

Received July 31, 2020, accepted August 21, 2020, date of publication August 27, 2020, date of current version September 10, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3019963

Towards Persistent Surveillance and Reconnaissance Using a Connected Swarm of Multiple UAVs

JEIHEE CHO^{ID}, (Student Member, IEEE), JAEYI SUNG^{ID}, JINYI YOON^{ID}, (Student Member, IEEE), AND HYUNGJUNE LEE^{ID}, (Member, IEEE)

Department of Computer Science and Engineering, Ewha Womans University, Seoul 03760, South Korea

Corresponding author: HyungJune Lee (hyungjune.lee@ewha.ac.kr)

This work was supported by the National Research Foundation (NRF) of Korea Grant funded by the Korea Government [Ministry of Science and ICT (MSIT)] under Grant NRF-2018R1A2B6004006.

ABSTRACT In situations where surveillance or communication infrastructure has collapsed, it is important to keep monitoring affected areas. We leverage unmanned aerial vehicles (UAVs) to collect and provide up-to-date on-site information to a data consumer in an efficient way, for later complete yet agile analysis. We propose a distributed dynamic data collection scheme for persistent surveillance and reconnaissance, using a swarm of connected UAVs with two phases of operation: 1) network formation; and 2) UAV traversal of a region of interest. The main task of a UAV is to continuously collect data within its sensing range, while the UAV swarm travels along the calculated paths. When UAVs are newly connected to form a swarm, or disconnected from an already-formed swarm, a formation phase begins. In the formation phase, UAVs become a single group and produce a compact, dynamically alternating formation called *DiagonalX* to cover broad areas, including boundary parts, in a fair and effective manner. During the traversal phase, each UAV swarm finds a simple yet efficient navigation path based on data freshness to cover sub-areas and continuously obtain up-to-date information evenly throughout the whole region of interest. Simulation experiments confirm that both formation and traversal procedures perform essential tasks in a distributed manner, while maintaining better data freshness than other counterpart algorithms, with a freshness factor of up to 5.77, and reasonable overheads. An additional feature, a dynamically aperiodic formation change, achieves a more stable performance.

INDEX TERMS Persistent surveillance, reconnaissance, connected UAVs, mobile sensor networks, swarm exploration.

I. INTRODUCTION

Even in severe situations, where surveillance or communication infrastructure is useless or no longer available, it is important to make contingency plans for keeping on-site information up-to-date. Particularly in cluttered areas or battlegrounds, the acquisition of reliable surveillance and monitoring at all times is critical. This kind of mission tends to require simultaneous event monitoring across a region of interest (RoI). Obtaining on-site information using cameras or sensors evenly over an area with a lack of surveillance and communication infrastructure is challenging.

The associate editor coordinating the review of this manuscript and approving it for publication was Yuyu Yin^{ID}.

The use of wireless sensor networks [1] has been proposed to solve the problem of ad-hoc sensing and surveillance, by using the sensors to construct low-power, multi-hop networks. Efficient sensor deployment should provide adequate sensing coverage and network connectivity [2], [3]. However, sensor deployment for target detection inevitably incurs deployment costs associated with utilizing movement-assisted agents [4]. Also, static localized deployment suffers from static and irregular collection of local information only in the vicinity of the constrained area in which the sensors are deployed [5].

Since real-time data rapidly becomes worthless in sensor networks, on-time data collection is crucial. Ensuring the timeliness of the data collected is an important design factor. A recent approach to this issue in stationary networks is the

leveraging of mobile agents such as unmanned aerial vehicles (UAVs) for target detection and sensing [6], [7].

Data collected by a single mobile agent remains isolated, and disassociated from other areas in the environment. Procedures for combining and maintaining all of the items of information collected over an RoI by scattered agents not been studied in detail. To address this problem, some researchers have leveraged a swarm of multiple mobile agents for deployment, an approach which necessitates path planning [8]–[10].

When deployed, more stationary UAVs than dynamically moving UAVs are required to cover a specific area. Therefore, it is important to take a systematic traversal approach to using a swarm of UAVs for consecutive data collection, while solving the problem of data consolidation and the synchronization of disconnected UAVs.

In this work, we solve the problem of persistent surveillance and reconnaissance, by constructing a connected swarm of UAVs. We place a swarm of connected UAVs in a specific formation and run a simple yet efficient traversal algorithm. The formation is determined to be a relative disposition within a group of UAVs which effectively and fairly covers the RoI for continuous data collection. Our system enables a connected UAV group to keep up-to-date on-site information from local terrestrial areas, and maintain collected information in a way which is efficient for later complete yet agile access.

To define the essential design requirements to achieve this goal, we first raise two key questions: 1) Is a mobile swarm better for maintaining persistent data collection than an individual UAV? and 2) If so, which formation is most efficient for sensing coverage and network communication?

First, we propose a persistent data collection scheme using multiple UAVs, which consists of two phases: UAV network formation, and UAV traversal of an RoI. We derive a simple yet efficient formation, *DiagonalX*, which forms alternately diagonal shapes of UAVs, while keeping neighboring UAVs connected. It enables effective navigation control as a group, to cover unvisited or outdated areas, and facilitates data sharing.

Second, we present a lightweight swarm traversal scheme that identifies sub-areas in an RoI to visit in a persistent manner. By considering both data freshness and trajectory efficiency, a connected UAV group can keep up-to-date information combined regularly and evenly throughout the RoI.

The main contributions of this work can be summarized as follows:

- Our approach offers a way for a group of UAVs to construct spatio-temporal information chunks over an RoI in a collaborative way, for a later extensive yet agile surveillance updates over the area.
- To continuously and evenly maintain the timeliness of locally collected data throughout the RoI, the task is divided into two core sub-procedures: UAV swarm formation, and UAV traversal.

- We propose an efficient UAV swarm formation shape with a diagonal distribution, with periodic or aperiodic alternate formation shifts to fairly cover the RoI, including the boundary areas, and a traversal strategy which takes into account data freshness.

II. RELATED WORK

The problem of persistent data collection using a swarm of connected UAVs has two major aspects: 1) the way in which a group of UAVs should behave in terms of connection and traversal, and 2) how to ensure the data collected over the RoI are up-to-date.

A. UAV SWARMS

The use of UAV swarms has drawn considerable attention in the areas of surveillance, rescue, and data collection. Previous studies have used graphs consisting of a fixed number of locations which mobile agents have to visit, either once [11]–[13], or in a periodic or persistent way [14], [15]. Various forms of optimization problems with different constraints to cover all locations via a shortest path have been solved with variants of the Travelling Salesman Problem (TSP), Vehicle Routing Problem (VRP), or by integer programming. However, many previous algorithms suffer from excessive computational overheads.

More relevant to this work, [16] has used multiple UAVs for infrastructure inspection, using a triangular formation for the UAV swarm positioning. Those researchers exploited an angle-coded particle swarm optimization approach to swarm path planning. Some researchers have suggested a reconnaissance method in which multiple UAVs follow optimized paths identified using distributed particle swarm optimization [17]. Researchers have also studied the optimal deployment of a UAV swarm in situations in which Internet of Things (IoT) devices randomly change their active status.

Although these projects have proposed approaches to optimal path planning for traveling, collision avoidance, and energy efficiency, they do not explicitly consider the newness or timeliness of the collected data.

B. UP-TO-DATE DATA COLLECTION

Since newness of data is critical to support real-time services or surveillance, it has been investigated in various fields. The problem in the fields of data analytics, knowledge discovery, and control systems has been investigated under the names of *timeliness* [18], [19] and the Age of Information (AoI) [20]–[22]. These approaches have measured the newness of data in terms of the elapsed time since an initial data collection or an update on the data has occurred.

