Energy-Efficient UAV Routing for Wireless Sensor Networks

Jaeuk Baek, *Student Member, IEEE*, Sang Ik Han, *Member, IEEE*, and Youngnam Han, *Senior Member, IEEE*

***Abstract*—Recently, an unmanned aerial vehicle (UAV) has been widely adopted to make efficient use of network resources in such areas as internet of things (IoT), sensor networks and three dimensional (3D) wireless networks. Especially, in wireless sensor networks (WSNs) where energy consumption of sensors in data transmission is the most conspicuous feature, data collection by UAV provides a promising solution. To address this issue, we consider a UAV-enabled WSN, where a UAV is dispatched to collect data from sensors distributed in networks. We formulate an optimization problem to maximize the minimum residual energy of sensors after data transmission for energy-efficient UAV routing subject to data collection and UAV traveling distance constraints. To solve the non-convex optimization problem, we first derive a feasible solution, i.e., the shortest UAV route that guarantees data collection at all the sensors, where a Voronoi diagram is modified to find a set of UAV hovering locations. The proposed algorithm preferentially determines each UAV hovering location at Voronoi vertex so that UAV can collect data from as many adjacent sensors as possible. Then with an initial shortest UAV route, a UAV route is proposed by adjusting each UAV hovering location sequentially based on sensor energy status, which is easily accomplished by the properties of Voronoi diagram. Lastly, to find the proposed solution more quickly, we propose a sensor-energy based initial UAV route determination method. Simulation results are provided to validate the performance of our proposed algorithm, and to compare with other UAV route determination schemes.**

***Index Terms*—UAV flight route, Wireless sensor network, Voronoi diagram, Energy-efficient data collection.**

I. Introduction

IN WIRELESS sensor networks (WSNs), a mobile data collector has been adopted for energy-efficient data transmission, particularly when sensors are sparsely spread over wide area [1]–[3]. The mobile collector moves close to each sensor to gather data and sends it to a remote control center, where each sensor does not need to communicate with the control center directly, resulting in significant energy savings in data transmission.

Manuscript received September 3, 2019; revised December 2, 2019; accepted December 10, 2019. Date of publication December 16, 2019; date of current version February 12, 2020. The review of this article was coordinated by Dr. B. Mao. *(Corresponding author: Youngnam Han.)*

J. Baek and Y. Han are with the School of Electrical Engineering, Korea Advanced Institute of Science and Technology, Daejeon 34141, South Korea (e-mail: doloveqw@kaist.ac.kr; ynhan@kaist.ac.kr).

S. I. Han is with Advanced Unmanned Vehicle Research Center (AUVRC) at Spacesoft Industries Co., Ltd., 34013 Daejeon, South Korea (e-mail: si.han@spacesofts.com).

Digital Object Identifier 10.1109/TVT.2019.2959808

Recently, an unmanned aerial vehicle (UAV) has drawn significant research interest as a flying data collector [4]–[6]. Compared to a ground data collector (e.g., mobile robot) whose mobility is limited by a number of obstacles, UAVs can be swiftly deployed and moved in three-dimensional (3D) free space. In addition, UAV enables short-range communication with ground nodes via a strong air-to-ground line-of-sight (LoS) link, which increases signal transmission reliability [7], [8]. However, overall UAV traveling distance is restricted due to the limited on-board energy of UAV [9], [10]. Hence, UAV route should be determined to guarantee a return trip after a UAV collects data from all the sensors.

UAV route design depends on the types of UAV, such as fixed-wing or rotary-wing UAVs. Fixed-wing UAVs can remain aloft while moving forward, so circular or rectilinear UAV routes are designed to provide wide coverage and backhaul links to other nodes through multi-hop transmission [11]–[14]. On the other hand, rotary-wing UAVs can flexibly move in any direction and hover at a given location, hence their route can be determined randomly over a surveillance region [15] or designed by finding and connecting a set of UAV hovering locations [16]–[18]. In WSNs where sensors are randomly deployed over surveillance area, it is better to utilize rotary-wing UAVs than fixed-wing UAVs for efficient data collection, since rotary-wing UAVs can move adaptively according to sensor distribution. In addition, by focusing on UAV hovering location that affects both sensor's energy consumption for data transmission and UAV traveling distance, UAV route can be optimized to improve network performance, especially to extend the lifetime of WSNs.

There are several studies to determine UAV hovering locations [16]–[18]. The "sensor visiting" approach is to utilize given sensor locations [16], where a UAV can visit each sensor node to minimize communication distance between the UAV and each sensor. This method extends network lifetime by minimizing energy consumption of all sensors for data transmission, but increases UAV traveling distance significantly. To reduce UAV traveling distance, some sensors are selected as cluster heads to transmit data of cluster members [17], [18]. But their proposed routes significantly increase energy consumption at cluster head sensors, and cannot be applicable when direct communication between sensors are not available.

Another approach is to determine UAV hovering locations considering sensor's communication range [19]–[24]. In this approach, UAV hovering locations are determined so that the UAV at each hovering location can communicate with multiple adjacent sensors by time-division multiple access (TDMA).

However, it is nontrivial to determine UAV route since all UAV hovering locations are interwined each other. [19] utilizes Welzl's algorithm [25] to find a set of sensors that can share a single hovering location. Then, each UAV hovering location is optimally modified to shorten UAV traveling distance without considering energy consumption of sensors in data transmission. [20]–[22] utilize a Voronoi diagram, where Voronoi edges are chosen and connected to find a set of closed UAV routes. Then, a genetic algorithm [26] and a k-shortest path finding method [27] are applied to find the shortest UAV route. However, the resultant UAV route is not optimal because for any given two hovering locations, combining Voronoi edges generally results in longer UAV route compared with a straight line connecting them. A grid-based map [23] and the geometry of sensor locations [24] have been proposed to find UAV hovering locations, but computationally inefficient due to exhaustive search.

In this paper, we propose a UAV route determination algorithm by modifying a Voronoi diagram reflecting sensor energy information for WSNs, where the UAV hovering location for low-energy sensors is preferentially placed to prolong network lifetime. Compared with other Voronoi diagram based UAV routes [20]–[22], ours provides shorter UAV route. Major results of this paper are as follows.

Firstly, we formulate a problem to maximize the minimum residual energy of sensors after data transmission under data collection and UAV traveling distance constraints. Our objective is to determine UAV route for energy savings in many sensors, which consequently results in energy-efficient data collection and longer network lifetime.

Second, to find feasible solutions, we consider the shortest UAV route guaranteeing data collection at all sensors. Intuitively, the number of UAV hovering locations supporting multiple nodes needs to be maximized to reduce overall UAV traveling distance. To this end, our proposed algorithm utilizes a modified Voronoi diagram to place UAV hovering locations at Voronoi vertex, Voronoi edge and the maximum horizontal communication distance from sensors. It is worth noting that we construct a modified Voronoi diagram so that each sensor can communicate with the UAV at the assigned hovering location.

Next, each UAV hovering location of the shortest UAV route is adjusted based on sensor energy status and Voronoi diagram. Since the adjustment increases UAV traveling distance, the route optimization procedure repeats until the UAV traveling distance constraint is met. Lastly, we utilize sensor energy information to find an initial UAV route for the efficient and faster solution.

The rest of our paper is organized as follows. Sec. II presents our system model and optimization problem of UAV-enabled WSN. Sec. III presents how feasible solutions are obtained. Sec. IV presents an algorithm to find the proposed UAV route among those feasible solutions. Sec. V provides numerical results to validate the performance of our proposed algorithm. Sec. VI draws conclusions.

Notations: Boldface letters refer to set. $\mathbf{A} \cup \mathbf{B} \triangleq \{x | x \in \mathbf{A} \text{ or } x \in \mathbf{B}\}$ and $\mathbf{A} \setminus \mathbf{B} \triangleq \{x \in \mathbf{A} | x \notin \mathbf{B}\}$. $\emptyset$ refers to an empty set. $d_{x,y}$ represents distance between two points, x and y.

TABLE I
EXAMPLES OF MAPPING BETWEEN SENSORS AND UAV HOVERING LOCATIONS

$N = 6$ and $K = 3$. UAV route: $(l_0, l_1, l_2, l_3, l_0)$.					
Sensor 1	Sensor 2	Sensor 3	Sensor 4	Sensor 5	Sensor 6
l_1			l_2		l_3
$\pi(1) = 1$	$\pi(2) = 1$	$\pi(3) = 1$	$\pi(4) = 2$	$\pi(5) = 2$	$\pi(6) = 3$

$N = K = 6$. UAV route: $(l_0, l_1, l_2, l_3, l_4, l_5, l_6, l_0)$.					
Sensor 1	Sensor 2	Sensor 3	Sensor 4	Sensor 5	Sensor 6
l_2	l_1	l_3	l_4	l_6	l_5
$\pi(1) = 2$	$\pi(2) = 1$	$\pi(3) = 3$	$\pi(4) = 4$	$\pi(5) = 6$	$\pi(6) = 5$

II. SYSTEM MODEL

We consider a UAV-enabled sparsely distributed WSN, where a rotary-wing UAV flies at a fixed height H to collect data from sensors deployed randomly. Depending on sensors' capability to exchange their data, UAV route can be designed differently. When no direct communication among sensors is available, UAV route needs to be determined for each node. On the other hand, when multi-hop transmission among sensors is possible, some sensors can be selected as sink nodes [17], [18], which can be considered as nodes in the case above. So, in general we assume that there are N sensors (or sink nodes) distributed in area $\mathbf{S}$. Each sensor $n \in \mathbf{N} \triangleq \{1, \ldots, N\}$ is located at a fixed location X_n. We assume that no direct communication link between any sensor and a control center is available.

