest points of their respective trajectories at time  $t$ , we must have

$$\theta_{u,0} + \frac{2\pi t}{T} \mod 2\pi = \theta_{uv}^* \mod 2\pi \quad (23)$$

and

$$\theta_{v,0} - \frac{2\pi t}{T} \mod 2\pi = \theta_{uv}^* + \pi \mod 2\pi. \quad (24)$$

By adding together (23) and (24), it follows that

$$\theta_{u,0} + \theta_{v,0} \mod 2\pi = 2\theta_{uv}^* + \pi \mod 2\pi. \quad (25)$$

The case where UAVs  $u$  and  $v$  fly in clockwise and counter-clockwise directions can be proved similarly. ■

Based on the above proposition, we can sequentially adjust the the initial phases of the UAVs level-by-level starting from the root node. In particular, suppose that UAV  $u$  is in level  $\ell$  and that UAV  $v$  is its parent node in level  $\ell - 1$ . Moreover, assume that the initial phase of UAV  $v$  has already been determined (e.g., by matching it to the initial phase of its parent node), and is given by  $\theta_{v,0}$ . Then, by Proposition 1, the initial phase of UAV  $u$  should be chosen as  $\theta_{u,0} = \theta_{v,0} + \pi \mod 2\pi$  if UAVs  $u$  and  $v$  are flying in the same direction, and as  $\theta_{u,0} = 2\theta_{uv}^* + \pi - \theta_{v,0} \mod 2\pi$  if they are flying in opposite directions. However, for a UAV in level 1, its parent is the root node (i.e., the destination  $d$ ) which stays at a static position. In this case, the UAV will always be able to meet its parent  $d$  at the closest point of its trajectory to  $d$ . The time

that this occurs depends on the initial phase of the UAV. More specifically, suppose that UAV  $u_1$  in level 1 reaches its closest point to the destination at time  $t_d$ . This implies that the initial phase of  $u_1$  is

$$\theta_{u_1,0} = \theta_{u_1 d}^* - \frac{2\pi t_d}{T} \mod 2\pi, \quad (26)$$

if UAV  $u_1$  flies in the counter-clockwise direction, and

$$\theta_{u_1,0} = \theta_{u_1 d}^* + \frac{2\pi t_d}{T} \mod 2\pi, \quad (27)$$

if it flies in the clockwise direction. Then, going back in time, the most recent time that UAV  $u_1$  has met its children node UAV  $u_2$  (in level 2) is  $t_d - (\theta_{u_1 d}^* - \pi - \theta_{u_2 u_1}^* \mod 2\pi) \frac{T}{2\pi}$  if UAV  $u_1$  is traveling in the counter-clockwise direction (and, thus, backwards in time in the clockwise direction) and  $t_d - (\theta_{u_2 u_1}^* + \pi - \theta_{u_1 d}^* \mod 2\pi) \frac{T}{2\pi}$  if UAV  $u_1$  is traveling in the clockwise direction. Let us define the binary variable  $o_u \in \{1, -1\}$ , where  $o_u = 1$  if UAV  $u$  is flying in the counter-clockwise direction and  $o_u = -1$  if UAV  $u$  is flying in the clockwise direction. Moreover, let  $\ell(u)$  be the level of UAV  $u$  and let  $(u, \text{pa}(u), \text{pa}^2(u), \dots, \text{pa}^{\ell(u)-1}(u), \text{pa}^{\ell(u)}(u) = d)$  be the sequence of UAVs on the path from node  $u$  to the destination  $d$  within the spanning tree, where  $\text{pa}^k(u)$  represents the ancestor of UAV  $u$  in level  $\ell(u) - k$ . Then, the time that UAV  $u$  meets its parent node at the closest point of its trajectory can be computed as

$$t_d - \sum_{k=1}^{\ell(u)-1} \left[ o_{\text{pa}^k(u)} \left( \theta_{\text{pa}^k(u)\text{pa}^{k+1}(u)}^* - \pi - \theta_{\text{pa}^{k-1}(u)\text{pa}^k(u)}^* \right) \mod 2\pi \right] \frac{T}{2\pi}. \quad (28)$$

