

Received January 22, 2018, accepted February 24, 2018, date of publication March 12, 2018, date of current version March 28, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2812896

Optimal UAV Path Planning: Sensing Data Acquisition Over IoT Sensor Networks Using Multi-Objective Bio-Inspired Algorithms

QIN YANG AND SANG-JO YOO[✉] (Member, IEEE)

Department of Information and Communication Engineering, Inha University, Incheon 402-751, South Korea

Corresponding author: Sang-Jo Yoo (sjyoo@inha.ac.kr)

This work was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT, and Future Planning under Grant NRF-2017R1A2B4003512.

ABSTRACT The use of unmanned aerial vehicles (UAVs) has been considered to be an efficient platform for monitoring critical infrastructures spanning over geographical areas. UAVs have also demonstrated exceptional feasibility when collecting data due to the wide wireless sensor networks in which they operate. Based on environmental information such as prohibited airspace, geo-locational conditions, flight risk, and sensor deployment statistics, we developed an optimal flight path planning mechanism by using multi-objective bio-inspired algorithms. In this paper, we first acquire data sensing points from the entire sensor field, in which UAV communicates with sensors to obtain sensor data, then we determine the best flight path between neighboring acquisition points. Using the proposed joint genetic algorithm and ant colony optimization from possible UAV flight paths, an optimal one is selected in accordance with sensing, energy, time, and risk utilities. The simulation results show that our method can obtain dynamic environmental adaptivity and high utility in various practical situations.

INDEX TERMS Bio-inspired algorithms, multi-objectives, optimal path, sensor networks, unmanned aerial vehicle.

I. INTRODUCTION

Unmanned aerial vehicles (UAVs), also known as drones, are expected for rapidly deployment in many sectors of our daily lives. UAVs now perform a wide range of activities, from package delivery to water submersion for underwater operations. There are also many applications operating in wireless networks that can employ UAVs in order to extend the range of communication [1], maximize network data communication capability by using vehicles as relay nodes [2], collect data from a wide area network in remote or harsh environments [3] and aid node localization in a mobile network [4]. Data collection from stationary sensor nodes is a typical application for Internet of Things (IoT) sensor networks. Given that sensors are randomly scattered throughout a vast area, it is not easy to obtain information from all sensors if they are not fully connected. UAVs are promising carriages for data gathering in sensor networks due to their direct communication abilities between the UAV and sensor nodes. Regarding the communication quality between sensor networks and UAVs, studies of [5] have shown that a lower

bit error rate (BER) is experienced if the UAV communicates with fewer sensor nodes at a given time. However, flight time will be significantly increased if the UAV needs to fly over all the nodes in the wireless sensor network. Ho *et al.* [6] introduced a new MAC protocol for communication between UAVs and sensor nodes in order to minimize energy usage and maximize network life.

The objective of this paper is to ascertain the optimal flight path for UAVs to maximize the value of gathered sensing information and to minimize the total cost in terms of UAV flying time, energy consumption and the operation risk for a UAV in its given environmental conditions. The environmental information for UAV path planning includes geographical topology, location dependent wireless communication channel statistics and flight risk, sensor node deployment and worth of sensing information for different sensor types. Generally, during UAV operation in a wide area sensor network, there are some constraints in terms of operation time, total energy consumption and prohibited flight areas so that deriving the optimal UAV flight path and the method

used to obtain the sensing information from a dedicated set of sensors are NP-hard problems. In this paper, we propose an evolutionary algorithm that jointly employs Genetic Algorithm (GA) and Ant Colony Optimization (ACO) algorithms. We have divided the entire sensor field into unit area cells, and first consider the set of sensing information gathering (SIG) cells. UAVs use SIG cells to communicate with sensors in order to obtain sensor data. We then derive the best flying cell sequence between neighboring SIG cells. Using the proposed joint GA and ACO evolutional optimization algorithms, an optimal flight path is selected from candidate UAV paths in accordance with proposed utility functions.

The rest of this paper is organized as follows. Related work is presented in Section II. In Section III, we present the proposed system structure, along with its detailed constraints and conditions. The utility function is described in Section IV. The proposed optimal UAV path finding scheme using joint evolutionary algorithm is presented in Section V. The performance evaluation with simulation results is revealed in Section VI. Conclusions are drawn in Section VII.

II. RELATED WORK

Path planning is one of the fundamental tasks for UAV operation which has been studied for decades [7]. A variety of algorithms and approaches have emerged. Dijkstra's algorithm [8] is for finding the shortest path between nodes in a graph. A* search algorithm [9] is one of the most common search algorithms for route planning that achieves better performance by using heuristics to guide a search in cases that involve a two-dimensional environment. Recently, these methods have been extended to work in three-dimensional (3D) geographical environments. The path planning problem in three-dimensional environments containing no obstacles was addressed by Kanchanavally *et al.* [10] and with static obstacles in 3-D path planning for the navigation of UAVs [11]. Nikolos and Valavanis [12] used B-spline curves to simulate the 3D flight trajectory of aircraft, and then used an evolutionary algorithm to optimize the B-spline curve control points. This method can generate a smooth 3D path, but will cause complexity when trying to avoid obstacles. The advent of intelligent optimization algorithm is foremost in solving the highly flexible problems of route determination depending on different objective functions. For example, evolutionary algorithms with binary coding have also been used for solving the path-finding problem in 3D environments for underwater vehicles, assuming that the path is a sequence of cells in a 3D grid. Doherty, *et al.* developed a method for controlling a quad-rotor vehicle to hover at a stable location [13]. These features enable UAVs to move vertically and hover at a stable location, which makes them suitable for scanning. In order to hover at a fixed altitude to perform accurate scanning, the results of altitude retention through Fuzzy Particle Swarm Optimization (PSO) method were measured. Tulum *et al.* [14] introduced an agent-based approach for the UAV mission route planning problem by using situation awareness algorithms. The approach

considers target, threat, terrain and airspace restrictions to compute the best set of waypoints for the UAV mission. In another study [15], deterministic and probabilistic path planning strategies for an autonomous UAV network are examined by exploring a given area of obstacles, and providing an image overview. This study presents algorithms for online and offline implementations, and concludes that the applied approach can provide a solution with a minimal number of pictures. This plan requires more knowledge and time to generate, while the probabilistic approaches are more flexible and adaptive.

In [16] a comprehensive survey on the UAV-based value-added IoT services is introduced. It mainly studied UAV collision avoidance techniques, wireless communication issues of flying ad-hoc network (FANET), on board UAV sensors and relevant data processing collected by UAV. L. Tong, *et al.* proposed an architecture called SENMA [17] for large low power sensor network with mobile agents. In SENMA, mobile agents are the receiving terminals in data collection and it offers considerable advantage in energy efficiency but only for flat ad hoc network. Ho *et al.* [18] dealt with selection of sensor network communication topology and the use of UAVs for data gathering. Their objective function is consisted of energy, communication quality and travelling time optimized with PSO method. Another novel bio-inspired multi-objective optimization algorithm named MOCAS [19] was proposed for wireless sensor network in terms of generational distance, error ratio, and spacing. This research is still facing that how to set algorithm parameters and handle constrained problems. A SWARM-based data delivery algorithm [20] was proposed to guarantee the connectivity among objects and people, fault tolerance routing and improved particle multi-swarm optimization performance.

In this paper, we have derived a set of optimal sensing points and flight paths for given UAV operational constraints in 3D environments. In our approach, distance is not the only cost-determinant variable we take into account. Multi-objective UAV operation, UAV energy consumption, sensing and flying time, UAV flying angle and environmental risk are jointly considered in this paper. This makes it unlike many other conventional approaches in UAV path planning.

III. SYSTEM MODELING

A. TOPOLOGY STRUCTURE

In this paper we divide the entire IoT sensor field into a multitude of small unit areas called cells as shown in Fig.1. Fig.1 shows UAV sensing and flying geographical environment. x -axis and y -axis represent location coordinates and z -axis denotes geographical height. The red curve represents one example of UAV flight path through cells which ends at the start point cell.

Our proposed path planning scheme includes two parts. First, we derive optimal sensing points to gather sensing information from different types of sensors on the ground.

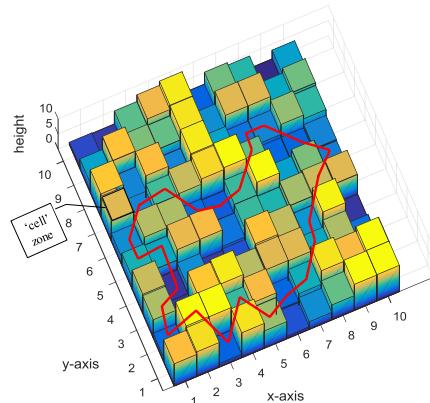


FIGURE 1. UAV sensing and flight path in 3D environment.

