IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 8, 2021, accepted January 19, 2021, date of publication January 22, 2021, date of current version February 2, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3053605

Bio-Inspired Approaches for Energy-Efficient Localization and Clustering in UAV Networks for Monitoring Wildfires in Remote Areas

MUHAMMAD YEASIR ARAFAT AND SANGMAN MOH, (Member, IEEE)
Department of Computer Engineering, Chosun University, Gwangju 61452, Republic of Korea

Corresponding author: Sangman Moh (smmoh@chosun.ac.kr)

This work was supported in part by the National Research Foundation of Korea (NRF) funded by the Korean Government (MIST) under Grant 2019R1F1A1060501.

ABSTRACT In dynamic unmanned aerial vehicle (UAV) networks, localization and clustering are fundamental functions for cooperative control. In this article, we propose bio-inspired localization (BIL) and clustering (BIC) schemes in UAV networks for wildfire detection and monitoring. First, we develop a hybrid gray wolf optimization (HGWO) method and propose an energy-efficient three-dimensional BIL algorithm based on the HGWO, which reduces localization errors, avoids flip ambiguity in bounded distance measurement errors, and achieves high localization accuracy. In BIL, the bounding cube method is applied to reduce the initial search space. Second, we propose an energy-efficient BIC algorithm based on the HGWO. The BIC algorithm utilizes the gray wolf leadership hierarchy to improve clustering efficiency. We also develop an analytical model for determining the optimal number of clusters that provide the minimum number of transmissions. Finally, we propose a GWO-based compressive sensing (CS-GWO) algorithm to transmit data from cluster heads (CHs) to the base station (BS). The proposed CS-GWO constructs an efficient routing tree from CHs to the BS, thereby reducing the routing delay and the number of transmissions. Our performance evaluation shows that the proposed BIL and BIC significantly outperform conventional schemes in terms of various performance metrics under different scenarios.

INDEX TERMS Bio-inspired algorithm, cluster head, clustering, energy efficiency, gray wolf optimization, localization, network lifetime, routing protocol, unmanned aerial vehicle network.

I. INTRODUCTION

Recently, a new type of ad hoc network known as flying ad hoc network (FANET) has drawn attention. Owing to the deployment of modern communication technologies such as sensors, artificial intelligence, low-cost Wi-Fi radio interfaces, control technologies, battery technologies, micro-embedded systems, and global positioning system (GPS), there has been a significant increase in the use of unmanned aerial vehicles (UAVs). Furthermore, UAVs have shown potential for application in civil and military domains because of their rapid deployment, self-organization, high flexibility, scalability, ease of acquisition, and low maintenance costs [1], [2].

In the last decade, wildfires have become one of the costliest and deadliest natural disasters worldwide. Wildfires have threatened the lives of thousands and caused significant damage to millions of hectares of forest resources, homes, and communication infrastructures [3]. According to a statistical review by the National Interagency Fire Center, since 2000, an annual average of 72,400 wildfires have burned an average of 7.0 million acres [4]. Additionally, because of climate change, wildfires have become a common occurrence. Although ground sensors, satellite imagery, and remotely piloted vehicles are used for detecting and monitoring fires, they are not quick and reliable. The major drawbacks of these technologies are the delays in detecting fires and the low accuracy of satellite images during nighttime and cloudy and foggy weather situations. Furthermore, ground-based forest fire suppression/management is time-consuming and costly, as it requires ground staff and aircraft pilots.

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

Recently, mini-UAVs or drones have become popular for civil applications. Generally, a mini-UAV has a weight of up to a few kilograms, and it is equipped with a camera, several sensors, a wireless communication system, and GPS [5], [6]. Mini-UAVs can transmit collected data (i.e., images and videos) to a ground station. Drone-based wildfire detection and monitoring offers a low-cost and rapid-imaging alternative in remote areas. UAV-based networks have been tested in several fire-monitoring missions [7]. Therefore, wildfire detection and post-fire monitoring are considered in this study.

In a multi-UAV network, UAV-to-UAV (U2U) and UAV-to-base station (U2BS) information sharing is an effective solution for high-quality network communication. Owing to the high mobility of UAVs, UAV localization is a major technical issue. Node localization is realized using GPS; however, GPS suffers from an average location error of 10–30 m [8]. In many cases, such as rural-area monitoring under bad weather conditions, the GPS signal is insufficient or completely absent. Therefore, several methods have been proposed to address this localization problem in UAV networks [9]. Most of these methods, which use the distance measurement method, are based on bilateration and trilateration [10]. However, flip ambiguity (FA) is a major problem in distance-measurement-based localization methods [11]. A majority of existing clustering methods for UAV networks employ GPS systems to locate node positions. However, GPS is expensive and energy consuming. To design an energy-efficient, flexible, and scalable clustering protocol, localization is required.

Recently, the clustering approach has been used by researchers to address the UAV routing problem [12]. In multi-UAV networks, clustering is used to control the scalability and stability of networks. Because of energy limitations, the network lifetime is a crucial parameter in UAV networks. Moreover, owing to the high mobility of UAVs, topology control is essential to reduce communication interference. The clustering mechanism solves the long-distance communication issue, increases network scalability, prolongs network lifetime, and increases the reliability of the entire network. Furthermore, clustering can provide efficient and steady routes with low communication overhead during the route discovery and forwarding processes [13].

Recently, bio-inspired algorithms have been commonly applied to solve the clustering problem in UAV networks. This is because of their simplicity, effectiveness in solving complex optimization problems, and local minimum avoidance. A majority of these algorithms are inspired by animal behavior, evolutionary concepts, and physical phenomena. Owing to their simplicity, researchers have tried to develop and propose new bio-inspired algorithms. In this domain, gray wolf optimization (GWO) is a relatively new bio-inspired algorithm. GWO is based on the hunting mechanism of gray wolves (i.e., searching, tracking, encircling, and attacking the prey). In contrast to other bio-inspired algorithms, it considers the leadership hierarchy. For the past two years, researchers have employed the GWO algorithm to solve localization [14], path planning [15], [16], topology control [17], [18], and clustering [19] problems in different types of networks.

A. CONTRIBUTION OF THE STUDY

In this article, an energy-efficient bio-inspired localization (BIL) algorithm, an energy-efficient bio-inspired clustering (BIC) algorithm, and a GWO-based compressive sensing (CS-GWO) algorithm are proposed for application in UAV networks to detect wildfires and for post-fire monitoring in urban areas. The primary contributions of our study are summarized as follows:

- We develop a hybrid GWO (HGWO) method and an energy-efficient, range-free distributed localization algorithm based on HGWO for UAV networks, which is called BIL. The bounding cube method is applied to improve the sampling efficiency by reducing the initial search space. The improved sampling efficiency significantly reduces the computational cost, and rapid convergence is achieved by using the boundary cube technique owing to the reduced initial search space. In the proposed algorithm, a bounding cube with hop count is used to reduce the distance estimation errors.
- The proposed BIL algorithm follows an optimization method that incorporates the essence of HGWO to detect the optimal position of the target UAV node. In the HGWO model, the leader wolf dynamically estimates the location of the prey, and each wolf moves toward the estimated location of the prey. According to the simulation results, our BIL algorithm has higher localization accuracy, lower computational cost, faster convergence, and reduced energy consumption compared to existing algorithms.
- We propose a clustering algorithm called BIC based on the HGWO to prolong the network lifetime. In dynamic UAV networks with high mobility, the cluster size and the number of clusters considerably affect the communication performance. We also propose an analytical model for determining the optimal number of clusters in a network. According to our evaluation, the proposed BIC can minimize the number of transmissions.
- In the proposed BIC, a novel approach is proposed to select the CH based on the fitness value of the UAV. A node with higher residual energy, additional neighbors, and shorter intra-cluster distance is selected as the CH node, which is in charge of data aggregation and transmission.
- We also propose the CS-GWO algorithm, which is employed in BIC to transmit data. Within a cluster, cluster members (CMs) transmit data to their CH without using CS-GWO. Subsequently, the CH uses CS-GWO routing to transmit the data from the CH to the BS. The proposed CS-GWO constructs a minimum-cost routing tree from the cluster head (CH) to the BS, thereby reducing the routing delay and the number of transmissions.

The simulation results show that the proposed scheme outperforms conventional protocols under various scenarios, indicating that BIC reduces the energy consumption of UAVs and improves the data transmission efficiency and network reliability.

- We conduct extensive simulations to evaluate the performance of the proposed BIC and CS-GWO under different scenarios. Thereafter, the performance is compared with that of existing clustering protocols for UAV networks. The simulation results show that the proposed scheme outperforms the conventional protocols under various scenarios; this indicates that BIC reduces the energy consumption of UAVs and improves data transmission efficiency and network reliability.

B. OUTLINE OF THE PAPER

The remainder of this article is organized as follows. The next section provides a summary of existing related studies. In Section III, the preliminaries including the motivation scenario, assumptions, and GWO algorithm are introduced. In Section IV, the proposed BIL algorithm is described along with the network model, energy consumption model, sampling region, building of the bounding cube, cooperative distance estimation model, and GWO-based location optimization. In Section V, the proposed BIC algorithm is presented with respect to the system model, channel modeling, energy modeling of CHs, optimal number of clusters, clustering algorithm, CH selection algorithm, and transmission-efficient routing. In Section VI, the performance evaluation of BIL, BIC, and CS-GWO via computer simulations is discussed and a comparison of their performances to those of conventional schemes is presented. Finally, the conclusions are provided in Section VII.

II. REVIEW OF RELATED STUDIES

Currently, owing to the availability of UAVs, UAV networks have attracted the attention of academic and industrial researchers. Recently, UAVs or drones have been widely used for wildlife applications [20]. Node localization is an essential requirement for UAV net-works due to their high mobility. In geographical-position-based clustering/routing, node location information is necessary to determine the next-hop relay node. The method via which nodes obtain their position is known as localization. For localization, nodes are classified into two types of anchor nodes and unknown nodes. Anchor nodes are awarded their own position, and unknown nodes need to determine their positions by using the localization algorithm. However, using GPS is an easy and simple technique to localize UAV nodes.

Localization algorithms are roughly classified into range-based and range-free localizations. Range-based localization algorithms measure the distance between the anchor and unknown nodes based on actual distance or angel information. The range-based localization requires additional hardware to measure distance and angel information. However, range-free localization algorithms estimate the distance based on node density in the network, hop count between nodes, and node connectivity. Thus, range-free localization is economical and does not require additional hardware; therefore, it has attracted attention in recent decades. In the after-math of a calamity, existing Internet providers might often be unavailable due to infrastructural damages.

Trotta *et al.* [21] proposed a GPS-free flocking model for UAV networks in disaster recovery scenarios. The flocking model intended to support post-disaster rescue operations in perilous situations where UAV nodes cannot depend on GPS. In GPS-denied environments such as urban areas and battlefields (enemy-controlled airspace), indoor environment, and mountain/forest area, accurate or meaningful location information is hard to obtain. Russell *et al.* [22] proposed a cooperative localization technique, where they considered GPS-denied situations as urban areas and battlefields (enemy-controlled airspace). She *et al.* [23] also proposed a relative localization for multi-UAV networks in GPS-denied environments. The authors in [24] considered GPS-refused situations in UAV networks due to net-work jamming and GPS spoofing.

The distance vector hop (DV-Hop) [25] is a popular localization algorithm that estimates the distance of unknown nodes based on the anchor nodes and hop count between the nodes. However, the location accuracy of DV-Hop is affected by estimation errors originating from inaccurate measurement methods. Recently, bio-inspired algorithms have been used to solve the localization problem, as they are highly accurate and less complex. Several bio-inspired-based localization algorithms have been proposed in previous studies [26]–[30]. Rajakumar *et al.* [26] utilized the GWO algorithm to address the localization problem and achieved high localization accuracies in locating unknown nodes. The output proves that the GWO-based localization (GWO-LPWSN) performs better than the particle swarm optimization (PSO) and modified bat algorithm, owing to its lower computational cost and higher success rate. Sharma and Kumar [27] pro-posed a three-dimensional (3D), range-free localization algorithm based on a genetic algorithm (GA) with improved DV-Hop, called 3D-GAIDV-Hop. The localization accuracy was improved by applying a generic algorithm, and the bounding box facilitated the accurate location of the target node. The simulation output shows that 3D-GAIDV-Hop performs better in terms of localization accuracy and position coverage compared to 3D-DV-Hop, 3D-GADV-Hop, and 3D-PSODV-Hop.

Cui *et al.* [28] used a differential evolution (DE) algorithm with an improved DV-Hop-based localization algorithm, called DECHDV-Hop, to localize unknown nodes. DECHDV-Hop comprises two phases. First, the unknown distance is estimated based on the location information of anchor nodes and hop counts. Second, the DE is applied instead of multilateration to estimate the location of unknown nodes. The simulation results demonstrate that DECHDV-Hop performs significantly better than DV-Hop, GADV-Hop, and PSO-DV-Hop un-der different conditions.

Raguraman *et al.* [29] used a hybrid-dimensionality-based PSO called HDPSO to solve localization errors. A dimensionally based estimation method was applied in HDPSO to achieve rapid convergence. The simulation results demonstrate that the HDPSO performs better than PSO and hybrid PSO in terms of localization error and computational cost. Swarm–intelligence-based 3D localization (SIL) was proposed by Arafat and Moh [30] based on the modified PSO algorithm. The SIL algorithm employed a grouping method to achieve rapid convergence. The bounding box method was used to reduce the initial search space of the particles.

