IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 18 December 2023, accepted 8 January 2024, date of publication 11 January 2024, date of current version 22 January 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3353193

RESEARCH ARTICLE

Energy Optimization and Trajectory Planning for Constrained Multi-UAV Data Collection in WSNs

AMIRA A. AMER[1], REEM AHMED[2], IRENE S. FAHIM[3], (Senior Member, IEEE), AND TAWFIK ISMAIL[1,4,5], (Senior Member, IEEE)

[1]Wireless Intelligent Networks Center (WINC), Nile University, Giza 12677, Egypt
[2]Department of Mathematics, Faculty of Science, Cairo University, Giza 12613, Egypt
[3]Smart Engineering System Research Center (SESC), Nile University, Giza 12677, Egypt
[4]National Institute of Laser Enhanced Sciences, Cairo University, Giza 12613, Egypt
[5]College of Engineering, Taibah University, Madinah 42353, Saudi Arabia

Corresponding author: Tawfik Ismail (tismail@cu.edu.eg)

This work was supported by the Science, Technology & Innovation Funding Authority (STIFA), Egypt, under Grant 43204.

ABSTRACT Wireless sensor networks (WSNs) deployed in remote areas face a challenge in uploading the collected data to data centers due to limited network coverage. Unmanned aerial vehicles (UAVs) can extend network coverage to remote WSNs by flying and communicating with WSN aggregator nodes to collect data. UAV-assisted data collection systems need to be carefully developed to collect all data efficiently while considering the UAV and WSN constraints. This paper provides an energy-efficient multi-UAV data collection framework for WSNs. We formulate the data collection system as a problem that jointly optimizes the system cost and energy consumption constrained by the communication power, UAV mission time, and memory size. The problem is resolved over two steps: First, the location and number of aggregators needed are determined using a triangulation-based K-means clustering that minimizes the number of aggregators used and the system cost. Second, the dockstation position that minimizes the energy consumption is obtained using the gaining-sharing knowledge (GSK) optimization algorithm. The optimum UAV trajectory for each GSK candidate solution is designed by solving a capacitated vehicle routing problem (CVRP) that combines heuristic and metaheuristic solving techniques. Simulations show that our framework outperforms other recent techniques by minimizing the overall system cost and energy consumption.

INDEX TERMS Unmanned aerial vehicles, wireless sensor networks, clustering, trajectory planning, energy optimization, data collection.

I. INTRODUCTION

Wireless sensor networks (WSNs) are a trending research area due to their usability in a wide range of applications, including industrial, telemedicine, traffic monitoring, disaster recovery, and agriculture [1], [2]. One challenge WSNs face is transferring the detected data to remote processing centers in an energy-efficient manner to extend the lifetime of the battery-powered sensors. Clustering is one of the most widely used WSN management techniques [3]. In clustering, sensors are separated into groups orchestrated by a cluster head (CH), also known as an aggregator. During their operation, the CHs collect data from their associated sensors and transmit that data to the surrounding station, referred to as cloud stations. In WSNs implemented in rural areas, aggregators may have difficulty in transmitting data since direct communication between CHs and the cloud station might be infeasible [4], [5].

Recently, unmanned aerial vehicles (UAVs) have attracted the widespread attention of researchers in wireless communications [6], [7]. In modern networks, the UAVs have been suggested as a solution for extending coverage to remote and rural areas either by serving as mobile base stations (BSs) or by collecting data to a remote cloud [8]. This paper focuses on WSNs that do not require real-time responses and can tolerate data uploading to processing centers (remote-cloud)

The associate editor coordinating the review of this manuscript and approving it for publication was Petros Nicopolitidis.

after UAVs reach their dockstation (charging station) by the end of a data collection mission. The UAVs can fly and hover over each aggregator to collect the data through the use of high data rates wireless transmission such as millimeter wave (mmWave) or free-space optical links [5], [9]. However, the UAVs are constrained by various parameters such as battery capacity, maximum flying speed, mission time, and memory capacity. Several studies have investigated the optimization of UAV trajectories and data collection schedules under a variety of limitations in order to minimize the amount of overall consumed energy. In this paper, additional optimization factors, such as the transmission power of aggregators and sensors, the flying speed of UAVs, and the placement of aggregators and dockstation, were taken into account in order to minimize the overall energy consumption of the data collection process.

The integration of WSNs and UAV networks presents a dynamic landscape of challenges and opportunities. The key challenges include ensuring reliable communication between sensors and UAVs in diverse and sometimes harsh environments, managing the energy constraints of both systems, and addressing security concerns inherent in wireless communications. However, the potential applications are vast and transformative. In agriculture, these integrated systems offer precision monitoring of soil conditions and crop health, optimizing resource usage. Environmental monitoring benefits from UAVs swiftly covering large areas to gather real-time data on ecosystems, wildlife, and pollution levels. For search and rescue operations, UAVs equipped with sensors enable rapid and effective coverage of expansive terrains, aiding in locating survivors. In disaster response, the seamless coordination between WSNs and UAVs provides crucial information on damage assessment, environmental conditions, and survivor locations, enhancing overall response efforts. The integration of WSNs and UAV networks thus holds promise across diverse domains, offering innovative solutions to longstanding challenges.

Energy-saving techniques for integrated WSNs and UAV networks are crucial for extending operational time and maintaining system functionality [10], [11]. One such technique is sleep scheduling, where nodes selectively deactivate during periods of inactivity, significantly reducing power consumption. Implementing this in UAV networks can be challenging due to the dynamic nature of UAVs and the need for immediate responsiveness. Algorithms must be designed to predict periods of low activity without compromising the network's ability to react to sudden changes or requests for data. Another technique is energy harvesting, which equips UAVs and sensors with capabilities to extract energy from environmental sources such as solar, wind, or vibrations. This can offset energy usage but may add weight and complexity to the UAVs [12]. Additionally, energy-efficient communication protocols can minimize the energy spent on data transmission by optimizing the routing of messages and reducing the number of transmissions. However, these protocols must balance energy savings with the potential increase in latency and the need for reliable data delivery [13]. Adaptive duty cycling, where the network's activity levels are adjusted based on current energy reserves and operational demands, can also help conserve energy but requires sophisticated management to avoid degrading network performance. Overall, the implementation of these techniques involves trade-offs between energy efficiency, reliability, and latency, necessitating a holistic approach to network design and management.

This research presents an innovative WSNs with a data collection framework, which aims to minimize the overall energy required by the system while considering all of the constraints for data collecting UAVs that have been previously discussed. To the best of our knowledge, this is the first study investigating multi-dimensional optimization, including dockstation placement, sensors power, and UAV trajectories. The main contributions of the paper are summarized as follows:

- We investigate large-scale heterogeneous WSNs, in which sensors transmit their data to a nearby aggregator, and multiple UAVs are employed to collect data from all aggregators at the same time.
- We formulate a problem to minimize the overall consumed energy and cost through optimizing the placement of aggregators, the position of the dockstation, the transmission power needed by aggregators and sensors, and the UAV trajectories.
- A newly proposed triangulation-based clustering technique is presented to minimize the number and placement of aggregators, as well as the associated sensors.
- The gaining-sharing knowledge (GSK) metaheuristic and capacitated vehicle routing problem (CVRP) algorithms are employed in order to determine the optimal dockstation placement, transmission power, and UAV trajectory.
- The optimization problems have been solved subject to mission time, maximum flying speed, UAV memory size, and sensors power constraints.
- The system was tested under different area sizes and different sensor densities to show the scalability of the proposed framework in minimizing the system energy.

The rest of the paper is organized as follows. Section II discusses the previous works on UAV mission planning and UAV data collection systems. Section III describes the system architecture used and formulates the data collection optimization problem. Section IV explains the proposed approach to solve the data collection problem. In Section V, the results of our work are displayed and compared to other data collection optimization approaches. Section VI concludes our work and discusses future work.

TABLE 1. UAV data collection literature review.

Reference #	Multi-UAV	Time Constrained	Memory Constrained	CH Placement	Dockstation Placement	A2G Power Optimization	Large WSNs
[14]	No	No	No	No	No	No	No
[15]	No	No	No	No	No	No	No
[2]	No	No	No	Yes	No	No	No
[16]	No	No	No	Yes	No	Yes	No
[17]	Yes	Yes	No	Yes	No	No	No
[18]	Yes	Yes	Yes	No	No	No	No
[19]	Yes	Yes	No	Yes	No	No	Yes
[20]	Yes	No	No	No	No	No	No
[21]	Yes	Yes	No	No	No	Yes	No
This work	Yes	Yes	Yes	Yes	Yes	Yes	Yes

II. LITERATURE SURVEY

This section discusses approaches for optimizing UAV mission planning that has been published in the literature. The majority of mission planning studies fall within the fly and communicate or fly-hover-communicate categories. The fly and communicate requires the UAV to communicate with users while it is in motion, and the challenge is to determine the optimal time slots to schedule the communication with each user. The challenge with the fly-hover communication strategy is determining the optimal hovering positions to service all users/sensors. Fly-hover-communicate entails UAVs flying to a specific area, hovering there, and communicating with users nearby to collect data. All techniques based on fly-hover-communicate employ multirotor UAVs capable of hovering, whereas continuous motion techniques may employ multirotor or fixed-wing UAVs. The following subsections include in-depth descriptions of the two techniques as well as a review of the relevant literature. Table 1 summarizes the UAV data collection literature review.

A. FLY AND COMMUNICATE TECHNIQUES

Qin et al. [22] investigated a system consisting of multiple fixed-wing UAVs that provide mobile edge computing services to ground users under a predefined mission schedule. Their research aims to minimize the energy consumption of both UAVs and users by optimizing UAV trajectory, communication bandwidth, CPU frequency, UAV-user association, data uploading schedule, and user transmission power. The optimization problem has been subdivided into two subproblems, each solved using a convex solver. Both problems are iteratively solved until convergence occurs. Sun et al. [23] developed a UAV energy optimization approach for a system similar to that described in [22], but with an additional deadline constraint on tasks. To solve the optimization problem, the researchers introduced slack variables to the non-convex optimization problem and solved the problem using a successive convex approximation (SCA)-based algorithm. The problem formulated in [22] and [23] both have the mission planning tightly coupled with offloading decisions and CPU frequency, resulting in solutions that are inapplicable in data collection scenarios where offloading decisions and the CPU frequency of devices are irrelevant.