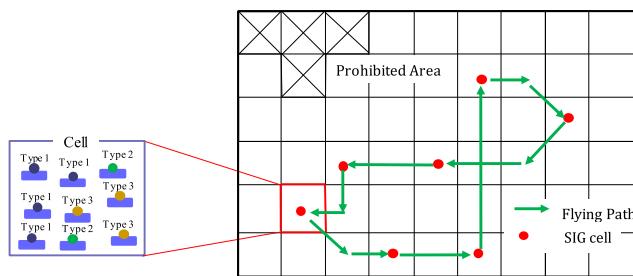


FIGURE 2. UAV search area topology: SIG cells and flying cells.

We assume that there are Q sensor types and the value (in other word operational worth) of the sensing information of each sensor type can be different. Second, we determine the best flight path between neighboring sensing points. In this paper, we define two cell types: sensing information gathering cells (SIG cells) and flying cells. In a SIG cell, the UAV stays on that cell area and communicates with ground sensors to obtain all sensing information. Each sensor type may have a different value of sensing information (VSI) which presents the rank of a sensor's informational importance. For example, in a certain application, the sensing information from a gas sensor may be worth more than the information from a temperature sensor.

The sensors can be deployed randomly or intentionally so that the sensor density of a cell can differ. Moreover, sensors have different types, and a cell may have multiple sensor types. As shown in Fig.2, the UAV simply passes through flying cells, and only communicates with terrestrial sensors to obtain the sensing information in SIG cells.

A complete flight path consists of a set of SIG cells (C_{SIG}) and a set of flying cells (C_{fly}) which denotes as

$$C_{SIG} = \{C_{SIG}^1, C_{SIG}^2, \dots, C_{SIG}^n, \dots, C_{SIG}^{N_{SIG}}\} \quad (1)$$

$$C_{fly} = \{S_{fly}^1, S_{fly}^2, \dots, S_{fly}^n, \dots, S_{fly}^{N_{SIG}}, S_{fly}^{N_{SIG}+1}\} \quad (2)$$

$$S_{fly}^n = \{C_{fly(n)}^1, C_{fly(n)}^2, \dots, C_{fly(n)}^i, \dots, C_{fly(n)}^{F(n)}\} \quad (3)$$

where C_{SIG}^n is the n th SIG cell and N_{SIG} is the total number of SIG cells. The set of flying cells, C_{fly} , includes flying subsets. The flying subset S_{fly}^n is the sub-path of flying cells from $(n-1)$ th SIG cell to n th SIG cell. Therefore, S_{fly}^1 is the sub-path from the starting point cell (C_{start}) to the first SIG cell (C_{SIG}^1) and $S_{fly}^{N_{SIG}+1}$ is the sub-path from the last SIG cell ($C_{SIG}^{N_{SIG}}$) to the end point cell (in the case that the UAV should return to the starting point, it is C_{start}). $C_{fly(n)}^i$ and $F(n)$ represent the i th flying cell and the total number of flying cells for n th flying subset, respectively.

B. CONSTRAINTS AND CONDITIONS

In this section, constraints and conditions for UAV operation are defined in accordance with the sensor network topology conducted in this paper. We focus on energy consumption, time expenditure, the flying pitch angle of UAV and prohibited area on topology.

1) ENERGY CONSTRAINTS

Small UAVs mostly operate on lithium-polymer batteries (Li-Po), while larger vehicles rely on the power of conventional airplane engines. Recently, hydrogen fuel batteries have been studied for using in UAV power systems, and have the potential to double or triple energy efficiency. In any power supply mechanism, a UAV has a finite amount of energy. This means we must ensure that a UAV finishes its task and returns safely to the predetermined end point before it runs out of energy. If a UAV falls to the ground due to energy expenditure on its return path after acquiring sensing information, the obtained information is lost, and there will be problems in subsequent UAV operations. In this paper, the energy constraint parameter is E^{limit} .

2) TIME CONSTRAINTS

After sensing information is acquired by a UAV, it should be delivered to the application system within a certain time period. Therefore, UAV flight is generally time-constrained (T^{limit}). The number of SIG cells, as well as their positions and flight paths between SIG cells, should be carefully designed to guarantee that a UAV operation should be completed within a certain time period.

3) PITCH ANGLE CONSTRAINTS

The roll, pitch and yaw angles in a three-dimensional environment are important flight dynamic parameters for most air vehicles. Rotation around the front-to-back axis is called roll, which is controlled by a vehicle's ailerons. Rotation around the side-to-side axis is called pitch, as in Fig.3, which is controlled by the vehicle's elevator. Rotation around vertical axis is called yaw, which is controlled by the vehicle's rudder. In this paper, we only consider the pitch motion for UAVs. Due to difficulties involved with vertical up and plummet, we assume that there is a predefined maximum pitch angle (ϑ^{limit}) for a given UAV's motion.

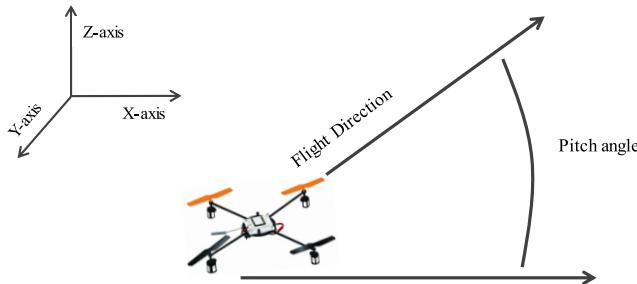


FIGURE 3. UAV pitch angle explanation.

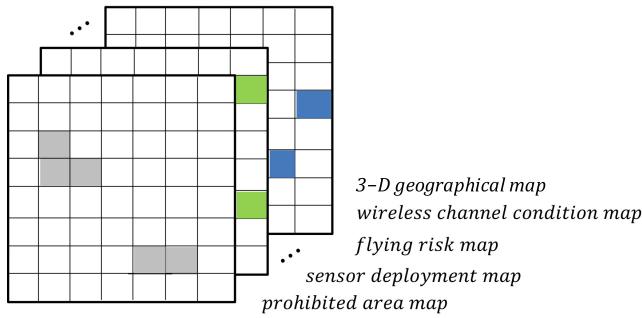


FIGURE 4. Multi-layer information maps.

4) PROHIBITED AREAS

In general, flying UAVs over designated facilities or areas such as government buildings or military facilities is prohibited by regulations. Therefore, we define prohibited areas in our model as shown in Fig.2. SIG cells that are inside a prohibited area should not be selected, and the UAV should move to the next SIG cell without passing any prohibited area cells. UAVs should take the minimum cost path that does not include a prohibited area during its flight.

IV. UTILITY FUNCTION FOR UAV OPTIMAL PATH PLANNING IN WIRELESS SENSOR NETWORKS

In this section, we propose a utility function for a UAV optimal path planning scheme that maximizes UAV operational utility and satisfies multiple constraints. The UAV operational goals are defined in the form of multi-objective utility functions such as energy, time, value of sensing information and risk. To maximize the total utility value, in this paper we apply bio-inspired algorithms to derive optimal SIG cell and flying cell sequences.

We assume that in setting up the optimal path, we already have information in the form of a multi-layer information map (MLImap) as shown in Fig.4. The MLImap consists of a prohibited-area map, sensor deployment map, flight risk map, wireless channel condition map and a three-dimensional geographical map. The maps can be given at initial UAV operation time and they are updated at every UAV flight operation. The entire given sensor field is subdivided into unit-sized cells, and each map has representative information for each cell unit.

A. UTILITY FUNCTION DESIGN

In sensing information acquisition using UAVs in wireless sensor networks, UAV sensing and flight details should be carefully planned for maximization of the utility to achieve the desired purpose. We propose four utility functions that capture sensing information values (sensing utility, U^S), required energy (energy utility, U^E), expected UAV operational time (time utility, U^T) and UAV flight risk (risk utility, U^R) [9]. The sensing utility has a positive value, but all other utilities have negative values. The total utility to evaluate the effectiveness of the candidate UAV path plan p is defined as a weighted sum of all utility values.

$$U_p = \omega_S \times \frac{U^S}{\max U^S} + \omega_E \times \frac{U^E}{\max U^E} + \omega_T \times \frac{U^T}{\max U^T} + \omega_R \times \frac{U^R}{\max U^R} \quad (4)$$

where, $\omega_S + \omega_E + \omega_T + \omega_R = 1$; $\max U^S$, $\max U^E$, $\max U^T$ and $\max U^R$ are the pre-defined maximum values.

The optimum path p^* from the all possible candidate paths can be derived from (5), in which p^* has the maximum utility value and satisfies the path constraints.

$$\begin{aligned} p^* &= \underset{p}{\operatorname{argmax}} U_p \\ &\text{subject to } U_{p^*}^E \leq E^{\text{limit}} \\ &\quad U_{p^*}^T \leq T^{\text{limit}} \end{aligned} \quad (5)$$

where, $U_{p^*}^E$ and $U_{p^*}^T$ are sensing and time utility of the optimum path, respectively. The pitch angle and prohibited area constraints are considered for candidate path derivation process that will be explained in Section V.