Note that  $\sum_{k=1}^{\ell(u)-1} \left[ o_{\text{pa}^k(u)} \left( \theta_{\text{pa}^k(u)\text{pa}^{k+1}(u)}^* - \pi - \theta_{\text{pa}^{k-1}(u)\text{pa}^k(u)}^* \right) \mod 2\pi \right] \frac{T}{2\pi}$  must be less than  $\lambda$  in order for the delay constraint to be satisfied. To enable more flexibility in the temporal path adjustment, we choose to set

$$t_d = t_0 + \max_{u \in \mathcal{U}} \sum_{k=1}^{\ell(u)-1} \left[ o_{\text{pa}^k(u)} \left( \theta_{\text{pa}^k(u)\text{pa}^{k+1}(u)}^* - \pi - \theta_{\text{pa}^{k-1}(u)\text{pa}^k(u)}^* \right) \mod 2\pi \right] \frac{T}{2\pi} \quad (29)$$

so that the first transmission between UAVs occur at time  $t_0$ . In this case, by substituting  $t_d$  into (26) and (27), the initial phase of UAV  $u$  in level 1 can be written as

$$\theta_{u,0} = \theta_{u d}^* - o_u \frac{2\pi t_0}{T} - o_u \max_{u \in \mathcal{U}} \sum_{k=1}^{\ell(u)-1} \left[ o_{\text{pa}^k(u)} \left( \theta_{\text{pa}^k(u)\text{pa}^{k+1}(u)}^* - \pi - \theta_{\text{pa}^{k-1}(u)\text{pa}^k(u)}^* \right) \mod 2\pi \right] \mod 2\pi. \quad (30)$$

Recall that, in our work, we assume that data is gathered periodically by the UAVs and, thus, the feedback transmission is initiated every integer multiple of  $T$ , as mentioned in Section II. In this case, the initial transmission time  $t_0$  is equal

**Algorithm 2** Trajectory Phase Readjustment Algorithm

**Input:** Temporal paths  $\text{Path}_{u \rightarrow d}$  for all  $u \in \mathcal{U}$ .

**Output:** Adjusted trajectory phases  $\theta_{u,0}$ , for all  $u \in \mathcal{U}$ .

- 1: Let  $\mathcal{G} = \cup_{u \in \mathcal{U}} \text{Path}_{u \rightarrow d}$  be the directed graph formed by the temporal paths from all UAVs to the destination. By reversing the direction of the edges, find a spanning tree following the depth-first search algorithm [38].
- 2: Divide the nodes into subsets  $\mathcal{U}_\ell$ , for  $\ell = 1, \dots, L$ , where  $\mathcal{U}_\ell$  is the subset that include nodes in level  $\ell$  of the tree and  $L$  is the maximum depth of the tree.
- 3: Set  $t_d \leftarrow t_0 + \max_{u \in \mathcal{U}} \sum_{k=1}^{\ell(u)-1} [o_{\text{pa}^k(u)} (\theta_{\text{pa}^k(u)}^* \text{pa}^{k+1}(u) - \pi - \theta_{\text{pa}^{k-1}(u)}^* \text{pa}^k(u)) \bmod 2\pi] \frac{T}{2\pi}$ , and  $\theta_{u,0} \leftarrow \theta_{ud}^* - o_u \frac{2\pi t_d}{T} \bmod 2\pi$ , for all  $u \in \mathcal{U}_1$ .
- 4: **for**  $\ell = 2$  to  $L$  **do**
- 5:     **for** node  $u$  in  $\mathcal{U}_\ell$  **do**
- 6:         Set  $\theta_{u,0} \leftarrow \theta_{\text{pa}(u),0} - \pi \bmod 2\pi$ , if UAVs  $u$  and  $v$  are flying in the same direction, and  $\theta_{u,0} \leftarrow 2\theta_{\text{upa}(u)}^* + \pi - \theta_{\text{pa}(u),0} \bmod 2\pi$ , if they are flying in opposite directions.
- 7:     **end for**
- 8: **end for**
- 9: **return**  $\theta_{u,0}$ , for all  $u \in \mathcal{U}$ .

to  $kT$ , for some integer  $k$ , and thus the initial phase of UAVs in level 1 can be reduced to

$$\theta_{u,0} = \theta_{ud}^* - o_u \max_{u \in \mathcal{U}} \sum_{k=1}^{\ell(u)-1} \left[ o_{\text{pa}^k(u)} \left( \theta_{\text{pa}^k(u)}^* \text{pa}^{k+1}(u) - \pi - \theta_{\text{pa}^{k-1}(u)}^* \text{pa}^k(u) \right) \bmod 2\pi \right] \bmod 2\pi. \quad (31)$$

The trajectory phase readjustment algorithm is summarized in Algorithm 2.

Furthermore, once the initial phases have been readjusted according to Algorithm 2, the PE temporal paths and the flight radii are updated following the algorithms proposed in Sections IV and III, respectively. The overall operation can be summarized as follows:

*Initialization:* Set random initial values for the initial phases  $\{\theta_{u,0}\}_{u \in \mathcal{V}}$ , and set the initial radii as  $r_u = R^*$ , where  $R^*$  is the radius of the ground coverage.