Li et al. [14] introduced an approach to minimize the mission time for a single multirotor UAV data collection system. The time is minimized by optimizing the UAV trajectory, the flying height, the UAV speed, and the transmission schedule. The solution is obtained by solving three problems in this order: determining the flying height, optimizing the trajectory, and optimizing the velocity and transmission scheduling jointly. Numerical techniques determine the optimal flying height by solving a signal-to-noise ratio (SNR) minimization problem. The trajectory optimization is accomplished based on the following steps. The order of data collection is determined by solving a trajectory length minimization problem using the traveling salesman problem (TSP) algorithm. The actual path of the UAV is expressed as an ordered set of waypoints, where the UAV moves from one waypoint to the next in a straight line. These waypoints are determined by using convex optimization techniques to solve a convex problem involving minimizing the length of the UAV trajectory. The UAV collects data from all users within its range at each trajectory segment. The flying velocity and the transmission schedule at each segment are obtained by solving mixed-integer nonlinear programming problems using block coordinate descent. The method introduced reduced the mission time significantly. However, energy and memory constraints were not considered. Although continuous motion-based data collection achieves a higher energy-efficient, fly-hover-communicate techniques are more practical [24] and have a lower complexity [25]. As a result, a fly-hover-communicate approach was adopted in this paper.

B. FLY-HOVER-COMMUNICATE TECHNIQUES

Wu et al. [15] suggested a trajectory planning method that aims to optimize both the energy consumption and mission time for a single UAV data collection system. The trajectory is represented as a set of ordered hovering points. They use an SCA-based algorithm to get the optimum hovering points. The authors of [2] presented a data collection framework to minimize the WSN energy and UAV travel time through optimizing the position of aggregators, hovering points, and

UAV trajectory. Aggregators are chosen using a center-biased hybrid energy-efficient distributed (CBHEED) algorithm, which is a modification to the HEED clustering algorithm. Hovering points are defined as the points with the highest received signal strength indicator (RSSI) for each aggregator and are found using tabu search. The optimum trajectory that minimizes the travel time and passes through all hovering points is obtained by solving a TSP problem using a genetic algorithm (GA). Wang et al. [16] considered a trajectory and sensor uploading power optimization problem to minimize the energy consumption of a WSN system with a single data collecting UAV. The problem was broken down into two sub-problems: a joint optimization of sensor power and hovering points problem and a data collection order problem. Given a serving order, the joint optimization of sensor power and hovering point problem was solved using a proposed algorithm based on SCA. Given the sensor power and hovering points, the data collection order was reduced to a TSP and solved using one of the TSP solvers. The optimum values for sensor power, hovering points, and collection order is obtained iteratively. All the works discussed consider only a system with a single UAV, which can only cover a small area before needing to recharge. The UAV needs to do multiple recharge stops to cover a large area, which is inefficient. Therefore, a multi-UAV system is considered in this paper.

Ghdiri et al. [17] proposed a WSN data collection framework with multiple UAVs. Their framework aimed to minimize the WSN deployment cost and the energy consumed by UAVs through optimizing the position of data aggregators and the UAV trajectory. The aggregator positions were determined using a constrained version of the K-means clustering algorithm. Given the aggregator positions, the UAV trajectory problem was modeled as a capacitated vehicle routing problem with time windows and solved using metaheuristics. The UAV trajectory was constrained by the battery capacity of the UAV and the time frame of each aggregator. The work in [17] ignores memory constraints and does not consider optimizing the energy consumption of aggregators as opposed to our work. Shen et al. [18] discussed an algorithm to minimize the UAV deployment and operation costs in a multi-UAV scenario. UAV cost minimization was achieved through optimizing the hovering points and UAV trajectory. The problem was constrained by the battery capacity and memory capacity of UAVs. The problem was broken into two sub-problems solved iteratively until convergence was reached. The first sub-problem is finding a trajectory given the hovering points. The trajectory optimization was modeled as a capacitated vehicle routing problem with time windows and solved using an ant colony-based approach. The simplicity of modeling multi-UAV trajectory planning using capacitated vehicle routing problems inspired us to adopt a similar approach for trajectory planning. The second sub-problem was finding hovering points given the UAV trajectory, which was solved using an SCA-based algorithm. This work only focuses on the costs from a UAV perspective but does not consider how to lower the energy consumption of IoT devices or minimize the cost of aggregators as opposed to our work.

Nguyen et al. [19] introduced a heuristic algorithm for jointly clustering and planning data collection missions in large WSNs to minimize the energy consumption and extend the WSN lifetime. Their heuristic iteratively clusters and plans the data collection mission until the UAV time constraints are not violated. Their clustering algorithm is based on constructing η-balanced Voronoi sets. For a given set of cluster heads, the trajectory of the UAVs is determined by solving a multiple TSP problem. Although their clustering technique gives solid results for energy saving for WSNs, they do not explore energy saving for cluster heads and UAVs. Pan et al. [20] suggested a deep learning-based method for multi-UAV trajectory planning to minimize the number of UAVs used and the trajectory length of UAVs. The path planning problem is modeled as a modification of TSP. The TSP is solved using GA with the UAV energy consumption as the fitness function. The output of the GA for different scenarios was used to train a convolution neural network, which is used later to plan the path of UAVs according to the requirements of sensor nodes. The downside of this work was that it ignored the memory and energy constraints of UAVs by modeling path planning as a TSP problem. Khodaparast et al. [21] studied collaborative energy-constrained multi-UAVs data gathering systems. They used a deep reinforcement learning (DRL) approach to determine the trajectory of UAVs and the communication power of sensors that minimizes the energy consumption of both UAVs and sensors. The problem was modeled as three Markov decision processes (MDP). The UAV trajectory and sensor power problems were each solved using a deep deterministic policy gradient. The scheduling of sensors to UAVs was solved using multi-agent deep Q-learning. However, their work did not consider memory constraints which are considered by our work. All the mentioned works also did not consider optimizing the dockstation position as opposed to our work.

III. SYSTEM MODEL AND PROBLEM FORMULATION

This section describes the two subsystems that make up the overall system under study. The first subsystem is the sensor network, which consists of heterogeneous sensing devices and their aggregators. The second subsystem is the UAV-Aggregator network, which has UAVs communicating with the aggregators to collect data. Fig. 1 shows an overview of the system and the interactions within each subsystem. We also describe the communication model, the delay model, and the energy model adopted. The notations that are used in this section are listed in Table 2.

A. SENSOR NETWORK MODEL

We consider a WSN having N sensors randomly placed over a given area of size G. The position of the sensors is given by a vector $m = [m_1, m_2, \ldots, m_N]$, where $m_i = \left[x_{m_i}, y_{m_i}, 0\right]$ is the 3D Cartesian coordinate of sensor i. Sensors are assumed to be heterogeneous, and therefore, the size of the data

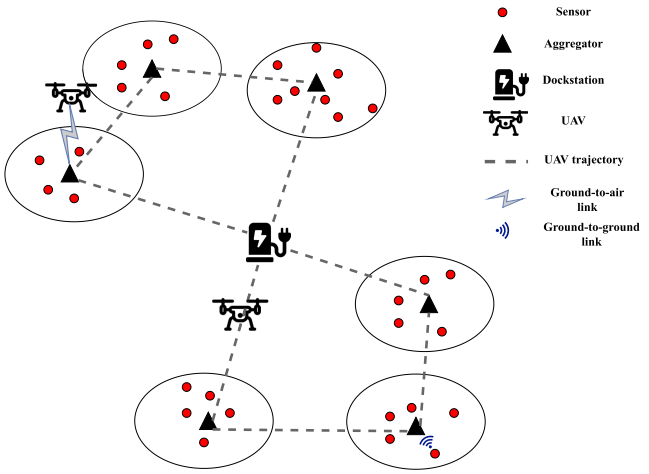


FIGURE 1. System overview.

sensed differs from one sensor to another. Due to the power constraints on sensors, they send their measured data at a particular time to a nearby aggregator. The symbol S_i denotes the amount of data generated by sensor i. The sensor i delivers data to its associated aggregator using the transmission power of P_i. The aggregator gathers and combines data from various sensors before sending it to a UAV that collects data. Given that different sensors generate data of varying sizes, it is reasonable to expect that sensors that generate large amounts of data will demand more power. The relationship between sensor power and data size could be represented as $P_i = P_{\text{kbit}} \times S_i$ where P_{kbit} is the required power to transmit 1 kbit of sensor data. Assuming that the sensors communicate with the aggregator using orthogonal channels, the maximum communication distance between a sensor i and its associated aggregator can be written as [17]:

$$d_{\text{th}}^{i} = \left(\frac{P_i}{\sigma_i^2 \times \gamma_{\text{th}}} \right)^{\frac{1}{\alpha}}, \tag{1}$$

where σ_i^2 is the noise power of the channel between the sensor and aggregator, γ_{th} is the minimum signal-to-noise ratio needed to send the data successfully, and α is the path loss exponent. Assuming the noise power is the same for all ground-to-ground links, denoted by σ_{G2G}^2, we can reformulate equation (1) to get the maximum communication threshold distance for all sensors as follows:

$$d_{\text{th}} = \left(\frac{P_{\text{kbit}}}{\sigma_{\text{G2G}}^2 \times \gamma_{\text{th}}} \right)^{\frac{1}{\alpha}}. \tag{2}$$

B. UAV-AGGREGATOR NETWORK MODEL

In order to collect the data from all aggregators, a set of K identical UAVs will fly over the area and hover over each aggregator for a specific period of time to collect the data. UAVs are assumed to start the mission from the same dockstation and to have identical properties following the assumptions stated in [17]. The maximum mission time and the maximum memory capacity of the UAVs are denoted by $T_{\max}$ and $C_{\max}$ respectively. The position of aggregators is represented as a vector $q = [q_1, q_2, \ldots, q_{N_{\text{ag}}}]$, where $q_j = [x_{q_j}, y_{q_j}, 0]$ is the 3D Cartesian coordinate of aggregator j. The position of an UAV k at time t that flies at an altitude H can be expressed as $u_k[t] = [x_{u_k}[t], y_{u_k}[t], H]$.

TABLE 2. Notations used.