Denote $P_{\boldsymbol{L}_\pi} \triangleq (l_0, l_1, \ldots, l_K, l_0)$ as the route traversed by the UAV during a single-round trip, which starts from and ends at the control center at l_0. $l_k \in \boldsymbol{L}_\pi \triangleq \{l_1, \ldots, l_K\}$ is the k^{th} visiting UAV hovering location, where $K \leq N$ since UAV at a single UAV hovering location can collect data from adjacent multiple sensors. The overall UAV traveling distance of a route $P_{\boldsymbol{L}_\pi}$ can be expressed as $D_{\boldsymbol{L}_\pi} \triangleq \sum_{i=0}^{K} d_{l_i, l_{i+1}}$, where $d_{l_i, l_{i+1}}$ is distance between l_i and l_{i+1}, and $l_0 = l_{K+1}$ for convenience.

We assume that sensor n transmits data to UAV at $l_{\pi(n)}$, where $\pi(n) \in \{1, \ldots, K\}$. That is, $l_{\pi(n)}$ is the $\pi(n)^{th}$ UAV hovering location to which data is transmitted from a sensor n (see Table I). It is worth noting that $\boldsymbol{L}_\pi$, $P_{\boldsymbol{L}_\pi}$ and $D_{\boldsymbol{L}_\pi}$ depend on the way how each $\pi(n)$ is determined.

We assume line-of-sight (LoS) link between the UAV and a sensor [28]. When a sensor n transmits data to UAV at $l_{\pi(n)}$, the signal to noise ratio (SNR) at the UAV is expressed as

$$SNR_n = \frac{p_n}{\tilde{d}^2_{X_n, l_{\pi(n)}} + H^2}, \tag{1}$$

where the free-space path loss model with pathloss exponent of 2 is assumed. p_n is the transmit power at sensor n. $\tilde{d}_{X_n, l_{\pi(n)}}$ and $\sqrt{\tilde{d}^2_{X_n, l_{\pi(n)}} + H^2}$ represent horizontal distance and communication distance between the UAV at $l_{\pi(n)}$ and sensor n, respectively.

Based on the first-order radio hardware energy dissipation model [1], we assume that a sensor n controls its transmit power p_n to guarantee an acceptable SNR at the UAV. Therefore, the

energy consumption of sensor n for transmitting b_n bits over a $\sqrt{\tilde{d}^2_{X_n,l_{\pi(n)}}+H^2}$ is written as

$$f(b_n, \sqrt{\tilde{d}^2_{X_n,l_{\pi(n)}}+H^2}) = b_n\left(\varepsilon_{el}+\varepsilon_{amp}\left(\tilde{d}^2_{X_n,l_{\pi(n)}}+H^2\right)\right)\text{[nJ]}, \quad (2)$$

where $\varepsilon_{el}=50$ [nJ/bit] to run a transmitter or a receiver circuit, and $\varepsilon_{amp}=0.1$ [nJ/bit/m^2] for a transmit amplifier [1].

Then, the residual energy of a sensor n after transmitting data to the UAV at $l_{\pi(n)}$ is expressed as

$$E_n^{res} = E_n - f\left(b_n, \sqrt{\tilde{d}^2_{X_n,l_{\pi(n)}}+H^2}\right), \quad (3)$$

where E_n is the energy of a sensor n before transmitting data to the UAV.

Note that for a given E_n, E_n^{res} can be maximized when $\tilde{d}_{X_n,l_{\pi(n)}}=0$ in (2), which means that $l_{\pi(n)}$ is located above X_n. In addition, due to its limited power, there exists a maximum communication distance such that each sensor can guarantee the required SNR at the UAV. Thus, we have $0\le\tilde{d}_{X_n,l_{\pi(n)}}\le d_n^{\max}$, where $d_n^{\max}$ denotes the maximum horizontal communication range (MHCR) of a sensor n and depends on a UAV altitude and an acceptable SNR.

A. Optimization Problem

In this paper, we investigate UAV routes to reduce energy consumption of sensors in data transmission. Specifically, we aim to maximize the minimum E_n^{res} of sensors $n\in\mathbf{I}\triangleq\mathbf{N}\setminus\{n|\tilde{d}_{X_n,l_{\pi(n)}}=0, n\in\mathbf{N}\}$ by optimizing $P_{\boldsymbol{L}_\pi}$, subject to data collection and UAV traveling distance constraints. Note that E_n^{res} can be increased for all $n\in\mathbf{I}$ (equivalently sensor n can reduce energy consumption for data transmission) as $\tilde{d}_{X_n,l_{\pi(n)}}$ decreases. On the other hand, for all $n\in\mathbf{N}\setminus\mathbf{I}$, the maximum E_n^{res} is achieved since $l_{\pi(n)}$ is placed above sensor location X_n. Therefore, $\mathbf{I}$ affects both $l_{\pi(n)}$ and E_n^{res} for all $n\in\mathbf{N}$.

So, an optimization problem can be formulated as

$$\max_{P_{\boldsymbol{L}_\pi}}\min_{n\in\mathbf{I}} \quad E_n^{res}=E_n-f\left(b_n,\sqrt{\tilde{d}^2_{X_n,l_{\pi(n)}}+H^2}\right) \quad (4)$$

$$\text{s.t.} \quad \mathbf{I}=\mathbf{N}\setminus\left\{n|\tilde{d}_{X_n,l_{\pi(n)}}=0, n\in\mathbf{N}\right\}, \quad (4a)$$

$$0\le\tilde{d}_{X_n,l_{\pi(n)}}\le d_n^{\max}, n\in\mathbf{N}, \quad (4b)$$

$$D_{\boldsymbol{L}_\pi}\le D_{tr}^{\max}, \quad (4c)$$

$$\pi(n)\in\{1,\ldots,K\}, n\in\mathbf{N}. \quad (4d)$$

(4b) is the constraint on $l_{\pi(n)}$ required to guarantee data collection at sensor n, where $l_{\pi(n)}$ is limited by the MHCR $d_n^{\max}$. (4c) is the constraint on UAV traveling distance, where $D_{tr}^{\max}$ is the maximum allowable UAV traveling distance limited by its total amount of energy.

When $D_{tr}^{\max}$ is sufficiently large such that $\tilde{d}_{X_n,l_{\pi(n)}}=0$ for all $n\in\mathbf{N}$ (i.e., the UAV can visit all sensor locations), $\mathbf{I}=\emptyset$ and the formulated problem becomes equivalent to finding a feasible UAV route that satisfies (4c), which is a well-known traveling salesman problem (TSP) [16]; given N sensor locations (i.e., cities to be visited) and the UAV (i.e., salesman), find a shortest closed route that a salesman visits all cities only once.

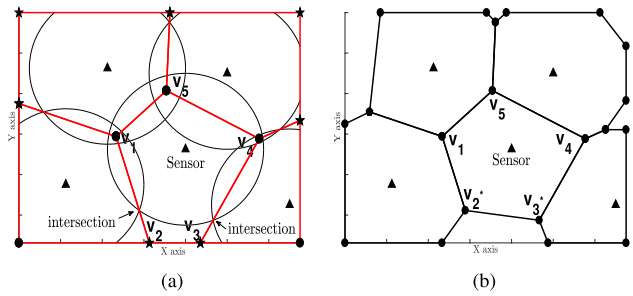


Fig. 1. Construction of $\mathbf{mVD}(\{d_n^{\max}\})$. (a) Before integration. (b) After integration.

For the case when a given $D_{tr}^{\max}$ cannot allow a TSP solution, sensor energy status and UAV traveling distance should be considered simultaneously to optimize $P_{\boldsymbol{L}_\pi}$, which is difficult to solve due to the non-convexity of the problem [29]. In addition, the formulated optimization problem is a variant of TSP, where finding the optimal solution is NP-complete [16]. In the following two sections, we propose polynomial algorithms for feasible and sub-optimal solutions.

III. Algorithm for Feasible Solutions

The set of feasible solutions can be obtained by finding the shortest UAV route that satisfies data collection constraints (4b). The proposed algorithm utilizes Voronoi diagram [30], [31] to reduce search space for UAV hovering locations, but MHCR of each sensor needs to be considered since the UAV hovering outside the MHCR cannot ensure reliable data gathering at sensors. In the following subsections, we construct a modified Voronoi diagram by taking MHCR into account, and propose an algorithm to find the shortest UAV route.