Jiang *et al.* [31] proposed a 3D mobile localization algorithm by using distance-only measurements for GPS-denied agents based on semidefinite optimization technique. Liu *et al.* [32] proposed a framework for developing a distributed 3D UAV-relative localization in UAV networks. Node-relative position is estimated by constrained optimization, and the performance of relative localization is improved by utilizing the Cramer-Rao bound method. Guo *et al.* [33] proposed a localization technique by the combination of distributed cooperative localization and distance-based formation control for UAV swarms in infrastructure-less GPS-free environments. Chen *et al.* [34] proposed a distributed and collaboration-based localization technique for multi-UAV swarms to address the low-accuracy of GPS signals. In [34], the authors utilized multidimensional scaling technique to get the relative coordinates of UAVs with GPS signals.

Clustering is used to efficiently manage networks. Cluster-based networks are more scalable, stable, and energy-efficient than flat networks. Node mobility makes the design of a clustering protocol challenging due to higher packet loss. Lee and Teng [35] proposed an adaptive hierarchy clustering algorithm for mobile sensor networks. The algorithm minimizes sensor energy consumption as well as packet loss in the network. Alami and Najid [36] utilized fuzzy logic to design an energy-efficient routing protocol. In [37], the authors proposed a clustering algorithm for homogeneous and heterogeneous networks by considering redundant data collection mechanism for overlapping and neighboring sensor nodes. The approach prolongs the network lifetime. Recently, UAV is utilized in sensor networks to collect the data. In our research, we focus on clustering and routing in the UAV-to-UAV communication rather than the UAV-to-ground communication. In the traditional systems, sensor nodes transmit the data to BS by multi-hop communication. In [38] and [39], UAV collect the data from the sensor nodes in an energy-efficient way, which reduce the energy consumption and prolong network lifetime.

Cluster-based routing protocols for UAV networks have been surveyed and compared in [40]. In the last two decades, several clustering protocols have been proposed. Shi and Luo [41] presented a weight-based mechanism for clustering, named cluster-based location-aided dynamic source routing (CBLDSR). In CBLDSR, the CH is elected based on node energy level, low relative speed, and large number of neighboring nodes. Zang and Zang [42] proposed a mobility prediction clustering algorithm for UAV net-works, which is called MPCA, where the link expiration time is combined with directional structure prediction. Recently, BIC mechanisms have been widely employed in UAV networks.

Aadil *et al.* [43] presented an energy-aware cluster-based routing (EALC) for UAV networks. The EALC addresses the energy limitation of UAVs and the inefficient routing in UAV networks. Khan *et al.* [44] proposed a BIC scheme for FANETs, called BICSF. The BICSF is a hybrid clustering protocol, which is a combination of krill herd (KH) and glowworm swarm optimization (GSO). The node with higher residual energy is elected as the CH, and the remaining nodes are considered as CMs. The KH algorithm is used for predicting cluster management. Khan *et al.* [45] proposed a self-organized-based clustering scheme (SOCS) for FANET. SOCS is based on GSO; the GSO algorithm is used for the cluster formation and management phases. The residual energy and luciferin value of the UAVs are considered for the CH election phase.

Bhandari *et al.* [46] proposed a location-aware *k*-means UAV clustering algorithm by considering the UAV mobility and relative locations. Their location-aware clustering scheme enhances the performance of UAV networks with limited resources. Jin *et al.* [47] proposed an optimization algorithm for an efficient and reliable UAV cluster-based video surveillance system in the heterogeneous communication of smart cities. You *et al.* [48] proposed a clustering scheme for UAV networks, which jointly optimizes UAV area coverage and mobile edge computing. The authors in [48] formulate jointly optimizing problem to maximize the coverage efficiency in UAV networks with lower delay and power constraints by considering the clustering, positions, and transmission power. Duan *et al.* [49] proposed a flexible network architecture and dynamic scheduling algorithm to address the fault tolerance in highly dynamic UAV networks.

Energy-efficient mechanism is widely used in ad-hoc networks and mobile edge networks. Zhou and Hu [50] proposed a framework to address the computation efficiency in wireless powered mobile edge networks. In [50], the authors considered time division multiple access and non-orthogonal multiple access to evaluate offloading. Zhang *et al.* [51] proposed an energy-efficient computation for UAV-aided mobile edge computing. In [51], the authors jointly considered transmission power of users and UAV trajectory to maximize computation efficiency. In [52], the authors proposed an energy-efficient computing for multi-UAV-aided mobile edge networks, in which the authors jointly optimized user association, spectral power and resource, and UAV trajectory scheduling. In [53], the authors studied a solar-powered UAV communication system to address the UAV optimal trajectory and resource allocation. In [54], the authors proposed a secure and energy-efficient UAV communication system based on multiple antennas.

III. PRELIMINARIES

In this section, the scenario and assumptions are de-scribed as the preliminaries of our study.

A. SCENARIO

This study assumes the typical UAV network model shown in Fig. 1. In the scenario, we assumed that communication facilities are unavailable in remote wild areas, where formal communication infrastructures do not exist. In this model, we demonstrated the use of a multi-UAV network for wildfire detection and monitoring.

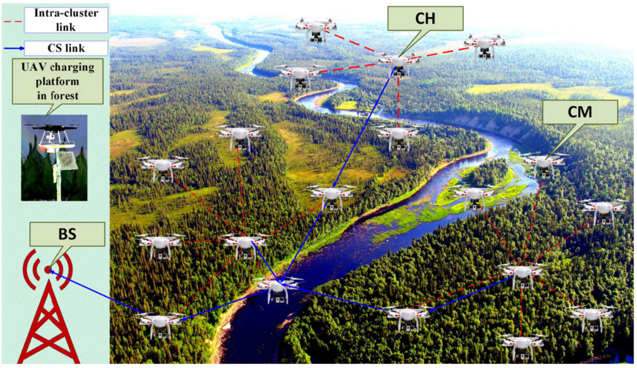

FIGURE 1. UAV-aided wildfire detection and monitoring: UAVs are organized into clusters. Data from cluster member UAVs are transferred to the BS using transmission-efficient routing.

Our proposed system utilizes the rapid mobility of UAVs with the advantage of providing an on-demand monitoring service that is faster than the current manned aircraft, remotely controlled UAV, and satellite imagery. The proposed system minimizes human intervention in risky wildfire areas. The purpose of this study is to simulate a UAV network model for wild areas. The UAVs are equipped with geo-tagged cameras. They scout fire locations by using cameras employing thermal imaging technology and assist the BS for further analyses. The UAVs can constantly monitor forest areas and observe variations in temperature to help prevent wildfires. Light detection and ranging–equipped UAVs have been widely used for fire prevention in previous studies [55]. The UAVs can be placed in standby mode in the BS or on weather-resistant charging pads situated across a forest. The UAVs can move around the forest and land automatically and periodically on pre-installed platforms for charging.

B. ASSUMPTIONS

We considered a set of rotary mini-drones or medium-sized drones. The speed of the UAVs is approximately 10–30 m/s. In our proposed model, few UAVs have GPS modules, but all UAVs have inertial measurement units that are provided for the motion sensing of the UAV. The UAVs are equipped with a wireless communication interface. The proposed system is based on offline and online phases, as illustrated in Fig. 2. The offline phase is the localization phase, where the localization algorithm is implemented to estimate the actual position of unknown UAV nodes. The offline phase provides an overview of the overall network topology. After localizing the node, the online phase is applied. The online phase contains two subsystems: clustering and data transmission. In the clustering process, all nodes are able to recognize their neighboring nodes, and a CH is elected for each of the defined clusters. The data transmission subphase is used for node communication and data aggregation between CMs, CHs, and BS. Details regarding each phase are provided in the following sections.

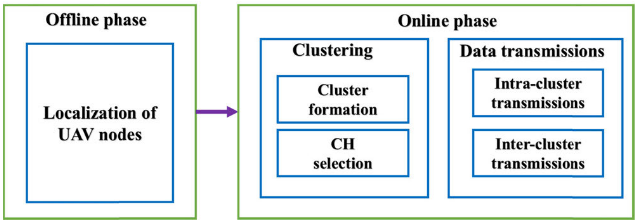

FIGURE 2. Block diagram of the proposed system.

C. GWO MODEL

The GWO model is a novel bio-inspired optimization algorithm proposed by Mirjalili *et al.* [56], which mainly mimics the leadership and hunting process of grey wolves in nature. By using this model, quick seeking speed and high–precision search can be achieved easily. The GWO algorithm has numerous applications, as it can be easily combined with particle engineering problems. Generally, the hunting and searching behaviors of animals are used to mimic swarm intelligence algorithms. However, the GWO simulates the internal leadership hierarchy between wolves. Therefore, during the searching process, the optimal solution can be comprehensively assessed and selected from three solutions, whereas other bio-inspired algorithms only provide a single solution. Therefore, GWO can significantly shrink the probability of premature and falling into the local optimum.

In the leadership hierarchy, wolf colonies hire four sorts of grey wolves of alpha, beta, delta, and omega. The alpha wolf is the dominant leader and the decision maker to manage the pack. This shows that discipline and organization in the pack are more important than strength. Beta is the second level leader in GWO hierarchy. The beta wolves help alphas in decision making, and lower-level wolves follow their orders. The beta replaces the alpha if the alpha passes away. The omega is a lower-level wolf. The omegas must follow the orders of the alpha and beta wolves. If the wolf is not an alpha, beta, or omega, it is called a subordinate wolf, i.e., delta. The delta wolves have to follow alphas and betas. The grey wolves also exhibit another interesting social behavior known as group hunting. The main stages of group hunting are as follows: searching for prey, encircling the prey, and attacking the prey. In this study, localization, clustering, CH selection, and data transmission are inspired by the structure and hierarchies of grey wolf packs. The similarity between the grey wolf pack and the UAV network are presented in Table 1.

TABLE 1. Similarity between grey wolf pack and UAV network.

Gray wolf packs	UAV networks
Grey wolf	UAV nodes
Food placement or prey	Objective function
The position of the prey in the area is not known by grey wolves.	Nodes do not know the cost function. They detect the changes in the values of the cost function.
Best value of the fitness function	Best node known as CH
The average number of wolves per group is between 8 and 12.	The ideal number of nodes per cluster is between 5 and 12.
Position of the wolves	Position of the UAVs (x, y, z)

IV. ENERGY-EFFICIENT BIO-INSPIRED LOCALIZATION

In this section, the BIL algorithm for UAV networks is proposed and discussed. The network model, energy model of the proposed localization algorithm, and pro-posed HGWO-based localization algorithm are presented in detail.

A. NETWORK MODEL

This section describes the network model and parameters used in the BIL. N is the total number of UAV nodes and N_a is the total number of anchors. The total number of unknown nodes is $N_u = N - N_a$ and the anchor ratio is $A_r = N_a/N$. S_n is a set of neighbor nodes within the sensing range of a node, which is also known as the neighbor set of the node. The transmission range or radius of each UAV node is R (m), and D_v is the deployment region in the cube (m^3). We assumed that the proposed network model is self-organized and uniformly distributed, without a central control for deploying UAV nodes in the network. There are N UAV nodes, and N_a anchor nodes are uniformly distributed in the network. The anchor nodes are provided with their exact position information at all times.

B. LOCALIZATION ENERGY CONSUMPTION

For the energy consumption model, we used the popular energy model proposed by Heinzelman *et al.* [57]. For an unlocalized node, the transmitted and received energies are represented by E_{Tx} and E_{Rx}, respectively, for a data size Ψ and transmission distance d. That is,

$$E_{Tx} = \Psi \times E_s + \Psi \times d^2 \times E_a, \tag{1}$$

and

$$E_{Rx} = \Psi \times E_s, \tag{2}$$

where E_s and E_a represent the energy dispersed in the transmitter electronics circuitry and radio frequency amplifiers for propagation loss, respectively.

C. SAMPLING REGION

In our model, two UAV nodes p and q can directly communicate with each other when they are within a communication range R. Therefore, if p and q are in the communication range R, node q is a one-hop neighbor. If p and q are in the communication range $2R$, node q is a two-hop neighbor of p. All nodes move within the maximum speed V_{max}. Assume that a node moves from time $t-1$ to time t at speed V_{max}. If the anchor node broadcasts the scanning signals periodically, the period T is defined as

$$T = \frac{2R}{V_{max}}. \tag{3}$$

The process of estimating the location of unknown nodes involves three phases: initial, prediction, and filtering phases. Initially, there is no information regarding the location of the target node. At the initial phase, Q sample points in the given region are randomly selected, and the possible location sample set is $l_t^k = \left\{l_t^0, l_t^1, \ldots, l_t^{Q-1}\right\}$, where l_t^k represents the possible location of the target node at time t and $k \in \{0, 1, \ldots, Q-1\}$.

Let R_{max} be the maximum moving distance of the target node at V_{max} speed and $d(l_t|l_{t-1})$ is the Euclidean distance between the two consecutive locations l_t and l_{t-1} of the target node at time t and $t-1$, respectively. Hence, the target node generates location samples through the following probability function:

$$p\left(l_t|l_{t-1}\right) = \begin{cases} {}^{1}\!/_{\pi R_{max}^2} & if\ d\left(l_t|l_{t-1}\right) < R_{max} \\ 0 & if\ d\left(l_t|l_{t-1}\right) \geq R_{max}, \end{cases} \tag{4}$$

where $p(l_t|l_{t-1})$ is the probability that the location prediction of the target node is correct. Based on the previous position, the probability $p\left(l_t|l_{t-1}\right)$ is used to predict the current location of the target node.