Notation	Definition
G	Size of the area covered
N	Number of sensors in system
S_i	Size of data generated by sensor i
P_{kbit}	Transmission power per 1 kbit of sensor data
α	Path loss exponent
d_{th}^{i}	Communication threshold distance between sensor i and its aggregator
γ_{th}	Minimum signal-to-noise ratio (SNR) needed to send sensor data successfully
K	Number of UAVs used in the system
$T_{\max}$	Maximum mission time for a UAV
$C_{\max}$	Memory capacity of a UAV
q_j	Position of aggregator j
H	Flying altitude of UAVs
$u_k[t]$	Position of UAV k at time t
$l_{kj}[t]$	Path loss of channel between UAV k and aggregator j at time t
$d_{kj}[t]$	Horizontal distance between UAV k and aggregator j at time t
η_{LoS}	Line of sight (LoS) connection loss
η_{NLoS}	Non-line of sight (NLoS) connection loss
$p_{\text{LoS}}^{kj}[t]$	LoS connectivity probability between UAV k and aggregator j at time t
f_c	Carrier frequency
c	Speed of light
$\theta_{kj}[t]$	Elevation angle between UAV k and aggregator j at time t
$h_{kj}[t]$	Channel power gain between UAV k and aggregator j at time t
R_{kj}	Data transmission rate between UAV k and aggregator j during hovering
B	Total bandwidth of the channel of ground-to-air channel
σ_{UAV}^2	Noise power of ground-to-air channel
L_j	Size of data to be sent by aggregator j
T_{kj}^{hover}	Time spent hovering by UAV k over aggregator j
T_k^{fly}	Time spent flying by UAV k
D_k	Distance of the trajectory of UAV k
v_k	Flying velocity of UAV k
$v_{\max}$	Maximum flying velocity of UAVs
E_{comm}^{j}	Communication energy consumed by aggregator j
P_{ag}	Transmission power of aggregators
E_{comm}	Communication energy consumed by system
$P_{\text{prop}}(v)$	Propulsion power of UAV at velocity v
ξ_I	Induced power
v_0	Mean rotor induced velocity
ξ_B	Blade profile power
Q_{tip}	Tip speed of the rotor blade
d_0	Fuselage drag ratio
r	Rotor solidity
ρ	Air density
A	Rotor disc area
E_{UAV}^{k}	Energy consumed by UAV k
E_{UAV}	Energy consumed by UAVs in the system
q	Set of aggregator positions
N_{ag}	Number of aggregators in the system
m_i	Position of sensor i
X_k	The trajectory of UAV k
X	Set of all UAV trajectories
q_0	Position of dockstation

Each UAV k is assigned to collect data from a given group of aggregators, denoted by the set ϑ_k. The trajectory of the UAV k mission must pass through all aggregators in ϑ_k. Hence, we simplify the definition of the UAV trajectory to be the aggregator positions placed in the order by which they will be visited. The trajectory of UAV k is written as $X_k = \left[q_{(k,1)}, q_{(k,2)}, \ldots, q_{(k,|\vartheta_k|)}\right]$, where $q_{(k,n)} \in \vartheta_k \forall k$, $q_{(k,n)}$ is the position of the nth point that will be visited by UAV k and $|\vartheta_k|$ is the size of set ϑ_k.

1) COMMUNICATION MODEL

The UAV receives data through a ground-to-air channel that considers line-of-sight (LoS) links and non-line-of-sight (NLoS) links. Therefore, the wireless channel path-loss between UAV k and aggregator j at time t is expressed by [22]:

$$l_{kj}[t] = 20\log_{10}\left(\sqrt{H^2 + d_{kj}[t]^2}\right) + (\eta_{\text{LoS}} - \eta_{\text{NLoS}}) \times p^{kj}_{\text{LoS}}[t] + 20\log_{10}\left(\frac{4\pi f_c}{c}\right) + \eta_{\text{NLoS}}, \quad (3)$$

where η_{LoS} is the line-of-sight connection loss, η_{NLoS} is non-line-of-sight connection loss, f_c is the carrier frequency, and c is speed of light. $d_{kj}[t]$ is the horizontal distance between UAV k and aggregator j at time t calculated by $d_{kj}[t] = \sqrt{\left(x_j - x_k[t]\right)^2 + \left(y_j - y_k[t]\right)^2}$. $p^{kj}_{\text{LoS}}[t]$ is the line-of-sight connectivity probability between UAV k and aggregator j at time t is expressed by [22]:

$$p^{kj}_{\text{LoS}}[t] = \left(1 + a\exp\left(-b \times \theta_{kj}[t]\right)\right)^{-1} \quad , \quad (4)$$

where a and b are environment constants and $\theta_{kj}[t] = \arctan\left(H/d_{kj}[t]\right)$ is the elevation angle between UAV k and aggregator j at time t. Given the path loss, the channel power gain between UAV k and an aggregator j at time t is given by [22]:

$$h_{kj}[t] = 10^{-l_{kj}[t]/10} \, . \quad (5)$$

In data collection, UAVs hover over aggregators and therefore the probability of the LoS link is much higher than the NLoS link. Also, the UAV position is constant during data collection, making the channel power gain between the aggregator and the UAV constant and can be denoted by h_{kj}. In this case the data transmission rate from aggregator j to UAV k can be expressed as:

$$R_{kj} = B\log_2\left(1 + \frac{P_{\text{ag}} \times h_{kj}}{\sigma^2_{\text{UAV}}}\right) \quad , \quad (6)$$

where B is the total bandwidth of the channel, P_{ag} is the transmission power of an aggregator and σ^2_{UAV} is the noise power of the ground-to-air channel.

2) DELAY MODEL

The required hovering time for the UAV to collect data from an aggregator is affected by both the data transmission rate and the size of the data to be sent. The size of data to be sent by an aggregator j, denoted by L_j, is estimated as $\sum_{i=1}^{N} S_i$ for all sensors associated with the aggregator j. The hovering time required for UAV k to collect data from aggregator j can be calculated as:

$$T^{\text{hover}}_{kj} = L_j / R_{kj} \quad , \quad (7)$$

Given the trajectory of UAV k, we can calculate the total distance of the UAV trajectory as:

$$D_k = \|q_{(k,1)} - q_0\| + \|q_0 - q_{(k,|\vartheta_k|)}\| + \sum_{n=2}^{|\vartheta_k|-1} \|q_{(k,n)} - q_{(k,n-1)}\| \quad , \quad (8)$$

where q_0 is the position of the dockstation from which all UAVs should start and end their missions. The flight time of a UAV k is given as

$$T^{\text{fly}}_k = \frac{D_k}{v_k} \quad , \quad (9)$$

where v_k is the flying velocity of UAV k throughout the mission. v_k of any UAV is constrained by the maximum flying velocity of UAVs, denoted by $v_{\max}$. The mission time of a UAV k is the sum of the flight time, represented as T^{fly}_k, and total time spent hovering by UAV k, written as T^{hover}_k.

3) ENERGY MODEL

In the system under consideration, the UAV propulsion and ground-to-air communication are the two most energy-consumption components. The communication energy consumed by an aggregator j assigned to UAV k can be calculated by:

$$E^j_{\text{comm}} = P_{\text{ag}} \times T^{\text{hover}}_{kj} \quad , \quad (10)$$

The total communication energy consumed by the system is equal to the sum of communication energy consumed by each aggregator, written as:

$$E_{\text{comm}} = \sum_{j=1}^{N_{\text{ag}}} E^j_{\text{comm}} \quad (11)$$

The energy consumption by the UAVs is attributed to the propulsion power during the hovering and motion. The propulsion power model for a rotatory-wing UAV is defined as [17], [26]:

$$P_{\text{prop}}(v) = \xi_I\left(\sqrt{1 + \frac{v^4}{4v_0^4}} - \frac{v^2}{2v_0^2}\right)^{\frac{1}{2}} + \xi_B\left(1 + \frac{3v^2}{Q^2_{tip}}\right) + \frac{1}{2}d_0 r\rho A v^3 \quad , \quad (12)$$

where ξ_I represents the induced power, v_0 is the mean rotor induced velocity, ξ_B is the blade profile power, Q_{tip} is the

tip speed of the rotor blade, d_0 is the fuselage drag ratio, r is the rotor solidity, ρ is the air density, and A is the rotor disc area. The hovering power of the UAV is equivalent to the propulsion power at a speed of 0, given as $P_{\text{hover}} = P_{\text{prop}}(0)$. The total amount of energy spent by a UAV k is determined as follows:

$$E^k_{\text{UAV}} = P_{\text{prop}}(v_k) \times T^{\text{fly}}_k + P_{\text{hover}} \times T^{\text{hover}}_k \quad , \tag{13}$$

The total energy consumption of UAVs is equal to the sum of the energy spent by each individual UAV, expressed as:

$$E_{\text{UAV}} = \sum_{k=1}^{K} E^k_{\text{UAV}} \tag{14}$$

Therefore, the overall energy consumed by the integrated UAV/aggregator networks is calculated as follows:

$$E_{\text{total}} = E_{\text{UAV}} + E_{\text{comm}} \tag{15}$$

C. PROBLEM FORMULATION

In this paper, our goal is to maximize the cost efficiency of the system by optimizing both the sensor and UAV/aggregator networks. For the sensor network, we aim to minimize the cost of installing aggregators for ground-to-ground data collection. This can be achieved by reducing the number of aggregators used, denoted as N_{ag}. However, we must ensure that the quality-of-service in the links connecting the sensors to the aggregators is not compromised. Hence, we impose a threshold on the distance between the aggregator and any associated sensor, as described by Eq. 2. There are two key design parameters for the sensor networks: aggregator positions and sensor-aggregator associations. The sensor-aggregator associations are represented as a vector $\beta = [\beta_1, \beta_2, \ldots, \beta_N]$, where $\beta_i = \left[\beta_{i1}, \beta_{i2}, \ldots, \beta_{iN_{\text{ag}}}\right]$ indicates the aggregator associated with sensor i and β_{ij} is a binary variable indicating whether sensor i is associated with aggregator j or not. To formulate this problem, we adopt the formalization presented in [17] as follow:

$$\min_{q,\beta} \quad N_{\text{ag}} \tag{16a}$$

$$\text{s.t.} \quad \beta_{ij} \in \{0, 1\} \qquad \forall\, i \tag{16b}$$

$$d_{ij} = \|m_i - q_j\| \leq d_{\text{th}} \qquad \forall\, \beta_{ij} = 1 \tag{16c}$$

$$\sum_{j=1}^{N_{\text{ag}}} \beta_{ij} = 1 \qquad \forall\, i \tag{16d}$$

where is the distance between sensor i and aggregator j, and β_{ij} is a boolean indicating whether sensor i is associated with aggregator j or not. Constraint (16c) assures that the distance between an aggregator and any of its associated sensors is within the maximum communication range to avoid degradation in the quality-of-service of links connecting aggregators and sensors. Constraint (16d) ensures that each sensor must be associated with exactly one aggregator. Hence, all sensor data is guaranteed to be collected.