A. Modified Voronoi Diagram

In Voronoi diagram, $P_{\boldsymbol{L}_\pi}$ can be determined by placing each $l_{\pi(n)}$ at Voronoi edge or vertex. Given N sensor locations $\{X_n, n\in\mathbf{N}\}$, the Voronoi diagram is defined as

$$\{x|d_{x,X_n}=d_{x,X_m},\ m\in\mathbf{A}(n), n\in\mathbf{A}(m), n,m\in\mathbf{N}, x\in\mathbf{S}\}, \quad (5)$$

where d_{x,X_n} is a distance between a point x and sensor n, and $\mathbf{A}(n)$ denotes a set of neighboring nodes to a sensor n.

However, note that UAV route determined based on the Voronoi diagram does not always guarantee data collection at all sensors, since $l_{\pi(n)}$ can be located at a Voronoi edge or vertex that is outside the communication range of a sensor n. As a resolution, we construct a modified Voronoi diagram $\mathbf{mVD}(\{d_n^{\max}\})$ by integrating MHCR $\{d_n^{\max}\}$ (e.g., circle in Fig. 1a) into Voronoi diagram, aiming to remove the portion of Voronoi edges that violates (4b). Specifically, Voronoi vertex that is not within any MHCR (see ★ in Fig. 1a) is substituted with the intersection of Voronoi edge and MHCR. For example, among Voronoi vertices v_j ($j=1,\ldots,5$) in Fig. 1a, v_2 and v_3 are not within any MHCR. Hence, they are replaced by v_2^* and

v_3^* in Fig. 1b, respectively. After replacing all Voronoi vertices that violate (4b), $\mathbf{mVD}(\{d_n^{\max}\})$ can be obtained as in Fig. 1b. Note that when there is no intersection of Voronoi edges and MHCR, the corresponding Voronoi edge (e.g., line connecting v_2 and v_3 in Fig. 1a) is eliminated from $\mathbf{mVD}(\{d_n^{\max}\})$.

B. Analysis of $\mathbf{mVD}(\{d_n^{\max}\})$

For a given $\mathbf{mVD}(\{d_n^{\max}\})$, let $\boldsymbol{L}^{(h)} \triangleq \{l_1^{(h)}, l_2^{(h)}, \ldots\}$ be the set of points that adjoin h Voronoi regions, where $h(\geq 2)$ is an integer. In addition, denote by $\boldsymbol{L}^{(1)} \triangleq \{l_1^{(1)}, l_2^{(1)}, \ldots\}$ the set of points involved in one Voronoi region. Then, we can draw the following observations.

- For $h \geq 3$, $\boldsymbol{L}^{(h)}$ consists of Voronoi vertices surrounded by h Voronoi regions. Each Voronoi vertex has more than three adjacent Voronoi regions, but we can rarely obtain $\boldsymbol{L}^{(h)} \neq \emptyset$ for $h \geq 4$ since more than four adjacent Voronoi centers may not have a common circumscribed circle[1] [30]. In most cases, Voronoi vertices are included in $\boldsymbol{L}^{(3)}$.
- $\boldsymbol{L}^{(2)}$ consists of points on the Voronoi edge. Since every two adjacent Voronoi regions have a common edge, we can always obtain $\boldsymbol{L}^{(2)} \neq \emptyset$.
- $\boldsymbol{L}^{(1)}$ consists of points inside the Voronoi regions. Especially, all Voronoi centers are included in $\boldsymbol{L}^{(1)}$ (i.e., $\{X_n | \forall n \in \mathbf{N}\} \subset \boldsymbol{L}^{(1)}$).

Note that for a given $\mathbf{mVD}(\{d_n^{\max}\})$, the maximum value of h (i.e., $h \leq h_{\max}$) can be determined by searching for all Voronoi vertices (e.g., $h_{\max} = 3$ in Figs. 1–3).

C. Shortest Route Determination

To shorten UAV traveling distance, the number of UAV hovering locations should be minimized. In other words, each UAV hovering location is required to include as many adjacent sensors as possible, whose data can be collected. Based on the fact that more than two adjacent Voronoi centers (i.e., sensors) exist for each Voronoi edge or vertex, $\boldsymbol{L}^{(h)}$ for $h = 1, \ldots, h_{\max}$ can be utilized to find the shortest route. However, it is a non-trivial combinatorial problem. To reduce computational complexity, we propose an algorithm for each UAV hovering location using the reference path.

1) Reference Path: A reference path is defined as the shortest route connecting all Voronoi centers, which is equivalent to a solution to TSP with Voronoi centers and the UAV [16]. It is worth noting that for static sensors, the reference path can be found by solving the TSP only once.

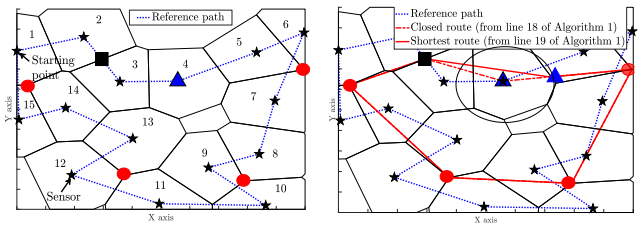
2) Algorithm Procedure: The proposed algorithm first assigns the numbers to each Voronoi region in an ascending order along the reference path. Then, to minimize the number of UAV hovering locations, the proposed algorithm determines UAV hovering locations using $\boldsymbol{L}^{(h_{\max})}, \boldsymbol{L}^{(h_{\max}-1)}, \ldots, \boldsymbol{L}^{(1)}$, sequentially. Specifically, for $h \geq 4$, all $l^{(h)} \in \boldsymbol{L}^{(h)}$ are selected as the UAV hovering locations due to observations in Sec. III-B. On the other hand, since there are a number of choices for the case of $h \leq 3$, some of $l^{(h)} \in \boldsymbol{L}^{(h)}$ are selected as the UAV hovering location by the numbers assigned to each Voronoi region. Once UAV hovering locations are determined, the visiting order of all the selected hovering locations is obtained along the order of reference path. Details of the shortest UAV route determination is summarized in Algorithm 1.

[1]The center of the circumscribed circle is equivalent to Voronoi vertex.

Algorithm 1: The Shortest Route Determination on $\mathbf{mVD}(\{d_n^{\max}\})$.

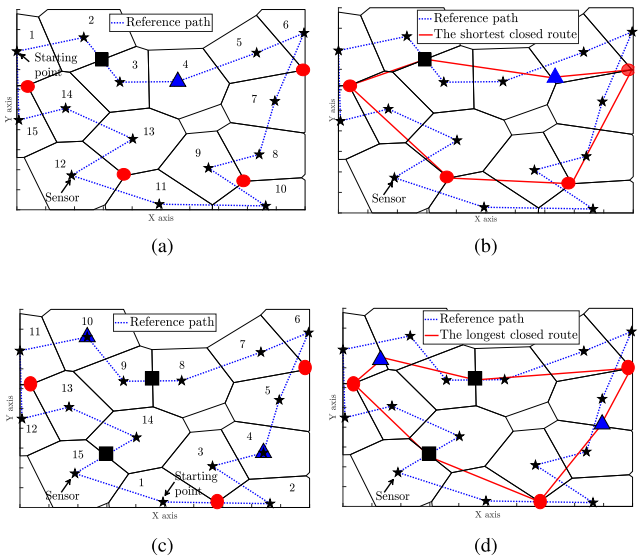
Input: Reference path, a starting point, route direction, $\boldsymbol{L}^{(h_{\max})}, \boldsymbol{L}^{(h_{\max}-1)}, \ldots, \boldsymbol{L}^{(1)}$.
Output: The set of UAV hovering locations $\boldsymbol{L}$.
Initialize: $\boldsymbol{\Lambda} \triangleq \{1, \ldots, N\}$, $\boldsymbol{L} = \emptyset$.
Step 1: Assign numbers to $\mathbf{mVD}(\{d_n^{\max}\})$
1: Assign numbers to Voronoi regions in an ascending order along the reference path and the route direction, from the starting point.
Step 2: Determine UAV hovering location, sequentially
2: $h \leftarrow h_{\max}$
3: **if** $h \geq 4$ **then**
4: **if** $\boldsymbol{L}^{(h)} \neq \emptyset$ **then**
5: $\boldsymbol{L} \leftarrow \boldsymbol{L} \cup \{l^{(h)} | \forall l^{(h)} \in \boldsymbol{L}^{(h)}\}$. For all $l^{(h)} \in \boldsymbol{L}^{(h)}$, numbers assigned to Voronoi regions adjacent to $l^{(h)}$ are removed from the set $\boldsymbol{\Lambda}$.
6: **end if**
7: $h \leftarrow h - 1$ and go to line 3.
8: **end if**
9: Let $l^{(3)} \in \boldsymbol{L}^{(3)}$ be the Voronoi vertex surrounded by *three* Voronoi regions with consecutive numbers.
10: **if** $l^{(3)}$ exists for numbers $\{i, i+1, i+2\} \in \boldsymbol{\Lambda}$ **then**
11: $\boldsymbol{L} \leftarrow \boldsymbol{L} \cup \{l^{(3)}\}$, $\boldsymbol{\Lambda} \leftarrow \boldsymbol{\Lambda} \setminus \{i, i+1, i+2\}$, Go to line 9.
12: **end if**
13: For *two* adjacent Voronoi regions with consecutive numbers, let $l^{(2)} \in \boldsymbol{L}^{(2)}$ be the intersection of Voronoi edge and the reference path.
14: **if** $l^{(2)}$ exists for numbers $\{i, i+1\} \in \boldsymbol{\Lambda}$ **then**
15: $\boldsymbol{L} \leftarrow \boldsymbol{L} \cup \{l^{(2)}\}$, $\boldsymbol{\Lambda} \leftarrow \boldsymbol{\Lambda} \setminus \{i, i+1\}$, Go to line 13.
16: **end if**
17: Let $l^{(1)} \in \boldsymbol{L}^{(1)}$ be the Voronoi center with number $i \in \boldsymbol{\Lambda}$.
$\boldsymbol{L} \leftarrow \boldsymbol{L} \cup \{l^{(1)} | \forall i \in \boldsymbol{\Lambda}\}$
Step 3: Find the shortest UAV route
18: All $l \in \boldsymbol{L}$ are connected along the order of the reference path to make a closed route.
19: For each $l^{(1)} \in \boldsymbol{L}$, replace $l^{(1)}$ with the intersection of the closed route and the MHCR of the corresponding sensor.