A concept known as the Age of Information has recently been employed to represent the freshness of the data in communication systems [20]–[22]. More closely related to our work involving UAVs, some researchers have solved the problem of collecting data from sensor nodes only once by generating two different trajectories for a single UAV, based on two different optimization metrics of the maximum AoI

and the average AoI. However, this work is limited to one mobile agent for data collection, and the feasibility of using multiple UAVs has not been investigated in detail.

There has been little work addressing both path planning for the connected UAVs and the up-to-dateness of the data that the UAV swarm has collected. Our work provides a specific way to solve the challenging problem of persistent surveillance and reconnaissance by multiple UAVs, by executing robust swarm formation and traversal.

III. SYSTEM OVERVIEW

We address the problem of on-site data collection for surveillance or sensing over a physical region of interest in a continuous manner. We consider a scenario in which persistent autonomous surveillance or on-site information over a certain area is necessary, and existing surveillance and communication infrastructure may be useless or no longer available. One question that we raise is: can UAVs be utilized for this mission, and if so, how they should perform over time to collect on-site information and provide the most up-to-date information to a surveillance officer or a military sergeant?

We assume that UAVs can navigate over a virtual grid topology by choosing either the north, east, south, or west direction without any battery outage or obstacle collision issues. The UAVs are equipped with a positioning system and sensor devices such as cameras and motion or environmental sensors with sufficient storage space that they can be aware of their own position and collect on-site information over cells within the sensing range at each grid point. UAVs can communicate with each other using a wireless radio, such as 802.11 within the radio range. We assume that the communication radio range is larger than the sensing range. It is also assumed that neither the number of UAVs nor the status of other UAVs beyond their direct UAV-to-UAV communication is known.

UAV navigation [23], [24], flight control model [25], data routing [26], and communication issues [27], [28] for UAV swarm management are important in practice. However, these issues comprise a separate problem, and are therefore not explicitly considered in this article. Also, since our work focuses on data collection, the fine control of UAV flight movement has not been considered.

The goal of this article is to find a cost-effective traversal and formation mechanism for UAVs, to maintain the most up-to-date grid-based information over space and time. We aim to collect the latest information over an RoI from a UAV or a connected set of UAVs. Each UAV is allowed to navigate over the area by itself or within a UAV group, by creating a formation with UAVs encountered within the communication range, in a distributed manner. A UAV or UAV swarm accumulates real-time information within the sensing range over time, according to its traversal and formation scheme. The data collected by a UAV or UAV swarm can be provided as time-series data at each specific location over the RoI, in an on-demand basis.

A. PROCEDURE

Our dynamic data collection scheme consists of two steps: 1) network formation and 2) traversal procedure by UAVs, as depicted in Figure 1.

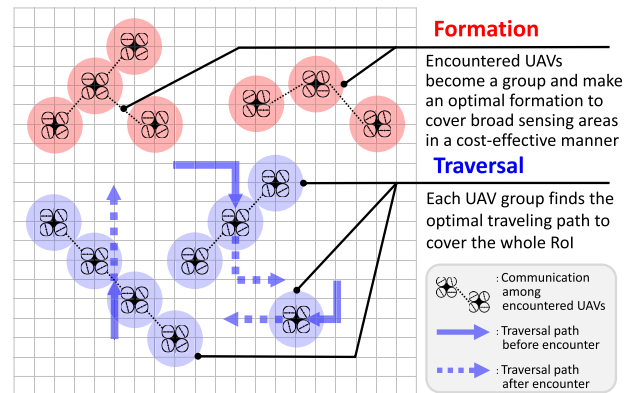


FIGURE 1. Overall procedure for persistent data collection by UAVs in two phases: UAV network formation, and UAV traversal.

Each UAV starts navigating over a designated RoI by finding a grid point to visit for retrieving the most useful up-to-date information among the candidate points. Upon reaching a specific grid point, the UAV updates the information collected from that point in its local storage. To check for the presence of other UAVs in its region, the UAV periodically broadcasts a hello packet. If one or more UAVs are found, each shares its own information collected from visited grid points, and merges these data to its local storage. Then, the UAVs create a specific formation, which can maximize the total utility of up-to-date information for the grid points covered by the group. The network formation procedure is described in Section IV.

After this procedure, a UAV that has not yet encountered other UAVs, or a group of UAVs after network formation, continues to traverse grid points in such a way, as to provide the maximum utility of up-to-date information. The UAV traversal procedure is described in Section V.

The overall procedure is described in detail in Algorithm 1, and the high-level workflow chart is presented in Figure 2.

IV. NETWORK FORMATION

In an infrastructure-free environment, where there are almost no cameras, sensors, or communication networks, we can utilize UAVs as information collectors on-the-go to gather pieces of information from an RoI. To maintain the latest information over a larger area, a swarm of connected UAVs is needed.

UAVs connected and organized in a specific formation, with the ability to communicate, can share of all the information collected to date, and can be ready to send that information to a data consumer whenever one is nearby. Depending upon the UAV formation, even the same traversal pattern can result in a different area sensed by a group of UAVs, and can accordingly provide different information.

Algorithm 1 Overall Procedure for Data Collection by UAVs

```

1: Input: this: This UAV, t: Current time, uavGroup(this):
   UAV group which this UAV belongs to, formation refresh
   period W, Grid topology of RoI
   // I. Gather information from the currently coverable area

2: Initialize sensingDatat;
3: for cell c ∈ coverableCellList(uavGroup(this)) do
4:   Update sensingDatat(c);
5: end for
   // II. Transition to formation upon an encounter with other
   UAVs
6: if isTraversalPhase() == true then
7:   Broadcast a packet to find nearby UAVs within com-
   munication range;
8:   if any UAVs with reply then
9:     for UAV u who has replied do
10:      uavGroup(this) = uavGroup(this) ∪
        uavGroup(u);
11:      sensingData(this) = sensingData(this) ∪
        sensingData(u);
12:    end for
13:    Invoke formation() to make the DiagonalX forma-
    tion;
14:  else
15:    Continue traversal();
16:  end if
17: end if
   // III. Alternate the DiagonalX formation for every W
18: if t mod W == 0 then
19:   Invoke formation() to change the formation;
20: end if

```

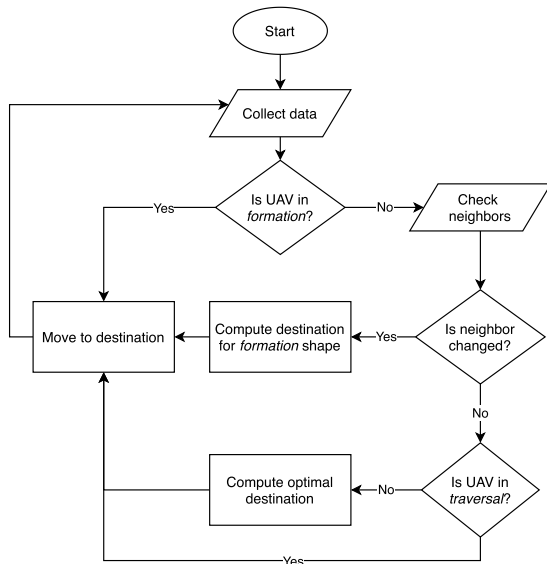


FIGURE 2. Workflow of the overall procedure in our proposed system.

A. MATHEMATICAL MOTIVATION

We present some underlying mathematical analysis on the data utility gain produced by the specific movement of a

UAV swarm in each direction: upward, downward, leftward, or rightward. By investigating the typical formations *Circle*, *Straight*, and *Diagonal*, we aim to derive a desirable formation structure that can outperform the baseline formations.

We suppose that a swarm comprises *K* UAVs, where $K \geq 2$. The covered size and the radius of the sensing area of a UAV are denoted as a_s and r_s , respectively, where $a_s = \pi r_s^2$. In this scenario, a UAV swarm is allowed to move $2r_s$ at most during a single time slot.