In the integrated UAV/aggregator network, the goal is to minimize the total amount of energy consumed by the network during the mission. This goal can be achieved by optimizing the UAV trajectories, aggregator transmission power, and dockstation location. This minimization problem can be formulated as follows:

$$\min_{X, P_{\text{ag}}, q_0} \quad E_{\text{total}} \tag{17a}$$

$$\text{s.t.} \quad T^{\text{fly}}_k + T^{\text{hover}}_k \leq T_{\max} \forall k \tag{17b}$$

$$\sum_{j \in \vartheta_k} L_j \leq C_{\max} \forall k \tag{17c}$$

$$P_{\text{ag}} \leq P^{\max}_{\text{ag}} \tag{17d}$$

where $X = [X_1, X_2, \ldots, X_K]$ is a decision variable representing the set of all UAV trajectories. Constraint (17b) ensures that the mission time for any UAV does not exceed the maximum allowed mission time, while (17c) guarantees that the total collected data size by any UAV does not exceed the memory capacity of the UAV. Constraint (17d) limits the value of P_{ag} to $P^{\max}_{\text{ag}}$.

IV. PROBLEM SOLUTION

In this section, we discuss the problems that were presented in (16a) and (17a), as well as the solutions that have been proposed to address those problems.

A. AGGREGATOR PLACEMENT OPTIMIZATION

The primary objective of the placement problem is to determine the optimal locations of aggregators and the associated sensors that require the smallest number of aggregators. This situation strongly matches the clustering problem discussed and solved in [17] and [27]. The authors in [27] performed K-means clustering repeatedly while increasing the value of N_{ag} until constraint (16c) was met. In [17], the constraint (16c) was embedded as a part of sensor association criteria in K-means to improve the clustering performance. However, in both techniques, the aggregator is located relative to the mean position of the associated sensors, which means that aggregators are placed in areas with a high sensor density. In contrast, the sensors located outside of these regions will require additional aggregators to cover them.

To overcome this problem, we propose a triangulation-based clustering method for determining the location of aggregators. In this method, the positions of the aggregators and the sensor associations are first determined by using common clustering algorithms. The aggregator position is chosen to be the cluster centroid. If the position of the aggregator for a given cluster violates the communication range constraint 16c, a new aggregator location for that cluster is determined by triangulation. If the new position of the aggregator still violates the communication range constraint, we repeat the clustering process with a larger number of clusters. The clustering algorithm is terminated once the position of aggregators in all clusters satisfy the communication range constraint.

The triangulation technique is used in a variety of fields, including wireless sensor networks [28], [29]. In geometry,

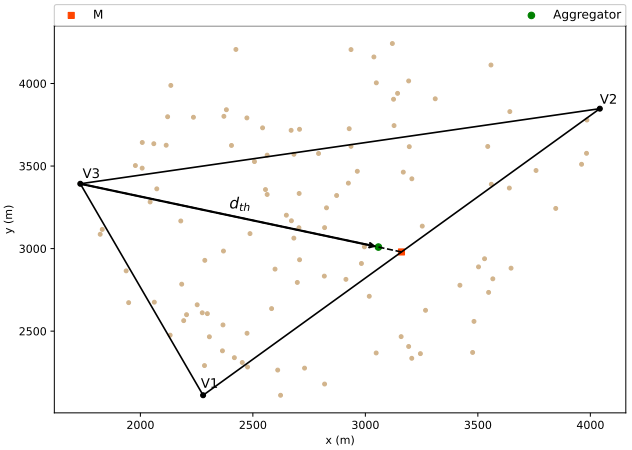


FIGURE 2. Triangulation-based cluster aggregator placement.

triangulation is dividing a polygon into a set of triangles. For triangles, a triangulation point is a point that is used to divide a triangle into three sub-triangles [30]. Triangulation points may have constraints that reduce their possible positions. A valid position for a constrained triangulation point can be found using geometric or heuristic approaches. In this paper, we are interested in a triangulation point whose distance to any vertex is less than or equal to d_{th}. Assuming a triangle with vertices V_1, V_2, and V_3, a simple method to find the desired triangulation point is to get the midpoint $M = (V_1 + V_2)/2$ between V_1 and V_2 and then move the point V_3 along the line V_3M for a distance equal to the maximum allowed value. Fig. 2 illustrates the process of placing an aggregator for a cluster utilizing the proposed triangulation-based methodology. This approach involves the use of triangulation techniques to determine the optimal location for the aggregator within the cluster.

In order to form a triangle, V_1 and V_2 are chosen to be the two sensors furthest away from each other. The third vertex is the sensor furthest away from the midpoint M. The triangulation-based clustering is performed iteratively while increasing N_{ag} similar to [17] and [27]. The implementation of triangulation-based clustering is shown in the Algorithm 1.

Algorithm 1 Triangulation-Based Clustering

Input : d_{th}, G, m
Output: β, q
set $N_{ag}= \left(G/\left(d_{\text{th}} \times 2\right)^2\right) - 1$;
while $d_{ij} > d_{th}$ *for any* $\beta_{ij} = 1$ **do**
  set N_{ag}=$N_{ag} + 1$;
  apply K-means clustering with K = N_{ag} ;
  for *each cluster* **do**
    if $\|m_i - centroid\| \leq d_{th}$ $\forall i \in cluster$ **then**
      continue ;
    get points V_1, V_2, and V_3 from cluster ;
    if $\|M - V_3\| > d_{th}$ **then**
      move point V_3 towards M for d_{th} units ;
      set cluster centroid = point V_3 ;
    else
      set cluster centroid = M ;

B. GAINING-SHARING KNOWLEDGE (GSK) OPTIMIZATION ALGORITHM

Gaining-sharing knowledge (GSK) is a human-related metaheuristic introduced by Mohamed et al. [31] and is inspired by how humans exchange knowledge over their lifetimes. In [31], GSK required fewer function evaluations than other metaheuristics, such as genetic algorithms and particle swarm optimization, to achieve equivalent or better results.

GSK aims to find the best combination of decision variables for a given problem. The algorithm utilizes a pool of candidate solutions (population) denoted by g_i, $i = 1, 2, \ldots, N_{\text{pop}}$, where N_{pop} is the population size. Each candidate solution has different combination of decision variables expressed by $g_i = (g_{i1}, g_{i2}, \ldots, g_{iD})$ where D is the problem dimension. The quality of each solution is assessed by the ability of the solution to minimize a user-defined objective function. Hence, the problem solved by GSK can be defined as:

$$\min_{g_i} f(g_i) \,. \tag{18}$$

where $f(g_i)$ is the value of the user-defined objective function when candidate solution i is applied. The objective function should relate to the value being optimized. In the case of system energy minimization, $f(g_i)$ is equal to the value of E_{total} if candidate solution i was applied.

The algorithm consists of two stages: the junior gaining and sharing stage and the senior gaining and sharing stage. The junior stage represents the early years of a person's life where they exchange knowledge within a small network, such as family. However, they occasionally exchange knowledge with random people due to curiosity. The algorithm executes the junior knowledge exchange by sorting the population according to the objective function. Then, each solution i exchanges knowledge with the nearest best solution, the nearest worse solution, and a random solution. The nearest best solution, the nearest worse solution, and the random solution are denoted by g_{i-1}, g_{i+1}, and g_r, respectively. The equation for knowledge exchange in the junior stage is expressed as follows:

$$g_{ij}^{\text{new}} = \begin{cases} g_{ij} + k_f \times \left(g_{(i-1)j} - g_{(i+1)j} + g_{rj} - g_{ij}\right) \\ \qquad\qquad \text{if } f(g_i) > f(g_r) \\ g_{ij} + k_f \times \left(g_{(i-1)j} - g_{(i+1)j} + g_{ij} - g_{rj}\right) \\ \qquad\qquad \text{if } f(g_i) < f(g_r) \end{cases} \tag{19}$$

where $k_f \in [0, 1]$ is known as the knowledge factor and it controls the amount of knowledge to be transmitted to each candidate solution.

The senior stage describes the late years of a person's life, where humans have enough experience in differentiating between good and bad people and have the ability to exchange knowledge with the appropriate people outside their small network. Therefore, updates are done by classifying the population into three categories according to their objective function value: best, middle, and worst. A solution exchanges knowledge with a random individual from the top $(100 \times p)\%$ solutions from each category, where $p \in [0, 1]$. The best, middle, and worst chosen for knowledge exchange are denoted by g_{best}, g_{mid}, g_{worst}. The equation for knowledge exchange in the senior stage is expressed as follows:

$$g_{ij}^{\text{new}} = \begin{cases} g_{ij} + k_f \times \left(g_{(\text{best})j} - g_{(\text{worst})j} + g_{(\text{mid})j} - g_{ij}\right) \\ \qquad\qquad \text{if } f(g_i) > f(g_{\text{mid}}) \\ g_{ij} + k_f \times \left(g_{(\text{best})j} - g_{(\text{worst})j} + g_{ij} - g_{(\text{mid})j}\right) \\ \qquad\qquad \text{if } f(g_i) < f(g_{\text{mid}}) \end{cases} \tag{20}$$

where g_{ij} represents the value of dimension j of solution i.

The value of a dimension is only updated in the junior and senior stages if a randomly generated number is less than the knowledge ratio k_r. As a result, k_r determines the amount of knowledge passed from one generation to another. The algorithm runs for I_{GSK} iterations (generations) and returns the solution with the lowest objective value across generations, written as g_{global}. The notations GSK uses are described in Table 3.

TABLE 3. GSK notations.

Notation	Definition
g_i	Candidate solution i
N_{pop}	Number of solutions in the population
g_{ij}	Value of dimension j in solution i
D	Number of problem dimensions
$f(g_i)$	Objective value of solution i
k_f	Knowledge factor
k_r	Knowledge ratio
I_{GSK}	Number of generations
g_{global}	Best solution

C. SYSTEM ENERGY OPTIMIZATION

To solve the energy optimization problem described by Equation (17a), we suggest using the GSK metaheuristic. However, the multi-dimensionality of the problem due to the coupling between UAV trajectories,colorblue denoted by X, and the values of P_{ag} and q_0, makes the problem challenging. Therefore, we follow a two-level optimization approach to decouple the variables and reduce the complexity of the problem. The outer optimization level uses GSK to optimize the values of P_{ag} and q_0, hence g_i for this problem has a dimension $D = 2$ and a candidate solution can be written as $g_i = \left(q_0[i], P_{\text{ag}}[i]\right)$. The random value of P_{ag} is restricted between 0 and $P_{\text{ag}}^{\max}$ to satisfy constraint (17d). The inner level is UAV trajectories optimization.