3) Example: Fig. 2 illustrates the procedure of Algorithm 1, where $h_{\max} = 3$ for the given $\mathbf{mVD}(\{d_n^{\max}\})$. In Step 1, each Voronoi region is numbered from the starting point along the reference path and the route direction (e.g., clockwise). Step 2 determines UAV hovering locations (i.e., '●,' '■' and '▲' in Fig. 2a). To be specific, lines 9-12 of Algorithm 1 select Voronoi vertices surrounded by Voronoi regions with numbers {5, 6, 7},



(a) (b)

Fig. 2. The shortest UAV route determination in $\mathbf{mVD}(\{d_n^{\max}\})$. (a) Number allocation. (b) The proposed route.



(a) (b)

(c) (d)

Fig. 3. Comparisons of the shortest and the longest closed-route. (a) Number allocation (route direction: clockwise). (b) The shortest closed route (route length: 514.35 (m)). (c) Number allocation (route direction: counterclockwise). (d) The longest closed route (route length: 536.59 (m)).[2]

$\{8, 9, 10\}, \{11, 12, 13\}$ and $\{14, 15, 1\}$. For remaining numbers (i.e., $\mathbf{\Lambda} = \{2, 3, 4\}$), lines 13-16 of Algorithm 1 determine the UAV hovering location using the Voronoi regions with numbers $\{2, 3\}$ as the intersection of Voronoi edge and the reference path. Lastly, line 17 of Algorithm 1 determines the UAV hovering location as the center of Voronoi region with number $\{4\}$. In Step 3, to make a closed route, all selected UAV hovering locations are connected according to the order of reference path. Note that line 19 of Algorithm 1 reduces the length of a closed route (See Fig. 2b). Specifically, the UAV hovering location in the Voronoi region with number $\{4\}$ is updated as the intersection of the closed route and MHCR (i.e., the dashed circle in Fig. 2b).

Remark 1: Once UAV hovering locations and its visiting order are determined, a round-trip UAV route, which starts from and ends at l_0, is determined in such a way that the UAV first visits its hovering location nearest to a control center.

Remark 2: From Algorithm 1, different starting points and route directions may result in different UAV hovering locations, consequently different UAV routes. To evaluate their effects on UAV route, we conduct simulations considering various starting points and route directions. Fig. 3 represents closed routes for an instance of sensor distribution, where the length of a closed route varies according to the starting location and route direction. It is shown from Fig. 3 that the difference between the longest and the shortest closed routes is 22.24 [m] (i.e., the ratio of length difference to the shortest closed route length is 4.32%). However, a major factor affecting $D_{\boldsymbol{L}_\pi}(=\sum_{i=0}^{K} d_{l_i,l_{i+1}})$ is the distance between a control center at l_0 and the surveillance region (see Table II). So, the effect of the starting point and route direction to Algorithm 1 can be negligible.

[2]In Fig. 3, we assume that 15 sensors are deployed in a square area centered at the origin X_0, the length of its side is 200 [m] and $d_n^{\max} = 35$ [m], $\forall n \in \mathbf{N}$.

TABLE II
THE DIFFERENCE OF UAV TRAVELING DISTANCES ACCORDING TO THE LOCATION OF CONTROL CENTER[3]

d_{l_0,X_0} [m]	200	300	400	500	600
$\frac{P_{tr}^L - P_{tr}^S}{P_{tr}^S}$ [%]	3.66	3.04	2.57	2.22	1.94

IV. ALGORITHM FOR SUB-OPTIMAL SOLUTION

In this section, the feasible UAV routes from Algorithm 1 are adjusted to find a sub-optimal solution. We first determine how to adjust each UAV hovering location reflecting sensor energy status, then apply a sequential algorithm to find the proposed UAV route.

A. Adjustment Scheme for UAV Hovering Location

Recall that Algorithm 1 determines each UAV hovering location using Voronoi vertex (i.e., $\boldsymbol{L}^{(h)}$ for $h \geq 3$), Voronoi edge (i.e., $\boldsymbol{L}^{(2)}$) and Voronoi center (i.e., $\boldsymbol{L}^{(1)}$). We provide adjustment schemes for each type of UAV hovering location (see Fig. 4). For the ease of description, let $n^* \triangleq \arg\min_{n\in\mathbf{I}} E_n^{res}$ and we increase $E_{n^*}^{res}$ to a target value E_{tgt} by adjusting $\tilde{d}_{X_{n^*},l_{\pi(n^*)}}$. Then, the exact location of the updated $l_{\pi(n^*)}$ is determined.

1) When $l_{\pi(n^*)} \in \boldsymbol{L}^{(1)}$: In this case, $l_{\pi(n^*)}$ is only involved in a sensor n^*. By solving $E_{n^*}^{res} = E_{tgt}$ using (3), $\tilde{d}_{X_{n^*},l_{\pi(n^*)}}$ can be updated as

$$\tilde{d}_{X_{n^*},l_{\pi(n^*)}} = \begin{cases} 0, & \text{if } E_{n^*} - b_{n^*}\varepsilon_{el} - b_{n^*}\varepsilon_{amp}H^2 < E_{tgt}, \\ \min\left(\sqrt{\frac{E_{n^*} - b_{n^*}\varepsilon_{el} - E_{tgt}}{b_{n^*}\varepsilon_{amp}} - H^2},\ d_{n^*}^{\max}\right), & \text{otherwise.} \end{cases} \tag{6}$$

Note that the maximum value of $E_{n^*}^{res}$ is obtained when $l_{\pi(n^*)}$ is placed above X_{n^*} (i.e., $\tilde{d}_{X_{n^*},l_{\pi(n^*)}} = 0$). So,

$$\begin{aligned} E_{n^*}^{res} &\leq E_{n^*} - f\left(b_{n^*}, \sqrt{0 + H^2}\right), \\ &= E_{n^*} - b_{n^*}\varepsilon_{el} - b_{n^*}\varepsilon_{amp}H^2. \end{aligned}$$

Therefore, when $E_{n^*} - b_{n^*}\varepsilon_{el} - b_{n^*}\varepsilon_{amp}H^2 < E_{tgt}$, we cannot increase $E_{n^*}^{res}$ to E_{tgt} by adjusting $\tilde{d}_{X_{n^*},l_{\pi(n^*)}}$. In this

[3]In Table III, we assume that the control center is located at $(-d_{l_0,X_0},0)$ in the XY plane. d_{l_0,X_0} [m] denotes distance between the control center and the origin. P_{tr}^L and P_{tr}^S denote round-trip UAV traveling distances corresponding to the longest and the shortest closed route, respectively.

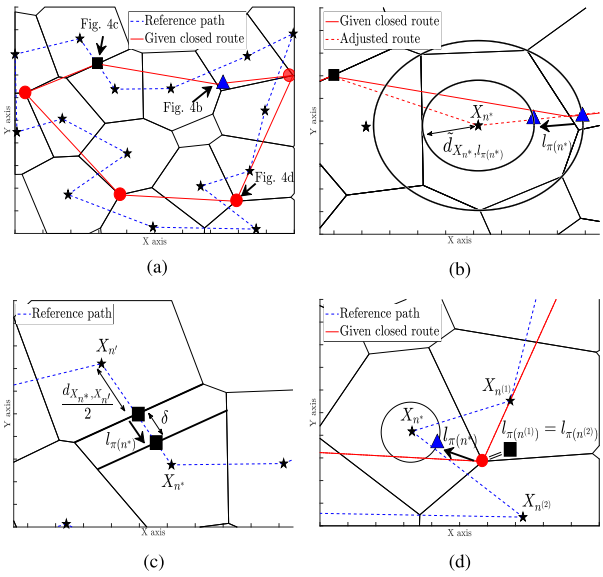


Fig. 4. Adjustment scheme for UAV hovering location. (a) A feasible UAV route from Algorithm 1. (b) The case of $l_{\pi(n^*)} \in \boldsymbol{L}^{(2)}$. (c) The case of $l_{\pi(n^*)} \in \boldsymbol{L}^{(2)}$. (d) The case of $l_{\pi(n^*)} \in \boldsymbol{L}^{(3)}$.

case, we set $\tilde{d}_{X_{n^*},l_{\pi(n^*)}} = 0$ and $\mathbf{I}$ in (4a) is updated as $\mathbf{I} = \mathbf{I} \setminus \{n^*\}$.