- (i) Determine the optimal temporal paths from all UAVs to the destination using Algorithm 1.
- (ii) Given the temporal paths, determine the optimal flight radii  $\{r_u^*\}_{u \in \mathcal{V}}$  by the total power minimization or the lifetime maximization strategies.
- (iii) Update the radii as  $r_u \leftarrow (1 - \alpha)r_u + \alpha r_u^*$ , for all  $u \in \mathcal{V}$ .
- (iv) Adjust the initial phases  $\{\theta_{u,0}\}_{u \in \mathcal{V}}$  using Algorithm 2.
- (v) Repeat (i)-(iv) until no further improvement can be observed in terms of either the total power or the network lifetime.

## VI. NUMERICAL RESULTS AND PERFORMANCE COMPARISONS

In this section, we demonstrate the effectiveness of the proposed temporal path and flight radius adjustment strategies through computer simulations using MATLAB. In these experiments, we assume that UAVs are deployed to gather data

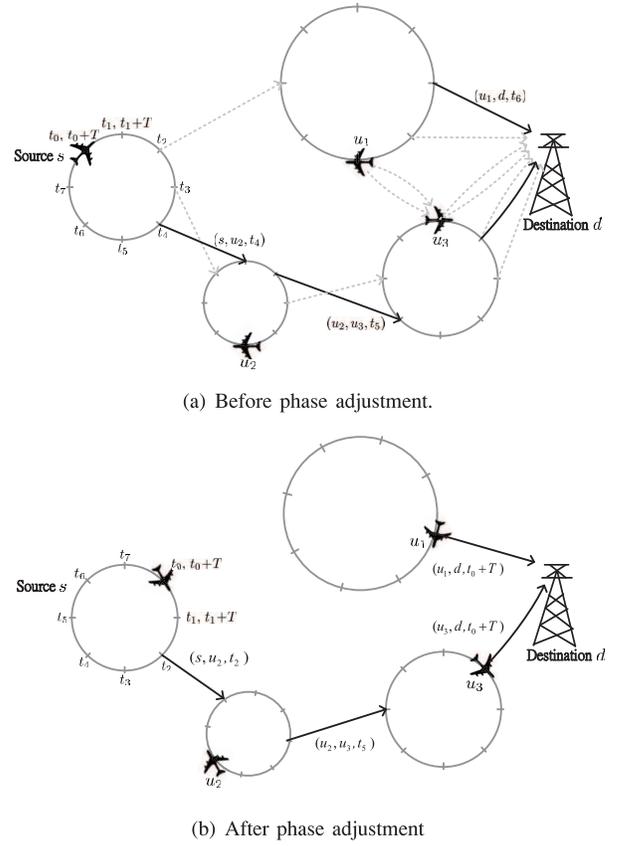


Fig. 7. Example of a temporal path before and after phase readjustment.

from  $M = 20000$  ground sensors placed randomly according to a uniform distribution within a  $1500 \times 1500 \text{ m}^2$  area. The sensor association is determined by the  $k$ -means clustering algorithm and the center of the UAVs' circular trajectories are placed at their respective centroid positions. By [35], it can be shown that the optimal altitude and ground coverage radius of the UAVs in the suburban environment is given by  $h^* = 40.8 \text{ m}$  and  $R^* = 155 \text{ m}$ , respectively, when the maximum acceptable pathloss is  $\Gamma = 85\text{dB}$  and the carrier frequency is  $2.5 \text{ GHz}$ . The upper and lower bounds of the flight radius  $r_u$  (i.e.,  $\bar{r}_{u,\text{max}}$  and  $\bar{r}_{u,\text{min}}$ ) can then be computed by (4). Here, we set  $\bar{r}_{\text{phy}} = 40 \text{ m}$ . The circulating directions of the UAVs (i.e., clockwise or counter-clockwise) are chosen randomly with equal probability. The battery has capacity equal to  $5300 \text{ mAh}$  and operates at  $18.5 \text{ V}$  (i.e.,  $352.89 \text{ kJ}$ ). Here, we consider cases where  $T = 20 \text{ s}$ ,  $\tau = 0.5 \text{ s}$ , and  $\delta_u = 220 \text{ m}$ , for all  $u \in \mathcal{U}$ . The curves are averaged over 100 network realizations with 95% confidence interval.

In Fig. 8, we show the average power consumption per UAV versus the number of UAVs for both the proposed PE temporal path algorithm and the earliest arrival time (EAT) temporal path algorithm proposed in [27]. The EAT algorithm aims to minimize the arrival time of each UAV's packet at the destination. In fact, it can be achieved by setting the accumulated cost in Algorithm 1 equal to the arrival time on each subpath, i.e.,  $\text{AccCost}[u] = \text{Time}[u]$ , for all  $u$ . Here, we set  $\lambda = 60 \text{ s}$ .