The inner level optimization is the problem of finding the optimum trajectory for data collecting UAVs, given the value of P_{ag} and q_0. In the data collection model, all UAVs are assumed to have identical ground-to-air channels, and all UAVs collect data only while hovering over aggregators directly. Therefore, the propulsion energy of hovering over an aggregator j is the same for any UAV k. Given that P_{ag} is also the same for any aggregator, the energy consumed for the communication between any aggregator and any UAV is the same. Hence, the inner optimization problem can be reduced to finding the set of trajectories that minimize E_{UAV} constrained by $C_{\max}$ and $T_{\max}$. The discussed UAV trajectory optimization problem matches the definition of CVRP description with the addition of a mission time constraint. The CVRP is a problem that seeks to find the optimal paths for multiple vehicles, such that all paths begin and end at the same point (dockstation) and that a single-vehicle visits each client. The total demand of clients along any path is constrained by the vehicle capacity [32] (similar to constraint (17c)). The objective of the problem is to minimize the total lengths of all paths and thus the energy/fuel cost. The CVRP problem has both heuristic and metaheuristic solutions in the literature that can be used to determine the optimal trajectories for any candidate solution [33]. There are also software CVRP solvers that implement heuristic or metaheuristic solutions. One example is the OR-Tools [34] developed by Google. OR-Tools allows defining customized constraints to the problem. Moreover, OR-Tools offers an option that allows the best metaheuristic to be used and allows the developer to set a timeout period, colorbluerepresented with T_{timeout}, for solving the problem to avoid running for long periods of time. We used OR-Tools as recommended by [17] but we added a capacity constraint on the vehicles to simulate the memory capacity constraints described in (17c). Algorithm 2 shows how the two-level optimization method works to get the optimum q_0,P_{ag} pair value (colorbluerepresented by g_{global}) and the optimum trajectories (colorbluedenoted by X_{global}) for this q_0,P_{ag} pair.

D. COMPLEXITY ANALYSIS

Line 1 generates N_{pop} candidate solutions, therefore its complexity is $O(N_{\text{pop}})$. Line 2 is $O(I_{\text{GSK}})$ and line 3 is $O(N_{\text{pop}})$. Line 4 is $O(T_{\text{timeout}})$ because OR-Tools take at most T_{timeout} seconds to get the routing solution for the CVRP problem. The complexity of line 5 is $O(N_{\text{ag}})$ to calculate the communication energy consumed by each aggregator and sum them to get E_{comm}. For E_{UAV} in line 6, we need to the sum of hovering times of all UAVs and the sum of flying times for all UAVs. The total hovering time was already calculated to get E_{comm}, hence we only need to calculate the total flying time of all UAVs. To calculate the flying time of all UAVs we need to know D_k for each UAV. For UAV k,

Algorithm 2 Dockstation Placement and Transmission Power Optimization Using GSK

Input : p,k_r,k_f,N_{pop},I_{GSK}
Output: g_{global},X_{global}
generate random N_{pop} pairs of q_0,P_{ag} ;
for I_{GSK} *iterations* **do**
  for $i = 0 \longrightarrow N_{pop}$ **do**
    Calculate $X[i]$ and T_k^{fly} for each UAV k using OR-Tools using $q_0[i]$,$P_{\text{ag}}[i]$;
    Calculate E_{comm} using (10);
    Calculate E_{UAV} using (13);
    $f(g_i) = E_{\text{comm}} + E_{\text{UAV}}$;
  Perform junior knowledge exchange;
  Perform senior knowledge exchange;
  for $i = 0 \longrightarrow N_{pop}$ **do**
    if $f(g_i^{new}) < f(g_i)$ **then**
      $g_i \leftarrow g_i^{\text{new}}$;
      $f(g_i) \leftarrow f(g_i^{\text{new}})$;
    if $f(g_i) < f(g_{global})$ **then**
      $g_{\text{global}} \leftarrow g_i$;
      $X_{\text{global}} \leftarrow X[i]$;
      $f(g_{\text{global}}) \leftarrow f(g_i)$;

D_k can be calculated by summing the distance between each two consecutive nodes along the trajectory of the UAV. For one trajectory, the number of segments contained is equal to the number of nodes it passes through which is the served aggregators plus the dockstation. Hence, for all trajectories where each aggregator is contained in one trajectory only, the total number of segments considered is $N_{\text{ag}} + K$. Therefore, the complexity of line 6 is $O(N_{\text{ag}}+K)$. Line 7 is an assignment which is $O(1)$. Lines 8 and 9 are $O(1)$ since both junior and senior stages consist of only assignments and if statements. 10 is $O(N_{\text{pop}})$ similar to line 3. Lines $11-17$ are all $O(1)$ similar to lines 8 and 9. Therefore, the complexity of Algorithm 2 is $O(I_{\text{GSK}}N_{\text{pop}}(T_{\text{timeout}} + N_{\text{ag}} + K))$.

V. SIMULATION RESULTS AND DISCUSSIONS

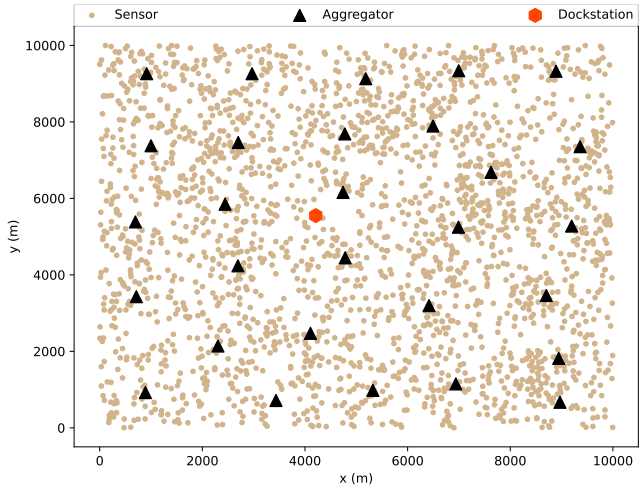
A. SIMULATION SETUP

We should first explore a sensor distribution technique to validate our approach and the developed algorithms across a range of sensor densities. The Homogeneous Poisson point process (HPPP) is a two-dimensional distribution that has been successfully used to represent the behavior of cellular users [35] and sensor nodes distributed over a geographical area [36]. The HPPP adopts the assumption that users/sensors intensity is consistent throughout the area, which does not take into consideration variances in population or necessary sensory data between different areas. To address this issue, the number and positioning of sensors are determined using a mixed Poisson point process (MPPP). The MPPP divides the overall area into smaller sub-areas and generates a random sensor intensity for each sub-area, expressed by the notation λ. While the sensors within each sub-area are distributed using the HPPP with an intensity value equal to the λ generated [37]. The value of λ generates using a gamma distribution with shape k_γ and a scale $\theta = \mu_\lambda/k_\gamma$, where μ_λ is the mean intensity per square meter. Therefore, the system will be evaluated at three densities (low: $\mu_\lambda = 2.5 \times 10^{-5}$, medium: $\mu_\lambda = 5 \times 10^{-5}$, and high: $\mu_\lambda = 10 \times 10^{-5}$). Fig. 3 illustrates an example of MPPP distributed sensors at low, medium, and high mean densities, as well as the aggregator and dockstation placement resulting from Algorithms 1 and 2 at the three mean densities for a (10×10) km area.

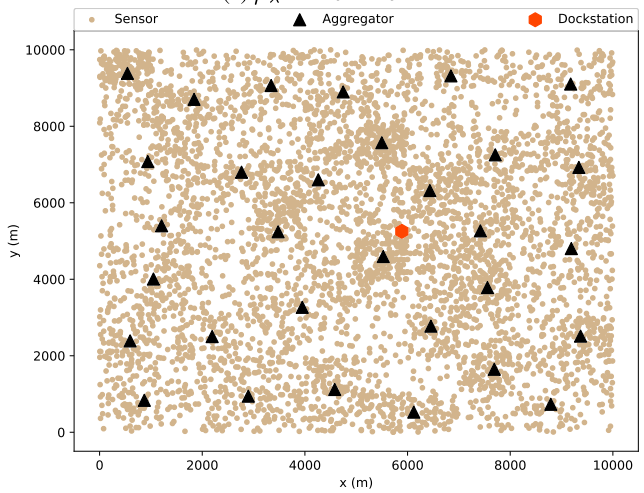
The system was developed using the Java Standard Edition 13 programming language, using OR-Tools to solve the CVRP problem. The simulations were conducted on a computing system consisting of a single processor Intel Core i5, which is equipped with four cores running at a clock speed of 1.6 GHz. The amount of traffic at each sensor S_i is generated using a continuous uniform distribution in the range [100, 1000] kbits following the sensor information volume found in [24] and [38]. After clustering is applied to the sensor network, the size of the data L_j at each aggregator is obtained. The UAV parameters used in the simulation are retrieved from [17]. To evaluate the system, three alternative area sizes, G, were used: 10×10 km, 15×15 km, and 20×20 km. Thus, the system was tested under six which involved the use of all possible combinations of μ_λ and G. This approach was adopted to ensure that the system's performance was evaluated comprehensively across a range of conditions.

Small UAVs typically have a limited flight time of 10-30 minutes, which is determined by the capacity of their batteries [39]. In order to reduce the cost of the system, UAVs with lower battery capacities are preferred. However, the use of UAVs with shorter flight times (e.g., 10 minutes) may not be sufficient to cover large areas, such as those measuring 15×15 km and 20×20 km. To address this issue, we have increased T_{max} to 900 s and 1200 s for areas of 15×15 km and 20×20 km, respectively. This modification has been made to ensure that the CVRP can be solved effectively. If UAVs with high battery capacity cannot be deployed, the large area can be divided into smaller sub-areas, where each sub-area has its dockstation to enable UAVs with small battery capacities to cover all aggregators. The optimization problem is then solved for each sub-area independently. However, this will incur extra costs for installing additional dockstations. The parameters used in the simulation are summarized in Table 4.

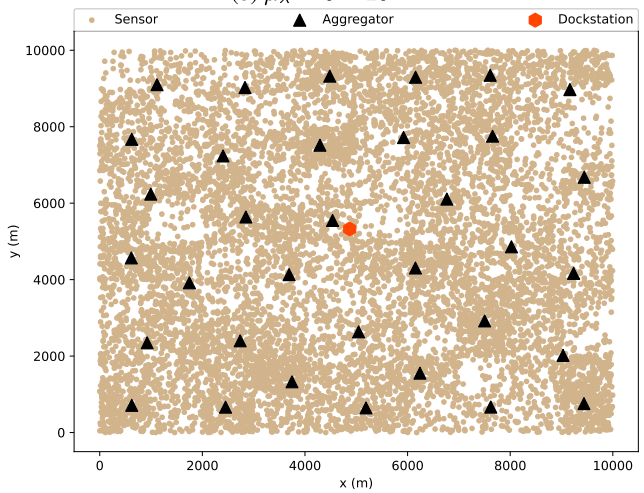
Given that the number of aggregators to be visited is relatively low (as seen in Table 5) and that Java is faster than Python, the CVRP problem can be solved with $T_{\text{timeout}} = 1$ sec. From the complexity analysis in section IV-D and the values of T_{timeout}, I_{GSK}, and N_{pop} we can conclude that the algorithm can get the optimum dockstation position and transmission power for a given aggregator distribution within



(a) $\mu_\lambda = 2.5 \times 10^{-5}$



(b) $\mu_\lambda = 5 \times 10^{-5}$



(c) $\mu_\lambda = 1 \times 10^{-4}$

FIGURE 3. Output of aggregator placement and dockstation placement given by Algorithms 1 and 2 at different values of μ_λ.