The target node filters the sampling point according to the information obtained from the one-hop and two-hop anchor nodes of the target node. For anchor node a, sampling particle l, one-hop anchor nodes set S_{a1}, and two-hop anchor nodes set S_{a2}, the filter condition of sampling particle l at time t can be expressed as

$$filter\left(l_t^i\right) = \left[\forall_a \in S_{a1}, d\left(l_t^i, a\right) \leq R\right] \cap \left[\forall_a \in S_{a2}, R < d\left(l_t^i, a\right) \leq 2R\right]. \tag{5}$$

D. BUILDING THE BOUNDING CUBE

Assume that p is a random unknown node; the neighbor nodes of node p have positions denoted as $A_{k_1}, \ldots, A_{k_m}$, for some $m > 0$. The coordinates of A_{k_i} are denoted as $A_{k_i} = [x_i, y_i, z_i]^T \in R^3$, where $(x_i, y_i, z_i) \in \{1, 2, \ldots, n\}$. For integers $1 \leq a \leq b \leq n$, $1 \leq c \leq d \leq n$, and $1 \leq e \leq f \leq n$, the cubic region $[a, b] \times [c, d] \times [e, f] \in D_v$ denotes the coordination of (i, j, k), where $a \leq i \leq b$, $c \leq j \leq d$, and $e \leq k \leq f$. The bounding cube zone (BC_i) for (x_i, y_i, z_i) coordinates is computed as

$$BC_i = [x_i-R, x_i+R] \times [y_i-R, y_i+R] \times [z_i-R, z_i+R]. \tag{6}$$

Equation (6) represents the communication region of anchor A_{k_i}. Then, for a random unknown node $p \in BC_i$, for all

$1 \le i \le m$, and therefore,

$$p \in D_v \cap \bigcap_{i=1}^{m} BC_i. \tag{7}$$

Equation (7) can be constructed for (x_i, y_i, z_i) as follows:

$$\begin{aligned} BC_i \cap BC_j \cap BC_k \\ &= \left[\max\left(x_i, x_j, x_k\right) - R, \min\left(x_i, x_j, x_k\right) + R\right] \\ &\quad \times \left[\max\left(y_i, y_j, y_k\right) - R, \min\left(y_i, y_j, y_k\right) + R\right] \\ &\quad \times [\max\left(z_i, z_j, z_k\right) - R, \min\left(z_i, z_j, z_k\right) + R]. \end{aligned} \tag{8}$$

Thus,

$$\begin{aligned} p \in D_v \cap [x_+ - R, x_- + R] \times [y_+ - R, y_- + R] \\ \times [z_+ - R, z_- + R], \end{aligned} \tag{9}$$

where $x_+ = \max(x_1, \ldots, x_m)$, and $x_- = \min(x_1, \ldots, x_m)$, and similarly for (y_i, z_i)'s. For $D_v = [1, n] \times [1, n] \times [1, n]$, the estimated position p is expressed as

$$\begin{aligned} p \in &[\max(x_+ - R, 1), \min(x_- + R, n)] \\ &\times [\max(y_+ - R, 1), \min(y_- + R, n)] \\ &\times [\max(z_+ - R, 1), \min(z_- + R, n)]. \end{aligned} \tag{10}$$

The communication region of node p for the coordinates of (x, y, z) can be represented as

$$\begin{aligned} BC\,(x, y, z) = [x - R, x + R] \times [y - R, y + R] \\ \times [z - R, z + R] \cap D_v. \end{aligned} \tag{11}$$

From (10) and (11),

$$\begin{aligned} BC\,(x, y, z) = &[\max(x - R, 1), \min(x + R, n)] \\ &\times [\max(y - R, 1), \min(y + R, n)] \\ &\times [\max(z - R, 1), \min(z + R, n)]. \end{aligned} \tag{12}$$

Assuming that a UAV node has n number of one-hop anchors, the bounding box can be built as

$$x_{min} = max_{i=1}^{n}\left\{x_i - R\right\}, \quad x_{max} = min_{i=1}^{n}\left\{x_i + R\right\}, \tag{13}$$

$$y_{min} = max_{i=1}^{n}\left\{y_i - R\right\}, \quad y_{max} = min_{i=1}^{n}\left\{y_i + R\right\}, \tag{14}$$

and

$$z_{min} = max_{i=1}^{n}\left\{z_i - R\right\}, \quad z_{max} = min_{i=1}^{n}\left\{z_i + R\right\}, \tag{15}$$

where (x_i, y_i, z_i) are the coordinates of the i-th anchor node, x_{min} and x_{max} denote the upper bound and lower bound of x direction, respectively, and y_{min}, y_{max}, z_{min} and z_{max} denote those for y and z coordinates. For two-hop anchor nodes, the size is reduced by replacing R with $2R$ in (13)–(15). The position of the previous time and the maximum speed of the node affect the prediction position.

Assume that B_l and B_u are the lower and upper boundaries of N_u, respectively. The feasible region of N_u is defined as FR_u. As depicted in Fig. 3a, the overlapping space (intersection) between the bounding cube FR_u consists of the maximum of the low coordinates and the minimum of the high coordinates of the bounding cube and is expressed as

$$FR_u = \bigcap_{i=1}^{m} BC_i = [B_l, B_u], \tag{16}$$

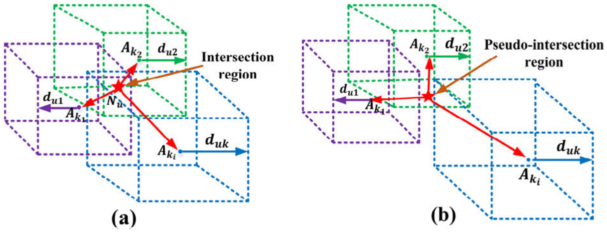

FIGURE 3. Bounding cube method for UAV localization: (a) intersection case and (b) pseudo-intersection case.

Here, $B_l = [\max(x_i - d_i), \max(y_i - d_i), \max(z_i - d_i)]^T$ and $B_u = [\min(x_i + d_i), \min(y_i + d_i), \min(z_i + d_i)]^T$, where d_i is the estimate of the distance from the unknown node to anchor A_{k_i}. During the pseudo-intersection, there should be no intersection between all bounding cubes, as presented in Fig. 3(b).

Assume that $B_l = [x_l, y_l, z_l]^T$ and $B_u = [x_u, y_u, z_u]^T$ are the lower and upper bounds of the deployed region, respectively. The bounding cube for the pseudo-intersection obeys the following conditions:

$$[B_l, B_u] = \begin{cases} if\,\{x_l < x_i - d_i\}, & set\{x_l = x_i - d_i\} \\ if\,\{x_u > x_i + d_i\}, & set\{x_u = x_i + d_i\}. \end{cases} \tag{17}$$

Similarly, for the y and z coordinates, B_l and B_u are denoted as $\{y_l < y_i - d_i, z_l < z_i - d_i\}$ and $\{y_u > y_i + d_i, z_l > z_i + d_i\}$, respectively. For $[\{x_u < x_l\}, \{y_u < y_l\}, \{z_u < z_l\}]$, the values of $[x_l, x_u]$, $[y_l, y_u]$, $[z_l, z_u]$ are changed.

E. COOPERATIVE DISTANCE ESTIMATION

Assume that the coordinates of the estimated and real positions of the unknown node are defined as (x_{ei}, y_{ei}, z_{ei}) and (x_{ri}, y_{ri}, z_{ri}), respectively. The localization error is defined as the distance between the real and estimated locations of the unknown node.

$$e_{ui} = \sqrt{(x_{ei} - x_{ri})^2 + (y_{ei} - y_{ri})^2 + (z_{ei} - z_{ri})^2}. \tag{18}$$

The position estimation of a particular unknown node can be formulated as an optimization problem. This optimization problem can be solved by minimizing the objective function. For a noisy measure distance $(\widehat{d_{ij}})$ between an unknown node N_i and anchor node A_j, the objective function of the localization error can be expressed as

$$f\,(x_{ei}, y_{ei}, z_{ei}) = \frac{1}{K} \sum_{k=1}^{K} \left(d_{ij} - \widehat{d_{ij}}\right)^2, \tag{19}$$

where K is the number of neighboring nodes and d_{ij} can be expressed as

$$d_{ij} = \sqrt{(x_{ei} - x_{aj})^2 + (y_{ei} - y_{aj})^2 + (z_{ei} - z_{aj})^2}. \tag{20}$$

The multihop algorithm is economical and does not require distance measurement devices. As displayed in Fig. 4, N_t is the target node, whose coordinates are (x_t, y_t, z_t). A_{k_1}, A_{k_2}, and A_{k_3} are three anchor nodes that are the neighbors of

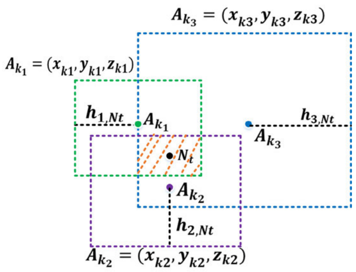
$A_{k_3} = (x_{k3}, y_{k3}, z_{k3})$
$A_{k_1} = (x_{k1}, y_{k1}, z_{k1})$
$h_{1,Nt}$
A_{k_1}
$h_{3,Nt}$
A_{k_3}
N_t
A_{k_2}
$h_{2,Nt}$
$A_{k_2} = (x_{k2}, y_{k2}, z_{k2})$

FIGURE 4. Feasible region with hop count in UAV localization.

N_t. The coordinates of A_{k_1},A_{k_2}, and A_{k_3} are (x_{k1}, y_{k1}, z_{k1}), (x_{k2}, y_{k2}, z_{k2}), and (x_{k3}, y_{k3}, z_{k3}), respectively. The maximum hop counts for the three anchor nodes are $h_{1,Nt}$, $h_{2,Nt}$, and $h_{3,Nt}$ respectively. The feasible region for N_t is shown in the inner cube of Fig. 4.

The feasible region is constructed with a radius of $R \times h_{i,Nt}$, where $h_{i,Nt}$ is the hop count from a particular anchor node $(i = 1, \ldots, 3)$. The feasible region can be bounded by bounding cubes, which are defined as follows:

$$\begin{aligned} x_{max} &= max_{i=1}^{n} \left\{x_i - R \times h_{i,Nt}\right\} \leq x_t \leq x_{max} \\ &= min_{i=1}^{n} \left\{x_i + R \times h_{i,Nt}\right\}, \end{aligned} \tag{21}$$

$$\begin{aligned} y_{max} &= max_{i=1}^{n} \left\{y_i - R \times h_{i,Nt}\right\} \leq y_t \leq y_{max} \\ &= min_{i=1}^{n} \left\{y_i + R \times h_{i,Nt}\right\}, \end{aligned} \tag{22}$$

and

$$\begin{aligned} z_{max} &= max_{i=1}^{n} \left\{z_i - R \times h_{i,Nt}\right\} \leq z_t \leq z_{max} \\ &= min_{i=1}^{n} \left\{z_i + R \times h_{i,Nt}\right\}. \end{aligned} \tag{23}$$

Localization is an iterative procedure. An unknown node has three anchor nodes that are first localized. The localized unknown node helps anchors to localize other unknown nodes. This process is repeated until all unknown nodes are localized. The initial search space of the network is fixed using Algorithm 1.

The anchor node broadcasts its locations. Then, the bounding cube method is applied to improve sampling efficiency by reducing the initial search space. An improved sampling efficiency significantly reduces computational cost, and fast convergence is achieved by using the boundary cube technique due to the reduced initial search space. In Algorithm 1, the unknown node has three anchor nodes that have intersections between each other and are localized by lines 3 to 8. The feasible regions are obtained by using the bounding cube method. In the case of pseudo-intersections (no intersections between the anchors), the search space is defined by lines 9 to 12. Finally, the bounding cube with hop count is used to reduce distance-estimation errors. In Step 4, we used hop counting for localizing the remaining unknown nodes. If the unknown node has three neighbor nodes, the search space of the target node is defined in lines 16–21.

Algorithm 1 Find the Search Space of a Network

```
Input: Unknown node $N_u$, anchor node $N_a$, transmission range $R$, deployed region $D_v$, and node maximum velocity $V_{max}$
Output: Search space
Procedure
1: Step 1: Initialize $N_i$, $N_a$, $R$, and $V_{max}$.
2: Step 2: Anchor node broadcasts its location information.
3: Step 3: Measure the distance between neighbor nodes using (19).
4: for all unknown UAV nodes $N_u$ do
5:     if an unknown node has $K > 3$ neighbor anchors, then
6:         if anchors have an intersection between each other, then
7:                 Obtain the search space using (8)
8:                 Call localization Algorithm 2
9:         else no intersections (pseudo-intersection) between anchor nodes
10:            Obtain the search space using (16)–(17)
11:            Call localization Algorithm 2
12:        end if
13:    end if
14: end for
15: Step 4: If an unknown node is not localized, then
16:            if an unknown node has $K > 3$ neighbor anchors, then
17:                Obtain the search space of each target node according to (21)–(23)
18:                Call localization Algorithm 2
19:            end if
20:        end if
21: Until no unknown nodes can be localized
22: end procedure
```

F. BIO-INSPIRED LOCALIZATION ESTIMATION

The GWO model is a novel bio-inspired optimization algorithm. The proposed localization is based on GWO. The main inspiration of the GWO algorithm is derived from the strict social hierarchy and social collaborative behavior of gray wolves during hunting. The population of gray wolves divided into subgroups during hunting is referred to as grades. In GWO, every gray wolf (alpha (α), beta (β), delta (δ), and omega (ω)) represents a role in the optimization process.