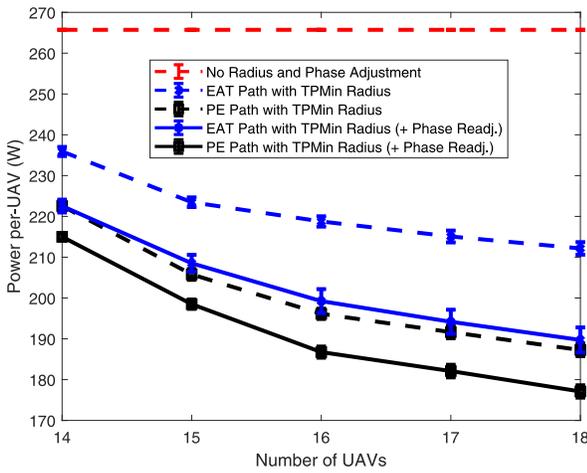
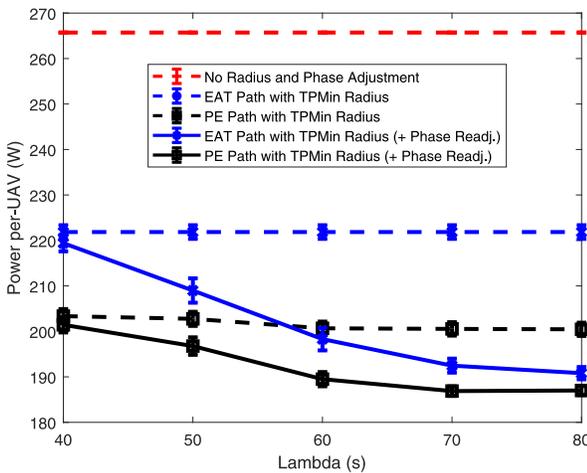


Fig. 8. Average per-UAV power consumption versus the number of UAVs.

Fig. 9. Average per-UAV power consumption versus  $\lambda$ .

For each temporal path algorithm, we consider both the case with only the total power minimization (TPMin) radius adjustment proposed in Section III and the case with both TPMin radius adjustment and also the iterative phase readjustment proposed in Section V. The case with no radius and phase adjustment (i.e., the case where all UAVs fly at the radius  $R^*$ , i.e.,  $r_u = R^*$ ,  $\forall u$ , with random initial phases) is also shown for comparison. We can observe that the proposed PE temporal path algorithm outperforms the EAT algorithm in terms of power savings since the EAT algorithm focuses only on minimizing the arrival time and does not consider the efficiency of flight power consumption. For a network with  $N = 18$  UAVs, we can see that our proposed scheme with only TPMin radius adjustment reduces the average power consumption by approximately 30.6% compared to the case with no adjustment and by 11.7% compared to the EAT algorithm. An additional 5.3% improvement can be observed by further taking into consideration the iterative phase readjustment. Moreover, we can see that the average power consumption per UAV decreases as the number of UAVs increases since the UAVs are closer in this case and can more flexibly adjust their radii.

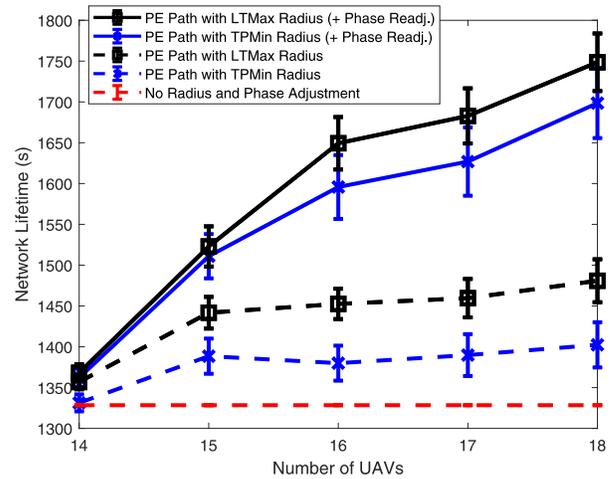
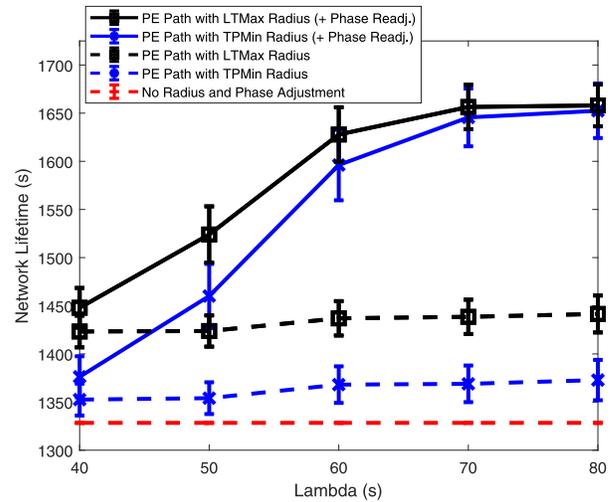


Fig. 10. Network lifetime versus the number of UAVs.