Once $\tilde{d}_{X_{n^*},l_{\pi(n^*)}} \neq 0$ is obtained from (6), the exact location of $l_{\pi(n^*)}$ can be determined by adjusting the given closed route and finding the intersection of the adjusted route and the circle with center X_{n^*} and radius $\tilde{d}_{X_{n^*},l_{\pi(n^*)}}$ (see Fig. 4b). Note that since (6) reduces $\tilde{d}_{X_{n^*},l_{\pi(n^*)}}$, there is a case when no intersecting point exists to determine $l_{\pi(n^*)}$. To this end, the given closed route is adjusted to visit the location of sensor n^* before finding an intersecting point.

2) When $l_{\pi(n^*)} \in \boldsymbol{L}^{(2)}$: In this case, two adjacent sensors are served by a UAV hovering location at Voronoi edge. Recall that Algorithm 1 determines each UAV hovering location in $\boldsymbol{L}^{(2)}$ as the intersection of Voronoi edge and line connecting two adjacent sensors. Therefore, before increasing $E^{res}_{n^*}$ to E_{tgt}, we can adjust Voronoi edges close to sensor n^* based on energy status of two adjacent sensors (See Fig. 4c).

Let sensors n^* and n' share the same UAV hovering location $l_{\pi(n^*)} = l_{\pi(n')}$. By introducing δ (≥ 0) and solving the equation $E_{n^*} - f(b_{n^*}, \sqrt{(\tilde{d}_{X_{n^*},l_{\pi(n^*)}} - \delta)^2 + H^2}) = E_{n'} - f(b_{n'}, \sqrt{(\tilde{d}_{X_{n'},l_{\pi(n^*)}} + \delta)^2 + H^2})$ for δ, we can obtain the following updated horizontal communication distances, i.e.,

$$\tilde{d}_{X_{n^*},l_{\pi(n^*)}} = \frac{d_{X_{n^*},X_{n'}}}{2} - \delta, \tag{7}$$

$$\tilde{d}_{X_{n'},l_{\pi(n^*)}} = \frac{d_{X_{n^*},X_{n'}}}{2} + \delta, \tag{8}$$

where $\delta = \max(\min(\delta, \frac{d_{X_{n^*},X_{n'}}}{2}), 0)$. It is worth noting that due to Voronoi diagram, $\tilde{d}_{X_{n^*},l_{\pi(n^*)}} + \tilde{d}_{X_{n'},l_{\pi(n^*)}} = d_{X_{n^*},X_{n'}}$ holds for any δ within $[0, \frac{d_{X_{n^*},X_{n'}}}{2}]$. In addition, based on (7) and (8), the newly updated $l_{\pi(n^*)}$ is uniquely determined.

If the newly updated $l_{\pi(n^*)}$ makes $E^{res}_{n^*} \geq E_{tgt}$ and $E^{res}_{n'} \geq E_{tgt}$, no more update for the horizontal communication distance is required. Otherwise, the adjustment of Voronoi edge cannot be a solution to achieve E_{tgt}. In this case, we can divide $\{n^*, n'\}$ into $\{n^*\}$ and $\{n'\}$, then apply the procedure in Sec. IV-A1 to sensor n^*.

3) When $l_{\pi(n^*)} \in \boldsymbol{L}^{(h)}$ for $h \geq 3$: In this case, more than three adjacent sensors are served by a UAV hovering location at the Voronoi vertex. Let h sensors (e.g., $n^*, n^{(1)}, n^{(2)}, \ldots, n^{(h-1)}$) share the same UAV hovering location $l_{\pi(n^*)} = l_{\pi(n^{(1)})} = \cdots = l_{\pi(n^{(h-1)})}$. In order to increase $E^{res}_{n^*}$ to E_{tgt}, the given UAV hovering location needs to be shifted to a new UAV hovering location x that satisfies

$$E_{n^*} - f\left(b_{n^*}, \sqrt{\tilde{d}^2_{X_{n^*},x} + H^2}\right) = E_{tgt}, \tag{9a}$$

$$\tilde{d}_{X_{n^{(1)}},x} = \tilde{d}_{X_{n^{(2)}},x} = \cdots = \tilde{d}_{X_{n^{(h-1)}},x}. \tag{9b}$$

(9a) makes $E^{res}_{n^*} = E_{tgt}$ for sensor n^*, and (9b) makes x equidistant to the sensors $n^{(1)}, n^{(2)}, \cdots, n^{(h-1)}$.

However, x that satisfies (9a) and (9b) does not always exist for $h \geq 4$, and it is nontrivial to find even if x exists. So, we only head to find a new UAV hovering location for sensor n^* by the procedure in Sec. IV-A1. On the other hand, the UAV hovering locations for remaining sensors (i.e., $n^{(1)}, n^{(2)}, \cdots, n^{(h-1)}$) are kept unchanged (see Fig. 4d for the case of $h = 3$).

B. Sequential Algorithm

Based on the adjustment schemes for UAV hovering location in Sec. IV-A, the feasible UAV route can be adjusted to find the proposed UAV route. However, route adjustment can increase UAV traveling distance. Hence, the proposed algorithm updates the initial feasible UAV route sequentially until the newly updated UAV route does not violate the UAV traveling distance constraint (4c).

Given the feasible UAV route, let $n^* = \arg\min_{n \in \mathbf{I}} E^{res}_n$ and $n^\dagger = \arg\min_{n \in \mathbf{I} \setminus \{n*\}} E^{res}_n$, respectively. To maximize the objective value in (4), the proposed sequential algorithm focuses on the sensor n^*, and increases $E^{res}_{n^*}$ to $E_{tgt} \triangleq E^{res}_{n^\dagger}$ by adjusting $l_{\pi(n^*)}$. If the resultant UAV route is feasible, the above procedures are repeated. Details of the proposed sequential algorithm are summarized in Algorithm 2.

Remark 3: It is noted that when $E^{res}_{n^*} = E_{tgt}$ and $\tilde{d}_{X_{n^*},l_{\pi(n^*)}} \neq 0$ are obtained as a result of updating $l_{\pi(n^*)}$, Algorithm 2 should adjust two hovering locations $l_{\pi(n^*)}$ and $l_{\pi(n^\dagger)}$ simultaneously in the next iteration to increase the objective value. Also, note that once $\mathbf{I}$ becomes empty, Algorithm 2 provides UAV route equivalent to the TSP solution (i.e., reference path in Sec. III-C1).

Remark 4: Algorithm 2 finds sensor n^* for each iteration, but the number of such iterations can be reduced by adjusting all the $l^{(2)} \in \boldsymbol{L}^{(2)}$ first. Specifically, when the shortest route from Algorithm 1 is given, the route optimization can initially adjust each $l^{(2)} \in \boldsymbol{L}^{(2)}$, instead of focusing on sensor n^*. Note that this process is equivalent to updating $\mathbf{mVD}(\{d^{\max}_n\})$ partially, i.e., adjusting Voronoi edges based on sensor energy status (see Fig. 4c).

Remark 5: When sensor energy information is not available, Algorithm 2 can be implemented by letting $E_n = 0$

Algorithm 2: Sequential Algorithm to Find the Proposed UAV Route.

Input: The shortest UAV route from Algorithm 1.
Output: The proposed UAV route P^{opt}.
Initialize: $P^{opt} \leftarrow$ the shortest route from Algorithm 1.
Step 1: Given P^{opt}, determine n^* and E_{tgt}
$n^* \leftarrow \arg\min_{n \in \mathbf{I}} E_n^{res}$.
$E_{tgt} \leftarrow E_{n^\dagger}^{res}$, where $n^\dagger \leftarrow \arg\min_{n \in \mathbf{I} \setminus \{n^*\}} E_n^{res}$.
Step 2: Update the UAV hovering location of sensor n^*
if $l_{\pi(n^*)} \in \boldsymbol{L}^{(1)}$ **then**
  Update $l_{\pi(n^*)}$ by the procedure in Sec. IV-A1.
end if
if $l_{\pi(n^*)} \in \boldsymbol{L}^{(2)}$ **then**
  Update $l_{\pi(n^*)}$ by the procedure in Sec. IV-A2.
end if
if $l_{\pi(n^*)} \in \boldsymbol{L}^{(h)}$ for $h \geq 3$ **then**
  Update $l_{\pi(n^*)}$ by the procedure in Sec. IV-A3.
end if
Step 3: Check the feasibility of the updated UAV route
if The updated UAV route satisfy (4c) **then**
  $P^{opt} \leftarrow$ updated UAV route.
else
  Terminate algorithm.
end if
if $\mathbf{I}$ is not empty **then**
  Go to step 1.
end if

Algorithm 3: Sensor Energy-Based Initial UAV Route Determination on $\mathbf{mVD}(\{d_n^{\max}\})$.