1) SENSING UTILITY

The sensing utility (U^S) is the sum of the value of sensing information (VSI) obtained from the sensors at all visiting SIG cells. In this paper, the VSI from each sensor can be different depending on the sensor type (e.g., temperature sensor, flame sensor, gas sensor, vibration sensor or light sensor) and it is also a function of historical observation results at that area. Let U_n^S be the sensing utility at SIG cell n ; Q be a number of sensor types in the sensor field; s_n^q be the number of sensor nodes of type q at SIG cell n ; $v_n^q(t)$ is the VSI of the sensor type q sensor node at SIG cell n at time t . The total sensing utility U^S obtained from all SIG cells is computed as,

$$U^S = \sum_{n=1}^{N_{SIG}} U_n^S \quad (6)$$

$$U_n^S = \sum_{q=1}^Q s_n^q v_n^q(t) \quad (7)$$

where, N_{SIG} is the number of SIG cells, $v_n^q(t)$ update rule is explained in the next section.

2) ENERGY UTILITY

The energy utility (U^E) is calculated by total energy consumption, as the sum of communication energy utility (U^{CE}), staying energy utility (U^{SE}) and flying energy utility (U^{FE}).

First, the communication energy is required to successfully transmit and receive sensing information data between sensors and the UAV. The UAV requires communication energy only at SIG cells because at flying cells it does not communicate with sensors. The communication energy utility U_n^{CE} at nth SIG cell is a function of the number of sensors, unit communication energy per single packet transmission, and wireless channel condition at the given area as in (8) and (9).

$$U^{CE} = - \sum_{n=1}^{N_{SIG}} U_n^{CE} \quad (8)$$

$$U_n^{CE} = P^c N_n^P \sum_{q=1}^Q t_e^q s_n^q \quad (9)$$

where P^c shows unit of communication energy for a unit time provided by the UAV power system in a cell; N_n^P is the required number of transmissions to successfully transfer a sensing packet in cell n ; t_e^q is the single sensing packet exchange time (in unit time) for the sensor type q ; s_n^q is the number of sensor nodes of type q in cell n . Depending on the sensing information length of each sensor type, t_e^q can be different. Wireless channel condition in sensor networks is a location dependent parameter that impacts packet transmission efficiency. In this paper, we assume that before t -th UAV path set up time, we already know the wireless channel condition from the multi-layer information maps. In our model, the location-dependent wireless channel condition is represented by the packet error probability, which can be obtained from the previous UAV flight trials, or measured in advance. For the given packet error probability p_n^e at cell n , the total required number of packet transmissions N_n^P to successfully transfer a packet can be defined as in (10) using geometric distribution [9].

$$N_n^P = \frac{1}{1 - p_n^e} \quad (10)$$

At each SIG cell, a UAV also needs some amount of energy to maintain its current flight altitude. The staying energy utility (U^{SE}) is defined as follows:

$$U^{SE} = - \sum_{n=1}^{N_{SIG}} U_n^{SE} \quad (11)$$

$$U_n^{SE} = P^s N_n^P \sum_{q=1}^Q t_e^q s_n^q \quad (12)$$

where P^s is the unit staying energy provided by UAV power system.

To move from its current cell to the next cell, the UAV consumes flying energy. The flying energy utility U^{FE} depends on the number of SIG cells and the number of flying cells between the SIG cells. It is also a function of locations of cells on the UAV path. Energy is location-dependent, and the fundamental concept of flying energy utility (U^{FE}) is the sum of energy utilities of all sub-paths of C_{fly} of (2), which is expressed as follows:

$$U^{FE} = - \sum_{n=1}^{N_{SIG}+1} U_n^{FE} \quad (13)$$

$$U_n^{FE} = \sum_{\forall C_{fly(n)}^k \in S_{fly}^n} e_{fly(n)}^{k,k+1} \quad (14)$$

where U_n^{FE} represents the flying energy utility from $(n-1)$ th SIG cell to n th SIG cell; $C_{fly(n)}^k$ is k th cell on the path from $(n-1)$ th SIG cell to n th SIG cell; S_{fly}^n is the set of all cells on the path from $(n-1)$ th SIG cell to n th SIG cell; $e_{fly(n)}^{k,k+1}$ is the flying energy from $C_{fly(n)}^k$ cell to $C_{fly(n)}^{k+1}$ cell. $e_{fly(n)}^{k,k+1}$ can be different depending on the UAV power supply mechanism. However, it definitely depends on the three-dimensional geographical conditions between neighboring cells. In our model, for simplicity $e_{fly(n)}^{k,k+1}$ is computed as in (15).

$$e_{fly(n)}^{k,k+1} = t_{fly(n)}^{k,k+1} \times P^f \quad (15)$$

$$t_{fly(n)}^{k,k+1} = \frac{d_{fly(n)}^{k,k+1}}{v_{fly(n)}^{k,k+1}} \quad (16)$$

where, $t_{fly(n)}^{k,k+1}$, $d_{fly(n)}^{k,k+1}$ and $v_{fly(n)}^{k,k+1}$ represent flying time, three-dimensional distance, and moving speed from k th cell to $(k+1)$ th cell on the path from $(n-1)$ th SIG cell to n th SIG cell, respectively. P^f is the unit flying energy provided by the UAV's power system. Finally, the energy utility is the sum of communication, staying and flying energy utilities.

$$U^E = U^{CE} + U^{SE} + U^{FE} \quad (17)$$

3) TIME UTILITY

Time is also a crucial element in the UAV flight path utility. The total time that the UAV spends in the sensor field includes the time required to obtain all sensing information values at every SIG cell and the time to fly through every cell on the path. Therefore, the time utility (U^T) can be represented as in (18)

$$U^T = - \left\{ \sum_{n=1}^{N_{SIG}} \left(N_n^P \sum_{q=1}^Q t_e^q s_n^q \right) + \sum_{n=1}^{N_{SIG}+1} \sum_{\forall C_{fly(n)}^k \in S_{fly}^n} t_{fly(n)}^{k,k+1} \right\} \quad (18)$$

4) RISK UTILITY

Staying risk and flying risk are location-dependent parameters and the risk increases when the UAV stays or flies for more time at a risky area. The total risk utility U^R can be computed as in (19).

$$U^R = - \left\{ \sum_{n=1}^{N_{SIG}} \left(r_n N_n^P \sum_{q=1}^Q t_e^q s_n^q \right) + \sum_{n=1}^{N_{SIG}+1} \right. \\ \left. \times \sum_{\forall C_{fly(n)}^k \in S_{fly}^n} \left(r_{fly(n)}^{k,k+1} \times t_{fly(n)}^{k,k+1} \right) \right\} \quad (19)$$

where, r_n is the staying risk at n th SIG cell; $r_{fly(n)}^{k,k+1}$ is the flying risk to move from $C_{fly(n)}^k$ cell to $C_{fly(n)}^{k+1}$ cell. $r_{fly(n)}^{k,k+1}$ is defined as in (20).

$$r_{fly(n)}^{k,k+1} = \omega_g v_{fly(n)}^{k,k+1} + \omega_h h_{fly(n)}^{k,k+1} + \omega_e v_{fly(n)}^{k,k+1} \quad (20)$$

where, $\vartheta_{fly(n)}^{k,k+1}$ is the pitch angle; $h_{fly(n)}^{k,k+1}$ is height difference; $v_{fly(n)}^{k,k+1}$ is environment risk; $\omega_g + \omega_h + \omega_e = 1$.

B. EVENT DETECTION-BASED VALUE OF SENSING INFORMATION UPDATE

In this paper, the environmental context awareness is taken into account for the value of sensing information (VSI) function design. In a real situation, if a sensing value (e.g., current temperature or wind speed) at a certain area is rapidly increasing or decreasing compared to previous observations, then we should be more attentive to that area. It indicates that there is a high demand to sense that area by sending a UAV so that the value of that area's sensing information needs to be adjusted to accommodate new event information. For example, in fire detection, if the measured value of a temperature sensor in a certain area has increased or decreased ten degrees when compared to previous measurements, then the VSI of the temperature sensors at that area needs to be adjusted in order to monitor the fire's movement more closely and precisely during the next UAV flight time. In addition, if an event was detected at a certain SIG cell during the previous observation, then we should need to observe the conditions of the event's neighboring cells. Therefore, the VSI of each neighbor cell also needs to be adjusted. On the other hand, the longer a certain cell has gone without observation, the higher the demand to monitor that area. We have defined three cases for VSI update:

- VSI value is dynamically updated depending on the difference between the latest observed sensing value and the previous historical average value.
- When an event is detected in a certain cell, the VSI values of the surrounding cells also need to be adjusted.
- When a cell is not measured longer than the number of threshold trials, the VSI value of the cell should be increased.

For the first case, the VSI update rule is defined as a function of previous observation history at a given area, as in (21).

$$v_n^q(t) = f(v_n^q(t-1), O_n^q(1), O_n^q(2), \dots, O_n^q(t-1)) \quad (21)$$

where, $v_n^q(t-1)$ and $O_n^q(t-1)$ are the VSI and observed sensing value of sensor type q at SIG cell n at $(t-1)$ th UAV flying time, respectively. Each sensor type has an initial VSI regardless of its position, such as $v_n^q(1) = v^q$. We have defined the dynamic VSI adjustment rule, as in (22) ~ (25).