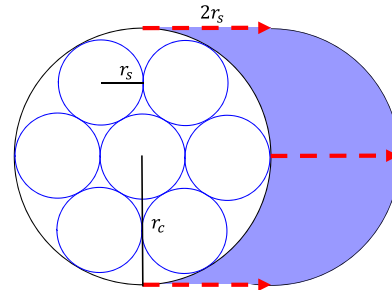


FIGURE 3. An example of a circle moving rightward. The blue colored area represents the newly covered area, and the small circle show the sensing area of a single UAV.

1) CIRCLE

The area which can be sensed by *K* UAVs can be approximated as a single large circle with a radius of r_c , as illustrated in Figure 3.

$$\pi r_c^2 \approx K \cdot \pi r_s^2. \quad (1)$$

The radius of the *swarm group* can be represented as follows:

$$r_c \approx \sqrt{K} \cdot r_s \quad (2)$$

In the *Circle* formation, the data gain for every direction is taken to be equal. In Figure 3, the data gain based on the *Circle* formation is marked with a blue area, as follows:

$$G_{circle} = 2r_s \cdot 2r_c = 4r_s \cdot r_c \approx 4\sqrt{K} \cdot r_s^2 \quad (3)$$

Since the actual coverage of the *Circle* formation is less than the ideal coverage, the data gain calculated in Equation 3 can be considered to be an upper bound of the actual *Circle* formation coverage.

2) STRAIGHT

In the *Straight* formation, a UAV swarm can obtain data gain equally for upward and downward, and also for rightward and leftward. If the *Straight* formation is horizontally positioned, the newly covered sensing area is denoted as follows:

$$G_{straight} = \begin{cases} K \cdot a_s = K \cdot \pi r_s^2 & \text{if upward or downward} \\ a_s = \pi r_s^2 & \text{otherwise} \end{cases} \quad (4)$$

If the UAV swarm moves in a parallel direction relative to the formation, the newly covered sensing area is obtained only by a single UAV within the swarm.

TABLE 1. Data gain over different formations for four directional movements where the UAV swarm is horizontally aligned in the *Straight* formation.

	Upward	Downward	Rightward	Leftward
<i>Circle</i>	$4\sqrt{K} \cdot r_s^2$			
<i>Straight</i>	$K\pi r_s^2$		πr_s^2	
<i>Diagonal</i>	$K\pi r_s^2$			

3) DIAGONAL

In the *Diagonal* formation, a swarm of UAVs form a down-left-to-up-right or up-left-to-down-right position at an angle of 45 degrees, with no overlap. The swarm can cover the new sensing area with a data gain as follows:

$$G_{diagonal} = K \cdot a_s = K \cdot \pi r_s^2 \quad (5)$$

According to Secs. IV-A1, IV-A2, and IV-A3, the data gain of each formation can be summarized as in Table 1. The data gain of the *Diagonal* is definitely larger than or equal to *Straight*. Also, the data gain of *Diagonal* formation is definitely larger than the gain of *Circle* under the following condition:

$$\begin{aligned} K \cdot \pi r_s^2 &> 4\sqrt{K} \cdot r_s^2 \\ &\equiv \sqrt{K} \geq \sqrt{2} > \frac{4}{\pi} \quad (\times \sqrt{K} \cdot \pi r_s^2) \\ &\equiv K \geq 2 \end{aligned} \quad (6)$$

As long as there are multiple UAVs to be used in a formation, selecting the *Diagonal* formation offers a higher data utility than the other formations. Thus, we choose the *Diagonal* formation as the base formation for our scheme.

B. *DiagonalX* FORMATION

To produce an effective formation of UAVs, we identify three requirements. We need a formation with: 1) minimum sensing range overlap among UAVs; 2) coupled interaction with its group traversal phase; and 3) minimum travel distance and time taken for the UAVs to relocate. To mitigate duplicate coverage from the first two perspectives, the distance between UAVs in the formation should be at least twice their sensing range, while they should be diagonally positioned according to their traversal direction, north, east, south, or west. The effort required to relocate from the current formation to our proposed formation should be minimized, to meet the last requirement.

We propose a *DiagonalX* formation to satisfy the above three requirements in an efficient manner. Our formation procedure is first launched upon an encounter between at least two UAVs within their communication range, or upon disconnection among UAVs. All of the UAVs encountered become part of the same group, and in case of separation due to UAV malfunction or battery outage, the current UAV swarm needs to be partitioned into separate UAV swarms.

First, the UAV located closest to the centroid of the group, which is the center of mass, is selected as the master UAV.

Then, the master UAV finds a grid point closest to the center location of the UAV swarm. The process of master UAV decision is conducted whenever there is any change in the UAV group. Second, it selects a diagonal formation, either down-right (Figure 4(a)) or up-right (Figure 4(b)), which can minimize the required movement distance of all UAVs, including itself. To quickly identify a one-to-one correspondence of which UAV to move to which position in the formation, we keep the same order in the x-coordinates of the current UAV positions for the x-coordinates of their future grid point locations in the selected formation. Third, the master UAV calculates all of the relocation grid points for the other UAVs, so that they can move to the calculated positions with the minimum movement overhead, based on the centroid of the above-obtained closest grid point. Lastly, after relaying the calculated relocation information to all other UAVs, the UAVs follow their own relocation procedures and complete the initial formation.

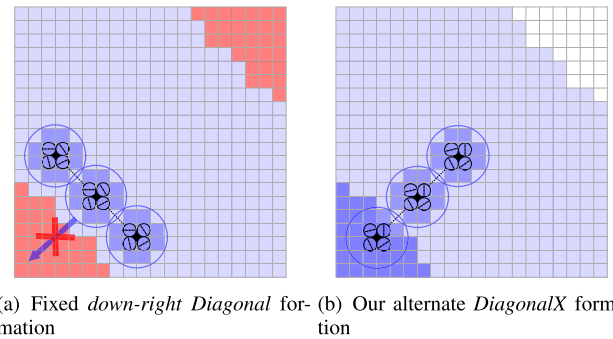


FIGURE 4. Two versions of diagonal formation for data collection from terrestrial local areas involving three UAVs with a sensing range of 12 grids.

The overall formation procedure is described in Algorithm 2.

If a group of UAVs keeps one diagonal formation, some boundary grid points (illustrated in red color in Figure 4(a)) would have lower priority, and could remain uncovered during traversal, even if there is sufficient time to cover the whole RoI. This problem arises because the coverage of some UAVs extends beyond the RoI. To overcome the inefficient boundary issue, we switch from one diagonal formation to the other, that is, from up-right diagonal to down-right diagonal, or vice versa, while rotating at an angle of 90 degrees with every specified period of time, the reformation period, denoted as W . Our *DiagonalX* formation can be applied to an arbitrary shape of RoI due to its innate adaptive property. The *DiagonalX* formation has two variants: the original *DiagonalX* formation makes a UAV swarm rotate in the clockwise direction, while the *Dynamic-with-Random* formation allows it to rotate randomly in either a clockwise or a counter-clockwise direction.