2 minutes. This time delay is acceptable when taking a one-time decision such as dockstation placement.

TABLE 4. Simulation parameters [17], [31].

Parameter	Value	Parameter	Value
G	[10x10, 15x15, 20x20] km	μ_λ	[10^{-4}, 5×10^{-5}, 2.5×10^{-5}] user/m^2
k_γ	5	γ_{th}	1 W
α	2.7	σ^2_{G2G}	10^{-14} W
$T_{\max}$	[600, 900, 1200] s	$C_{\max}$	256 MB
H	100 m	η_{LoS}	1
η_{NLoS}	20	f_c	2 GHz
c	3×10^8 m/s	B	10 MHz
σ^2_{UAV}	-109 dbm	ξ_I	118 W
v_0	5.4 m/s	ξ_B	3.4 W
Q_{tip}	60 m/s	d_0	0.3
r	0.03	ρ	1.225 kg/m^3
A	0.28 m^2	a	9.61
b	0.16	v_k	30 m/s
k_f	0.5	p	0.1
k_r	0.9	N_{pop}	20
I_{GSK}	50	T_{timeout}	1 sec

B. SIMULATION RESULTS

To evaluate the validity of the proposed methodology and algorithms, we study the effect of P_{kbit}, G and μ_λ on UAV power consumption. Fig. 4 illustrates the UAV power consumption when various combinations of P_{kbit}, G and μ_λ values are used. Each redult in the figure is further labelled with the average number of UAVs required, denoted by K, at each P_{kbit}, G, and μ_λ. For a given sensor distribution, the value of K is an integer number; however, because the sensor distribution and traffic change from one trial to another, the trajectories and number of UAVs required also change with each trial. We calculate the average value of K throughout all simulation trials for comparison purposes. The results show that as the value of P_{kbit} increases for a given G and μ_λ, the value of K reduces, and therefore the amount UAV power consumed decreases as well. These results are consistent with the fact that increasing P_{kbit} increases the value of d_{th} implying that a smaller number of aggregators can service the same number of sensors. Reducing the number of aggregators enables the CVRP solver to find shorter trajectories and cover all aggregators using fewer UAVs. Additionally, It can be observed that the UAV power decreases as G decreases for a given μ_λ and P_{kbit}. This observation is justified since the increase in the coverage area leads to the use of additional UAVs in order to cover all aggregators within the time constraint. As a result, as the number of UAVs increases, the overall power consumed by UAVs increases. As well as, the overall power consumed by UAVs could be increased by raising μ_λ for a given value of P_{kbit} and G.

The number of aggregators for the same G is nearly the same for different values of μ_λ as observed in Fig. 3. The increase in the number of sensors with higher μ_λ means that more sensors are associated with a single aggregator, which leads to an increase in the amount of traffic collected by each aggregator L_j. The increase in L_j increases T_{kj}^{hover} which also increases the energy consumed by the UAVs or the number of UAVs, K, to satisfy the memory constraints. In our remaining

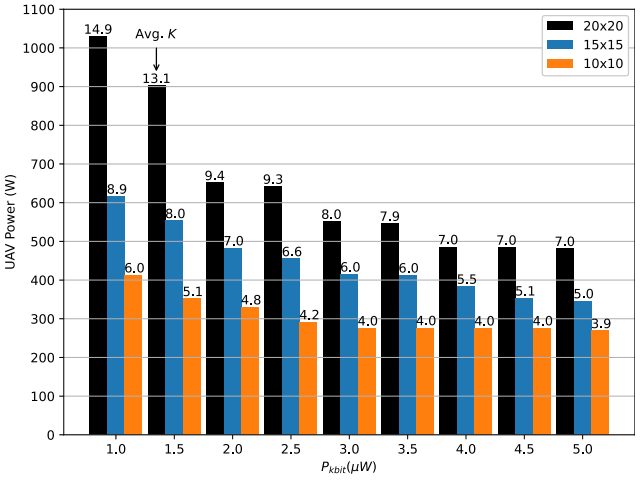
simulations, the value of P_{kbit} for a certain μ_λ, G is chosen using Fig. 4 as the value of P_{kbit} at which the UAV power reaches a steady value. For example, the P_{kbit} for $G = 10 \times 10$ at $\mu_\lambda = 1 \times 10^{-4}$ is 3 μW.

In order to demonstrate the effectiveness of triangulation-based clustering in terms of reducing N_{ag}, we simply compare the number of aggregators obtained by our approach to those provided by other algorithms. The average number of aggregators is calculated over 5000 different random trials, where the placement and traffic of sensors are determined as described in section V-A. The value of N_{ag} does not differ significantly for different μ_λ at the same value of G. Hence, the comparison was carried out at $\mu_\lambda = 2.5 \times 10^{-5}$ for different values of coverage area G. Referring to Fig. 4, the values of P_{kbit} were 3, 4.5 and 4 μW for $G = 10 \times 10, 15 \times 15, 20 \times 20$, respectively. For the purposes of comparison, the following five alternative clustering methods are taken into consideration:

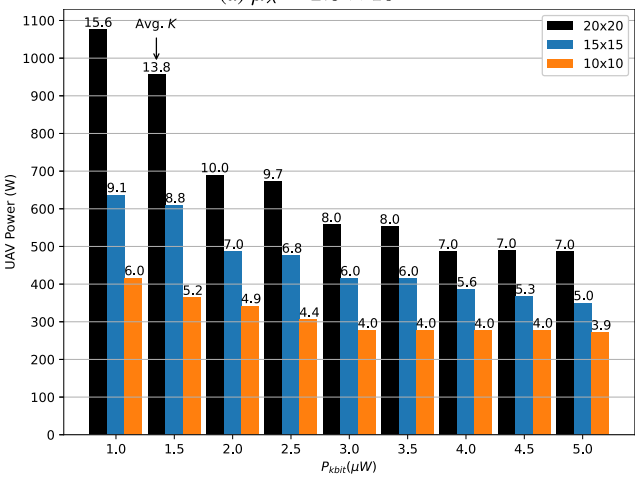
1) Triangulation-based K-means (TK-means) described in Algorithm 1
2) Triangulation-based hierarchical agglomerative clustering (THAC), which replaces the K-means in Algorithm 1 with hierarchical agglomerative clustering (HAC) [40]
3) Triangulation-based Gaussian mixture model (TGMM), which replaces the K-means in Algorithm 1 with Gaussian mixture model (GMM) [41]
4) Triangulation-based spectral clustering (TSpectral), which replaces the K-means in Algorithm 1 with spectral clustering [42]
5) Constrained K-means clustering introduced in [17]

The results of this comparison are presented in Table 5. Similar to the results in [17], THAC clustering showed worse performance than TK-means clustering. Additionally, the TK-means performed much better than TGMM and TSpectral clustering algorithms and therefore it was adopted as the clustering technique for Algorithm 1. Although constrained K-means successfully minimized N_{ag} compared to K-means based clustering presented in [27], TK-means successfully minimized the number of aggregators for various coverage area, G, up to 20% compared to constrained K-means proposed in [17].

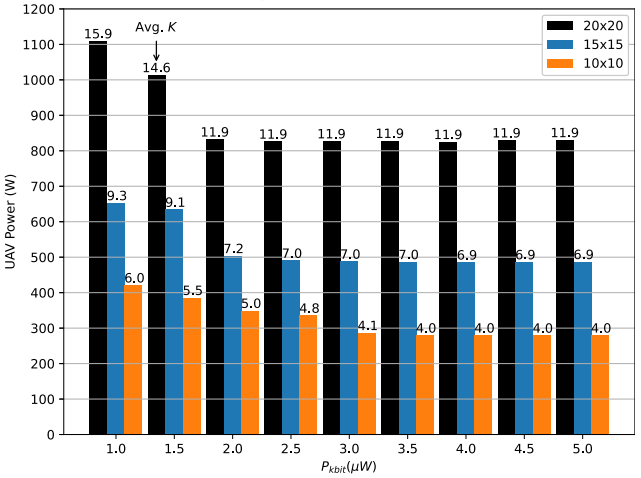
We compared the total energy consumption of applying TK-means with GSK optimization framework with the total energy consumption of two other methods to evaluate the efficiency of our proposed framework in terms of minimizing the total energy consumption of the system. One method is the offline data collection method presented in [17]. The other method utilizes the proposed TK-means clustering, but with fixed values for P_{ag} and q_0 and gets the UAV trajectory by solving the CVRP problem. The fixed values for P_{ag} and q_0 are 15 dBm, and the midpoint of the area, respectively, similar to the values used in [17]. The latter model is used to present the energy saving gained by utilizing GSK for optimizing the values of P_{ag} and q_0.



(a) $\mu_\lambda = 2.5 \times 10^{-5}$



(b) $\mu_\lambda = 5 \times 10^{-5}$



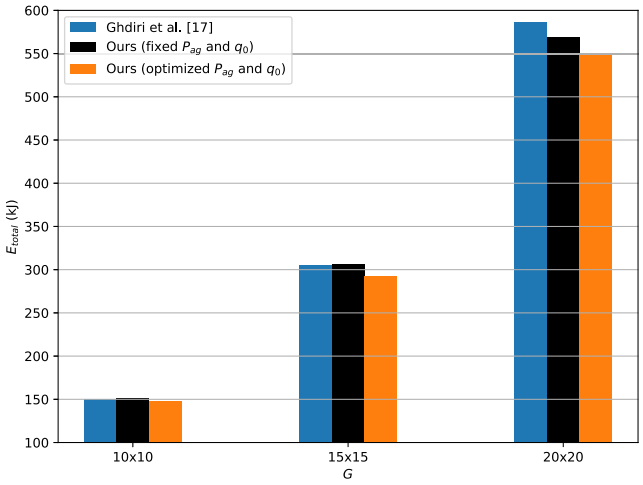
(c) $\mu_\lambda = 1 \times 10^{-4}$

FIGURE 4. UAV power consumption and K versus P_{kbit} at different values of G and μ_λ.