Fig. 11. Network lifetime versus  $\lambda$ .

In Fig. 9, we show the average power consumption per UAV versus the delay tolerance  $\lambda$  for the proposed PE and the EAT temporal path algorithms. Here, the radius adjustment is performed by the TPMin strategy and the number of UAVs is  $N = 16$ . We can see that the average power consumption decreases as  $\lambda$  increases in all cases. The power reduction is most evident when phase readjustment is further applied since, in this case, the probability that the delay constraint is violated after phase readjustment is significantly reduced. That is, UAVs have more flexibility in choosing longer routes that may benefit more from the phase readjustment.

In Fig. 10, we show the network lifetime versus the number of UAVs for both the lifetime maximization (LTMMax) and the TPMin radius adjustment strategies. Here, we adopt the PE temporal path algorithm in both cases and set  $\lambda = 60$  s. We can see that the network lifetime increases with the number of UAVs in all of the proposed schemes. The improvement is particularly significant when phase readjustment is further applied since UAVs can more flexibly adjust their radii in this case. In fact, under the LTMMax strategy, the network lifetime is improved by 31.6% compared to the case with no radius

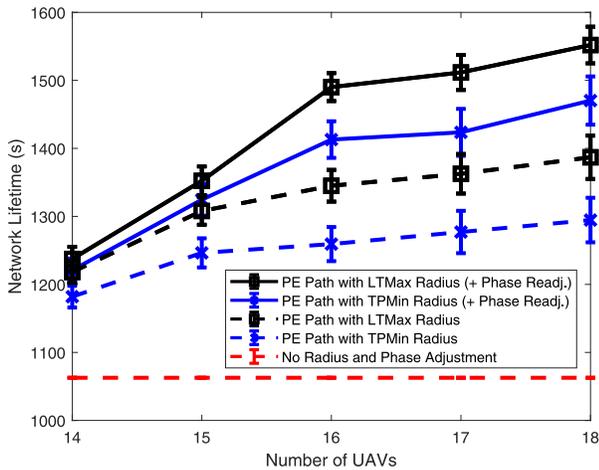


Fig. 12. Network lifetime versus the number of UAVs with unequal energy.

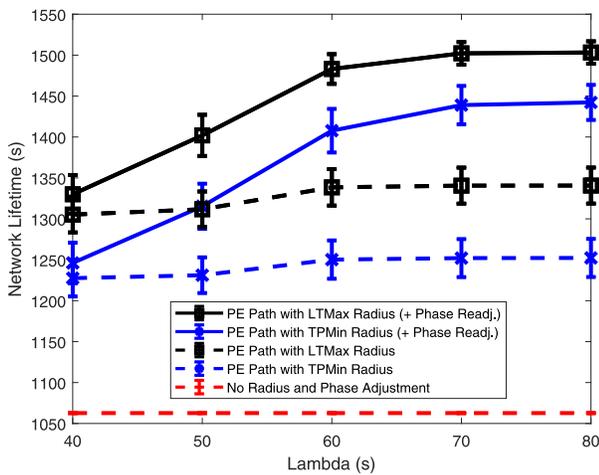


Fig. 13. Network lifetime versus  $\lambda$  with unequal energy.

and phase adjustment, and by 18.1% compared to the case with only radius adjustment. Furthermore, we can see that the LTMax strategy is able to yield a longer lifetime than the TPMin strategy since the former focuses more on reducing the power consumption of the bottleneck UAV instead of reducing the total power consumption.

Similarly, in Fig. 11, we show the network lifetime versus  $\lambda$  for both LTMax and TPMin radius adjustment strategies. Here, we again adopt the PE temporal path algorithm and set  $N = 16$ . We can see that network lifetime increases as  $\lambda$  increases since more path options become available in this case. Moreover, similar to Fig. 9, we can see that the increase is more substantial when phase readjustment is further applied since the UAVs have more flexibility in choosing potentially longer routes that may benefit from the phase readjustment.