If the absolute difference between the average observed sensing value and the last observed value is less than the predetermined threshold value t_{OSV} , then the VSI for the next flying time is equal to the initial value $v_n^q(t) = v^q$. Otherwise, the next VSI is updated with $z_n^q(t)$. Fig. 5 shows the example VSI update procedure. In this paper, for VSI updating, we defined three different $z_n^q(t)$ functions. Depending on the sensor network applications, one of the three functions can be applied by the UAV operator. Fig. 6 shows the VSI dynamics for the three functions in accordance with the observed

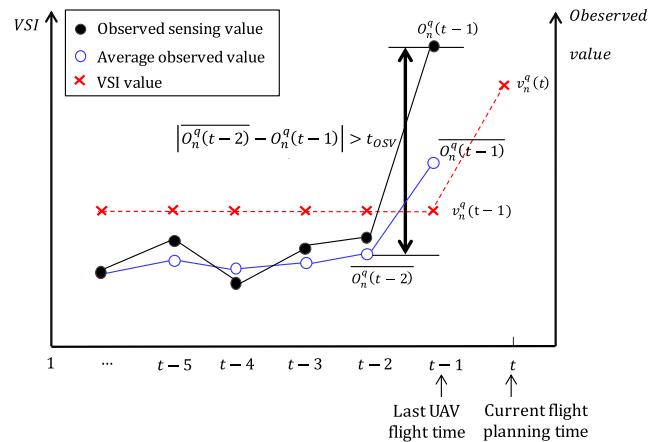


FIGURE 5. Time sequence to update VSI.

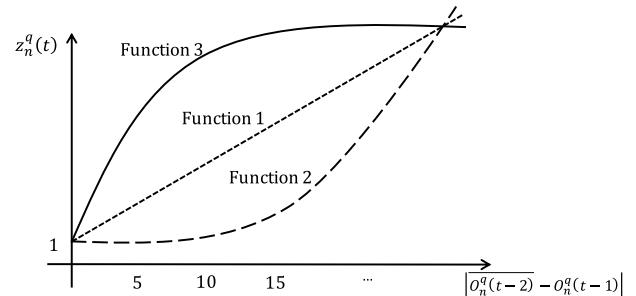


FIGURE 6. Three VSI dynamics in accordance with the observed sensing value changes.

sensing value changes, $|\overline{O_n^q(t-2)} - O_n^q(t-1)|$. To compute the average observed sensing value of sensor type q at SIG cell n up to $(t-2)$ th UAV operation time, $\overline{O_n^q(t-2)}$, exponentially weighted moving average (EWMA) is used as in (22). A higher weight parameter γ discounts older observations faster. In Fig. 6, it is assumed that $v_n^q(t-1)$ is equal to 1.

$$\overline{O_n^q(t-2)} = \gamma \overline{O_n^q(t-3)} + (1 - \gamma) O_n^q(t-2) \quad (22)$$

$$v_n^q(t) = \max [v_{\max}^q, w_n^q(t)] \quad (23)$$

$$w_n^q(t) = \begin{cases} v^q, & \text{if } |\overline{O_n^q(t-2)} - O_n^q(t-1)| < t_{OSV} \\ z_n^q(t), & \text{else} \end{cases} \quad (24)$$

$$z_n^q(t) = \begin{cases} \eta \times |\overline{O_n^q(t-2)} - O_n^q(t-1)| \times v_n^q(t-1) + 1, & \text{function 1} \\ \eta \times |\overline{O_n^q(t-2)} - O_n^q(t-1)| \times v_n^q(t-1), & \text{function 2} \\ \log(\eta \times |\overline{O_n^q(t-2)} - O_n^q(t-1)| + 1) \times v_n^q(t-1) + 1, & \text{function 3} \end{cases} \quad (25)$$

where, η is constant control parameter value; v_{\max}^q is the maximum VSI value for sensor type q .

For the second case, if a certain cell's VSI change, $|v_h^q(t) - v_h^q(t-1)|$, is greater than threshold t_{event} , then we can say that an event has been detected at the cell. In this case, the VSI value of the sensor type q of every surrounding neighbor cell x , $v_x^q(t)$ needs to be updated, as in (26).

$$v_x^q(t) = \max [v_{max}^q, \delta_1 \times v_x^q(t-1)], \quad \delta_1 > 1 \quad (26)$$

For the last case, if a cell y has not been selected as an SIG cell more than threshold t_{SIG} times, then the VSI values of any sensor type u in that cell are updated as (27).

$$v_y^u(t) = \max [v_{max}^u, \delta_2 \times v_y^u(t-1)], \quad \delta_2 > 1 \quad (27)$$

V. OPTIMAL PATH FINDING USING BIO-INSPIRED ALGORITHMS

In this section, optimal path finding with an evolutionary algorithm that jointly uses GA and ACO algorithms will be introduced. We have divided the entire sensor field into unit area cells, and first consider the set of SIG cells in which UAVs communicate with sensors to obtain sensor data. The optimization process will settle SIG cells first, and then derive the best flying cell sequence between neighboring SIG cells. From possible UAV paths, an optimal one is selected in accordance with the proposed multi-objective utility functions.

A. INITIAL SIG CELLS SEQUENCE SELECTION

Before UAV flight, we need to know the positions of the SIG cells that the UAV will acquire sensor data. Since there is no benefit acquiring sensing data from the same SIG cell more than once at one flight time and the UAV is not allowed to enter prohibited areas, the SIG cell sequence should include no duplicates and no prohibited cells. It should be noted that any SIG cell can be selected as a flying cell on the flying path between other SIG cell pair. It should be noted that any SIG cell can be selected as a flying cell on the flying path between other SIG cell pair. The initial SIG cell sequences are randomly selected with population size M . Population size means the number of SIG cell sequences to be evaluated. The number of SIG cells to acquire sensor data is determined as an operating parameter in practice. The possible SIG cell ratio to the total number of cells of the entire sensor field is determined as the range $[R_{sig}^{min}, R_{sig}^{max}]$. The SIG cell selection

probability, P_{sig} , is randomly selected within the range with stochastic choice before generating SIG cells for each chromosome. For example, when we have a 10×10 searching cell area and the range of the SIG cell ratio is $[0.05, 0.1]$, P_{sig} for a SIG cell sequence for each possible UAV path candidate can be chosen between 0.05 and 0.1. Therefore, it means that in average 7.5 SIG cells will be included in the initial SIG cell sequences. We then select the probability P_{sig} and generate the next SIG cell sequence until the number of SIG cell sequences reaches to the predetermined population size M .

B. FLYING CELL OPTIMIZATION USING ACO

Once the SIG cell sequences are generated, ACO is introduced for determining flight cell sequences. Real ants are capable of finding the shortest path from a food source to their nest without using visual cues, but by instead exploiting pheromone information. While walking, ants deposit pheromones onto the ground and follow, using probability, the pheromones previously deposited by other ants. In our approach, we use ACO to mimic this behavior by having "simulated ants" walk around the graph representing the problem. ACO selects optimum flight cell paths between one SIG cell and the next, which globalizes cell optimization.

In the construction of a solution, ants select the following cell to be visited through a stochastic mechanism. When ant k is in cell $C_{fly(n)}^i$, the probability of going to the next cell $C_{fly(n)}^j$ is given by:

$$P_{fly(n)}^{i,j}(k) = \begin{cases} \frac{\left(\tau_{fly(n)}^{i,j}\right)^{\alpha} \times \left(\eta_{fly(n)}^{i,j}\right)^{\beta}}{\sum_{i,l \in allowed} \left\{ \left(\tau_{fly(n)}^{i,l}\right)^{\alpha} \times \left(\eta_{fly(n)}^{i,l}\right)^{\beta} \right\}} & \text{if } (i, l) \in allowed \\ 0 & \text{otherwise} \end{cases} \quad (28)$$

where $allowed$ denotes the set of feasible cells that are not included in the prohibited area and the pitch angle between cell i and j satisfies the pitch angle constraint, $\vartheta^{i,j} \leq \vartheta^{limit}$. While β controls the relative importance of the heuristic information $\eta_{fly(n)}^{i,j}$ and α controls the relative importance of the pheromone versus $\tau_{fly(n)}^{i,j}$ which is defined respectively as:

$$\eta_{fly(n)}^{i,j} = \frac{1}{cst_{fly(n)}^{i,j}} \quad (29)$$

$$\tau_{fly(n)}^{i,j} \leftarrow (1 - \rho) \times \tau_{fly(n)}^{i,j} + \sum_{k=1}^m \Delta \tau_{fly(n)}^{i,j}(k) \quad (30)$$

where $cst_{fly(n)}^{i,j}$ is the cost between two adjacent cells $C_{fly(n)}^i$ and $C_{fly(n)}^j$; $\tau_{fly(n)}^{i,j}$ symbolizes pheromone value on this neighbor edge by ants. ρ is the evaporation rate and m is the number of ants; $\Delta \tau_{fly(n)}^{i,j}(k)$ is the quantity of pheromones laid on edge $\{C_{fly(n)}^i, C_{fly(n)}^j\}$ by ant k , which is designed as:

$$\Delta \tau_{fly(n)}^{i,j}(k) = \begin{cases} \frac{C}{L_k} & \text{if ant } k \text{ passes } (C_{fly(n)}^i, C_{fly(n)}^j) \text{ in its tour} \\ & \text{from the current SIG cell to next SIG cell.} \\ 0 & \text{otherwise} \end{cases} \quad (31)$$

$$L_k = \sum_{\{C_{fly(n)}^i, C_{fly(n)}^j\} \in S_{fly}^n} cst_{fly(n)}^{i,j} \quad (32)$$

$$cst_{fly(n)}^{i,j} = \omega_e \times e_{fly(n)}^{i,j} + \omega_t \times t_{fly(n)}^{i,j} + \omega_r \times r_{fly(n)}^{i,j} \quad (33)$$

where C is a constant value; L_k presents the total cost constructed by ant k in this iteration.