To mathematically model the social grade of wolves, we considered the fittest solutions when designing the GWO. The best solution is considered as an α wolf, and this is the outcome of the optimization process. Consequently, the second, third, and fourth fittest solutions are referred to as β, δ, and ω, respectively. The hunting behavior of gray wolves involves three steps: finding, encircling, and attacking the prey. As this predatory approach is very effective, the mathematical model of the GWO is developed as follows:

The population of $\mathcal{W}$ wolves is denoted as $\mathbb{U} = [\mathbb{U}_1, \ldots, \mathbb{U}_w, \ldots, \mathbb{U}_W]$. The position of the wth wolf in dimensional solution space d can be represented as $\mathbb{U}_w = \left[\mathbb{U}_w^1, \ldots, \mathbb{U}_w^d, \ldots, \mathbb{U}_w^D\right]^T$. The predatory process of the gray wolf behavior can be defined as

$$\mathbb{U}_w^d(t+1) = \mathbb{U}_p^d(t) - A_w^d \left|C_w^d \cdot \mathbb{U}_p^d(t) - \mathbb{U}_w^d(t)\right|, \tag{24}$$

where t is the current iteration, $\mathbb{U}_p^d$ is the position of the prey in dimension d, and $A_w^d \left|C_w^d \cdot \mathbb{U}_p^d(t) - \mathbb{U}_w^d(t)\right|$ is the size of the encirclement. A_w^d and C_w^d are coefficient vectors represented by

$$A_w^d = 2e \cdot r_1 - e \tag{25}$$

and

$$C_w^d = 2e \cdot r_2, \tag{26}$$

where r_1 and r_2 are random numbers between [0, 1], and variables e (decreasing linearly from 2 to 0) are the functions of the iteration steps:

$$e = 2 - \frac{t}{t_{max}}, \tag{27}$$

where t_{max} is the maximum iteration. From (24)–(27), the exploitation and exploration can be concurrently confirmed, which are known as local search and global search, respectively.

In the prediction phase, gray wolves usually know the position of the prey $\mathbb{U}_p = \left[\mathbb{U}_p^1, \ldots, \mathbb{U}_p^d, \ldots, \mathbb{U}_P^D\right]^T$. However, the desired position is unknown to us throughout the optimization process. Generally, $\mathbb{U}_\alpha$, $\mathbb{U}_\beta$, and $\mathbb{U}_\delta$ are near the potential position of the prey (α, β, and δ). For every iteration, assume that the desired position can be updated based on these positions:

$$\mathbb{U}_{w,\alpha}^d(t+1) = \mathbb{U}_\alpha^d(t) - A_{w,\alpha}^d \left|C_{w,\alpha}^d \cdot \mathbb{U}_\alpha^d(t) - \mathbb{U}_w^d(t)\right|, \tag{28}$$

$$\mathbb{U}_{w,\beta}^d(t+1) = \mathbb{U}_\beta^d(t) - A_{w,\beta}^d \left|C_{w,\beta}^d \cdot \mathbb{U}_\beta^d(t) - \mathbb{U}_w^d(t)\right|, \tag{29}$$

$$\mathbb{U}_{w,\delta}^d(t+1) = \mathbb{U}_\delta^d(t) - A_{w,\delta}^d \left|C_{w,\delta}^d \cdot \mathbb{U}_\delta^d(t) - \mathbb{U}_w^d(t)\right|, \tag{30}$$

$$\mathbb{U}_w^d(t+1) = \frac{1}{k} \sum\nolimits_{k\in\alpha,\beta,\delta} \mathbb{U}_{w,k}^d(t+1), \tag{31}$$

where k represents the three wolves α, β, and δ, and $k = 3$.

The conventional GWO searching technique is based on group communication and social learning. However, in conventional GWO, the individual experiences of wolves are ignored, and only the global best position is considered. To explore the best position, the personal best historical positions are considered. In our proposed HGWO, a wolf individual memory is added. The PSO algorithm is as follows:

$$\mathbb{U}_w^d(t+1) = \eta_1 \cdot \frac{1}{k} \sum\nolimits_{k\in\alpha,\beta,\delta} \mathbb{U}_{w,k}^d(t+1) + \eta_2 \cdot r_3 \cdot \frac{1}{k} \sum\nolimits_{k\in\alpha,\beta,\delta} (\mathbb{U}_{w,best}^d - \mathbb{U}_{w,k}^d(t)), \tag{32}$$

where η_1 and η_2 denote the social and individual learning factors of gray wolves, respectively, r_3 is a random value with range [0, 1], and $\mathbb{U}_{w,best}^d$ represents the personal historical best solution of the wth wolf. In (32), the update of the position of the wolf is inspired by the PSO algorithm.

In this step, the "greedy choice" approach of DE is applied to the selection operation. After the mutation operation, the crossover operation is performed by the individual wolf $\mathbb{U}_w^d(t+1)$, and the completely new position $\mathbb{U}_{w,new}^d(t+1)$ can be expressed as

$$\mathbb{U}_w^d(t+1) = \begin{cases} \mathbb{U}_w(t), & f\left(\mathbb{U}_{w,new}^d(t+1)\right) > f(\mathbb{U}_w(t)) \\ & and\ r_4 < F_\chi \\ \mathbb{U}_{w,new}^d(t+1), & \\ & othewise, \end{cases} \tag{33}$$

where $\mathbb{U}_w(t)$ is the position of the last iteration, $\mathbb{U}_{w,new}^d(t+1)$ represents the new position, and r_4 is a random value with range [0, 1]. In the DE algorithm, the principle of existence of the fittest value is implemented with a probability of F_χ. Equation (33) expresses that if the new position of the wolf based on (32) is worse than that of the last generation, it will be abandoned. The results converge to the optimal solution when the greedy method is used. In addition, the position of the wolf is refined, thereby reducing the possibility of FA in the position estimation. The step-by-step process of HGWO is presented in Algorithm 2.

According to Algorithm 2, the HGWO algorithm initializes the related parameters in lines 1–3. In HGWO, the first position of the UAV node is measured based on the Euclidian distance in lines 4–7. After that, grey wolf fitness value is utilized to determine the next UAV position in lines 8–22. Next, the algorithm reiterates the proposed hunting operations of the gray wolf in lines 8–22 until the maximum iteration is reached or the other ending criterion is fulfilled. The proposed algorithm searches for the positions of $\mathbb{U}_\alpha$, $\mathbb{U}_\beta$, and $\mathbb{U}_\delta$. In HGWO, we applied PSO to the individual memories of each search agent. In the second phase, we utilized PSO to calculate each wolf fitness value and compared it with the previous fitness value in line 15. The experience of each search agent was used to determine the optimal position. Finally, we applied DE algorithm to check the optimality of the previous position in line 16. The DE algorithm redefines the position and eliminates the FA error.

V. ENERGY-EFFICIENT BIO-INSPIRED CLUSTERING

Clustering is essential in UAV networks to efficiently manage the network topology. In this section, we mathematically model the proposed clustering scheme. Furthermore, we analyze and describe the proposed clustering scheme, CH election, and data transmission phase.

A. SYSTEM MODEL

In the clustering scheme, multiple UAV groups are formed and deployed to the wild area [58]. Assume that a number

Algorithm 2 UAV Localization Based on HGWO

```
Input: Population size $\mathcal{W}$, maximum number of iterations ($t_{max}$), dimension ($d$), and coefficients$r_1$, $r_2$, $r_3$, and $r_4$.
Output: Optimal position of the unknown node
/* Initialization*/
Initialize the gray wolf pack $\mathbb{U} = [\mathbb{U}_1, \ldots, \mathbb{U}_w, \ldots, \mathbb{U}_W]$.
Initialize the GWO parameters ($e$, $A^d_w$, $C^d_w$).
Initialize the fitness value ($\mathbb{U}_\alpha$, $\mathbb{U}_\beta$, $\mathbb{U}_\delta$).
/* Computation*//*
while ($t \leq t_{max}$) do
    for each wolf $w = 1$: $W$ do
        Update the current search agent position using (19)
end for
/*HGWO loop*/
    for each wolf $w = 1$: $W$ do
        Evaluate the fitness value and update ($\mathbb{U}_\alpha$, $\mathbb{U}_\beta$, $\mathbb{U}_\delta$)
        Obtain the variable $e$ based on (27).
    end for
    for each wolf $w = 1$: $W$ do
        Calculate $A^d_w$ and $C^d_w$ based on (25)–26)
        Update the position of wolf $w$ by (28)–(31)
        Redefine the position of wolf $w$ by (32)
        Evaluate whether the new position of wolf $w$ is acceptable by (33)
    end for
$w = w + 1$
$t = t + 1$
end while
Terminate the process and output the optimal position by $\mathbb{U}_\alpha$
```

of UAVs U_k are deployed in the area. The set of clusters C_s is defined as $\{C^1_s, \ldots, C^M_s\}$, where M is the number of clusters. The set of UAVs is defined as $\mathcal{N} = \{U_1, U_2 \ldots, U_k\}$. The network graph is presented as $\mathcal{G}(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of vertices of U_k UAVs, and $\mathcal{E}$ represents the set of edges. Assume that two UAVs (U_i andU_j) are neighbors when (U_i, $U_j \in \mathcal{E}$) are contingent on a minimum threshold value of the signal-to-noise ratio (SNR) between U_i andU_j [59]. The 3D coordinates of U_i at time t are defined as (x^{uav}_i, y^{uav}_i, h^{uav}_i). The distance between two UAVs (U_i andU_j) is defined as follows:

$$d^{i,j}\left\{U_i(t), U_j(t)\right\} = \sqrt{(x^{uav}_i - x^{uav}_j)^2 + (y^{uav}_i - y^{uav}_j)^2 + (h^{uav}_i - h^{uav}_j)^2}. \quad (34)$$

B. CHANNEL MODEL

In this subsection, the channel model for the UAV is presented. First, we present the U2U channel model. When U_i transmits the signals toU_j, the received power at U_j from U_i can be expressed as in [60]:

$$P^{r}_{U_{i,j}} = P_{i,j} p^{g}_{i,j} d^{-\gamma}_{i,j}, \quad (35)$$

where $P_{i,j}$ is the transmitted power from U_i to U_j, $p^g_{i,j}$ is the power gain of the small-scale fading channel, $d_{i,j}$ is the distance between U_i from U_j, and γ represents the mean path loss exponent. The SNR value from U_i to U_j is expressed as

$$\mathcal{SNR}_{U_{i,j}} = \frac{P_{i,j} p^{g}_{i,j} d^{-\gamma}_{i,j}}{N^w_G}, \quad (36)$$

where N^w_G is the additive white Gaussian noise. The U2U channels are generally controlled by a line-of-sight (LoS) link. For this reason, the path loss between U_i to U_j can be considered as a free space propagation, which is presented as follows:

$$L^{U_{i,j}}_{LoS} = 20\log d_{i,j} + 20\log f_0 + 20\log({}^{4\pi}/_{c}), \quad (37)$$

where f_0 is the carrier frequency of the U2U channel, and c is the speed of light.

Second, we present the U2BS channel model. For the U2BS channel mode, we consider that communication links are either LoS or non-line-of-sight (NLoS), based on some probability. The probability parameters are the UAV's altitude and elevation angle between the UAV and the BS. Assuming that U_i has an altitude h_i and BS has a distance $r_{U_i,BS}$ from the estimated position of the UAV, the probability of LoS is given by

$$p^{LoS}_{U_i,BS}(r_{U_i,BS}) = \frac{1}{1 + \psi \cdot \exp\left(-\phi \frac{180}{\pi} arctan \frac{\sqrt{r^2_{U_i,BS} - d^2_{U_i,BS}}}{d_{U_i,BS}} - \psi\right)}, \quad (38)$$

where $d_{U_i,BS}$ is the distance between U_i and BS, and ψ and ϕ are the environment-dependent parameters. The path loss between U_i and BS can be written as

$$L_{U_i,BS}\left(r_{U_i,BS}, d_{U_i,BS}\right) = \left(\frac{4\pi f r_{U_i,BS}}{c}\right)^{-\varsigma} \{\varrho_{LoS} p^{LoS}_{U_i,BS}\left(r_{U_i,BS}\right) + \varrho_{NLoS}(1 - p^{LoS}_{U_i,BS}\left(r_{U_i,BS}\right))\}^{-1}, \quad (39)$$

where f is the carrier frequency, c is the speed of light, ς is the path loss exponent, and ϱ_{LoS} and ϱ_{NLoS} are the additional losses due to the LoS and NLoS links, respectively.