In Figs. 12 and 13, we examine the impact of unequal battery energy on the lifetime of the UAVs. Here, the initial battery energy of UAVs are chosen uniformly from the interval  $[(1 - \beta) E_{\text{batt}}, (1 + \beta) E_{\text{batt}}]$ , where  $E_{\text{batt}} = 352.89$  kJ and  $\beta = 0.2$ . Specifically, in Fig. 12, we show the network lifetime versus the number of UAVs for both LTMax and TPMin radius adjustment strategies in the case with unequal battery

energies at the UAVs. Here, we again examine the case with  $\lambda = 60$  s. We can see that the advantage of the LTMax radius adjustment strategy over the TPMin strategy is even more pronounced in the unequal energy scenario since variations in the battery energy may further limit the lifetime of the bottleneck UAV. Similarly, in Fig. 13, we show the network lifetime versus  $\lambda$  for both LTMax and TPMin radius adjustment strategies in the same unequal battery energy scenario. Here, we set  $N = 16$ . Notice that, compared to Fig. 11, the difference in network lifetime between LTMax and TPMin is also more significant due to further limitations of the battery energy at the bottleneck UAV.

## VII. CONCLUSION

In this work, we proposed flight radius adjustment and temporal path algorithms for a network of fixed-wing UAVs to perform power-efficient feedback transmission from the UAVs to a data-gathering node. Each UAV follows a circular trajectory to maintain coverage over its responsible sensing area. Given the temporal paths between all UAVs and the destination, two radius adjustment strategies were proposed, namely, the TPMin and LTMax strategies. The former aims to minimize the total flight power consumption of all UAVs whereas the latter aims to maximize the time until a UAV's battery is depleted. By establishing the relationship between routing in this highly dynamic UAV network and that in temporal graphs, the PE temporal path algorithm was proposed to further improve upon the power efficiency by taking into account each UAV's flexibility in adjusting its flight-radius. Furthermore, an iterative phase readjustment algorithm was proposed to refine the phases of the UAVs' circular trajectories to further improve the power efficiency. The phase readjustment is accompanied by updates to the flight-radii and temporal paths in an alternating fashion. Finally, numerical simulations were performed to demonstrate the effectiveness of the proposed schemes in terms of reducing the total power consumption and extending network lifetime.

## REFERENCES

- [1] N. H. Motlagh, M. Bagaa, and T. Taleb, "UAV-based IoT platform: A crowd surveillance use case," *IEEE Commun. Mag.*, vol. 55, no. 2, pp. 128–134, Feb. 2017.
- [2] H. X. Pham, H. M. La, D. Feil-Seifer, and M. C. Deans, "A distributed control framework of multiple unmanned aerial vehicles for dynamic wildfire tracking," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 50, no. 4, pp. 1537–1548, Apr. 2020.
- [3] J. Sun, B. Li, Y. Jiang, and C.-Y. Wen, "A camera-based target detection and positioning UAV system for search and rescue (SAR) purposes," *Sensors*, vol. 16, no. 11, p. 1778, 2016.
- [4] Y. Zhong *et al.*, "Mini-UAV-borne hyperspectral remote sensing: From observation and processing to applications," *IEEE Geosci. Remote Sens. Mag.*, vol. 6, no. 4, pp. 46–62, Dec. 2018.
- [5] M. Erdelj, E. Natalizio, K. R. Chowdhury, and I. F. Akyildiz, "Help from the sky: Leveraging UAVs for disaster management," *IEEE Pervasive Comput.*, vol. 16, no. 1, pp. 24–32, Jan.–Mar. 2017.
- [6] J. Deruyck, M. Wyckmans, W. Joseph, and L. Martens, "Designing UAV-aided emergency networks for large-scale disaster scenarios," *EURASIP J. Wireless Commun. Netw.*, p. 79, Apr. 2018.
- [7] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless communications with unmanned aerial vehicles: Opportunities and challenges," *IEEE Commun. Mag.*, vol. 54, no. 5, pp. 36–42, May 2016.