If two different UAV groups that have already been formed encounter each other, we choose one diagonal form from

Algorithm 2 formation() by a Connected UAV Group

```

1: Input: this: This UAV, t: Current time, uavGroup(this):
   UAV group which this UAV belongs to
2: Output: nextGridpointToVisit
   // I. Select master UAV among group members
3: for all UAV u ∈ uavGroup(this) do
4:   calculate distance center;
5:   set master UAV;
6: end for
   // II. Find destination for new formation shape
7: if this == masterUAV then
8:   Select next formation shape
9:   Calculate relocation positions of each UAV in
   uavGroup(this);
10:  Pass the relocation position to each UAV in
   uavGroup(this);
11: end if
   // III. Move to relocation position
12: if currentPosition != relocationPosition(this) then
13:   Randomly chose left x, y grid coordinate toward relo-
   cationPosition(this);
14:   Move selected grid coordinate;
15: end if

```

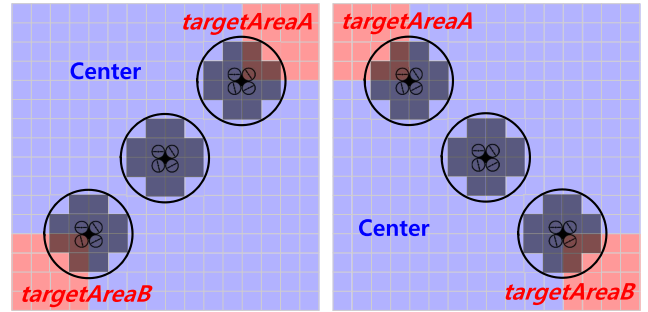
the larger UAV group, to minimize the relocation overhead. If the group sizes are the same, we randomly choose one form. If a UAV group whose formation is still in progress encounters another UAV or UAV group, it ignores the new encounter event and continues to finish its original formation procedure. When UAVs that belong to a UAV swarm become disconnected for any reason, including battery outage or UAV malfunction, the original UAV swarm is divided into separate UAV swarms, while keeping the original formation.

C. DYNAMIC DiagonalX FORMATION

We propose a *dynamic* version of *DiagonalX*, which adaptively changes the reformation period *W* to produce fair coverage between boundary and non-boundary areas.

We first define the boundary areas denoted *targetAreaA* and *targetAreaB* as illustrated with red colored cells in Figure 5. The boundary areas are located at the vertices of the RoI (for example, at the four corners in a squared RoI), and are covered by the sensing range of a UAV. All of the rest of the areas are defined as non-boundary areas, denoted *center*, and belong to neither *targetAreaA* nor *targetAreaB* of the RoI.

Depending on whether the *DiagonalX* formation is in the down-left (Figure 5(a)) or down-right position (Figure 5(b)), the target area varies. With the down-left position, *targetArea* is divided into the south-west and north-east corners, denoted as *targetAreaA* and *targetAreaB*, respectively. With the down-right position, *targetArea* consists of the north-west and south-east corners.



(a) *targetArea* of down-left DiagonalX formation (b) *targetArea* of down-right DiagonalX formation

FIGURE 5. The *targetArea* for each diagonal formation using the *DiagonalX* formation with three UAVs with a sensing range of 12 grid squares.

1) DATA FRESHNESS

We introduce a data freshness measure, *dataFreshness* that can be used as a metric of how up-to-date the data collected from a certain cell are at a specific time. As the first data freshness measure, we adopt the *Age-of-Information* with an exponential decay penalty [29]. In our context, the *age* is defined as the amount of time elapsed since the last data update from a previously sensed cell. We accordingly denote the data freshness for cell *c* at the current time *t* as follows:

$$dataFreshness_t(c) = \alpha^{t-t_{last}(c)} \tag{7}$$

where the last updated time from cell *c* is denoted as *t_{last}(c)*. By applying an exponential penalty to the age, the data freshness metric decays toward 0 as the collected data becomes staler over time.

As an example, in Figure 6 the dark gray-colored cells are the already-sensed ones covered by a group of two UAVs, where the centroid point of the UAV group is marked by a red circle. The data freshness of these cells at this time is 1, whereas that of unvisited cells is 0.

2) FORMATION CHANGE

It is usually harder to explore the boundary cells located at the corners than those near the center using a swarm formation. If the cells in both *targetAreaA* and *targetAreaB* areas have a higher data freshness value than those in the *center* area on average, we make a formation transition from down-left to down-right, or vice versa.

Each UAV swarm decides to change one diagonal formation to the other if two conditions are satisfied:

$$\overline{dataFreshness_t(c_{targetAreaA})} > \overline{dataFreshness_t(c_{center})} \tag{8}$$

$$\overline{dataFreshness_t(c_{targetAreaB})} > \overline{dataFreshness_t(c_{center})} \tag{9}$$

where *t* is the current time, and *c_{targetAreaA}*, *c_{targetAreaB}*, and *c_{center}* are the cells in the *targetAreaA*, *targetAreaB*, and *center* areas, respectively.

The direction of transitional rotation is randomly determined between clockwise and counter-clockwise. This fea-

ture helps a UAV swarm to cover the given area in a relatively fair fashion.

V. UAV TRAVERSAL

Regardless of the optimality of a specific formation, it is still challenging to cover the whole RoI using only a limited number of UAVs. To continuously maintain up-to-date information throughout the area, a UAV or a UAV swarm needs to embed its own efficient path planning algorithm. It should be able to navigate evenly over grid points taking into account the time of the last visit to them for data collection.

During the traversal phase, our UAV traversal algorithm selects a grid point to visit for the next timeslot considering two factors: 1) up-to-dateness of the data that have been collected at visited points and 2) traveling distance from the currently visiting point to a candidate point.

Our traversal scheme first calculates the data utility based on how incoming new data collected by a UAV or a UAV group improves the up-to-dateness of grid points. Then, considering both the data utility and the traveling distance, it selects a cost-effective grid point to visit for the next timeslot in a greedy manner, using a point-mass mobility model.

A. CALCULATING DATA UTILITY

Based on the data freshness measure Equation 7 over all of the cells in the RoI, we calculate an expected data utility by sensing the cells within the sensing range if a UAV or a UAV group visits a specific grid point. We expect to earn more utility if a UAV or a UAV group determines a grid point for which the cells within the sensing range have the lowest data freshness. The data utility metric at the current time t is calculated by accumulating the complement of data freshness over relevant cells, $1 - dataFreshness_t(c)$:

$$dataUtility_t(g) = \sum_{c \in coverableCellList(g)} (1 - dataFreshness_t(c)) \quad (10)$$

As in Figure 6, a grid point candidate located at (5, 4) has the $dataUtility_t(g)$ value of $8 \times (1 - 0) = 8$. The data utility for grid points outside the RoI covered by part of the UAVs is assigned to 0.

B. SELECTING THE NEXT DESTINATION OF A CONNECTED UAV GROUP

If we consider only the data utility measure to determine a grid point with the largest value, irrespective of its location from the grid point currently being visited, a UAV or a connected UAV group may wander around the RoI for greedily pursuing high profit without considering any loss incurred during the procedure. To balance between profit and loss when choosing a cost-effective next grid point to visit, we introduce a penalty factor of distance, to reflect relocation effort. Let us define $priorityScore_t(g)$ for a grid point candidate g at the current time t as follows:

$$priorityScore_t(g) = dataUtility_t(g) - \beta \cdot D(g_{cur}, g) \quad (11)$$

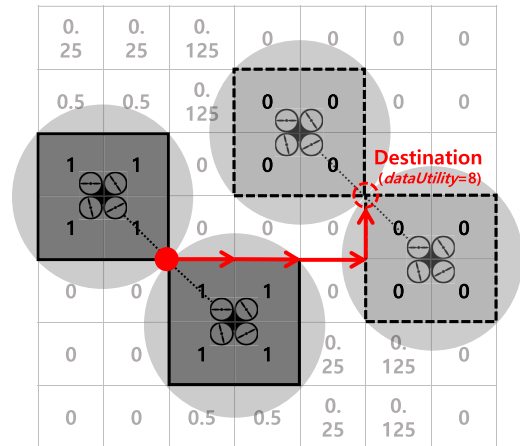


FIGURE 6. An example of finding the next grid point for a group of two UAVs to visit by calculating the $priorityScore$ based on the $dataFreshness$ of each cell.

where β is a penalizing factor, and $D(g_{cur}, g)$ is the total number of passing edges required to move from the currently visiting grid point g_{cur} to a grid candidate g , in the xy grid coordinate.