C. ENERGY CONSUMPTION OF CLUSTER HEADS

The node energy consumption is based on signal transmission and reception. The energy model used in this study is the same as that in [3]. The free-space multipath fading channel model, which is based on the distance between the transmitter and receiver is used. If the distance is d (Euclidean distance between the transmitter and receiver), and the distance threshold is d_{th} for transmitting l-bit data, the energy consumption is designated as

$$E_{Tx}(l,d) = \begin{cases} (E_{elec} + l \cdot \varepsilon f_s \cdot d^2) \times l, & d < d_{th} \\ \left(E_{elec} + l \cdot \varepsilon f_s \cdot d^4\right) \times l, & d \geq d_{th}, \end{cases} \tag{40}$$

where E_{elec} denotes the parameters of electronic energy consumption depending on the energy dissipated per bit to run the transmitter or the receiver. $\varepsilon f_s \cdot d^2$ and $\varepsilon f_s \cdot d^4$ are the amplifier energies under the two communication modes depending on the distance between the transmitter and receiver, respectively. The energy consumption for receiving l-bit messages can be calculated as

$$E_{Rx}(l) = E_{elec} \times l. \tag{41}$$

The energy consumption of UAVs is based on their flights and communications with other UAVs and the BS. The energy consumed during flight depends on the flying and motion, and it can be represented as

$$E_{fly} = \sqrt{\frac{(m_{tot}g)^3}{2\pi p_r^n r_p \rho}}, \tag{42}$$

where m_{tot} is the mass of UAVs, g and ρ are the earth gravity and air density, respectively, while p_r^n and r_p are the number of propellers and radii, respectively.

Assume that the average energy consumed per cluster is E_C for an average number of cluster C, and the average number of nodes per cluster is $\frac{N}{C}$, where N is the total number of nodes. The average energy for one round is

$$E_{1r} = c_r^g \times C \times E_c, \tag{43}$$

where c_r^g represents the quality of the communication channel. The average energy consumption per cluster is expressed as

$$\begin{aligned} E_C &= E_{CH} + \Gamma c_r^g \left(\frac{N}{C} - 1\right) E_{CM_{fs}} + (1-\Gamma) c_r^g \\ &\quad \times \left(\frac{N}{C} - 1\right) E_{CM_{mp}} \\ &\approx E_{CH} + \Gamma c_r^g \left(\frac{N}{C}\right) E_{CM_{fs}} + (1-\Gamma) c_r^g \left(\frac{N}{C}\right) E_{CM_{mp}}, \end{aligned} \tag{44}$$

where E_{CH} represents the energy dissipated by the CH, $E_{CM_{fs}}$ and $E_{CM_{mp}}$ are the energy dissipated by the CM nodes in the free space and multipath amplification models, respectively, and Γ and $(1-\Gamma)$ are the fractions of the member node that transmit using the free space model and the multipath model, respectively. E_{CH} is given as

$$\begin{aligned} E_{CH} &= E_{RX}(l) + E_{agg-tot} + E_{TX-amp}(l,d) \\ &= c_r^g \left(\frac{N}{C}\right) E_{elcc} + c_r^g \left(\frac{N}{C}\right) E_{agg} + l\varepsilon_{mp} d_{BS}^4, \end{aligned} \tag{45}$$

where d_{BS} is the average distance from the CH to the BS. $E_{CM_{fs}}$ and $E_{CM_{mp}}$ can be defined as

$$\begin{aligned} E_{CM_{fs}} &= E_{TX}(l,d) = E_{TX-elec}(l) + E_{TX-amp}(l,d) \\ &= lE_{elec} + l \cdot \varepsilon f_s \cdot d_{CH_{fs}}^4 \end{aligned} \tag{46}$$

$$\begin{aligned} E_{CM_{mp}} &= E_{TX}(l,d) = E_{TX-elec}(l) + E_{TX-amp}(l,d) \\ &= lE_{elec} + l \cdot \varepsilon f_s \cdot d_{CH_{mp}}^4 \end{aligned} \tag{47}$$

where $d_{CH_{fs}}$ and $d_{CH_{mp}}$ are the average distances from the CM to their corresponding CH in the free space and multipath models, respectively. For the case of $d \geq d_{th}$ from (40), substituting (45), (46), and (47), we obtain

$$\begin{aligned} E_C &= c_r^g \left(\frac{N}{C}\right) l \cdot E_{elec} + c_r^g \left(\frac{N}{C}\right) lE_{agg} + l\varepsilon_{mp} d_{BS}^4 \\ &\quad + \Gamma c_r^g \left(\frac{N}{C}\right) l \cdot E_{elec} + \Gamma c_r^g \left(\frac{N}{C}\right) l \cdot \varepsilon f_s \cdot d_{CH_{fs}}^4 \\ &\quad + (1-\Gamma) c_r^g \left(\frac{N}{C}\right) l \cdot E_{elec} + (1-\Gamma) c_r^g \left(\frac{N}{C}\right) l \\ &\quad \cdot \varepsilon f_s \cdot d_{CH_{mp}}^4. \end{aligned} \tag{48}$$

Substituting (43) into (48), we obtain

$$\begin{aligned} E_{1r} &= c_r^{g^2} N \cdot l(2E_{elec} + E_{agg} + \frac{C}{c_r^{g^2} N} \varepsilon_{mp} d_{BS}^4 \\ &\quad + \Gamma \varepsilon f_s \cdot d_{CH_{fs}}^2 + (1-\Gamma)\, \varepsilon_{mp} \cdot d_{CH_{mp}}^4). \end{aligned} \tag{49}$$

D. OPTIMAL NUMBER OF CLUSTERS

In general, the total energy consumption is determined by the number of clusters in the network. An optimal number of clusters can reduce the energy consumption and balance the network. We determined the optimal number of clusters via mathematical modeling. According to the basic geometric relationship, the volumes of the 3D cubic network region can be represented as

$$V_c = M \times M \times M = M^3, \tag{50}$$

where M is the length of the cube. The average cluster region C in the network can be expressed as

$$V_c = \frac{M^3}{C}. \tag{51}$$

The 3D cluster region with node distribution can be represented as $f_{XYH}(x,y,h)$. Let the CH position be the center of mass of the cluster region. The free space estimated value of the node distances to the CH can be calculated as

$$\begin{aligned} E\left[d_{CH_{fs}}^2\right] &= \iiint \left(x^2 + y^2 + h^2\right) f_{XYH}(x,y,h)\, dxdydh \\ &= \iiint r^2 f_{R\theta\emptyset}(r,\theta,\emptyset)\, r^2 \sin(\emptyset)\, drd\theta d\emptyset \\ &= \iiint r^4 f_{R\theta\emptyset}(r,\theta) \sin(\emptyset)\, drd\theta d\emptyset. \end{aligned} \tag{52}$$

By assuming that the region of the cluster is spherical, and $f_{R\theta\emptyset}(r,\theta,\emptyset)$ is the constant of the cluster region, the radius of the sphere and node density are expressed as (53) and (54), respectively.

$$\frac{M^3}{C} = \frac{4}{3}\pi R_c^3,\, R_c = \sqrt[3]{\frac{3}{4\pi C}M} \tag{53}$$

$$f_{R\theta\emptyset}(r,\theta,\emptyset) = \frac{1}{V_c} = \frac{C}{M^3}. \tag{54}$$

Equation (52) can be rewritten as

$$E\left[d_{CH_{fs}}^{2}\right] = \int_{0}^{\pi}\int_{0}^{2\pi}\int_{0}^{\sqrt[3]{\frac{3}{4\pi C}M}} r^{4}\frac{C}{M^{3}}\sin(\emptyset)\,drd\theta d\emptyset$$
$$= \frac{3}{10}\left(\frac{3}{\sqrt{2\pi C}}\right)^{\frac{2}{3}}M^{2}. \tag{55}$$

Substituting (55) in (49), we obtain the optimal number of clusters for the network:

$$\frac{\partial E_{1r}}{\partial C} = 0. \tag{56}$$

$$C_{opt} = \left(\frac{4}{3\pi^{2}}\right)^{1/5}\left(\Gamma c_{r}^{g}N\right)\frac{\varepsilon f_{s}}{\varepsilon_{mp}}\frac{M^{2}}{d_{BS}^{4}})^{3/5}. \tag{57}$$

The probability of selecting the optimal number of CHs can be calculated as

$$CH_{opt} = \frac{C_{opt}}{\Gamma c_{r}^{g}N} = \left(\frac{4}{3\pi^{2}}\right)^{1/5}\left(\frac{1}{\Gamma c_{r}^{g}N}\right)^{2/5}\frac{\varepsilon f_{s}}{\varepsilon_{mp}}\frac{M^{2}}{d_{BS}^{4}})^{3/5}. \tag{58}$$

E. CLUSTERING ALGORITHM

In this subsection, the clustering in UAV networks is presented. The proposed clustering is based on the HGWO algorithm. In HGWO clustering, the aim is to partition N nodes into predetermined or optimal C_{opt} clusters. During clustering, the proximate nodes are assigned to the cluster with the nearest mean using Euclidean distance. This ensures a low transmission range to reduce energy consumption. However, it is difficult to determine the location in high-mobility scenarios. To address this issue, the node distance is measured using the proposed HGWO algorithm. First, we group the UAV nodes based on their geographical locations by minimizing the sum of the squared errors as in (59).

$$\mathbb{X}_{i,C_h}^{t} = \arg\min \|\mathbb{U}_{w,i}^{d} - \mathbb{C}_{w}^{C_h}\|_{2}^{2}, \tag{59}$$

where $\mathbb{X}_{i,C_h}^{t}$ is the uniqueness of the wolf-to-cluster association, $\mathbb{C}_{w}^{h}$ is the set of wolves that are associated with the cluster C_h, and $\mathbb{U}_{w,i}^{d}$ is the position of wolf $\mathbb{U}_i$ in dimension$d$. If the new position (refer to (32)) of the gray wolf is beyond the search bounds, (60) moves the wolves toward the exceeded bound. It is more precisely represented as

$$\mathbb{U}_{w,i}^{d}(t+1) = \begin{cases} \mathbb{U}_{w,i}^{d}(t) + \ell \times (U^{d} - \mathbb{U}_{w,i}^{d}(t), \\ \quad \mathit{if}\ \mathbb{U}_{w,i}^{d}(t+1) > U^{d} \\ \mathbb{U}_{w,i}^{d}(t)\,\mathbb{U}_{w,i}^{d}(t) + \ell \times (L^{d} - \mathbb{U}_{w,i}^{d}(t), \\ \quad \mathit{if}\ \mathbb{U}_{w,i}^{d}(t+1) < L^{d} \end{cases} \tag{60}$$

where U^d and L^d are the upper and lower boundaries of the search area, respectively, and ℓ is a random number in [0, 1]. In the HGWO clustering, the position of the alpha wolf is the centroid of the cluster. The HGWO-based clustering scheme is described in Algorithm 3.

Algorithm 3 first initializes the parameters (lines 1–3). It assigns each node or wolf to a certain group based on the geographical locations of the wolves (lines 6–9). Thereafter, the HGWO algorithm is applied to fix the centroid of the cluster. The proposed algorithm estimates the potential location of the wolf by using (31) (lines 11–14). The individual position experience of the wolf is applied using (32) to obtain the optimal location (line 15). Finally, the wolf position is determined using the search bounds (line 16). After the iteration reaches the maximum number, the algorithm returns to the position of alpha, which is the final centroid of the cluster.

Algorithm 3 UAV Clustering Based on HGWO

```
Input: Number of clusters $C_{opt}$ and number of wolf packs $\mathbb{U}$
Output: Final centroid positions of clusters $C_m$, $m = 1, 2, \ldots, M$
/* Initialization Phase*/
1: Initialize the position of wolf pack $\mathbb{U} = [\mathbb{U}_1, \ldots, \mathbb{U}_w, \ldots, \mathbb{U}_W]$.
2: Initialize the GWO parameters $(e, A_w^d, C_w^d)$.
3: Initialize the number of clusters $C_m \leftarrow \emptyset$.
/* Computation*/
4: repeat
5:    $C_m \leftarrow C'_m$
6:    for each wolf $w = 1, 2, \ldots, W$ do
7:        for each cluster $C_{opt} = \{C_1, C_2, \ldots, C_h, \ldots, C_k\}$ do
8:            Update the wolf-to-cluster association using (1).
9:        end for
10:   end for
11: while ($t = 1$ to $t_{max}$) do
12:     for each wolf $w = 1$ to $W$ do
13:         for each wolf, $d = 1$ to $D$ do
14:             Update the position of each wolf defined by (31).
15:             Redefine the position of wolf $w$ using (32).
16:             Evaluate the new position of wolf $w$ defined by (60).
17:             Update the values of $(e, A_w^d, C_w^d)$.
18:             Calculate the fitness of each gray wolf.
19:             Update $\mathbb{U}_{w,\alpha}^d$, $\mathbb{U}_{w,\beta}^d$, and $\mathbb{U}_{w,\delta}^d$.
20:         end for
21:     end for
22:     $t = t + 1$
23: end while
24: Return $\mathbb{U}_{w,\alpha}^d$ /*the position of alpha is the final centroid position*/
25: until ($C_m = C'_m$) /*no change in the cluster centroids*/
```

F. CLUSTER HEAD SELECTION

This subsection describes the application of HGWO for selecting CHs in the network. The proposed CH selection process is presented in Algorithm 4. In the GWO algorithm, the three best solutions are alpha, beta, and delta (in that sequence) and they have the best idea of the prey location. In the first round, the positions of the wolves are updated