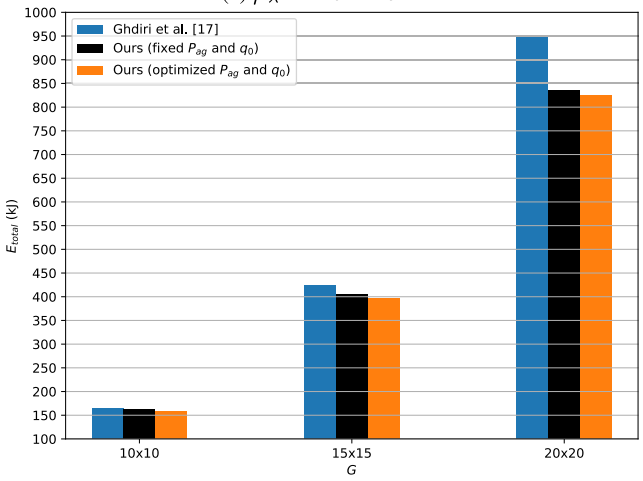
Fig. 5 shows the total energy consumption given by each algorithm at different values of G and μ_λ. At $\mu_\lambda = 2.5 \times 10^{-5}$, the values of P_{kbit} used were 3, 4.5 and 4 μW for

TABLE 5. Average N_{ag} for each algorithm with various coverage area, G, values.

Method \ Coverage Area G	10×10	15×15	20×20
TK-means (Proposed)	29.4	50.2	99.6
THAC	39.4	69.0	142.9
TGMM	33.7	55.9	112.1
TSpectral	40.9	67.0	137.8
Constrained K-means [17]	36.4	59.1	115.1

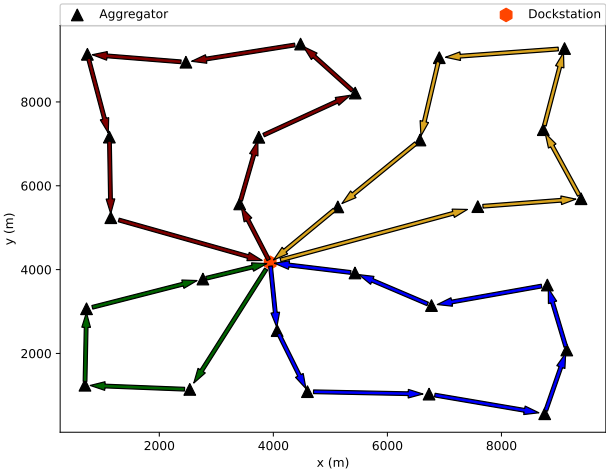


(a) $\mu_\lambda = 2.5 \times 10^{-5}$

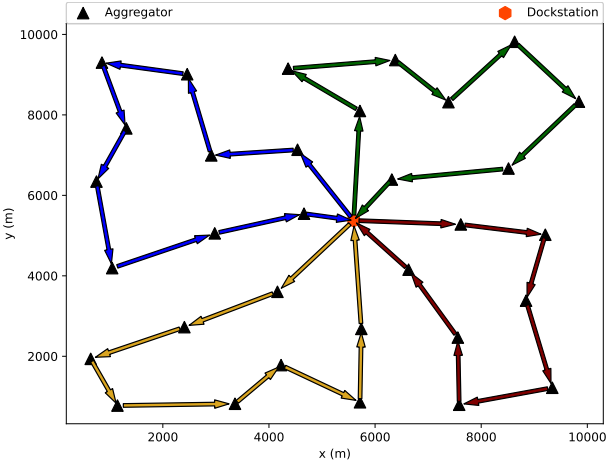


(b) $\mu_\lambda = 1 \times 10^{-4}$

FIGURE 5. Total energy consumption at different G and μ_λ.



(a) $\mu_\lambda = 2.5 \times 10^{-5}$



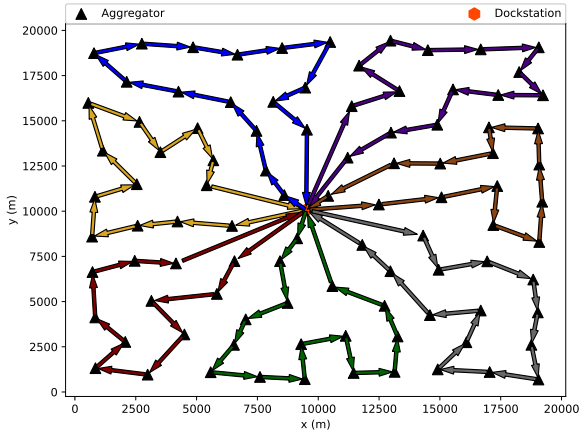
(b) $\mu_\lambda = 1 \times 10^{-4}$

FIGURE 6. UAV trajectories for $G = 10 \times 10$ at different μ_λ.

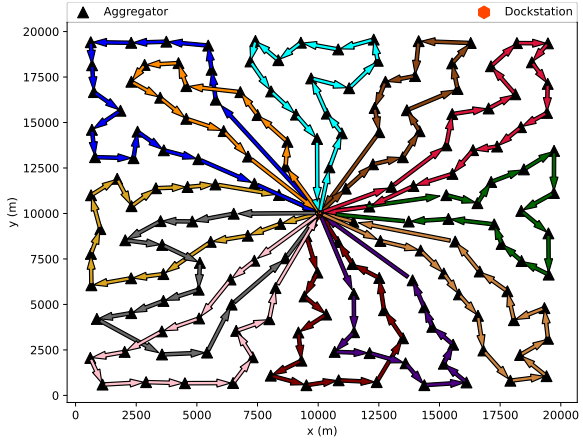
$G = 10 \times 10$, 15×15 and 20×20, respectively. For $\mu_\lambda = 1 \times 10^{-4}$, the values of P_{kbit} used were 3, 2.5 and 2 μW for G 10×10, 15×15 and 20×20, respectively. The results show that the TK-means with GSK optimization framework consumes the least amount of energy across all areas and sensor densities and decreases the energy consumption by up to 13.25% compared to [17].

Additionally, it is demonstrated that the proposed P_{ag} and q_0 optimization are more effective at smaller μ_λ. Applying the GSK optimization after TK-means reduced the total energy consumed for $G = 10 \times 10$, 15×15 and 20×20 by 2.00%, 4.72% and 3.81%, respectively, compared to utilizing TK-means alone. At a low μ_λ value, it is more feasible that sensors concentrate at specific patches while leaving others nearly empty. In this scenario, locating the dockstation near densely populated areas is preferable. At large μ_λ, approximately all sections contain sensors, so the best dockstation location is near the center of the area, as seen in Fig. 3. Since the fixed P_{ag} and dockstation technique places the dockstation at the midpoint of the area, the difference between the optimized method and the fixed value method at large μ_λ values is only the aggregator power.

As can be shown, the approach described in [17] and the proposed TK-means with fixed P_{ag} and q_0 perform similarly at $G = 10 \times 10$ at both high and low sensor densities and at $G = 15 \times 15$ at low sensor density. At the mentioned G and μ_λ values, the difference between N_{ag} calculated using TK-means and constrained K-means is not significant. Therefore, the energy consumed by visiting extra points is compensated in [17] by moving aggregators towards the dockstation to shorten the trajectories. Nonetheless,



(a) $\mu_\lambda = 2.5 \times 10^{-5}$



(b) $\mu_\lambda = 1 \times 10^{-4}$

FIGURE 7. UAV trajectories for $G = 20 \times 20$ at different μ_λ.

executing GSK optimization for P_{ag} and q_0 provides better energy savings compared to shifting aggregators towards the dockstation by 1.30%, 4.21%, and 4.33% for $G = 10 \times 10$ with low sensor density, $G = 10 \times 10$ at high sensor density, and $G = 15 \times 15$ at low sensor density, respectively. Additionally, splitting a 20×20 area into four 10×10 areas and solving each independently might yield lower energy consumption per trip. However, the maintenance and deployment costs would grow due to the increased requirement for dockstations and UAVs.

Figures 6 and 7 illustrate the results of the placement of aggregator positions, dockstations, and UAV trajectories in various scenarios with different values of G and μ_λ. The aggregators, dockstation, and trajectories of multiple UAVs are represented by different colors. The figures demonstrate that the algorithm successfully collects data from all aggregators while ensuring that no aggregator is accessed more than once. However, as the number of UAVs required increases, the probability of having UAVs with intersecting trajectories increases, as demonstrated in Fig. 7b. The issue of ensuring that UAV trajectories do not result in collisions was studied in [43]. Nonetheless, the integration between our work and [43] remains a topic for future research. These findings contribute to the scientific understanding of the performance of the algorithm and provide insights into the challenges that may arise when scaling up the system.

VI. CONCLUSION AND FUTURE WORK

In this paper, we presented an approach to optimize the energy consumption and cost of UAV-assisted WSNs for data collection, considering the time and memory constraints of UAVs. The problem was broken down into two sub-problems. First, the number and position of aggregators and the sensor-aggregator associations were determined using triangulation-based K-means clustering to minimize the number of aggregators used. The maximum communication power-constrained this problem for sensors. The results showed that triangulation-based K-means minimized the number of needed aggregators compared to other triangulation-based clustering algorithms found in the literature up to 28.12%. Second, the dockstation position, the aggregator communication power, and the UAV trajectories are optimized on multiple steps. Using the GSK optimization algorithm, a population of dockstation position and aggregator power pairs are generated. For each dockstation position and aggregator power pair, the optimum UAV trajectory is designed by solving a CVRP problem, and the total power consumed by the solution is calculated. The GSK updates the population according to the total energy consumption of each solution. The process is repeated for a given number of generations, and the best solution across all generations is returned. Results displayed that our framework successfully minimized the energy consumption per mission compared to other frameworks.

In the future, we will consider integrating our work with a crash avoidance technique such that energy consumption is still minimized. Other optimization factors such as UAV speed and hovering positions were not considered in this work and can be explored in the future. In this paper, the data collection framework considers non-real-time applications which have relaxed time constraints and do not wait for a response. To extend this framework to include real-time application, connections to high-altitude platforms (HAPs) need to be investigated. Moreover, balancing between incoming real-time tasks and non-real-time tasks should be considered to maintain a satisfactory quality of service for such a system.

REFERENCES

[1] M. T. Nguyen, C. V. Nguyen, H. T. Do, H. T. Hua, T. A. Tran, A. D. Nguyen, G. Ala, and F. Viola, "UAV-assisted data collection in wireless sensor networks: A comprehensive survey," *Electronics*, vol. 10, no. 21, p. 2603, Oct. 2021.