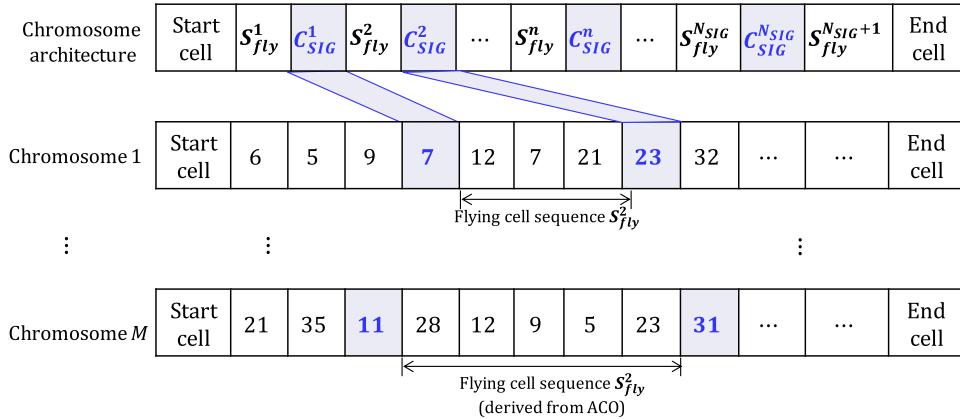


FIGURE 7. Chromosome of the proposed architecture.

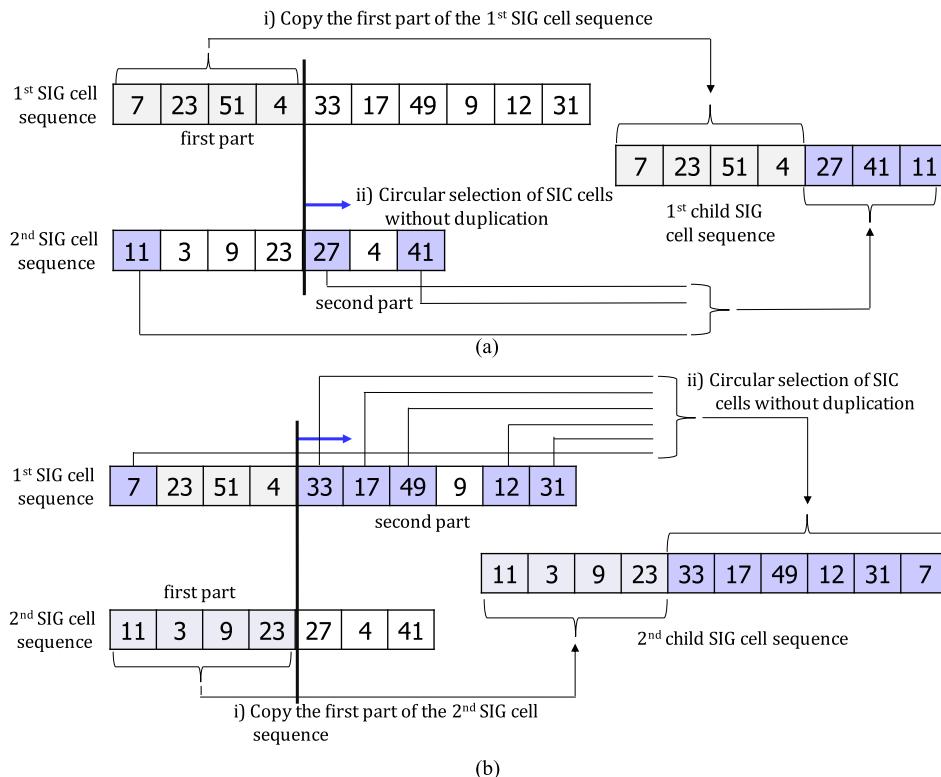


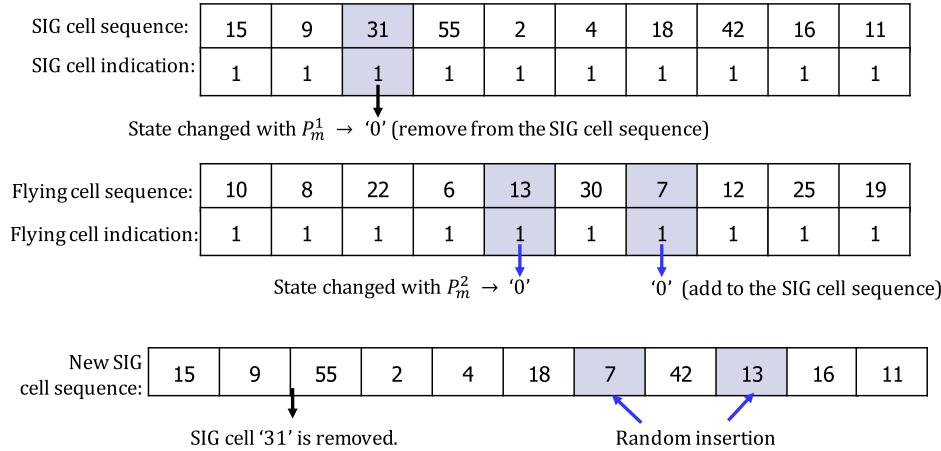
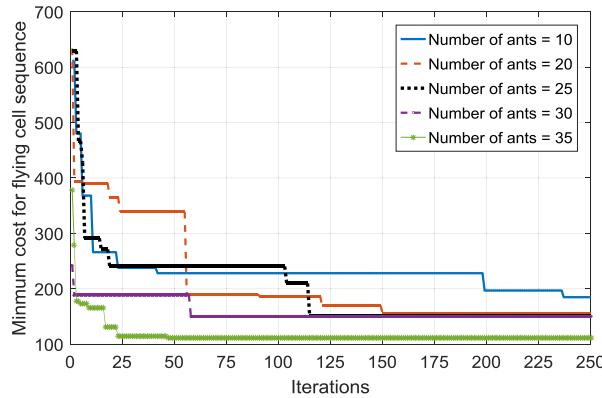
FIGURE 8. The proposed crossover strategy of SIG cell sequence.

C. UAV PATH OPTIMIZATION USING GA

The optimal path selection algorithm in this paper combines both of GA and ACO bio-inspired algorithms. Each GA chromosome represents a complete path from the start cell to the end cell and it consists of SIG cell sequence and flying cell sequences, in which a flying cell sequence a trajectory between two neighboring SIG cells. The proposed utility function for the optimal path selection is applied to whole chromosomes. Unlike the conventional GA method, crossover and mutation processes for the next chromosome generation only use subset sequences consisting of parent SIG cells because the selection of SIG cells and their visiting

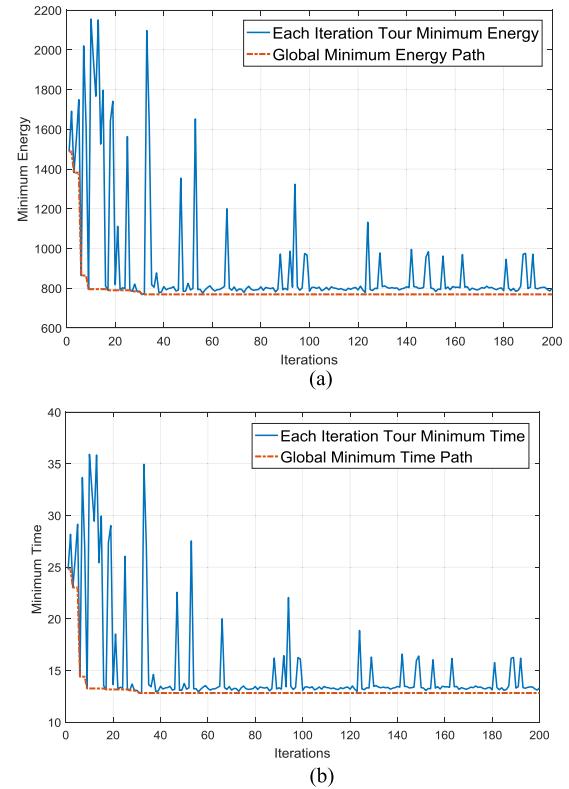
order are important to achieve the purpose of UAV sensing data acquisition operation. The proposed operation can reduce the search range and achieve faster convergence. As in conventional GA, if we perform crossover and mutation for the complete chromosomes, then the SIG cell and flying cell concept and their different characteristics cannot be effectively kept.