Algorithm 4 CH Selection Based on HGWO

```
Input: Clusters $C_m$ and number of nodes $\mathbb{U}$
Output: Cluster head $CH_m$, where $m = 1, 2, \ldots, M$.
/* Initialization phase*/
1: Initialize the position of the wolf pack $\mathbb{U} = [\mathbb{U}_1, \ldots, \mathbb{U}_w, \ldots, \mathbb{U}_W]$.
2: Initialize the GWO parameters $(e, A_w^d, C_w^d)$.
3: Calculate the fitness $(F)$ of each gray wolf.
4: Choose the three best solutions $\mathbb{U}_{w,\alpha}^d$, $\mathbb{U}_{w,\beta}^d$, and $\mathbb{U}_{w,\delta}^d$.
/* Computation*/
5: while $(t{+}{+} < t_{max})$, do
6:     for each search agent, do
7:         Update the position of the current search agent.
8:         if round = 0 then
9:             Update the position using (28)–(31).
10:        else
11:            Update the position using (61)–(73). /*apply cases 1–4*/
12:        end if
13:    end for
14:    Update $(e, A_w^d, C_w^d)$.
15:    Calculate the fitness $(F)$ of all search agents.
16:    Update $\mathbb{U}_{w,\alpha}^d$, $\mathbb{U}_{w,\beta}^d$, and $\mathbb{U}_{w,\delta}^d$
17:    $t = t + 1$
18: end while
19: return CHs
```

according to (28)–(31). Subsequently, the positions of the wolves are updated based on the fitness values. The process of measuring the fitness value is as follows:

For case 1:

$$D_{w,\alpha}^d = \left|C_{w,\alpha}^d \times (F_\alpha - F_w)\right|, \tag{61}$$

$$D_{w,\beta}^d = \left|C_{w,\beta}^d \times (F_\beta - F_w)\right|, \tag{62}$$

$$D_{w,\delta}^d = \left|C_{w,\delta}^d \times (F_\delta - F_w)\right|. \tag{63}$$

Therefore,

$$\mathbb{U}_{w,\alpha}^d = F_\alpha - A_{w,\alpha}^d \times D_{w,\alpha}^d, \tag{64}$$

$$\mathbb{U}_{w,\beta}^d = F_\beta - A_{w,\beta}^d \times D_{w,\beta}^d, \tag{65}$$

$$\mathbb{U}_{w,\delta}^d = F_\delta - A_{w,\delta}^d \times D_{w,\delta}^d, \tag{66}$$

$$\mathbb{U}_w^d (t+1) = \frac{1}{k} \sum\nolimits_{k \in \alpha,\beta,\delta} \mathbb{U}_{w,k}^d, \tag{67}$$

where D_w^d is the vector that depends on the location of the prey, and F is the fitness value of the wolf.

For case 2:

$$\mathbb{U}_{w,\alpha}^d = |F_\alpha - F_w|, \tag{68}$$

$$\mathbb{U}_{w,\beta}^d = |F_\beta - F_w|, \tag{69}$$

$$\mathbb{U}_{w,\delta}^d = |F_\delta - F_w|, \tag{70}$$

$$\mathbb{U}_w^d (t+1) = \frac{1}{k} \sum\nolimits_{k \in \alpha,\beta,\delta} \mathbb{U}_{w,k}^d. \tag{71}$$

For case 3:

$$\mathbb{U}_w^d (t+1) = \frac{1}{k}^{\frac{F_\alpha - F_w}{F_\alpha - F_o}}, \tag{72}$$

where k is the iteration and F_o is the worst fitness value.

For case 4:

$$\mathbb{U}_w^d (t+1) = \left|\frac{F_\alpha - F_w}{F_\alpha - F_o}\right|. \tag{73}$$

For the selection of CHs, several fitness functions are considered, and our goal is to maximize the fitness function. In our model fitness, the functions are intra-cluster distance (ICD), number of neighbors (NoN), and residual power of the UAVs (RP).

$$F(i) = coef_1 \times \text{ICD} + coef_2 \times \text{NoN} + coef_3 \times \text{RP}, \tag{74}$$

where $coef_1$, $coef_2$, and $coef_3$ are 0.2, 0.3, and 0.5, respectively. The coefficients are chosen based on priority. The average distance between the UAV nodes and the initial CH is the fitness coefficient ICD, which is defined as

$$\text{ICD} = \frac{1}{N_m}\left(\sum\nolimits_{m=1}^{N_m} \|U_n - CH_m\|\right), \tag{75}$$

where $\|\mathbb{U} - CH_m\|$ is the distance between the UAV node $\mathbb{U}$ to CH_m, and N_m is the total number of UAV nodes in the cluster. The second fitness function NoN is calculated as

$$\text{NoN} = R = \sqrt{\frac{M^3}{{}^4\!/_3\pi \times C_{opt}}}, \tag{76}$$

where R is the radius and C_{opt} is the optimal cluster (see (57). The third fitness function is calculated as follows:

$$\text{RP} = \frac{1}{\sum_{m=1}^{M} E_R CH_m}, \tag{77}$$

where $E_R CH_m$ is the current residual energy level of CH_m, and $1 \leq m \leq M$. The detailed process of CH selection is presented in Algorithm 4.

The position of the wolf pack and the GWO parameters are initialized at the beginning of Algorithm 4 in lines 1–2. The algorithm measures the fitness of the wolves and chooses the three-best wolves based on higher fitness value in lines 3–4. At the initial phase, wolves' fitness value is calculated based on equations (28)–(31) in lines 5–9. After that, the algorithm utilizes case 1–4 to measure the wolves' fitness value in line 11. The formula for calculating the fitness value is defined in equation (18) in line 15. The detailed procedure of the optimization stages in HGWO is provided in lines 16–19.

G. TRANSMISSION-EFFICIENT ROUTING

In this subsection, we present the transmission-efficient routing protocol for UAV networks. In this study, a transmission-efficient routing denotes routing with a minimum number of transmissions. A majority of the data transmission schemes for UAV networks are single-hop or multihop-based methods. However, the existing schemes are disadvantageous because of their considerable delays and large number of transmissions, which cause the battery of UAVs to drain quickly. In this article, we propose a GWO-based CS scheme named CS-GWO for UAV networks. This scheme significantly reduces the number of data transmissions and balances the traffic load of the entire network. The CM nodes transmit data to the CH in the same cluster without using CS-GWO.

The CHs employ CS-GWO routing to transmit data to the BS or to the nearest CH. In CS-GWO, we constructed a minimum-cost algorithm that connects all CHs to the BS, where the cost is represented by the number of hops. The CHs use this backbone routing path to transmit data to the BS. The backbone routing path is updated based on node fitness, as proposed in GWO. In our algorithm, which is inspired by GWO, we classified all nodes into three categories: alpha, beta, and delta. The backbone tree is constructed using the three types of nodes. An example of the proposed routing scheme is presented in Fig. 5.

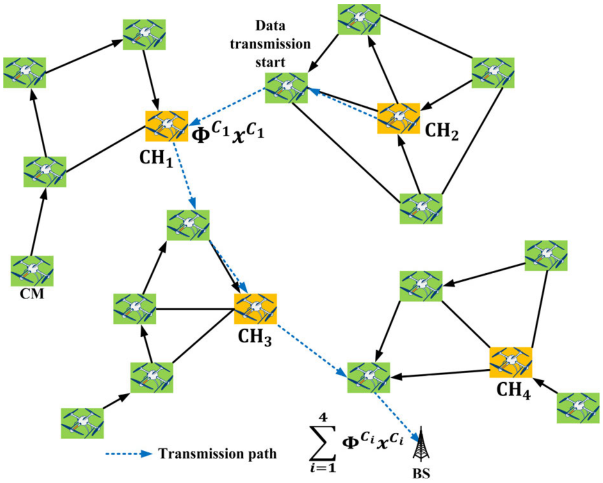

FIGURE 5. Example of transmission-efficient routing.

Algorithm 5 CS-GWO Routing

```
Input: CS matrix $\Phi_{M\times N}$, measurement vector $x$, joint sparsity $K$, set limiting parameter $S_l$, stopping criterion $\Omega$, and wolf set $\mathbb{U} = [\mathbb{U}_1, \ldots, \mathbb{U}_w, \ldots, \mathbb{U}_W]$
Output: Estimated solution set $x'$
/* Initialization*/ Initialize support set $\mathcal{R}$, stopping criterion $\Omega$, and estimated solution $x' = \emptyset$.
Initialize gray wolf population matrix $\mathbb{U}_{i\times K}$ with random integers between $[1, N]$.
Initialize $(\mathbb{U}_\alpha, \mathbb{U}_\beta, \mathbb{U}_\delta)$ as vector $(1 \times K)$ with all its components equal to 0.
Initialize $L^{t+1}_{best1} = L^{t+1}_{best2} = L^{t+1}_{best3} = \infty$.
Initialize iteration $t = 1$ and allow maximum iteration number $t_{max}$.
/* Computation and reconstruction*/
while ($t \le t_{max} \parallel L^{t+1}_{best1} > \Omega$) do
    for each wolf $w = 1$: $W$ do
        Update the wolves' positions
    end for
    for each row $i$ of matrix $\mathbb{U}_{i\times k}$
        Create a search set $\mathbb{C}$, where $\mathbb{C} = \mathcal{R} \cup \{\text{Row}\,iof\,\mathbb{U}_{i\times K}\}$
        $I \triangleq$ {indices of the $K$ largest magnitude entries in $\Phi^c x$
        $L = \Phi^I$
        Calculate fitness $f(L)$ using (79)
        if ($L^{t+1}_{best1} > f(L)$) then
            $\mathbb{U}_\alpha = I$
        else if ($L^{t+1}_{best1} < f(L)$) && ($L^{t+1}_{best2} > f(L)$) then
            $L^{t+1}_{best1} = f(L)$ and $\mathbb{U}_\beta = I$
        else if ($L^{t+1}_{best1} < f(L)$)&& $\left(L^{t+1}_{best2} < f(L)\right)$ && $\left(L^{t+1}_{best3} > f(L)\right)$ then
            $L^{t+1}_{best3} = f(L)$ and $\mathbb{U}_\delta = I$
        end if
        Set $\mathcal{R} = I$
    end for
    Update $(e, A^d_w, C^d_w)$.
    for each search agent, do
        Update the position of the current search agent by (31).
    end for
    $t = t + 1$
end while
return $x'$
```

Assume that the original data $\mathbb{X} = [x_1, x_2, \ldots, x_N]^T$ have a K-sparse symbol of matrix Φ, where Φ is a $M \times N$ matrix. In addition, assume that the CH is aware of the projection vectors of all nodes within the cluster. A measurement coefficient Φ_{ij} can be randomly generated to identify noden_j. Thus, it is easy to identify nodes in the network and generate a measurement matrix for the CHs. Matrix Φ can be divided into several submatrices for each cluster. Assume that Φ^{C_i} is a submatrix of cluster i, CH$_i$ is the cluster head, and the data vector of cluster CH$_i$ isx^{C_i}. Therefore, the collection data x^{C_i} from CMs with a cluster of submatrices can be defined as $\Phi^{C_i}x^{C_i}$. As depicted in Fig. 5, the UAV nodes are divided into four clusters and cluster heads. CH$_1$, CH$_2$, CH$_3$, and CH$_4$ are connected to the BS through a backbone tree. Vector x can be decomposed into $\left[x^{C_1}, x^{C_2}, x^{C_3}, x^{C_4}\right]^T$, and matrix Φ is defined as $\left[\Phi^{C_1}, \Phi^{C_2}, \Phi^{C_3}, \Phi^{C_4}\right]$. The projections of all data in the network, which are generated from the clusters, can be defined as

$$\begin{aligned} y &= \Phi x \\ &= \left[\Phi^{C_1}\Phi^{C_2}, \Phi^{C_3}, \Phi^{C_4}\right]\begin{pmatrix} x^{C_1} \\ x^{C_2} \\ x^{C_3} \\ x^{C_4} \end{pmatrix} \\ &= \sum\nolimits_{i=1}^{4} \Phi^{C_i}x^{C_i}. \end{aligned} \tag{78}$$

The CS-GWO routing is presented in Algorithm 5. A support set $\mathscr{R}$, which represents Φ^T columns, is initialized at the start of the algorithm. Assume that the search agents (positions of the wolves) are defined as matrix $\mathbb{U}_{i \times j}$, where i is the number of wolves, and j is the joint sparsity K. Each matrix has a randomly selected integer [1, N], where N represents the number of columns in Φ. $\mathbb{U}_\alpha$, $\mathbb{U}_\beta$, and $\mathbb{U}_\delta$ are the three best wolves represented in vector ($1 \times K$) (lines 1–4). The three best fitness values are initialized in line 4. Thereafter, the initialized parameters are updated at all positions. For each row i in matrix $\mathbb{U}$, a search set $\mathbb{C}$ is created, where $\mathbb{C} = \mathcal{R} \cup \{\text{Row}\#i \text{ of } \mathbb{U}_{i \times K}\}$. Subsequently, a submatrix Φ^c is created from the CS matrix Φ (lines 6–12). At line 13, a submatrix L is created, where $L = \Phi^I$ is the column of matrix Φ. The cost function, which is measured using the fitness value $f(L)$, is expressed as follows:

$$f(L) = \|L \cdot \Phi x - x\|_2. \tag{79}$$

The best wolves are found using lines 15–23, where L^{t+1}_{best1}, L^{t+1}_{best2}, and L^{t+1}_{best3} are the three best solutions. Finally, the positions of the wolves are updated based on (30).