The master UAV calculates the $priorityScore_t(g)$ of all possible grid point candidates, and selects the grid point with the largest value. If there are multiple candidates with the same maximum score, it randomly selects one of them. Once the next destination of the UAV group centroid is chosen, the destination grid points of all UAVs belonging to swarm are calculated. The connected UAV group leaves the current location for the calculated destination. It continues to travel along the shortest path by moving in a specific direction (north, east, south, or west), while keeping its formation until it arrives at the destination. The overall traversal procedure is described in Algorithm 3.

VI. EVALUATION

We evaluate our persistent data collection scheme over an RoI of $300 \times 300 m^2$ with a virtual grid topology of 75×75 cells of size $4 \times 4 m^2$ in a simulation environment (Table 2). The UAV flying speed is assumed to be $11 m/s$ (as per the Parrot AR.Drone 2.0 specification), and an altitude of $4 m$ is used in the simulation experiments. A timeslot (ts) is defined as the time to fly from one grid point to another adjacent grid point. A communication radio range of $30 m$ and a sensing range of $10 m$ are used in the experiments.

The design parameters β , and W are set to 10^{-5} , and 200 timeslots, respectively. The penalty base α ranges from 0 to 1, and is a fixed environmental factor determined by the application type, where an α value closer to 1 indicates that the data utility is less sensitive over time. If α is near 0, the data utility is more sensitively affected by time lapse, and the selection of the next visiting point is more affected by its physical distance, as well as the acquisition of timely information.

Algorithm 3 traversal() by a Connected UAV Group

```

1: Input: this: This UAV, t: Current time, uavGroup(this):
   UAV group which this UAV belongs to
2: Output: nextGridpointToVisit
   // I. Find a new destination only when there is no designat-
   ed destination
3: if isEmpty(dest) == true then
4:   Calculate dataFreshnesst, dataUtilityt, and
   priorityScoret;
5:   if there are multiple grid points with the maximum
   priorityScore then
6:     Randomly select one grid point among them and
     update dest;
7:   else
8:     dest = a grid point with the maximum
     priorityScore;
9:   end if
10: end if
   // II. Find a next passing grid point toward the selected
   destination
11: Calculate the number of edges in the xy grid coordinate
   toward dest;
12: passingGridpoint = a passing grid point on the shortest
   path toward dest;
13: if there are multiple passingGridpoint then
14:   Randomly select one among them and update
   passingGridpoint;
15: else
16:   nextGridpointToVisit = passingGridpoint;
17: end if

```

TABLE 2. Simulation environment and parameters.

Simulation Environment	
Territory Area	300 × 300 m ²
Grid Size	4 m
Timeslot <i>ts</i>	0.36 sec
UAV Speed	11 m/s
UAV Altitude	4 m
# of UAVs	1~10
Communication Radio Range	30 m
Sensing Range	10 m
Simulation Parameter	
Penalty Base α in <i>dataFreshness</i>	0.99
Penalty Slope β in <i>priorityScore</i>	10 ⁻⁵
Reformation Period <i>W</i> in <i>DiagonalX</i>	200 <i>ts</i>

We evaluate dynamic data collection performance in terms of the *dataFreshness*, network overhead, and fairness. We run 10 different simulations with randomly selected starting points and quantify the performance by taking the average value.

We validate our approach from the two perspectives of network formation and traversal. To verify how good a specific formation is in terms of data freshness in each cell, our proposed *DiagonalX* is compared against the formation shapes *Straight*, *Circle*, and *Diagonal*. *Straight* is a formation where UAVs in a group are aligned horizontally along the *X*-axis. *Circle* is one of the most popular formation shapes, considered to be an optimal compact packing algorithm [30]. *Diagonal* is a preliminary version of *DiagonalX* with the fixed up-right diagonal shape without any alternating reformation.

To examine the performance depending on the traversal decision, we compared our traversal algorithm with a classic TSP algorithm [31], which finds a sub-optimal path for a single agent to cover the whole area while minimizing its travel time. We let the TSP-based traversal run alone without forming a group, called *TSP w/o Group*, and also ran it combined with our *DiagonalX* formation, called *TSP w/ DiagonalX*.

We investigate how the data freshness measure improves over time using eight UAVs in our *DiagonalX* (Figure 7). We show the minimum, the average, and the maximum of the *dataFreshness* measure among nine UAVs. As UAVs encounter each other and start forming a group with the *DiagonalX* formation, at 37, 41, 79, and 95 *ts*, the data freshness score significantly improves. Once all of the UAVs are united, at 108 *ts*, they maintain the same cell information after forming a swarm, with the three minimum, average, maximum values the same. Since our *DiagonalX* alternates its diagonal shape with every 200 *ts*, some oscillatory behavior is observed, improving the data freshness by changing to another diagonal shape to better cover the otherwise less-prioritized areas. Eventually, at 1000 *ts*, the data freshness reaches around 0.42, and the average elapsed time since the last visit at each grid point is only 85.6 *ts* ($0.42 \approx \alpha^{85.6}$).

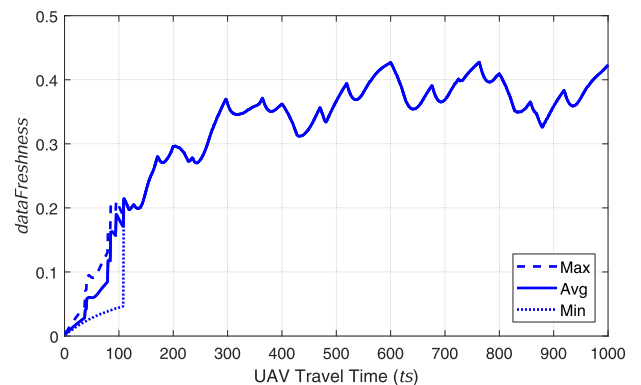


FIGURE 7. Performance dynamics of the data freshness measure over UAV travel time with eight UAVs using *DiagonalX*.

We visualize how recently a cell is visited and show its cumulative distribution in Figure 8. In Figures 8(a) and 8(c), a normalized time elapsed at present, 1000 *ts* is linearly mapped to a 0-1 grey level scale with 0 as black and 1 as white, while unvisited cells are colored red. The *Circle*

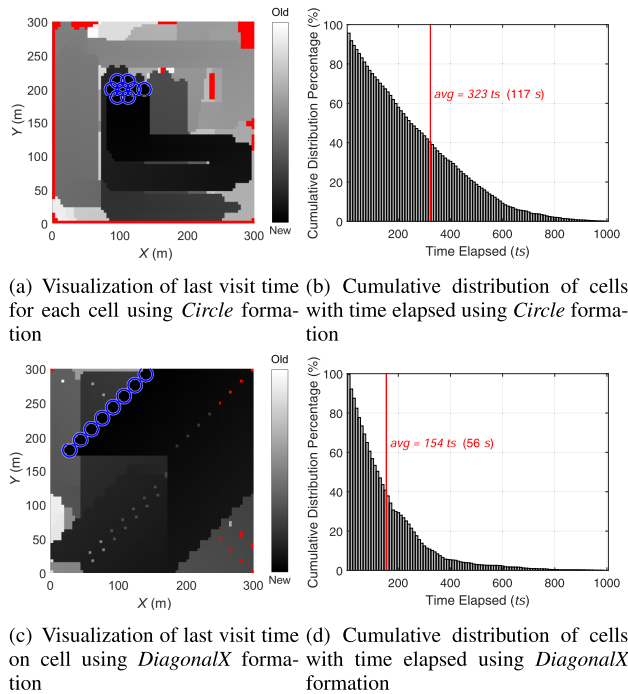


FIGURE 8. Visualization and cumulative distribution of 75×75 cells based on the last visit time using the *Circle* and *DiagonalX* formations with eight UAVs at 1000 ts. The darker grey color indicates more recently visited cells, while the red colored cells show never-visited ones, and the final sensing coverage of UAVs is shown with a blue circle.

formation results in 283 unvisited cells, 5.03 % of the total 75×75 cells, whereas the *DiagonalX* formation has only 13 unvisited cells, which is 0.23 %. *DiagonalX* has larger dark areas in broader regions than *Circle*. If the cells are sorted in the order of time elapsed and accumulated, *DiagonalX* has a significantly larger portion of cells with less time elapsed compared to *Circle*. In *DiagonalX*, the average time elapsed since the last visit time at a cell is 154 ts, which corresponds to 56 s. This observation indicates that an alternating diagonal formation within a group enables the collection and maintenance of data that were collected only 56 seconds ago on average from all of the cells.