Input: Reference path, a starting point, route direction, λ, $\boldsymbol{L}^{(h_{\max})}, \boldsymbol{L}^{(h_{\max}-1)}, \ldots, \boldsymbol{L}^{(1)}$.
Output: The set of UAV hovering locations $\boldsymbol{L}$.
Initialize: $\boldsymbol{\Lambda} \triangleq \{1, \ldots, N\}$, $\boldsymbol{L} = \emptyset$.
1: Conduct Step 1 of Algorithm 1.
2: $h \leftarrow h_{\max}$
3: **if** $h \geq 4$ **then**
4: **if** $\boldsymbol{L}^{(h)} \neq \emptyset$ **then**
5: $\boldsymbol{L} \leftarrow \boldsymbol{L} \cup \{l^{(h)} | E_n \geq \lambda$ for all sensors n adjacent to $l^{(h)} \in \boldsymbol{L}^{(h)}\}$.
6: **end if**
7: $h \leftarrow h - 1$ and go to line 3.
8: **end if**
9: For all $l^{(h)} \in \boldsymbol{L}$, numbers assigned to Voronoi regions adjacent to $l^{(h)}$ are removed from the set $\boldsymbol{\Lambda}$.
10: Let $l^{(3)} \in \boldsymbol{L}^{(3)}$ be the Voronoi vertex surrounded by *three* Voronoi regions with consecutive numbers.
11: **if** $l^{(3)}$ exists for numbers $\{i, i+1, i+2\} \in \boldsymbol{\Lambda}$ and $E_n \geq \lambda$ for all sensors n adjacent to $l^{(3)}$ **then**
12: $\boldsymbol{L} \leftarrow \boldsymbol{L} \cup \{l^{(3)}\}$, $\boldsymbol{\Lambda} \leftarrow \boldsymbol{\Lambda} \setminus \{i, i+1, i+2\}$. Go to line 9.
13: **end if**
14: Conduct Lines 13 to 19 of Algorithm 1.

for all $n \in \mathbf{N}$. In this case, the proposed algorithm utilizes $f(b_n, \sqrt{\tilde{d}^2_{X_n, l_{\pi(n)}} + H^2})$, rather than E_n^{res} of sensors $n \in \mathbf{I}$.

C. Sensor Energy-Based Initial UAV Route

In this section, a sensor energy-based initial UAV route determination scheme is proposed. Recall that Algorithm 1 does not utilize the sensor energy information in route determination. Therefore, when a number of low-energy sensors are served by the UAV hovering location at Voronoi vertex (i.e., $\boldsymbol{L}^{(h)}$ for $h \geq 3$), the corresponding initial route will be adjusted several times in route optimization. To find the proposed solution quickly, we can modify Algorithm 1 using the energy threshold value λ (≥ 0). Specifically, the proposed algorithm generates the UAV hovering location at Voronoi vertex to be kept unchanged as much as possible in route optimization. Details of procedures are summarized in Algorithm 3.

Remark 6: The value of λ affects the feasibility of a resultant UAV route. As λ increases, the number of UAV hovering locations serving more than three sensors decreases and the corresponding UAV route may not satisfy the UAV traveling distance constraint. Therefore, when the shortest UAV route is required, we can set $\lambda = 0$ to obtain the same result as Algorithm 1. Otherwise, the typical value of λ can be determined as $\lambda = \sum_{n=1}^{N} E_n / N$.

Remark 7: The difference between Algorithms 1 and 3 is the availability of sensor energy information. Without this information, Algorithm 3 cannot be conducted. However, once UAV makes a return trip and gets a sensor energy information, the next data collection route can be determined quickly by applying Algorithm 3 and Algorithm 2 sequentially.

V. Numerical Results

In this section, numerical results are provided to verify our proposed UAV route determination algorithm. In simulations, we assume that 15 sensors are randomly deployed over a square area $\mathbf{S}$ of 300 × 300 [m^2]. The UAV altitude H is 50 [m]. Each sensor energy E_n is randomly determined within [0.05, 0.1] [J]. $d_n^{\max} = 50$ [m] and $b_n = 300$ [bytes] for all $n \in \mathbf{N}$. We conduct 10,000 simulations for different sensor deployments and energy, and the average objective value is utilized as a performance metric. Note that the objective value is set to zero when the proposed algorithm cannot find the feasible UAV route under the UAV traveling distance constraint. Also, under the considered setup, the minimum value of E_n^{res} is 0.0487 [J] since the minimum value of E_n is 0.05 [J] and the maximum value of $f(b_n, \sqrt{\tilde{d}^2_{X_n, l_{\pi(n)}} + H^2})$ is 0.0013 [J].

Fig. 5 compares UAV traveling distance of the proposed algorithm with those of other algorithms in terms of the MHCR of each sensor. We consider four schemes as baseline for comparisons: TSP solution [16], the CSS algorithm [19], the Voronoi edge based method (VM) [20] and the exhaustive search. The TSP solution is equivalent to UAV route that connects all sensor locations with minimum length. The CSS algorithm determines UAV route using communication range of sensors, which is superior to other heuristic algorithms with respect to route length. VM determines UAV route by connecting Voronoi edges and the exhaustive search finds the shortest UAV route by searching for

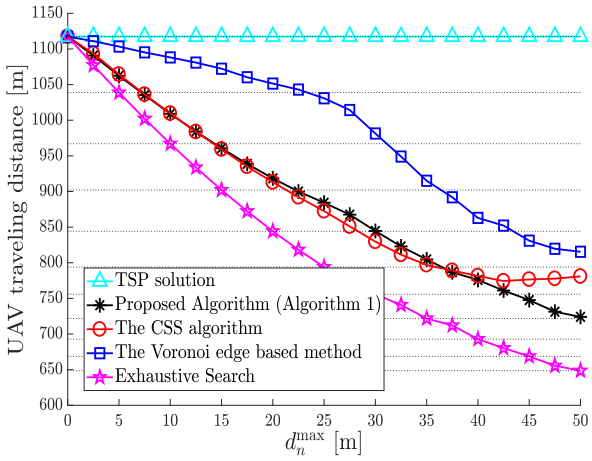


Fig. 5. Comparisons of UAV traveling distances when $d_n^{\max}$ is the same for all $n \in \mathbf{N}$.

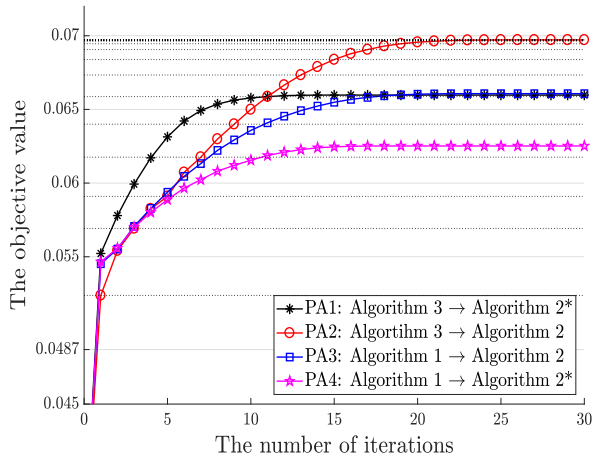


Fig. 6. Convergence performance of proposed algorithms.

TABLE III
TIME COMPLEXITY OF ALGORITHMS

Algorithm	Time complexity
TSP solution	$\mathcal{O}(N^2 2^N)$
Proposed Algorithm (Algorithm 1)	$\mathcal{O}(N^2 2^N)+\mathcal{O}(N \log N)+\mathcal{O}(N)$
The CSS algorithm	$\mathcal{O}(N^2 2^N)+\mathcal{O}(N^3 \log N)+\mathcal{O}(N^2 \log \frac{1}{\delta})$
The Voronoi edge based method	$\mathcal{O}(N^2 2^N)+\mathcal{O}(N \log N)+\mathcal{O}(M \log M)$
Exhaustive search	$\mathcal{O}(N^2(2\pi(d_n^{\max})^2)^N)$

TABLE IV
PERFORMANCE COMPARISONS AMONG THE PROPOSED ALGORITHM AND EXHAUSTIVE SEARCH

Problem (4) when **I=N**	UAV traveling distance [m]	Objective value [mJ]	Δ [%]
PA1	791.07	52.2	28.6
PA2	776.15	51.6	29.3
PA3	744.47	49.7	32.1
PA4	788.22	52.3	28.5
Problem (4)	UAV traveling distance [m]	Objective value [mJ]	Δ [%]
PA1	875.97	66.0	9.8
PA2	878.24	69.7	4.6
PA3	872.24	66.1	9.6
PA4	871.92	62.5	14.5
Exhaustive search	863.77	73.1	-

Δ denotes the ratio of difference in objective value to the maximum objective value obtained from the exhaustive search.

all possible UAV hovering locations. It is observed from Fig. 5 that when $d_n^{\max} = 0$ for all sensors, $\mathbf{I} = \emptyset$ and all algorithms find the shortest UAV route (i.e., the TSP solution). But this is not the case for $d_n^{\max} > 0$. In addition, it is observed from Fig. 5 that both our proposed algorithm and VM utilize the Voronoi diagram in route determination, but our proposed algorithm outperforms VM. Lastly, it is shown from Fig. 5 that our proposed algorithm provides UAV traveling distance similar to the CSS algorithm. It is worth noting that although the exhaustive search method provides the shortest UAV route, it cannot be implemented due to extremely high computational complexity (see Table III, and Appendix A for analysis).