H. COMPUTATIONAL COMPLEXITY

The computational or time complexity of proposed HGWO is represented as $O(\mathcal{W} * d * t_{max})$, where $\mathcal{W}$ is the size of the population, d the dimension of the optimization problem to be solved, and t_{max} the maximum number of iterations. The time complexity of HGWO contains the three phases of O(population initialization) $+ O$(calculation of the fitness value for entire population) $+ O$(updating the population position). In the first phase, HGWO have the complexity of $O(\mathcal{W} * d)$. Second, the time complexity of fitness value of entire population is $O(\mathcal{W} * d * t_{max})$. Finally, updating the population position contains the two parts of O(position update during crossover operation) and O(position update during mutation). Hence, the computational complexity of the third phase is $O(\mathcal{W} * d * t_{max})$. Therefore, total computational complexity of HGWO is $O(\mathcal{W} * d + \mathcal{W} * d * t_{max} + \mathcal{W} * d * t_{max}) = O((2t_{max} + 1)\mathcal{W} * d)$.

In the BIL algorithm, HGWO algorithm dominates and, thus, the computational complexity of BIL is the same as that of HGWO. That is, the computational complexity of BIL is $O((2t_{max} + 1)\mathcal{W} * d)$. The BIC and CS-GWO algorithms have the same time complexity. In the initialization phase, the algorithm requires $O(\mathcal{W} * d)$ time to calculate the control parameters. After that, the algorithm requires $O(\mathcal{W} * d)$ to update three best solutions in each iteration. For the maximum number of iterations, the total time complexity is $O(\mathcal{W} * d * t_{max})$.

VI. PERFORMANCE EVALUATION

We conducted extensive simulations using MATLAB [61] to evaluate the performance of the proposed BIL and BIC under different scenarios. First, we investigated the performance of BIL and compared it with those of the DV-Hop, GWO-LPWSN, 3D-GAIDV-Hop, DECHDV-Hop, and HDPSO algorithms. Then, the performance of BIC was evaluated and compared with those of CBLADSR, MPCA, EALC, BICSF, and SOCS.

TABLE 2. Simulation parameters.

Parameter	Value
Simulator	MATLAB
Network area	1000 m × 1000 m
Height of UAVs	125–200 m
UAV transmission range (R)	250 m
Sensing range of UAVs	250 m
Desired value of neighboring distance	115 m
Speed	10–30 m/s
Carrier frequency U2U	2.4 GHz
UAV transmission power	0.5 W
UAV flying power	300 W
Path loss under LoS	1 dB
Path loss under NLoS	20 dB
SINR threshold	−7 dB
Environmental parameters (ψ and ϕ)	1, 0.5
MAC protocol	IEEE 802.11n
Maximum number of UAVs	100
Number of ground stations	1
Initial energy of node	5 J
Transmitter/Receiver electronics	50 nJ/bit
Transmitter amplifier (free space)	100 pJ/bit
Transmitter amplifier (multipath)	0.010 pJ/bit
Traffic type	CBR
CBR rate	2 Mbps
Number of rounds	1000
Traffic load	5 messages/s
Message size	250 kB
Maximum number of iterations	Variable

A. SIMULATION ENVIRONMENT

In our simulation, an area of 1000 m × 1000 m was considered, where the UAVs were uniformly distributed in the network. The altitude range of the UAVs was approximately 125–200 m. The BS was placed beyond the simulation area. The minimum and maximum mobility's of the UAVs ranged from 10 to 30 m/s. The simulation parameters are summarized in Table 2.

In the first phase, we evaluated the performance of the BIL algorithm in terms of the performance metrics for localization such as accuracy of localization, mean localization error (MLE), and algorithm convergence cost. The methods for calculating the localization error, MLE, and localization success ratio were the same as those in [30]. In the second phase, we compared our proposed BIC algorithm with other algorithms. The performance metrics used are the number of clusters, cluster building time, and energy consumption. The performance metrics were observed by varying the number of nodes and rounds.

B. SIMULATION RESULTS AND DISCUSSION ON LOCALIZATION

In this subsection, the performance of the BIL algorithm is analyzed, and the results of the comparison with other localization algorithms are discussed. The data are summarized in Table 3, where a few key parameters for localization performance, such as the maximum localization error,

TABLE 3. Comparison results in terms of localization error.

Method	Total number of nodes (unknown + anchor)	Number of unknown nodes	Number of anchor nodes	Maximum localization error (m)	Minimum localization error (m)	Mean localization error (m)	Localization success rate (%)	Time required per node (ms)	Iterations required
BIL	150	120	30	1.51	0.15	0.32	98.23	44	28
DV-Hop	150	120	30	9.72	3.25	4.96	55.53	110	88
GWO-LPWSN	150	120	30	7.35	3.05	3.38	80.74	58	33
GAIDV-Hop	150	120	30	5.87	2.47	3.12	78.43	89	52
DECHDV-Hop	150	120	30	5.11	2.39	2.71	83.22	78	43
HDPSO	150	120	30	3.91	1.35	1.65	85.12	61	35

minimum localization error, and MLE, are also presented. The comparison results show that the proposed BIL has a more efficient performance than the other algorithms. Table 3 also presents the comparison results of the algorithm efficiency in terms of the number of iterations required to achieve the goal, time taken to localize each node, and percentage value of the localization success rate. From Table 3, it is evident that the proposed BIL can localize a larger number of nodes within a shorter time. In addition, the proposed approach achieves higher localization rate compared to other algorithms by considering the hop count.

The BIL performs better than other algorithms when identifying the location of the unknown nodes. The bounding cube method reduces the initial search space, thereby reducing the localization time as well as the computing time required for localization. In addition, due to the leadership and hunting mechanisms, HGWO helps to identify unknown nodes. Besides, HGWO optimizes the coordinates of the unknown nodes. To enhance the performance of the local searches (when the unknown node is near the anchor node) and the global searches (when the unknown node is far away from the anchor node), the parameter "e" is decreased gradually.

Owing to the mobility of the UAVs, the positions of UAV nodes, network topology, and frequency change constantly. Fig. 6(a) depicts the positions of the UAV nodes at time $t + 1$, where the black asterisks ($*$) indicate the anchor nodes, and the red circles ($\circ$) represent the unknown nodes. Fig. 6(b) illustrates the node connectivity between anchors and unknown nodes at time $t+1$. The localization error at time $t + 1$ is presented in Fig. 6(c), where the straight blue lines denote the localization errors at a distance from the actual position to the estimated position.

To reduce the localization error, we applied the bounding cube method to increase the sampling efficiency. A higher sampling efficiency reduces the node localization error, size of the searching region, and number of iterations. Thus, the proposed BIL consumes less energy. In our proposed algorithm, we applied sample filtering to significantly reduce FA. During the distance estimation, as shown in Fig. 6(c), the BIL algorithm has fewer localization errors and locates the nodes at higher accuracy than the other methods.

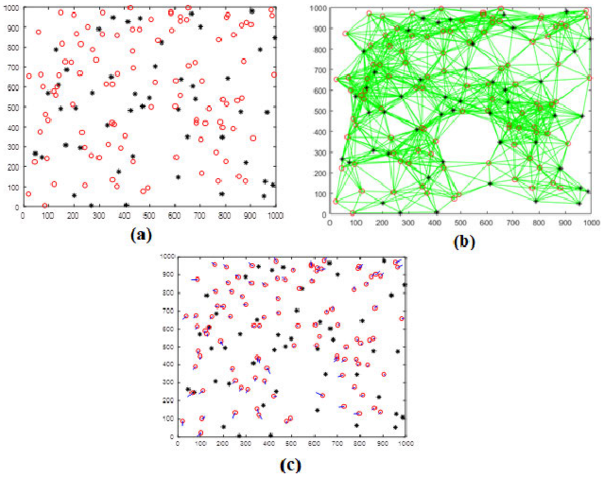

FIGURE 6. (a) Node position at time *t* + *1*, (b) node connectivity at time *t* + *1*, and (c) node localization error at time *t* + *1*, when using the BIL algorithm.

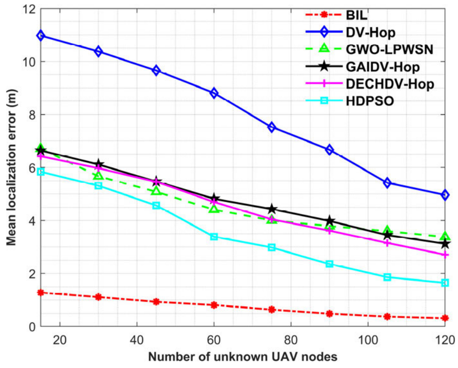

FIGURE 7. MLE vs. number of unknown UAV nodes.

1) MLE VS. NUMBER OF UNKNOWN UAV NODES

As depicted in Fig. 7, an increase in the number of nodes reduces the localization error. Furthermore, it increases the

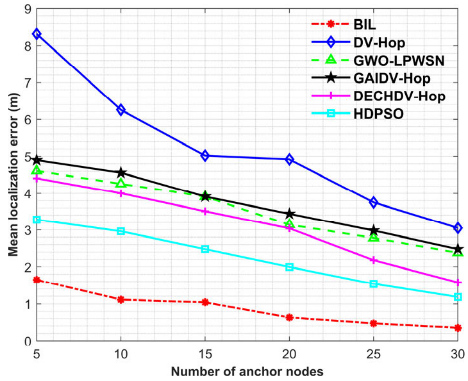

FIGURE 8. **MLE vs. number of anchor nodes.**

number of reference points, thereby facilitating the localization of additional nodes with fewer errors. Moreover, after an unknown node is localized, it can be used to localize the remaining unknown nodes. It can be observed from Fig. 7 that BIL has the lowest MLE among all algorithms.

2) MLE VS. NUMBER OF ANCHOR UAV NODES

As depicted in Fig. 8, as the number of anchor nodes increases, the MLE decreases. As the anchor nodes increase, the reference points of the unknown nodes also increase, and the distance between the anchor and unknown nodes is reduced. The bounding cube scheme increases the highly valid samples and, thus, the localization accuracy is increased in the BIL algorithm. Thus, the MLE is reduced for all algorithms; however, among the six algorithms, the MLE of BIL is the lowest for all cases.

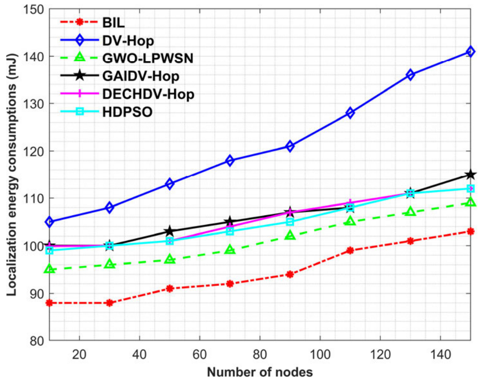

FIGURE 9. **Localization energy consumption vs. number of UAV nodes.**

3) LOCALIZATION ENERGY CONSUMPTION VS. NUMBER OF NODES

Fig. 9 shows that the proposed BIL consumes less energy for node localization compared to other algorithms. This is because BIL requires fewer anchor nodes to localize a large number of unknown nodes. The anchor nodes transmit their position information to unknown nodes only when they are within the sensing range. In other algorithms, the anchor nodes transmit position information whenever they are within the communication range. For BIL, communication cost is calculated by the number of transmitted control packets to localize the entire network. Each anchor node broadcast its location. In BIL, however, node location upgrade, location optimization, and hop count calculation are processed at target nodes. Therefore, energy consumption is significantly reduced for localization in BIL. In addition, because the HGWO algorithm quickly converges, the BIL algorithm requires less energy than the other algorithms.

C. SIMULATION RESULTS AND DISCUSSION ON CLUSTERING

The performance of BIC was evaluated and compared with those of other BIC protocols, namely, CBLADSR, MPCA, EALC, BICSF, and SOCS. The results of the comparison are presented in this subsection.

1) NUMBER OF CLUSTERS VS. TOTAL NUMBER OF UAV NODES

In the clustering method, network energy consumption is depends on the number of clusters in the network. The optimal number of clusters may reduce the energy consumption in the network. Therefore, it is essential to determine the optimal number of clusters in a network. Fig. 10 presents the relationship between the number of clusters and the number of nodes.

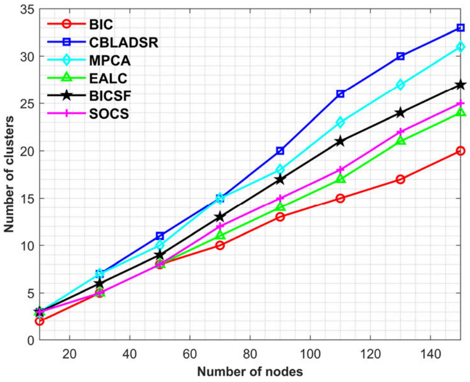

FIGURE 10. **Number of clusters vs. number of UAV nodes.**

The optimal number of clusters is the main factor that reduces energy consumption. In this study, we analytically modeled the optimal number of clusters, which ensures minimal energy consumption in the network. If the cluster size is significantly large, the number of intra-cluster transmissions will be large. However, if the cluster size is significantly small, the number of clusters will be large. Hence, if the number of clusters is not optimal, a few nodes may not join the cluster. This may increase the number of clusters due to the single-node cluster problem. For optimal clustering, it is necessary that a node should only be in one cluster. Our proposed clustering scheme prevents single-node clusters and optimizes the network topology. Therefore, the BIC performs better than the other clustering algorithms.