We compare our algorithm with seven different counterpart algorithms in terms of data freshness and network overhead in Figure 9. We quantify network overhead using two categories, control and data. The control overhead includes all packet transmissions for 1) sending a periodic hello packet to check the presence of nearby UAVs, 2) notifying a formation position, and 3) informing about the next cell to visit for traversal. The data overhead quantifies all packet transmissions for data exchange upon receiving a data query at 1000 ts except for *w/o Group*, which should receive data as soon as the UAVs are encountered, because they traverse individually without forming a group.

As Figure 9(a) shows, *DiagonalX* achieves the highest *dataFreshness*, outperforming all the other algorithms by a factor of up to 5.77. This observation implies that forming a group and making a specific formation shape is a key to

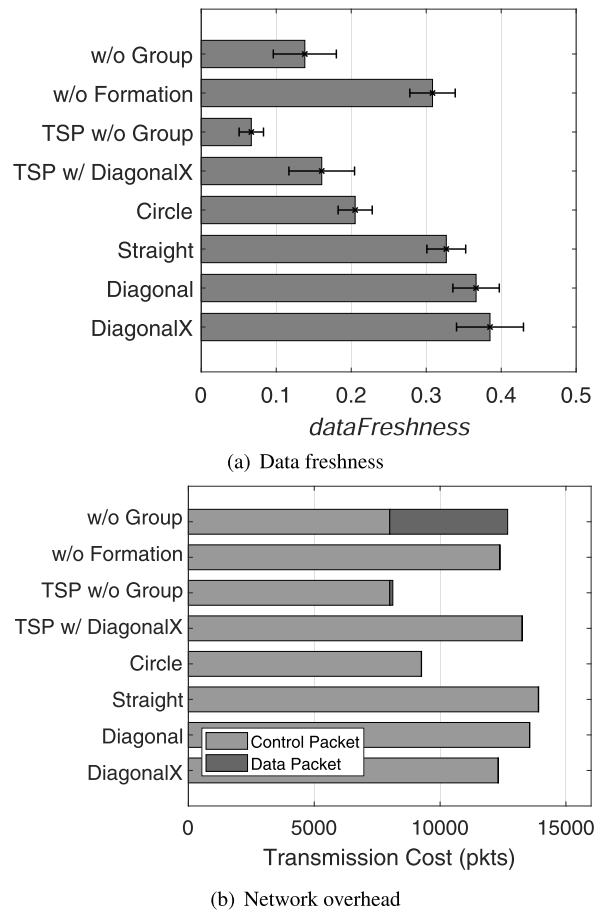


FIGURE 9. Network performance in terms of data freshness and network overhead using eight UAVs at 1000 ts.

maintaining a collection of the most up-to-date local data. The performance gap between the *w/o Group* and the *TSP w/o Group* is due to the fact that the *w/o Group*, which is our proposed traversal algorithm without forming a group, still works well, since it prefers to visit some common cells more often than the others and localizes its movement within the area, making more frequent encounters possible.

As shown in Figure 9(b), *DiagonalX* attains a similar network overhead as the other algorithms. Although *Circle* achieves relatively low network overhead compared to *DiagonalX*, due to its innate compact formation toward a center in terms of information exchange, it turns out not to be an efficient formation in traversal for persistent data collection. Also, the performance disparity between *TSP w/ DiagonalX* and *DiagonalX* shows that a desirable traversal algorithm focuses not only on travel efficiency, but also on information up-to-dateness.

We examine the effectiveness of our algorithm using another measure of the timeliness of information called *peak age of information (peak AoI)* [22]. The peak AoI is defined as the worst-case age of information at target cell i , denoted as A_i^p . When a UAV visits cell i , A_i is set to 1, and the peak value occurs immediately before visiting the cell.

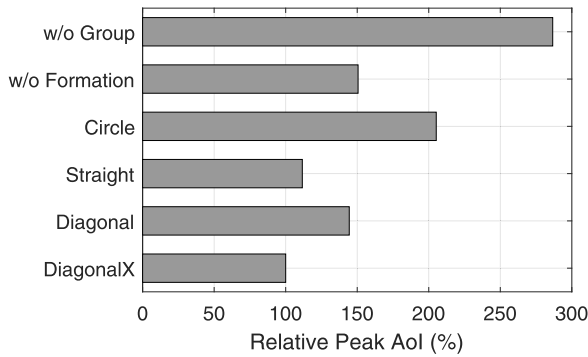
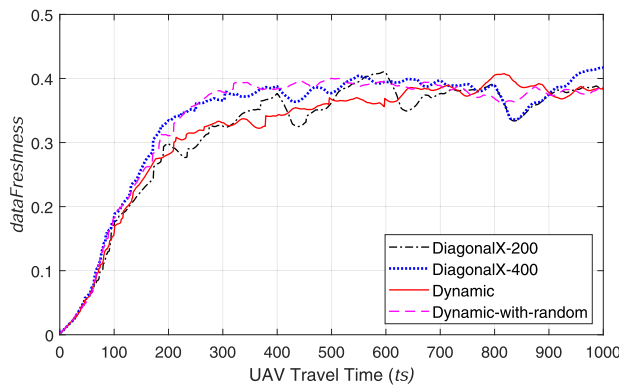
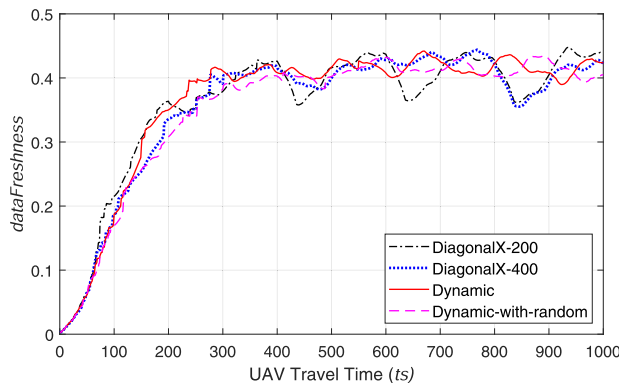


FIGURE 10. Relative peak age of information normalized by the *DiagonalX* formation performance using eight UAVs at 1000ts.



(a) 8 UAVs



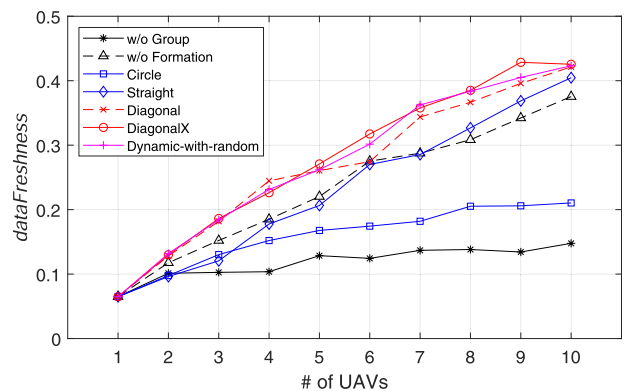
(b) 10 UAVs

FIGURE 11. Dynamic data freshness performance of *DiagonalX* with W of 200ts, 400ts, *Dynamic* with a fixed direction, and *Dynamic-with-random* with a random direction, with respect to UAV travel time.