Fig. 6 shows the convergence performance of the proposed algorithms. Depending on initial UAV route determination (i.e., Algorithms 1 and 3) and route optimization process (i.e., Algorithms 2 and 2*), four algorithms can be obtained, where Algorithm 2* represents a variant of Algorithm 2 that adjusts Voronoi edges first (see Remark 4). It is observed from Fig. 6 that UAV route from PA1, which reflects sensor energy status in initial route determination and route optimization, converges faster than other algorithms. However, it cannot maximize the objective value since the adjustment of Voronoi edge at the beginning of route optimization does not consider sensor $n^*(= \arg\min_{n \in \mathbf{I}} E_n^{res})$ preferentially. Also, it is observed that PA2 provides the maximum objective value since Algorithm 3 provides an initial UAV route such that UAV hovering location at Voronoi vertex is kept unchanged as much as possible in route optimization. Therefore, many UAV hovering locations (especially for the low-energy sensors) can be adjusted by Algorithm 2 to maximize the objective value under UAV traveling distance constraint.

Table IV shows the performance comparisons among the proposed algorithm and exhaustive search, where $D_{tr}^{\max} = 900$ [m]. It is observed from Table IV that the objective value of PA2 is similar to that of exhaustive search ($\Delta = 4.6$ [%]). Despite the existence of an optimality gap, the proposed algorithm significantly shortens the computation time required to find the proposed UAV route (see Fig. 6). In addition, it is observed from Table IV that compared to problem (4) when $\mathbf{I} = \mathbf{N}$, determining $\mathbf{I}$ by UAV hovering location gives higher objective value. In other words, the constraint (4a) is critical to find the "sensor energy saving" UAV route where the UAV needs to move closer to many sensors as possible for energy-efficient data collection.

Fig. 7 represents the average objective value of the proposed algorithms under different UAV traveling distance constraints, where the interval of $D_{Tr}^{\max}$ follows the results in Fig. 5. Note that when the average objective value is lower than 0.0487, there is a case when the proposed algorithm cannot find a feasible UAV route to problem (4) due to UAV traveling distance constraint. It is observed from Fig. 7 that when $D_{tr}^{\max}$ is small, PA3 and

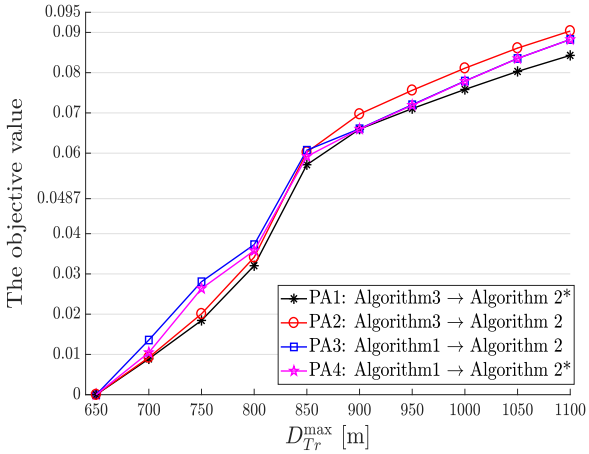


Fig. 7. The objective value of the proposed algorithms versus the maximum allowable UAV traveling distance.

PA4 have the average objective value higher than PA1 and PA2. This is because an initial UAV route from Algorithm 3 does not minimize UAV traveling distance, which results in an infeasible UAV route under UAV traveling distance constraint. However, as $D_{tr}^{\max}$ gets larger, Algorithm 3 can provide feasible UAV route and hence PA2 works better than other algorithms by the same reason as in Fig. 6.

VI. Conclusion

In this paper, we investigated UAV-enabled WSNs, for which a Voronoi diagram based algorithm is proposed for UAV route to maximize the minimum residual energy of sensors after data transmission. For practical system operation, we formulated an optimization problem constrained by data collection and UAV traveling distance. Two different schemes for an initial feasible UAV route are provided, where a Voronoi diagram can provide UAV hovering locations with low computational complexity. Then, the proposed UAV route is obtained by sequentially adjusting each UAV hovering location considering sensor energy status. Simulation results demonstrate that the performance of our proposed algorithm is almost the same as that of exhaustive search with very low computational complexity. In addition, the proposed UAV route can reduce energy consumption of many sensors resulting in energy-efficient data collection, which increases network lifetime considerably.

Appendix A
Analysis of Time Complexity in Table III

- *TSP solution:* TSP can be solved with time complexity of $\mathcal{O}(N^2 2^N)$ [16]. Note that heuristic algorithms can be applied to reduce time complexity, but cannot guarantee an optimal solution.
- *Proposed algorithm (Algorithm 1):* It utilizes a reference path and $\mathbf{mVD}(\{d_n^{\max}\})$ to find a shortest UAV route, where the reference path is obtained by solving TSP with time complexity of $\mathcal{O}(N^2 2^N)$. $\mathbf{mVD}(\{d_n^{\max}\})$ can be determined by constructing Voronoi diagram and comparing Voronoi vertices with MHCR. Since Voroni diagram can be obtained with the time complexity of $\mathcal{O}(N \log N)$ in a centralized fashion [31] and the number of Voronoi vertices in Voronoi diagram is at most $2N-5$ (i.e., $\mathcal{O}(N)$) [30], the total time complexity to construct $\mathbf{mVD}(\{d_n^{\max}\})$ is $\mathcal{O}(N \log N)+\mathcal{O}(N) \approx \mathcal{O}(N \log N)$. Once a reference path and $\mathbf{mVD}(\{d_n^{\max}\})$ are given, steps 1–3 of Algorithm 1 are conducted, where step 1 can assign numbers to each Voronoi region requiring time complexity of $\mathcal{O}(N)$, step 2 searches for $\mathcal{O}(N)$ Voronoi vertices to determine UAV hovering locations, and $\mathcal{O}(1)$ time complexity is needed in step 3 for the termination. Therefore, the overall time complexity is $\mathcal{O}(N^2 2^N)+\mathcal{O}(N \log N)+\mathcal{O}(N)$.
- *The CSS algorithm:* It consists of three steps; (i) find a TSP solution, (ii) combine adjacent sensors, (iii) skip and substitute UAV hovering locations. As mentioned in [19], the overall time complexity is $\mathcal{O}(N^2 2^N)+\mathcal{O}(N^3 \log N)+\mathcal{O}(N^2 \log \frac{1}{\delta})$. Note that the last step involves a bisection method [26] with termination criteria $\delta(>0)$.
- *The Voronoi edge based method:* It connects a set of Voronoi edges to find a UAV route. Based on the TSP solution, the visiting order of sensors can be obtained, followed by a parameter-less genetic algorithm with time complexity of $\mathcal{O}(M \log M)$ [32] to minimize UAV traveling distance, where M is the number of fitness evaluations in the genetic algorithm. Therefore, the overall time complexity is $\mathcal{O}(N^2 2^N)+\mathcal{O}(N \log N)+\mathcal{O}(M \log M)$. Note that $M \gg N$, since genetic algorithm requires sufficient number of fitness evaluations to get a solution of required quality.
- *Exhaustive search:* It finds a shortest UAV route by searching for all possible concatenations of UAV hovering locations. To be specific, a grid with consistent spacing width 1 [m] is applied to regions within the MHCR of each sensor, where the number of possible UAV hovering locations is approximately $\pi(d_n^{\max})^2$ for $n \in \{1,\ldots,N\}$ [33]. By selecting N UAV hovering locations (with time complexity $\mathcal{O}((\pi(d_n^{\max})^2)^N)$) and solving TSP, a shortest UAV route can be obtained. Therefore, the overall time complexity is expressed as $N^2 2^N \mathcal{O}((\pi(d_n^{\max})^2)^N) = \mathcal{O}(N^2(2\pi(d_n^{\max})^2)^N)$.