[2] R. A. Nazib and S. Moh, "Energy-efficient and fast data collection in UAV-aided wireless sensor networks for hilly terrains," *IEEE Access*, vol. 9, pp. 23168–23190, 2021.

[3] A. Shahraki, A. Taherkordi, Ø. Haugen, and F. Eliassen, "A survey and future directions on clustering: From WSNs to IoT and modern networking paradigms," *IEEE Trans. Netw. Service Manage.*, vol. 18, no. 2, pp. 2242–2274, Jun. 2021.

[4] S. Fu, Y. Tang, Y. Wu, N. Zhang, H. Gu, C. Chen, and M. Liu, "Energy-efficient UAV-enabled data collection via wireless charging: A reinforcement learning approach," *IEEE Internet Things J.*, vol. 8, no. 12, pp. 10209–10219, Jun. 2021.

[5] T. Feng, L. Xie, J. Yao, and J. Xu, "UAV-enabled data collection for wireless sensor networks with distributed beamforming," *IEEE Trans. Wireless Commun.*, vol. 21, no. 2, pp. 1347–1361, Feb. 2022.

[6] N. Elmeseiry, N. Alshaer, and T. Ismail, "A detailed survey and future directions of unmanned aerial vehicles (UAVs) with potential applications," *Aerospace*, vol. 8, no. 12, pp. 1–29, 2021.

[7] N. Elmeseiry, A. I. Salama, N. Alshaer, and T. Ismail, "Design and analysis of a reliable quadcopter UAV for wireless communication purposes," in *Proc. 3rd Novel Intell. Lead. Emerg. Sci. Conf. (NILES)*, Oct. 2021, pp. 193–198.

[8] A. Manzoor, K. Kim, S. R. Pandey, S. M. A. Kazmi, N. H. Tran, W. Saad, and C. S. Hong, "Ruin theory for energy-efficient resource allocation in UAV-assisted cellular networks," *IEEE Trans. Commun.*, vol. 69, no. 6, pp. 3943–3956, Jun. 2021.

[9] J. Ji, K. Zhu, D. Niyato, and R. Wang, "Joint cache placement, flight trajectory, and transmission power optimization for multi-UAV assisted wireless networks," *IEEE Trans. Wireless Commun.*, vol. 19, no. 8, pp. 5389–5403, Aug. 2020.

[10] S. K. Haider, A. Jiang, A. Almogren, A. U. Rehman, A. Ahmed, W. U. Khan, and H. Hamam, "Energy efficient UAV flight path model for cluster head selection in next-generation wireless sensor networks," *Sensors*, vol. 21, no. 24, pp. 1–22, 2021.

[11] M. A. Alanezi, A. F. Salami, Y. A. Sha'aban, H. R. E. H. Bouchekara, M. S. Shahriar, M. Khodja, and M. K. Smail, "UBER: UAV-based energy-efficient reconfigurable routing scheme for smart wireless livestock sensor network," *Sensors*, vol. 17, no. 22, pp. 1–20, 2022.

[12] R. Citroni, F. Di Paolo, and P. Livreri, "A novel energy harvester for powering small UAVs: Performance analysis, model validation and flight results," *Sensors*, vol. 19, no. 8, pp. 1–21, 2019.

[13] A. Osman and M. Alijani, "Energy-efficient techniques for UAVs in communication-based applications," in *Proc. 26th Int. Conf. Autom. Comput. (ICAC)*, Sep. 2021, pp. 1–6.

[14] J. Li, H. Zhao, H. Wang, F. Gu, J. Wei, H. Yin, and B. Ren, "Joint optimization on trajectory, altitude, velocity, and link scheduling for minimum mission time in UAV-aided data collection," *IEEE Internet Things J.*, vol. 7, no. 2, pp. 1464–1475, Feb. 2020.

[15] F. Wu, D. Yang, L. Xiao, and L. Cuthbert, "Energy consumption and completion time tradeoff in rotary-wing UAV enabled WPCN," *IEEE Access*, vol. 7, pp. 79617–79635, 2019.

[16] Y. Wang, M. Chen, C. Pan, K. Wang, and Y. Pan, "Joint optimization of UAV trajectory and sensor uploading powers for UAV-assisted data collection in wireless sensor networks," *IEEE Internet Things J.*, vol. 9, no. 13, pp. 11214–11226, Jul. 2022.

[17] O. Ghdiri, W. Jaafar, S. Alfattani, J. B. Abderrazak, and H. Yanikomeroglu, "Offline and online UAV-enabled data collection in time-constrained IoT networks," *IEEE Trans. Green Commun. Netw.*, vol. 5, no. 4, pp. 1918–1933, Dec. 2021.

[18] S. Shen, K. Yang, K. Wang, G. Zhang, and H. Mei, "Number and operation time minimization for multi-UAV-enabled data collection system with time windows," *IEEE Internet Things J.*, vol. 9, no. 12, pp. 10149–10161, Jun. 2022.

[19] K.-V. Nguyen, C.-H. Nguyen, T. V. Do, and C. Rotter, "Efficient multi-UAV assisted data gathering schemes for maximizing the operation time of wireless sensor networks in precision farming," *IEEE Trans. Ind. Informat.*, vol. 19, no. 12, pp. 11664–11674, 2023.

[20] Y. Pan, Y. Yang, and W. Li, "A deep learning trained by genetic algorithm to improve the efficiency of path planning for data collection with multi-UAV," *IEEE Access*, vol. 9, pp. 7994–8005, 2021.

[21] S. S. Khodaparast, X. Lu, P. Wang, and U. T. Nguyen, "Deep reinforcement learning based energy efficient multi-UAV data collection for IoT networks," *IEEE Open J. Veh. Technol.*, vol. 2, pp. 249–260, 2021.

[22] X. Qin, Z. Song, Y. Hao, and X. Sun, "Joint resource allocation and trajectory optimization for multi-UAV-assisted multi-access mobile edge computing," *IEEE Wireless Commun. Lett.*, vol. 10, no. 7, pp. 1400–1404, Jul. 2021.

[23] C. Sun, W. Ni, and X. Wang, "Joint computation offloading and trajectory planning for UAV-assisted edge computing," *IEEE Trans. Wireless Commun.*, vol. 20, no. 8, pp. 5343–5358, Aug. 2021.

[24] 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.

[25] G. Yang, R. Dai, and Y.-C. Liang, "Energy-efficient UAV backscatter communication with joint trajectory design and resource optimization," *IEEE Trans. Wireless Commun.*, vol. 20, no. 2, pp. 926–941, Feb. 2021.

[26] W.-T. Li, M. Zhao, Y.-H. Wu, J.-J. Yu, L.-Y. Bao, H. Yang, and D. Liu, "Collaborative offloading for UAV-enabled time-sensitive MEC networks," *EURASIP J. Wireless Commun. Netw.*, vol. 2021, no. 1, pp. 1–17, 2021.

[27] S. Alfattani, W. Jaafar, H. Yanikomeroglu, and A. Yongacoglu, "Multi-UAV data collection framework for wireless sensor networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2019, pp. 1–6.

[28] C. S. J. Rabaey et al., "Robust positioning algorithms for distributed ad-hoc wireless sensor networks," in *Proc. USENIX Tech. Annu. Conf.*, 2002, pp. 317–327.

[29] A. Kanavalli, G. Bharath, P. D. Shenoy, K. Venugopal, and L. Patnaik, "Triangulation based clustering for improving network lifetime in wireless sensor networks," *Trends Netw. Commun.*, pp. 272–284, 2011.

[30] E. W. Weisstein. *Triangulation Point*. [Online]. Available: https://mathworld.wolfram.com/TriangulationPoint.html

[31] A. W. Mohamed, A. A. Hadi, and A. K. Mohamed, "Gaining-sharing knowledge based algorithm for solving optimization problems: A novel nature-inspired algorithm," *Int. J. Mach. Learn. Cybern.*, vol. 11, no. 7, pp. 1501–1529, Jul. 2020.

[32] R. Fukasawa, H. Longo, J. Lysgaard, M. P. D. Aragão, M. Reis, E. Uchoa, and R. F. Werneck, "Robust branch-and-cut-and-price for the capacitated vehicle routing problem," *Math. Program.*, vol. 106, no. 3, pp. 491–511, May 2006.

[33] K. Braekers, K. Ramaekers, and I. Van Nieuwenhuyse, "The vehicle routing problem: State of the art classification and review," *Comput. Ind. Eng.*, vol. 99, pp. 300–313, Sep. 2016.

[34] L. Perron and V. Furnon, "Or-tools," Google, Tech. Rep., 2019. [Online]. Available: https://developers.google.com/optimization/

[35] F. Qamar, K. Dimyati, M. N. Hindia, K. A. Noordin, and I. S. Amiri, "A stochastically geometrical Poisson point process approach for the future 5G D2D enabled cooperative cellular network," *IEEE Access*, vol. 7, pp. 60465–60485, 2019.

[36] T. Kwon and J. M. Cioffi, "Random deployment of data collectors for serving randomly-located sensors," *IEEE Trans. Wireless Commun.*, vol. 12, no. 6, pp. 2556–2565, Jun. 2013.

[37] K. Burnecki, W. Härdle, and R. Weron, "An introduction to simulation of risk processes," *Encyclopedia of Actuarial Science*. Chichester, U.K.: Wiley, 2004, pp. 1564–1570.

[38] N. Hirosawa, H. Iimori, K. Ishibashi, and G. T. F. D. Abreu, "Minimizing age of information in energy harvesting wireless sensor networks," *IEEE Access*, vol. 8, pp. 219934–219945, 2020.

[39] P. Chandhar and E. G. Larsson, "Massive MIMO for connectivity with drones: Case studies and future directions," *IEEE Access*, vol. 7, pp. 94676–94691, 2019.

[40] C.-H. Lung and C. Zhou, "Using hierarchical agglomerative clustering in wireless sensor networks: An energy-efficient and flexible approach," *Ad Hoc Netw.*, vol. 8, no. 3, pp. 328–344, May 2010.

[41] H. Hojjatinia, M. Jahanshahi, and S. Shehnepoor, "Improving lifetime of wireless sensor networks based on nodes' distribution using Gaussian mixture model in multi-mobile sink approach," *Telecommun. Syst.*, vol. 77, no. 1, pp. 255–268, May 2021.

[42] J. Liu and J. Han, "Spectral clustering," in *Data Clustering*. London, U.K.: Chapman & Hall, 2018, pp. 177–200.

[43] A.-D. Tang, T. Han, H. Zhou, and L. Xie, "An improved equilibrium optimizer with application in unmanned aerial vehicle path planning," *Sensors*, vol. 21, no. 5, p. 1814, Mar. 2021.