From section A and section B, we can obtain a string of integrated sequences called a chromosome. Fig. 7 shows the architecture of chromosomes in our approach. A complete chromosome begins from the start cell and ends at the endpoint cell (in the case that UAV returns to the starting point,

**FIGURE 9.** The proposed mutation strategy of SIG cell sequence.**FIGURE 10.** Minimum cost for flying cell sequence for different number of iterations and ant agents.

the end cell will be the start cell). In a chromosome path, there are selected SIG cells and between the adjacent SIG cells flying cell sequence exists. Each chromosome presents a possible solution to the given UAV path planning problem.

Basically, GA follows the sequence as generation of the initial population, evaluation, selection, genetic operators and regeneration. First, the initial population generated by randomly selecting SIG cells with P_{sig} probability and filling the flying cells between SIG cells with ACO algorithm, in which prohibited cells are excluded. Once chromosomes are generated, they are evaluated by the utility function we derived, such as the sensing utility, energy utility, time utility and risk utility. Next, two genetic operators, crossover and mutation, alter the composition of genes to create new chromosomes called offspring. The selection operator with certain probability is an artificial version of natural selection (a Darwinian “survival of the fittest” mechanism among populations) that creates generational populations containing chromosomes with successively better fitness values that then have a higher probability of being selected in the next generation. It is worth mentioning that both crossover and mutation are aimed at the SIG cells of chromosomes, which means that we make changes in SIG cells, and only then

**FIGURE 11.** ACO flying cell sequence optimization performance in terms of energy and time.

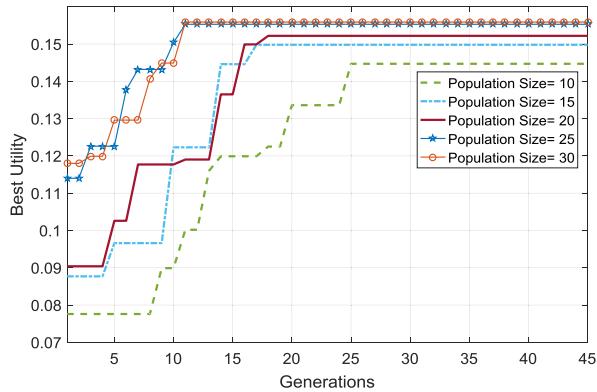
perform flight cell optimization using ACO to fill in all chromosomes.

D. PROPOSED MUTATION AND CROSSOVER MECHANISMS

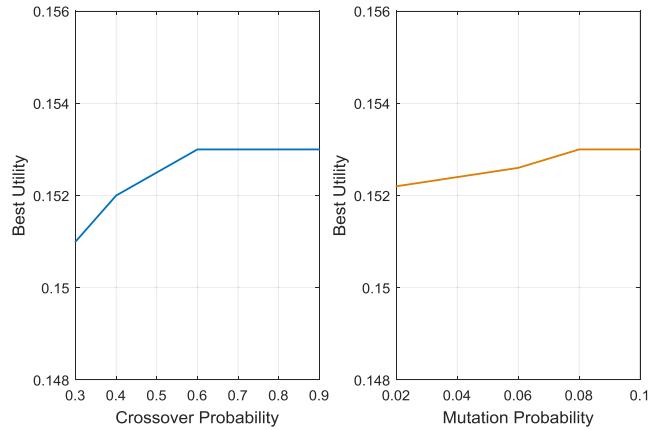
Fig. 8 shows an example scenario of the proposed strategy for crossover, and each number represents an SIG cell identification. First, pick out SIG cells from any two generated chromosomes with probability P_c . Then randomly draw a line dividing each sequence into two parts in an arbitrary

TABLE 1. Parameter definitions.

Parameters	Symbols	Values	Parameters	Symbols	Values
sensor network cell size	—	10×10	UAV time constraint	T^{limit}	35
cell height	—	0~1km	UAV energy constraint	E^{limit}	2200
UAV flight times	—	10	pitch angle constraint	ϑ^{limit}	$ \vartheta \leq 80^\circ$
environment risk	r_n, r_{fly}	0~20	packet error prob.	p_n^e	0.05~0.1
number of sensor types	Q	5	Initial VSI value	v^q	1~20
number of sensors for each sensor type q	s_n^q	0~10	maximum VSI value for each sensor type q	v_{max}^q	20
unit energy	P^c, P^s, P^f	20	event detection threshold	t_{event}	2
VSI update threshold	t_{osv}	5	population size (GA)	M	10~30
not sensed time threshold	t_{SIG}	5	crossover probability(GA)	P_c	0.3~0.9
SIG cell selection probability	P_{sig}	0.02~0.1	number of ant agents (ACO)	m	10~35
mutation probability (GA)	P_m^1, P_m^2	0.08	evaporation rate (ACO)	ρ	0.05
heuristic factor (ACO)	α, β	1	EWMA weight	γ	0.5
constant (ACO)	C	70	VSI event update control	η	1
VSI update control	δ_1, δ_2	1.5			

**FIGURE 12.** The optimum utility values for different GA population sizes and generations.

position like Fig. 8 (a). We put the first part of the first SIG cell sequence into the left part of the first child sequence, and fill in the right part of the first child sequence from the cut point of the second SIG cell sequence without duplication. Since the number of SIG cells of second part of the second SIG cell sequence is three in this example, the SIG cells that are selected from the second SIG cell sequence with circular selection is also three. Similarly, a second child sequence starts from the first part of the second SIG cell sequence by filling in with the first SIG cell sequence in the same way as shown in Fig. 8 (b). On the mutational part, one example is also given to explain our strategy on Fig. 9. With probability P_m^1 , among the remained flying cells that are not included in the SIG cell sequence, we change n_1 number of

**FIGURE 13.** The optimum utility values for different crossover and mutation probabilities.

SIG cells indication state to flying cells, then take it out of the SIG cell sequence. Meanwhile, with the probability P_m^2 , we change n_2 number of flying cells' indication state to SIG cells, then locate them to the SIG cell sequence randomly. After crossover and mutation is complete, it will go to the next iteration of GA.

E. OVERALL OPTIMIZATION ALGORITHM

In this paper, we use bio-inspired algorithms to optimize the flight path of a UAV. Following is the overall algorithm pseudocode. First, an initial M SIG cell sequences are selected randomly without prohibited areas and duplicates. For each SIG cell sequence's neighbor SIG cell pair, using

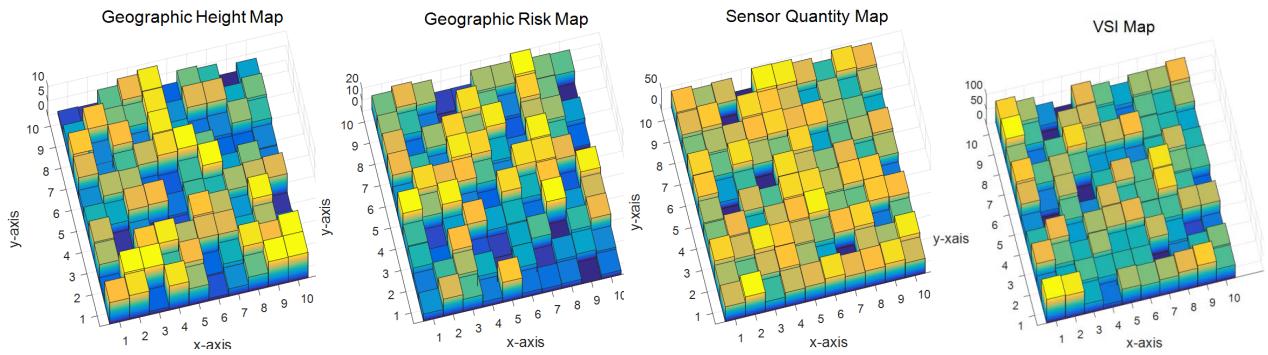


FIGURE 14. Simulation environmental information maps.