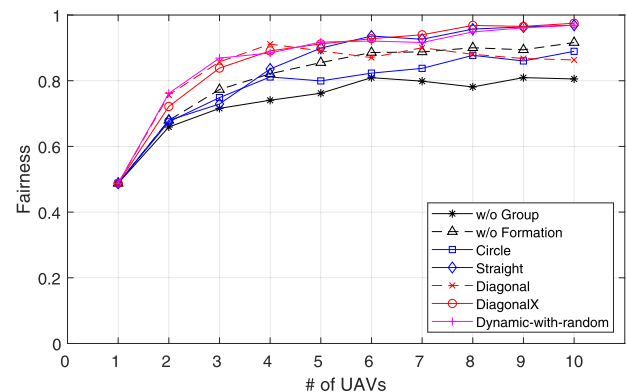
We measure the relative peak age normalized with the *DiagonalX* performance after 1000ts, and take the average of 10 different cases. As shown in Figure 10, *DiagonalX* outperforms the other formations: *w/o Group*, *w/o Formation*, *Circle*, *Straight*, and *Diagonal*, by a factor of 2.86, 1.50, 2.05, 1.11, and 1.44, respectively. The result shows that *DiagonalX* formation along with the proposed traversal method is suitable for both rapidly decaying data collection and age-sensitive data collection scenarios.

Lastly, we investigate the effect of dynamic reformation called *Dynamic* on top of *DiagonalX*. We show two types of *Dynamic*: with a fixed, or with a random direction of rotation for the reformation transition. As shown in Figure 11(a) and Figure 11(b), *Dynamic* and *Dynamic-with-random* show relatively more stable performance with less fluctuation, compared to *DiagonalX* with a constant reformation period W .

We examine the way in which the number of participating UAVs affects the data freshness and fairness in Figure 12. We vary the number of UAVs from 1 to 10, and measure those metrics at 1000 ts. For the fairness metric, we use Jain's fairness index [32] ranging from $1/(\# \text{ of cells})$ (worst case) to 1 (best case) to calculate the latest visit time, normalized by the end time at 1000 ts for all the cells. A larger fairness index means a fairer number of visits to cells on average.



(a) Data freshness



(b) Fairness of the normalized latest visit time among cells

FIGURE 12. Data freshness and fairness performance with respect to the number of UAVs.

Since the TSP-based approaches of *TSP w/o Group* and *TSP w/ DiagonalX* show the poorest performance, we report the results of all other algorithms except these two. As in Figure 12(a), we validate that flying in a swarm is essential for keeping data as up-to-date as possible. This is because the *w/o Group* shows the worst marginal performance improvement, even after engaging more UAV resources. Among the various formations within a group, *DiagonalX* achieves the highest data freshness, with a factor of 2.08 compared with *Circle* (with a factor of 3.19 compared with *w/o Group*). In the case

of sufficient UAV resource usage relative to the size of the RoI (e.g., 10 UAVs), most of the formation-based approaches show similar performance compared to *DiagonalX*, regardless of the formation shape. In Figure 12(b), both *DiagonalX* and *Straight* show the highest fairness. This finding implies that, considering both data freshness and fairness aspects, *DiagonalX* formation keeps covering a given broad sensing area with high fidelity and fairness, while incurring a reasonable network overhead. A further improvement with adaptive change in the reformation period via *Dynamic-with-random* produces both data freshness and fairness that are evenly higher than the other approaches across the number of UAVs.

We also investigate the effect of RoI size on *dataFreshness*, using eight UAVs at 1000 *ts* for 10 different cases (Figure 13). It is inevitable that *dataFreshness* declines as the RoI size increases, because a given number of UAVs are not sufficient to cover a larger area. However, we demonstrate that *DiagonalX* and *Dynamic-with-random* show stable up-to-date data collection performance compared to the other approaches.

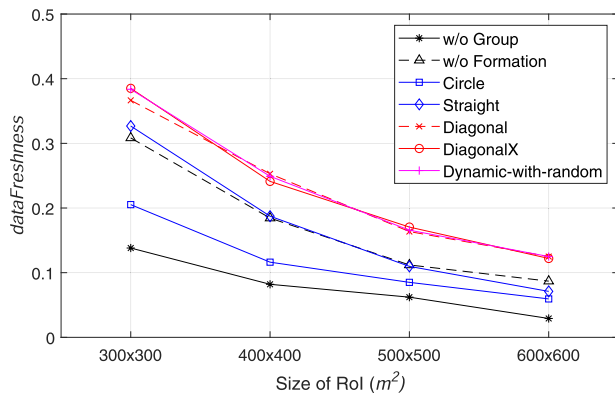


FIGURE 13. Data freshness performance with respect to the size of RoI using eight UAVs.

VII. CONCLUSION

We have presented a lightweight yet effective data collection approach using UAVs for persistent surveillance and reconnaissance. We engage a connected swarm of multiple UAVs with two phases – network formation and traversal – to keep information up-to-date over time evenly throughout the RoI.

We have validated our original claims: 1) a swarm movement with UAVs connected within a group was shown to be better than an individual UAV for maintaining persistent data collection over time; 2) a diagonal formation with alternating reformation either periodically or adaptively, for even fairer coverage across the areas, is a simple yet efficient swarm structure, and is beneficial for navigation control and data sharing.

For future work, we may extend our solution to the persistent data collection problem in two dimensions to three dimensions, in an RoI consisting of cluttered buildings or obstacles with different heights. A computationally feasible path planning algorithm over a more complex search

space would have to be derived under practical constraints by taking into account the control, navigation, and communication dynamics for more resilient UAV swarm management.