- [8] L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in UAV communication networks," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1123–1152, 2nd Quart., 2016.
- [9] D.-T. Ho, E. I. Grøtli, P. B. Sujit, T. A. Johansen, and J. A. B. Sousa, "Optimization of wireless sensor network and UAV data acquisition," *J. Intell. Robot. Syst.*, vol. 78, no. 1, pp. 159–179, Apr. 2015.
- [10] C. Zhan, Y. Zeng, and R. Zhang, "Energy-efficient data collection in UAV enabled wireless sensor network," *IEEE Wireless Commun. Lett.*, vol. 7, no. 3, pp. 328–331, Jun. 2018.
- [11] J. Liu, X. Wang, B. Bai, and H. Dai, "Age-optimal trajectory planning for UAV-assisted data collection," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM Workshops)*, Honolulu, HI, USA, Apr. 2018, pp. 553–558.
- [12] J. Gong, T. Chang, C. Shen, and X. Chen, "Flight time minimization of UAV for data collection over wireless sensor networks," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 1942–1954, Sep. 2018.
- [13] C. Caillouet, F. Giroire, and T. Razafindralambo, "Optimization of mobile sensor coverage with UAVs," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPs)*, Apr. 2018, pp. 622–627.
- [14] H. J. Na and S. Yoo, "PSO-based dynamic UAV positioning algorithm for sensing information acquisition in wireless sensor networks," *IEEE Access*, vol. 7, pp. 77499–77513, 2019.
- [15] C. Zhan and Y. Zeng, "Aerial-ground cost tradeoff for multi-UAV enabled data collection in wireless sensor networks," *IEEE Trans. Commun.*, vol. 68, no. 3, pp. 1937–1950, Mar. 2020.
- [16] Y. Chen, N. Zhao, Z. Ding, and M. Alouini, "Multiple UAVs as relays: Multi-hop single link versus multiple dual-hop links," *IEEE Trans. Wireless Commun.*, vol. 17, no. 9, pp. 6348–6359, Sep. 2018.
- [17] G. Zhang, H. Yan, Y. Zeng, M. Cui, and Y. Liu, "Trajectory optimization and power allocation for multi-hop UAV relaying communications," *IEEE Access*, vol. 6, pp. 48566–48576, 2018.
- [18] S. Rosati, K. Kruzelecki, L. Traynard, and B. R. Mobile, "Speed-aware routing for UAV ad-hoc networks," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2013, pp. 1367–1373.
- [19] Y. Zheng, Y. Wang, Z. Li, L. Dong, Y. Jiang, and H. Zhang, "A mobility and load aware OLSR routing protocol for UAV mobile ad-hoc networks," in *Proc. Int. Conf. Inf. Commun. Technol. (ICT)*, May 2014, pp. 1–7.
- [20] M. Song, J. Liu, and S. Yang, "A mobility prediction and delay prediction routing protocol for UAV networks," in *Proc. 10th Int. Conf. Wireless Commun. Signal Process. (WCSP)*, Oct. 2018, pp. 1–6.
- [21] B. Sliwa, D. Behnke, C. Ide, and C. Wietfeld, "B.A.T. mobile: Leveraging mobility control knowledge for efficient routing in mobile robotic networks," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2016, pp. 1–6.
- [22] R. Shirani, "Reactive-greedy-reactive in unmanned aeronautical ad-hoc networks: A combinational routing mechanism," M.S. thesis, Dept. Comput. Sci., Carleton Univ., Ottawa, ON, Canada, Aug. 2011.
- [23] J. M. M. Biomo, T. Kunz, and M. St-Hilaire, "Routing in unmanned aerial ad hoc networks: Introducing a route reliability criterion," in *Proc. IFIP Wireless Mobile Netw. Conf. (WMNC)*, May 2014, pp. 1–7.
- [24] G. Gankhuyag, A. P. Shrestha, and S. Yoo, "Robust and reliable predictive routing strategy for flying ad-hoc networks," *IEEE Access*, vol. 5, pp. 643–654, 2017.
- [25] J. Jiang and G. Han, "Routing protocols for unmanned aerial vehicles," *IEEE Commun. Mag.*, vol. 56, no. 1, pp. 58–63, Jan. 2018.
- [26] M. Y. Arafat and S. Moh, "Routing protocols for unmanned aerial vehicle networks: A survey," *IEEE Access*, vol. 7, pp. 99694–99720, 2019.
- [27] H. Wu, J. Cheng, S. Huang, Y. Ke, Y. Lu, and Y. Xu, "Path problems in temporal graphs," *Proc. VLDB Endow.*, vol. 7, no. 9, pp. 721–732, May 2014.
- [28] M. Grant and S. Boyd. (Mar. 2014). *CVX: MATLAB Software for Disciplined Convex Programming, Version 2.1*. [Online]. Available: <http://cvxr.com/cvx>
- [29] B. R. Marks and G. P. Wright, "A general inner approximation algorithm for nonconvex mathematical programs," *Oper. Res.*, vol. 26, no. 4, pp. 681–683, 1978.
- [30] Y. Zeng and R. Zhang, "Energy-efficient UAV communication with trajectory optimization," *IEEE Trans. Wireless Commun.*, vol. 16, no. 6, pp. 3747–3760, Jun. 2017.
- [31] D. Yang, Q. Wu, Y. Zeng, and R. Zhang, "Energy tradeoff in ground-to-UAV communication via trajectory design," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 6721–6726, Jul. 2018.
- [32] F. Ono, H. Ochiai, and R. Miura, "A wireless relay network based on unmanned aircraft system with rate optimization," *IEEE Trans. Wireless Commun.*, vol. 15, no. 11, pp. 7699–7708, Nov. 2016.
- [33] Z. M. Fadlullah, D. Takaishi, H. Nishiyama, N. Kato, and R. Miura, "A dynamic trajectory control algorithm for improving the communication throughput and delay in UAV-aided networks," *IEEE Netw.*, vol. 30, no. 1, pp. 100–105, Jan. 2016.
- [34] R. I. Bor-Yaliniz, A. El-Keyi, and H. Yanikomeroglu, "Efficient 3-D placement of an aerial base station in next generation cellular networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2016, pp. 1–5.
- [35] M. Alzenad, A. El-Keyi, and H. Yanikomeroglu, "3-D placement of an unmanned aerial vehicle base station for maximum coverage of users with different QoS requirements," *IEEE Wireless Commun. Lett.*, vol. 7, no. 1, pp. 38–41, Feb. 2018.
- [36] K. Huang and N. D. Sidiropoulos, "Consensus-ADMM for general quadratically constrained quadratic programming," *IEEE Trans. Signal Process.*, vol. 64, no. 20, pp. 5297–5310, Jan. 2016.
- [37] T.-H. Chang, M. Hong, and X. Wang, "Multi-agent distributed optimization via inexact consensus ADMM," *IEEE Trans. Signal Process.*, vol. 63, no. 2, pp. 482–497, Jan. 2015.
- [38] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 3rd ed. Cambridge, MA, USA: MIT Press, 2009.