References

[1] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proc. 33rd Annu. Hawaii Int. Conf. Syst. Sci.*, Jan. 2000, vol. 2, p. 10.
[2] Y. Tirta, Z. Li, Y.-H. Lu, and S. Bagchi, "Efficient collection of sensor data in remote fields using mobile collectors," in *Proc. 13th Int. Conf. Comput. Commun. Netw. (IEEE Cat. No.04EX969)*, Oct. 2004, pp. 515–519.
[3] M. Ma, Y. Yang, and M. Zhao, "Tour planning for mobile data-gathering mechanisms in wireless sensor networks," *IEEE Trans. Veh. Technol.*, vol. 62, no. 4, pp. 1472–1483, May 2013.
[4] 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.
[5] M. Dong, K. Ota, M. Lin, Z. Tang, S. Du, and H. Zhu, "UAV-assisted data gathering in wireless sensor networks," *J. Supercomputing*, vol. 70, no. 3, pp. 1142–1155, 2014.
[6] M. B. Ghorbel, D. Rodríguez-Duarte, H. Ghazzai, M. J. Hossain, and H. Menouar, "Joint position and travel path optimization for energy efficient wireless data gathering using unmanned aerial vehicles," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2165–2175, Mar. 2019.

[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] 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.
[9] H. Ghazzai, M. B. Ghorbel, A. Kadri, M. J. Hossain, and H. Menouar, "Energy-efficient management of unmanned aerial vehicles for underlay cognitive radio systems," *IEEE Trans. Green Commun. Netw.*, vol. 1, no. 4, pp. 434–443, Dec. 2017.
[10] Y. Zeng, J. Xu, and R. Zhang, "Energy minimization for wireless communication with rotary-wing UAV," *IEEE Trans. Wireless Commun.*, vol. 18, no. 4, pp. 2329–2345, Apr. 2019.
[11] Q. Zhang, M. Jiang, Z. Feng, W. Li, W. Zhang, and M. Pan, "IoT enabled UAV: Network architecture and routing algorithm," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3727–3742, Apr. 2019.
[12] 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.
[13] S. Rosati, K. Kruelecki, G. Heitz, D. Floreano, and B. Rimoldi, "Dynamic routing for flying ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 3, pp. 1690–1700, Mar. 2016.
[14] B. Mao *et al.*, "Routing or computing? The paradigm shift towards intelligent computer network packet transmission based on deep learning," *IEEE Trans. Comput.*, vol. 66, no. 11, pp. 1946–1960, Nov. 2017.
[15] O. S. Oubbati, N. Chaib, A. Lakas, P. Lorenz, and A. Rachedi, "UAV-assisted supporting services connectivity in urban VANETs," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3944–3951, Apr. 2019.
[16] E. L. Lawler *et al.*, *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optim.*, vol. 3. New York, NY, USA: Wiley, 1985.
[17] G. Xing, M. Li, T. Wang, W. Jia, and J. Huang, "Efficient rendezvous algorithms for mobility-enabled wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 11, no. 1, pp. 47–60, Jan. 2012.
[18] M. Zhao, Y. Yang, and C. Wang, "Mobile data gathering with load balanced clustering and dual data uploading in wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 14, no. 4, pp. 770–785, Apr. 2014.
[19] L. He, J. Pan, and J. Xu, "Reducing data collection latency in wireless sensor networks with mobile elements," in *Proc. IEEE Conf. Comput. Commun. Workshops*, Apr. 2011, pp. 572–577.
[20] Y. Eun and H. Bang, "Cooperative task assignment/path planning of multiple unmanned aerial vehicles using genetic algorithm," *J. Aircr.*, vol. 46, no. 1, pp. 338–343, 2009.
[21] Y. V. Pehlivanoglu, "A new vibrational genetic algorithm enhanced with a Voronoi diagram for path planning of autonomous UAV," *Aerosp. Sci. Technol.*, vol. 16, no. 1, pp. 47–55, 2012.
[22] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. B. Srivastava, "Coverage problems in wireless ad-hoc sensor networks," in *Proc. IEEE INFOCOM'01, Conf. Comput. Commun., 20th Annu. Joint Conf. IEEE Comput. Commun. Soc. (Cat. No.01CH37213)*, Anchorage, AK, USA, 2001, vol. 3, pp. 1380–1387.
[23] M. Jun and R. D'Andrea, "Path planning for unmanned aerial vehicles in uncertain and adversarial environments," in *Cooperative Control: Models, Applications and Algorithms*. New York, NY, USA: Springer, 2003, pp. 95–110.
[24] X. Zhang and H. Duan, "An improved constrained differential evolution algorithm for unmanned aerial vehicle global route planning," *Appl. Soft Comput.*, vol. 26, pp. 270–284, 2015.
[25] E. Welzl, "Smallest enclosing disks (balls and ellipsoids)," in *New Results and New Trends in Computer Science*. New York, NY, USA: Springer, 1991, pp. 359–370.
[26] E. K. Chong and S. H. Zak, *An Introduction to Optimization*, vol. 76. New York, NY, USA: Wiley, 2013.
[27] J. Y. Yen, "Finding the k shortest loopless paths in a network," *Manage. Sci.*, vol. 17, no. 11, pp. 712–716, 1971.
[28] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge, U.K.: Cambridge Univ. Press, 2005.
[29] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
[30] F. Aurenhammer, "Voronoi diagrams—A survey of a fundamental geometric data structure," *ACM Comput. Surv.*, vol. 23, no. 3, pp. 345–405, 1991.
[31] W. Alsalih, K. Islam, Y. N. Rodríguez, and H. Xiao, "Distributed Voronoi diagram computation in wireless sensor networks," in *Proc. Symp. Parallelism Algorithms Architectures*, 2008, p. 364.
[32] M. Pelikan and F. G. Lobo, "Parameter-less genetic algorithm: A worst-case time and space complexity analysis," in *Proc. Genetic Evol. Comput. Conf.*, Citeseer, 2000, p. 370.
[33] D. Hilbert and S. Cohn-Vossen, *Geometry and the Imagination*, no. 87. American Mathematical Soc., 1999.

Jaeuk Baek (S'19) received the B.S. degree in electrical engineering from Hanyang University, Seoul, Korea, in 2015, and is currently working toward Ph.D. degree in electrical engineering with the Korea Advanced Institute of Science and Technology, Daejeon, Korea. His research interests include resource management in wireless communication, optimization, UAV networks, and wireless sensor networks.

Sang Ik Han (S'11–M'16) received the B.S. degree in electrical and electronics engineering from Chung-Ang University, Seoul, South Korea, in 2006, the M.S. degree in optics from the University of Central Florida, Orlando, in 2009, and the Ph.D. degree in electrical engineering from the University of Texas at Dallas, Richardson, in 2014. He was a Postdoc Research Fellow with the School of Electrical Engineering at the Korea Advanced Institute of Science and Technology, Daejeon, South Korea. He is currently a CEO with Spacesoft Industries Co., Ltd., Daejeon, South Korea, where the universal platform for unmanned vehicles (e.g., drone or UAV) including hardware and software is being developed for various purposes. His research interests include 5G communication systems, UAV networks, vehicular communications, and delay-tolerant network. Dr. Han serves as a Reviewer for IEEE COMMUNICATIONS MAGAZINE, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, and IEEE ACCESS.

Youngnam Han (SM'99) received the B.S. and M.S. degrees in electrical engineering from Seoul National University, Seoul, Korea, in 1978 and 1980, respectively, and the Ph.D. degree in electrical engineering from the University of Massachusetts, Amherst, MA, USA, in 1992. He was a Principal Engineer with the Electronics and Telecommunications Research Institute, Daejeon, South Korea, from 1992 to 1997, managing the project of design and performance analysis of radio transmission technology for DCN, PCS, and IMT-2000. He was actively involved in research and development for IS-95 digital cellular system in Korea deployed nationwide in 1995 and for IMT-2000 standards activities as a Delegate at ITU-R representing Korea. He has been with Information and Communications University, Daejeon, South Korea, as a Faculty Member, since 1998, where he served as the Dean of Engineering and the Dean of Academic and Student Affairs. He was a Principal Engineer with Qualcomm, Inc., San Diego, CA, USA, from 2001 to 2002, where he was involved in the 3G standards, CDMA2000 1xEV. Since 2009, he has been with the Department of Electrical Engineering, Korea Advanced Institute of Science and Technology, Daejeon, South Korea, as a Professor. His research interests include performance evaluation of mobile communication systems, radio resource management, and optimization of mobile systems operations. Dr. Han is a Life-Long Member of Korea Information and Communications Society. He served on many conferences as a TPC member and the Organizing Chair, including the TPC Chair for VTC 2003 (Spring). He was the Chairman of BoG and the IEEE VTS Asia Pacific Wireless Communication Symposium from 2009 to 2010. He served as the General Chair of VTC 2014 (Spring), Seoul, South Korea. From 2013 to 2017, he was the Chair on the 5G Forum Steering Committee in Korea and leading research and development activities on 5G. He was a recipient of the Best Paper Award at the IEEE VTC 2000 in Tokyo.