REFERENCES

- [1] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *Proc. 1st ACM Int. Workshop Wireless Sensor Netw. Appl. (WSNA)*, 2002, pp. 88–97.
- [2] M. Rebai, M. Le Berre, H. Snoussi, F. Hnaïen, and L. Khoukhi, "Sensor deployment optimization methods to achieve both coverage and connectivity in wireless sensor networks," *Comput. Oper. Res.*, vol. 59, pp. 11–21, Jul. 2015.
- [3] Y.-C. Wang, C.-C. Hu, and Y.-C. Tseng, "Efficient deployment algorithms for ensuring coverage and connectivity of Wireless sensor networks," in *Proc. 1st Int. Conf. Wireless Internet (WICON)*, 2005, pp. 114–121.
- [4] G. Wang, G. Cao, and T. F. La Porta, "Movement-assisted sensor deployment," *IEEE Trans. Mobile Comput.*, vol. 5, no. 6, pp. 640–652, Jun. 2006.
- [5] W. Li and W. Zhang, "Coverage hole and boundary nodes detection in wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 48, pp. 35–43, Feb. 2015.
- [6] I. Maza and A. Ollero, "Multiple UAV cooperative searching operation using polygon area decomposition and efficient coverage algorithms," in *Distributed Autonomous Robotic Systems 6*. Tokyo, Japan: Springer, 2007.
- [7] S. Waharte, N. Trigoni, and S. Julier, "Coordinated search with a swarm of UAVs," in *Proc. 6th IEEE Annu. Commun. Soc. Conf. Sensor, Mesh Ad Hoc Commun. Netw. Workshops*, Jun. 2009, pp. 1–3.
- [8] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Mobile unmanned aerial vehicles (UAVs) for energy-efficient Internet of Things communications," *IEEE Trans. Wireless Commun.*, vol. 16, no. 11, pp. 7574–7589, Nov. 2017.
- [9] H. Shakhatreh, A. Khreishah, A. Alsarhan, I. Khalil, A. Sawalmeh, and N. S. Othman, "Efficient 3D placement of a UAV using particle swarm optimization," in *Proc. 8th Int. Conf. Inf. Commun. Syst. (ICICS)*, Apr. 2017, pp. 258–263.
- [10] M. Mozaffari, A. T. Z. Kasgari, W. Saad, M. Bennis, and M. Debbah, "Beyond 5G with UAVs: Foundations of a 3D wireless cellular network," *IEEE Trans. Wireless Commun.*, vol. 18, no. 1, pp. 357–372, Jan. 2019, doi: 10.1109/TWC.2018.2879940.
- [11] L. Lin and M. A. Goodrich, "UAV intelligent path planning for wilderness search and rescue," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2009, pp. 709–714.
- [12] S. G. Manyam, D. W. Casbeer, and K. Sundar, "Path planning for cooperative routing of air-ground vehicles," in *Proc. Amer. Control Conf. (ACC)*, Jul. 2016, pp. 4630–4635.
- [13] A. A. Somasundara, A. Ramamoorthy, and M. B. Srivastava, "Mobile element scheduling for efficient data collection in wireless sensor networks with dynamic deadlines," in *Proc. 25th IEEE Int. Real-Time Syst. Symp.*, 2004, pp. 296–305.
- [14] S. Alamdari, E. Fata, and S. L. Smith, "Persistent monitoring in discrete environments: Minimizing the maximum weighted latency between observations," *Int. J. Robot. Res.*, vol. 33, no. 1, pp. 138–154, Jan. 2014.
- [15] F. Pasqualetti, A. Franchi, and F. Bullo, "On cooperative patrolling: Optimal trajectories, complexity analysis, and approximation algorithms," *IEEE Trans. Robot.*, vol. 28, no. 3, pp. 592–606, Jun. 2012.
- [16] V. T. Hoang, M. D. Phung, T. H. Dinh, and Q. P. Ha, "System architecture for real-time surface inspection using multiple UAVs," *IEEE Syst. J.*, vol. 14, no. 2, pp. 2925–2936, Jun. 2020.
- [17] Y. Wang, P. Bai, X. Liang, W. Wang, J. Zhang, and Q. Fu, "Reconnaissance mission conducted by UAV swarms based on distributed PSO path planning algorithms," *IEEE Access*, vol. 7, pp. 105086–105099, 2019.
- [18] A. F. A. Fadzliah, "A mathematical model for measuring the timeliness of handheld application usage," in *Proc. Int. Conf. Informat. Appl. (ICIA)*, 2012, pp. 302–308.
- [19] B. Heinrich, M. Kaiser, and M. Klier, "How to measure data quality? A metric-based approach," in *Proc. 28th Int. Conf. Inf. Syst. (ICIS)*, S. Rivard and J. Webster, Eds. Montreal, QC, Canada: Queen's Univ., 2007, pp. 1–15. [Online]. Available: <https://epub.uni-regensburg.de/23633/>
- [20] S. Kaul, M. Gruteser, V. Rai, and J. Kenney, "Minimizing age of information in vehicular networks," in *Proc. 8th Annu. IEEE Commun. Soc. Conf. Sensor, Mesh Ad Hoc Commun. Netw.*, Jun. 2011, pp. 350–358.
- [21] A. Kosta, N. Pappas, and V. Angelakis, *Age of Information: A New Concept, Metric, and Tool*. Delft, The Netherlands: Now, 2017.

[22] V. Tripathi, R. Talak, and E. Modiano, "Age optimal information gathering and dissemination on graphs," in *Proc. IEEE INFOCOM-IEEE Conf. Comput. Commun.*, Apr. 2019, pp. 2422–2430, doi: [10.1109/INFOCOM.2019.8737642](https://doi.org/10.1109/INFOCOM.2019.8737642).

[23] S.-C. Han, H. Bang, and C.-S. Yoo, "Proportional navigation-based collision avoidance for UAVs," *Int. J. Control, Automat. Syst.*, vol. 7, no. 4, pp. 553–565, 2009.

[24] M. R. Brust and B. M. Strimbu, "A networked swarm model for UAV deployment in the assessment of forest environments," in *Proc. IEEE 10th Int. Conf. Intell. Sensors, Sensor Netw. Inf. Process. (ISSNIP)*, Apr. 2015, pp. 1–6.

[25] O. Bouachir, A. Abrassart, F. Garcia, and N. Larrieu, "A mobility model for UAV ad hoc network," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, May 2014, pp. 383–388.

[26] H. Gao, C. Liu, Y. Li, and X. Yang, "V2 VR: Reliable hybrid-network-oriented V2 V data transmission and routing considering RSUs and connectivity probability," *IEEE Trans. Intell. Transp. Syst.*, early access, Apr. 13, 2020, doi: [10.1109/TITS.2020.2983835](https://doi.org/10.1109/TITS.2020.2983835).

[27] H. Gao, Y. Duan, L. Shao, and X. Sun, "Transformation-based processing of typed resources for multimedia sources in the IoT environment," *Wireless Netw.*, pp. 1–7, Nov. 2019, doi: [10.1007/s11276-019-02200-6](https://doi.org/10.1007/s11276-019-02200-6).

[28] I. Jawhar, N. Mohamed, J. Al-Jaroodi, D. P. Agrawal, and S. Zhang, "Communication and networking of UAV-based systems: Classification and associated architectures," *J. Netw. Comput. Appl.*, vol. 84, pp. 93–108, Apr. 2017.

[29] Y. Sun, E. Uysal-Biyikoglu, R. D. Yates, C. E. Koksal, and N. B. Shroff, "Update or wait: How to keep your data fresh," *IEEE Trans. Inf. Theory*, vol. 63, no. 11, pp. 7492–7508, Nov. 2017.

[30] R. L. Graham, B. D. Lubachevsky, K. J. Nurmela, and P. R. J. Östergård, "Dense packings of congruent circles in a circle," *Discrete Math.*, vol. 181, nos. 1–3, pp. 139–154, Feb. 1998.

[31] J. Kirk. (2014). *Traveling Salesman Problem—Genetic Algorithm*. [Online]. Available: <https://www.mathworks.com/matlabcentral/fileexchange/13680-traveling-salesman-problem-genetic-algorithm>

[32] R. K. Jain, D.-M. W. Chiu, and W. R. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer system," *ACM Trans. Comput. Syst.*, vol. 2, no. 1, p. 301, 1984.



JAEYI SUNG is currently pursuing the B.S. degrees in computer science and engineering, and psychology with Ewha Womans University, Seoul, South Korea. Her current research interests include wireless sensor networks, machine learning, and deep learning.



JINYI YOON (Student Member, IEEE) received the B.S. and M.S. degrees in computer science and engineering from Ewha Womans University, Seoul, South Korea, in 2017 and 2019, respectively, where she is currently pursuing the Ph.D. degree. Her current research interests include wireless *ad hoc* networks, vehicular networks, the IoT security systems, and deep learning.



HYUNGJUNE LEE (Member, IEEE) received the B.S. degree in electrical engineering from Seoul National University, Seoul, South Korea, in 2001, and the M.S. and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA, USA, in 2006 and 2010, respectively. He joined Broadcom as a Senior Staff Scientist for working on research and development of 60 GHz 802.11ad SoC MAC. He also worked for AT&T Laboratories as a Principal Member of Technical Staff with the involvement of LTE overload estimation, LTE-WiFi interworking, and heterogeneous networks. He is currently an Associate Professor with the Department of Computer Science and Engineering, Ewha Womans University. His current research interests include future wireless networks on the IoT, fog computing, VANET, and machine learning-driven network system design.



JEIHEE CHO (Student Member, IEEE) is currently pursuing the B.S. degree in computer science and engineering with Ewha Womans University, Seoul, South Korea. Her current research interests include the IoT, sensor networks, wireless *ad hoc* networks, and deep learning.

...