**Y.-W. Peter Hong** (Senior Member, IEEE) received the B.S. degree in electrical engineering from National Taiwan University, Taipei, Taiwan, in 1999, and the Ph.D. degree in electrical engineering from Cornell University, Ithaca, NY, USA, in 2005.

He joined the Institute of Communications Engineering and the Department of Electrical Engineering, National Tsing Hua University (NTHU), Hsinchu, Taiwan, in 2005, where he is a Full Professor. He also currently serves as an Associate Vice President of Research and Development with NTHU. His research interests include multiuser communications, physical layer security, machine learning for wireless communications, and distributed signal processing for IoT and sensor networks. He received the IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award in 2010, the Y. Z. Hsu Scientific Paper Award in 2011, the National Science Council Wu Ta-You Memorial Award in 2011, the Chinese Institute of Electrical Engineering Outstanding Young Electrical Engineer Award in 2012, the Best Paper Award from the Asia-Pacific Signal and Information Processing Association Annual Summit and Conference in 2013, and the Ministry of Science and Technology Outstanding Research Award in 2019. He served as an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING and the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, and is currently an Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS. He was the Chair of the IEEE ComSoc Taipei Chapter in 2017–2018, and the Co-Chair of the Technical Affairs Committee, the Information Services Committee, and the Chapter Coordination Committee of the IEEE ComSoc Asia-Pacific Board in 2014–2015, 2016–2019, and 2020–2021, respectively.



**Ray-Hsiang Cheng** received the M.S. degree in computer science from National Tsing Hua University, Hsinchu, Taiwan, in 2018. His research interests include unmanned aerial vehicle networks, routing protocols, and power-saving algorithms. He is currently an Engineer with MediaTek.



**Yu-Cheng Hsiao** received the B.S. degree in electrical engineering from National Tsing Hua University, Hsinchu, Taiwan, in 2020. He is currently a Research Assistant with the Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan. His research interests include cooperative communications and signal processing for wireless networks.



**Jang-Ping Sheu** (Fellow, IEEE) received the B.S. degree in computer science from Tamkang University, Taiwan, in 1981, and the M.S. and Ph.D. degrees in computer science from National Tsing Hua University, Taiwan, in 1983 and 1987, respectively.

He is currently the Chair Professor with the Department of Computer Science, National Tsing Hua University. He was the Chair with the Department of Computer Science and Information Engineering, National Central University from 1997 to 1999, where he was the Director with Computer Center from 2003 to 2006. He was the Director of the Computer and Communication Research Center, National Tsing Hua University from 2009 to 2015, where he was an Associate Dean with the College of Electrical and Computer Science from 2016 to 2017. His current research interests include wireless communications, mobile computing, Internet of Things, and UAV-assisted communication systems. He was an Associate Editor of the *IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS* and the *International Journal of Sensor Networks*. He is an Advisory Board Member of the *International Journal of Ad Hoc and Ubiquitous Computing* and the *International Journal of Vehicle Information and Communication Systems*. He received the Distinguished Research Awards of the National Science Council of the Republic of China in 1993-1994, 1995-1996, and 1997-1998, the Distinguished Engineering Professor Award of the Chinese Institute of Engineers in 2003, the K.-T. Li Research Breakthrough Award of the Institute of Information and Computing Machinery in 2007, the Y. Z. Hsu Scientific Chair Professor Award and Pan Wen Yuan Outstanding Research Award in 2009 and 2014, respectively, the Academic Award in Engineering from the Ministry of Education and Medal of Honor in Information Sciences from the Institute of Information and Computing Machinery in 2016 and 2017, respectively, and the TECO Awards in 2019. He is a member of the Phi Tau Phi Society.