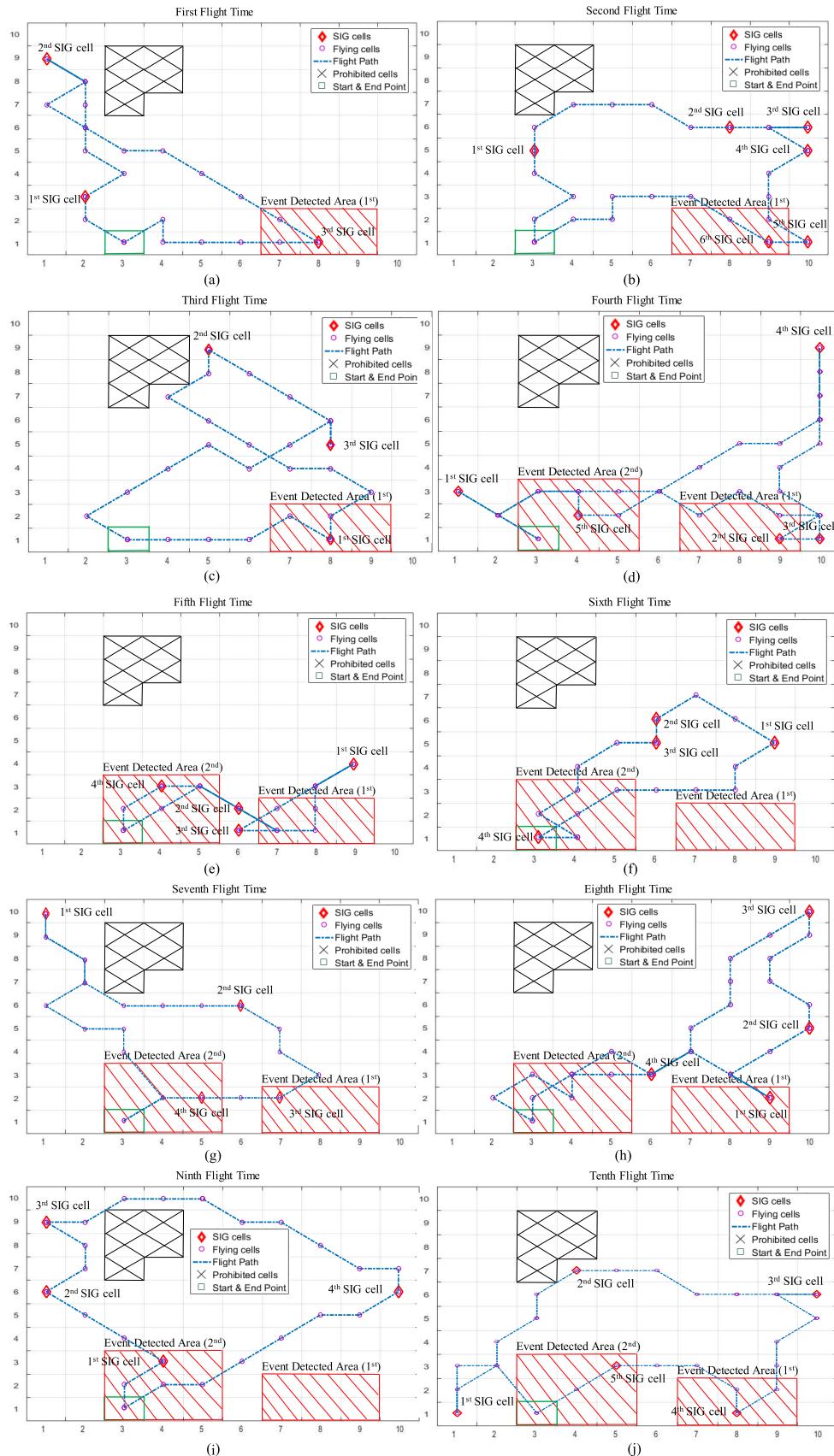
Algorithm 1 Pseudocode of the Proposed UAV Path Optimization Algorithm

```

1: /* initialization*/
2: Set population size =  $M$ ;
3: Generate initial  $M$  SIG cell sequences randomly;
4: Set  $iteration = 0$ ; /* iteration number */
5: while ( $iteration \leq$  predetermined maximum iteration number)
6: {
7:     for  $m$  SIG cell sequence,  $m = 1$  to  $M$ 
8:     {
9:         for neighboring SIG pair,  $n = 1$  to (SIG cell quantity+1)
10:        {
11:            find the optimal flying cell sequence between neighbor SIG cell pair using ACO;
12:        }
13:    }
14:    For each  $m$ , generate a complete chromosome  $i$  by combining SIG cell sequence and flying cell sequences;
15:    Evaluate utility of every complete chromosome;
16: /* using the utility function which derived equations (4)~(19)*/
17:  $opt\_chromosome(iteration) = (utility\_chromosome(i))$ 
18: /* in the current  $iteration$ , pick out the path sequences corresponding maximum utility */
19: if ( $utility\_chromosome(i) > opt\_utility$ )
20: {
21:      $opt\_utility = utility\_sequence(i);$ 
22:      $opt\_path = opt\_chromosome(iteration);$ 
23:     keep  $q$  chromosomes based on utility value order;
24: }
25: Generate new  $(M - q)$  SIG cell sequences by following GA operators;
26: for SIG cell sequence,  $m = 1$  to  $(M - q)$ 
27: {
28:     Perform crossover with probability  $P_c$ ;
29:     Perform mutation with probability  $P_m^1$  and  $P_m^2$ ;
30: }
31:  $iterations++;$ 
32: }
```

ACO we find flying cell sequences respectively. Then, the combined complete SIG cell and flying cell sequences are made and they are called chromosomes, and every chromosome is evaluated by the proposed utility functions. We target the chromosome with the maximum utility value as the optimal path on the current iteration. Next, we select

and keep certain number of chromosomes based on utility value order. Pick out SIG cell sequences and flying cell sequences from the remaining unselected chromosomes, and apply the proposed crossover and mutation schemes to generate new SIG cell sequences within the same population. After that, go back to the step for ACO to fill up flying

**FIGURE 15.** Ten times flight records.

cell sequences until the predetermined iteration is satisfied. In the end, the chromosome with maximal utility value is output, namely the optimal UAV flight path. The proposed path planning procedure is performed offline before UAV actually leaves the start point and VSI updates are based on previous UAV flight results. Generally bio-inspired algorithms (GA and ACO in this paper) require many iterations so that computation complexity and the required time to converge to the optimum value may not be ignorable. Since the proposed method proceeds offline, the difficulty of real-time calculation is not considered to be a serious problem.

VI. SIMULATION RESULTS

In this section, we evaluate the performance of the proposed UAV path planning optimization algorithm. First, we evaluate UAV path optimization solution and then we will prove the rationality of the proposed event detection based VSI update and path finding. In Table 1, the main parameters used for this simulation study are given. We define a $1\text{km} \times 1\text{km}$ square area that consists of 100 unit cells. In the initial generation, we randomly choose SIG cell positions, excluding prohibited area, and use the equations we derived from ACO to fill in among SIG cells. In each generation, the fitness of every individual chromosome in the population is evaluated based on the utility functions. If any chromosomes take energy and time more than the constraint thresholds, they are eliminated.

In Fig. 10, we show the minimum cost for flying sequence optimization using ACO by increasing iteration number and the number of ants. As we expected, as increasing the number of iterations and the number of ants, the obtained cost for the best flying cell sequence is reduced and converges to the optimum value faster. Fig.11 shows the energy consumption and time expenditure values of each iteration for ACO-based flying cell sequence optimization, in which the number of the ant agents is 35. We can see that under the influence of ant pheromones, the cost values of flight energy and UAV flight time converge gradually as increasing iteration number.

Next, by completing SIG cell and flying cell UAV path chromosome, using GA we evaluate the utility value obtained. In most cases, GA converges to the stable value by 10~20 generations. Fig.12 shows the how the population size affects the best utility of the proposed utility functions. Fig.13 shows the best utility values by adjusting the crossover and mutation probabilities. In our simulation, the best crossover and mutation probabilities are 0.6 and 0.08, respectively. As increasing the population size, we can obtain higher the best utility value. The proposed mechanism is not quite sensitive to the crossover and mutation probabilities as shown in Fig. 13. The best performance was obtained at 0.5 and 0.08 for crossover probability and mutation probability, respectively.

To verify the effectiveness of the proposed dynamic VSI update mechanism by detecting events in the sensor field, simulation environmental information maps were generated as in Fig. 14. As explained in Fig. 4, UAV operator has multi-layer information maps such as geographical location map,

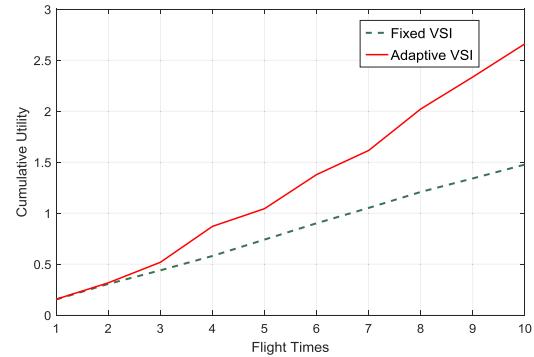


FIGURE 16. CUMULATIVE UTILITY COMPARISON ON FIXED VSI AND ADAPTIVE VSI.

UAV operational risk map, sensor deployment map and initial VSI map. In Fig. 14, z-axis represents the value of each map at every unit cell area. We randomly determine the geographic height and risk. A number of sensors with different sensor types are also randomly scattered in the entire sensor field. The obtainable VSI at each cell was computed based on the number of sensors installed and initial VSI value of each sensor. The VSI map is updated after every UAV flight time based on the observed sensing information changes.

In Fig.15, we show the UAV flight path changes in accordance with the event detections for 10 UAV flight trials. We choose Function 3 in Fig.6, the logarithmic relationship to update VSI in our simulation. We assume the third SIG cell is the event point, and red lines mark the event area. In the first flight time, three SIG cells are stochastically selected as shown in Fig.15 (a). The VSI of this event area is updated by the proposed rule. Fig.15 (b) represents the second flight time, and there are 5 SIG cells. After flight path optimization, the best utility route shows that UAV revisits the event detection area. Fig.15 (c) shows the third flight time route including three SIG cells, in which the first SIG cell passes through and senses the previous event detection area. In the fourth flight time, we triggered one more event at the 5th SIG cell point, VSI of the new event area is updated by our method and continues to the next flight time, which can be noticed on Fig.15 (d). At the fifth flight time UAV visits the second event area, but just passes through the first event area displayed on Fig.15 (e). From sixth to tenth flight time, UAV visits at least one of event areas so that we can gather more information from the event detected areas as shown in Fig.15 (f) ~ (j).

In Fig. 16 we compared the accumulated utility values of the selected paths with the fixed VSI case to verify the effectiveness of the dynamic VSI method, when UAV flight times increase. The cumulative utility value of the adaptive VSI case grows more rapidly, which indicates that the proposed dynamic VSI update method helps to find more valuable sensing points in accordance with dynamic environmental condition changes.

In Fig. 17, to compare with conventional bio-inspired architecture, we evaluated the utility values of the optimum path with A* algorithm [21] and A*+GA algorithm, in which the both compared methods also use the proposed utility

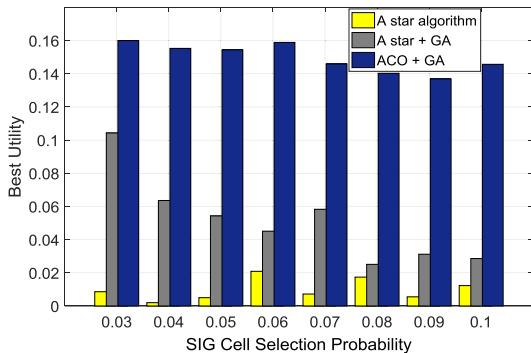


FIGURE 17. The optimal utility comparison on proposed method and other combinations.

function. For ‘A*+GA’, the proposed crossover and mutation method is applied. For all SIG cell selection probability conditions, the proposed method (‘ACO+GA’) shows the highest best utility values.

VII. CONCLUSION

In this paper, we propose an algorithm for UAV data acquisition in wide IoT sensor networks. During UAV operation, there are some constraints in terms of operation time, total energy consumption and prohibited flight areas so that we generate multi-objective utility functions for deriving the optimal UAV flight path and the method used to obtain the sensing information from a dedicated set of sensors. To maximize the total utility value, in this paper we apply bio-inspired algorithms to derive optimal SIG cell and flying cell sequences. Using the proposed joint GA and ACO from possible UAV flight paths, an optimal one is selected in accordance with sensing, energy, time and risk utilities. In this paper, we also presented the dynamic VSI value update mechanism to effectively consider new events in the sensor field. The simulation results show that our method can obtain dynamic environmental adaptivity and high utility in various practical situations.

REFERENCES

- [1] T. Brown, B. Argrow, C. Dixon, S. Doshi, R. Thekkunnel, and D. Henkel, “Ad hoc UAV-ground network (AUGNet),” in *Proc. AIAA 3rd Unmanned Unlimited Tech. Conf.*, 2004, pp. 1–11.
- [2] D.-T. Ho, E. I. Grøtli, S. Shimamoto, and T. A. Johansen, “Optimal relay path selection and cooperative communication protocol for a swarm of UAVs,” in *Proc. 3rd Int. Globecom Workshop*, Dec. 2012, pp. 1585–1590.
- [3] P. B. Sujit, D. E. Lucani, and J. B. Sousa, “Bridging cooperative sensing and route planning of autonomous vehicles,” *IEEE J. Sel. Areas Commun.*, vol. 30, no. 5, pp. 912–922, Jun. 2012.
- [4] J. T. Isaacs, S. Venkateswaran, J. Hespanha, U. Madhow, J. Burman, and T. Pham, “Multiple event localization in a sparse acoustic sensor network using UAVs as data mules,” in *Proc. Int. Globecom Workshop*, Dec. 2012, pp. 1562–1567.
- [5] Q. Zhao and L. Tong, “Quality-of-service specific information retrieval for densely deployed sensor networks,” in *Proc. IEEE Military Commun. Conf.*, Oct. 2003, pp. 591–596.
- [6] D. T. Ho and S. Shimamoto, “Highly reliable communication protocol for WSN-UAV system employing TDMA and PFS scheme,” in *Proc. Int. GLOBECOM Workshop*, Dec. 2011, pp. 1320–1324.
- [7] G. Gankhuyag, A. P. Shrestha, and S.-J. Yoo, “Robust and reliable predictive routing strategy for flying ad-hoc networks,” *IEEE Access*, vol. 5, pp. 643–654, 2017.
- [8] E. W. Dijkstra, “A note on two problems in connexion with graphs,” *Numer. Math.*, vol. 1, no. 1, pp. 269–271, 1959.
- [9] S.-J. Yoo, J.-H. Park, S.-H. Kim, and A. Shrestha, “Flying path optimization in UAV-assisted IoT sensor networks,” *ICTExp.*, vol. 2, no. 3, pp. 140–144, Sep. 2016.
- [10] S. Kanchanavally, R. Ordóñez, and C. J. Schumacher, “Path planning in three dimensional environment using feedback linearization,” in *Proc. Amer. Control Conf.*, Jun. 2006, pp. 3545–3550.
- [11] I. Hasircioglu, H. R. Topcuoglu, and M. Ermis, “3-D path planning for the navigation of unmanned aerial vehicles by using evolutionary algorithms,” in *Proc. Conf. Genetic Evol. Comput.*, 2008, pp. 1499–1506.
- [12] I. K. Nikolos, K. P. Valavanis, N. C. Tsourveloudis, and A. N. Kostaras, “Evolutionary algorithm based offline/online path planner for UAV navigation,” *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 33, no. 6, pp. 898–912, Dec. 2003.
- [13] P. Doherty *et al.*, “A distributed architecture for autonomous unmanned aerial vehicle experimentation,” in *Proc. 7th Int. Symp. Distrib. Auto. Robot. Syst.*, 2004, pp. 221–230.
- [14] K. Tulum, U. Durak, and S. K. Yder, “Situation aware UAV mission route planning,” in *Proc. IEEE Aerosp. Conf.*, Mar. 2009, pp. 1–12.
- [15] E. Yannaz, R. Kuschning, M. Quaritsch, C. Bettstetter, and B. Rinner, “On path planning strategies for networked unmanned aerial vehicles,” in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Apr. 2011, pp. 212–216.
- [16] N. H. Motlagh, T. Taleb, and O. Arouk, “Low-altitude unmanned aerial vehicles-based Internet of Things services: Comprehensive survey and future perspectives,” *IEEE Internet Things J.*, vol. 3, no. 6, pp. 899–922, Dec. 2016.
- [17] L. Tong, Q. Zhao, and S. Adireddy, “Sensor networks with mobile agents,” in *Proc. IEEE Military Commun. Conf. (MILCOM)*, Oct. 2003, pp. 688–693.
- [18] D.-T. Ho, E. I. Grøtli, P. B. Sujit, T. A. Johansen, and J. B. Sousa, “Optimization of wireless sensor network and UAV data acquisition,” *J. Intell. Robot. Syst.*, vol. 78, no. 1, pp. 159–179, 2015.
- [19] J. Huang, L. Xu, C.-C. Xing, and Q. Duan, “A novel bioinspired multiobjective optimization algorithm for designing wireless sensor networks in the Internet of Things,” *J. Sensors*, vol. 2015, Dec. 2014, Art. no. 192194.
- [20] M. Z. Hasan and F. Al-Turjman, “SWARM-based data delivery in social Internet of Things,” *Future Generat. Comput. Syst.*, to be published.
- [21] I.-C. Chang, H.-T. Tai, F.-H. Yeh, D.-L. Hsieh, and S.-H. Chang, “A VANET-based a* route planning algorithm for travelling time- and energy-efficient GPS navigation app,” *Int. J. Distrib. Sensor Netw.*, vol. 2013, Jun. 2013, Art. no. 794521.



QIN YANG received the B.E. degree in communication and information engineering from Chongqing University of Posts and Telecommunications in 2016. She is currently pursuing the M.S. degree in information and communication engineering with the Multimedia Network Laboratory, Inha University, South Korea. Her research interests include wireless sensor network, FANET, and machine learning.



SANG-JO YOO received the B.S. degree in electronic communication engineering from Hanyang University, Seoul, South Korea, in 1988 and the M.S. and Ph.D. degrees in electrical engineering from Korea Advanced Institute of Science and Technology, in 1990 and 2000, respectively. From 1990 to 2001, he was a Member of Technical Staff with the Korea Telecom Research and Development Group, where he was involved in communication protocol conformance testing and network design fields. From 1994 to 1995 and from 2007 to 2008, he was a Guest Researcher with the National Institute Standards and Technology, USA. Since 2001, he has been with Inha University, where he is currently a Professor with the Department of Information and Communication Engineering. His current research interests include cognitive radio network protocols, adhoc wireless network, MAC and routing protocol design, wireless network QoS, and wireless sensor networks.