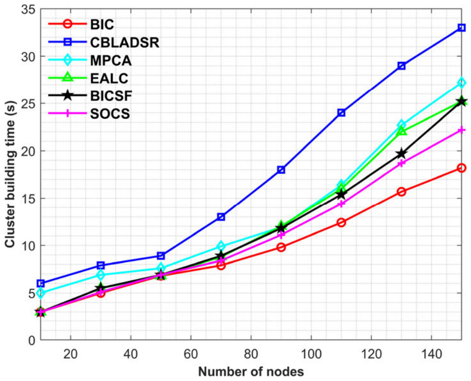

FIGURE 11. Cluster building time vs. number of UAV nodes.

2) CLUSTER BUILDING TIME VS. TOTAL NUMBER OF UAV NODES

In the clustering process, every node broadcasts its position and energy level at the beginning of each round, and each CH node broadcasts message to introduce itself in the network. In BIC, therefore, the number of transmitted control packets depends on the selection of CH and the distance between nodes and BS. The cluster building time, which refers to the time taken by the clustering algorithm for cluster formation and CH selection, contributes to the computational complexity of the clustering algorithm. A long cluster building time consumes additional energy, thereby reducing the network lifetime. Due to the energy limitation of UAVs, it is necessary to build up a cluster within a short period of time. As shown in Fig. 11, as the number of nodes in the network increases, the time taken by the clustering algorithm to build the cluster increases. In HGWO, we utilized PSO algorithm to avoid the wolf pack being trapped in local optima. As a result, more suitable CH and relay nodes are selected, which make BIC more energy efficient. Therefore, our proposed scheme performs better than the other algorithms due to its shorter cluster building time, which reduces the route selection delay.

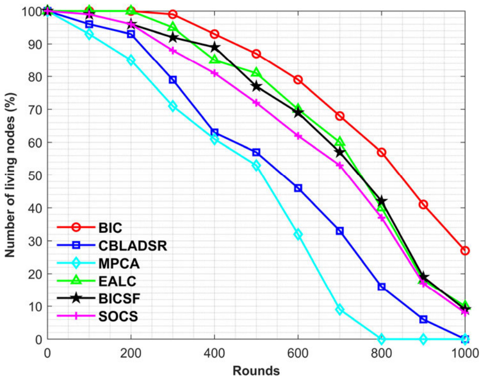

FIGURE 12. Number of living nodes vs. rounds.

3) NUMBER OF LIVING NODES VS. ROUNDS

Fig. 12 shows the variation curve of the living rate of nodes and the network life cycle. After each cycle, a few UAVs die because of the drainage of their battery power. From Fig. 12, we can observe that the number of living nodes of BIL is maintained longer and is more stable than those of other algorithms. Furthermore, the proposed BIL consumes the least amount of energy among the six algorithms.

4) TOTAL ENERGY CONSUMPTION VS. ROUNDS

Fig. 13 depicts the variation curve of energy consumption and rounds. Compared to the other algorithms, the slopes of the BIC curve are smaller, which indicate that the dying process of the nodes is comparatively milder. However, as the number of rounds increases, the energy consumption of BIC decreases because of the optimal clustering and selection of the optimal number of CHs.

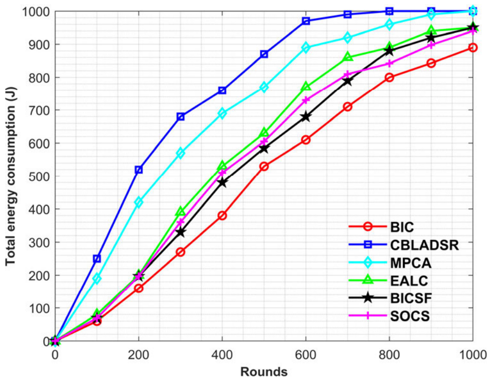

FIGURE 13. Total energy consumption vs. rounds.

D. SIMULATION RESULTS AND DISCUSSION ON ROUTING

In this subsection, we report the performance of the proposed CS-GWO routing. The UAV nodes were organized into clusters, and each cluster has a CH, which is indicated by the solid squares in Fig. 14. The blue links indicate the shortest routing path (backbone tree) for inter-cluster transmission links from CHs to the BS.

In most of the existing cluster-based routing algorithms, each cluster head (CH) node transmits data to the base station (BS) or to the nearest CH. In dynamic UAV networks, links between CHs may be long and thus need more power to transmit the data. Also, maintaining links between CHs is a difficult task due to high mobility. Our proposed CS-GWO utilizes relay nodes to transmit data to other CHs and the BS, which reduces the load from CH and prolongs the CH lifetime. Besides, in the existing clustering methods, the CH node near to the BS dies early due to the shortest path routing, which makes routing hole problem. In our approach, we utilize CM nodes as relays, which transfer the data to the BS and achieve energy balancing between the UAV nodes.

1) NUMBER OF TRANSMISSIONS VS. NUMBER OF UAV NODES

Fig. 15 compares our scheme of clustering without CS and the shortest path tree (SPT) in terms of the number

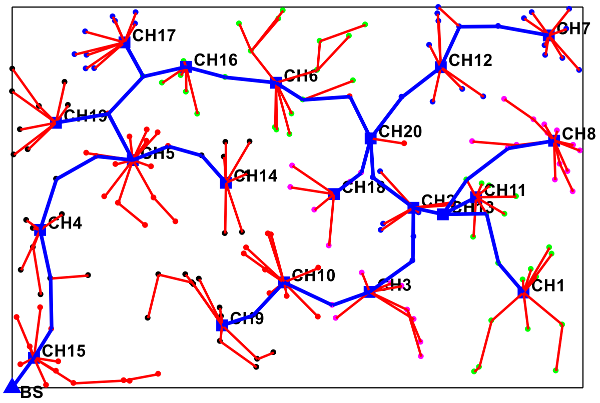

FIGURE 14. CS-GWO data transmission method.

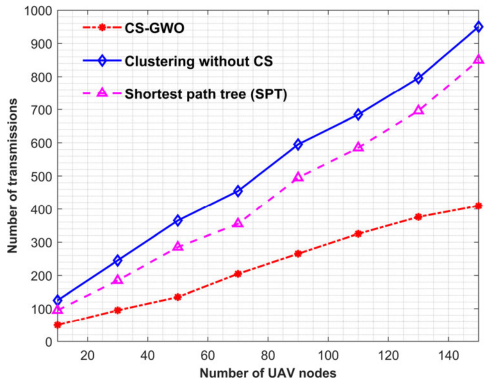

FIGURE 15. Number of transmissions vs. number of UAV nodes.

of transmissions. The number of transmissions in our scheme is significantly lower than those in other methods. In the proposed method, data were compressed using the CS technique at the CHs, thereby reducing the number of transmissions. The transmission of inter-cluster data proceeds through the backbone routing tree. The transmissions in our scheme are significantly lower than those in clustering without CS. In addition, the number of transmissions in our method is evidently smaller than that in SPT. This is because our scheme is based on the cluster structure, in which nodes transmit their data to the CH and the CH is located around the center of the cluster. Contrarily, in PST, the UAV nodes transmit their data to the nodes near the BS.

VII. CONCLUSION

In this article, we have proposed the BIL and BIC algorithms in UAV networks for wildfire detection and monitoring applications in remote areas. In BIL, the initial search space is limited by using the bounding cube method, thereby increasing the sampling efficiency and reducing the computational cost. The developed HGWO algorithm is incorporated to determine the actual location of unknown nodes. The simulation results show that BIL is superior in terms of various performance metrics under different scenarios compared to the conventional schemes. The proposed BIC is inspired by the gray wolf leadership hierarchy and focuses on improving energy efficiency. It reduces energy consumption by considering various issues such as the number of clusters, cluster size, cluster stability, and number of transmissions. We have also developed an analytical model for determining the optimal number of clusters. The HGWO algorithm optimizes the formation of UAV clusters and the selection of CH. The simulation results show that BIC achieves a significant performance improvement in terms of cluster building time, number of clusters, cluster lifetime, and energy consumption. Finally, we have proposed the CS-GWO routing, which reduces the number of transmissions in UAV networks.

Our study has a limitation because the proposed approach is based on multi-hop communication, where nodes are in remote area and the BS is out of node transmission range. In our approach, we have proposed CS links in multi-hop communication to reduce the UAV nodes' transmission power. Due to multi-hop communication, the proposed approach may increase the number of hops to transmit the data from CM to BS.

The use of multiple antennas can be adopted to improve the performance of UAV networks even though there are some burden on cost and complexity. We are going to consider the use of multiple antennas as a future work.

ACKNOWLEDGMENT

The authors thank the editor and anonymous referees for their constructive comments for improving the quality of this article.

REFERENCES

[1] M. Y. Arafat and S. Moh, "Location-aided delay tolerant routing protocol in UAV networks for post-disaster operation," *IEEE Access*, vol. 6, pp. 59891–59906, 2018, doi: 10.1109/access.2018.2875739.

[2] M. Y. Arafat and S. Moh, "Routing protocols for unmanned aerial vehicle networks: A survey," *IEEE Access*, vol. 7, pp. 99694–99720, 2019, doi: 10.1109/ACCESS.2019.2930813.

[3] D. S. Thomas, D. T. Butry, S. W. Gilbert, D. H. Webb, and J. F. Fung, "The costs and losses of wildfires," Nat. Institute Standard Technol. (NIST), Gaithersburg, MD, USA, Tech. Rep. 1215, 2017.

[4] K. Hoover and L. A. Hanson. (2019). *Wildfire Statistics 2019 | Centre for Research on the Epidemiology of Disasters*. Federal Assistance for Wildfire Response and Recovery. Accessed: May 26, 2019. [Online]. Available: https://fas.org/sgp/crs/misc/IF10244.pdf

[5] 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, doi: 10.1109/jiot.2016.2612119.

[6] M. Y. Arafat, S. Poudel, and S. Moh, "Medium access control protocols for flying ad hoc networks: A review," *IEEE Sensors J.*, vol. 21, no. 4, pp. 4097–4121, Feb. 2021, doi: 10.1109/jsen.2020.3034600.

[7] Z. Lin, H. H. T. Liu, and M. Wotton, "Kalman filter-based large-scale wildfire monitoring with a system of UAVs," *IEEE Trans. Ind. Electron.*, vol. 66, no. 1, pp. 606–615, Jan. 2019, doi: 10.1109/tie.2018.2823658.

[8] Q. Guo, Y. Zhang, J. Lloret, B. Kantarci, and W. K. G. Seah, "A localization method avoiding flip ambiguities for micro-UAVs with bounded distance measurement errors," *IEEE Trans. Mobile Comput.*, vol. 18, no. 8, pp. 1718–1730, Aug. 2019, doi: 10.1109/tmc.2018.2865462.

[9] F. Khelifi, A. Bradai, K. Singh, and M. Atri, "Localization and energy-efficient data routing for unmanned aerial vehicles: Fuzzy-logic-based approach," *IEEE Commun. Mag.*, vol. 56, no. 4, pp. 129–133, Apr. 2018, doi: 10.1109/mcom.2018.1700453.

[10] A. Boukerche and E. Nakamura, "Localization systems for wireless sensor networks," *IEEE Wireless Commun.*, vol. 14, no. 6, pp. 6–12, Dec. 2007, doi: 10.1109/mwc.2007.4407221.

[11] S. Yang, D. Enqing, L. Wei, and P. Xue, "An iterative method of processing node flip ambiguity in wireless sensor networks node localization," in *Proc. Int. Conf. Inf. Netw. (ICOIN)*, Jan. 2016, pp. 92–97, doi: 10.1109/icoin.2016.7427094.

[12] M. Y. Arafat, M. A. Habib, and S. Moh, "Routing protocols for UAV-aided wireless sensor networks," *Appl. Sci.*, vol. 10, no. 12, p. 4077, Jun. 2020, doi: 10.3390/app10124077.

[13] A. A. Khan, M. Abolhasan, and W. Ni, "An evolutionary game theoretic approach for stable and optimized clustering in VANETs," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4501–4513, May 2018, doi: 10.1109/tvt.2018.2790391.

[14] W. Zhang, G. Han, X. Wang, M. Guizani, K. Fan, and L. Shu, "A node location algorithm based on node movement prediction in underwater acoustic sensor networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 3, pp. 3166–3178, Mar. 2020, doi: 10.1109/tvt.2019.2963406.

[15] S. Zhang, Y. Zhou, Z. Li, and W. Pan, "Grey wolf optimizer for unmanned combat aerial vehicle path planning," *Adv. Eng. Softw.*, vol. 99, pp. 121–136, Sep. 2016, doi: 10.1016/j.advengsoft.2016.05.015.

[16] R. Dewangan, A. Shukla, and W. Godfrey, "Three dimensional path planning using grey wolf optimizer for UAVs," *Appl. Intell.*, vol. 49, no. 6, pp. 2201–2217, 2019, doi: 10.1007/s10489-018-1384-y.

[17] K. Albina and S. G. Lee, "Hybrid stochastic exploration using grey wolf optimizer and coordinated multi-robot exploration algorithms," *IEEE Access*, vol. 7, pp. 14246–14255, 2019, doi: 10.1109/access.2019.2894524.

[18] Y. Wang, T. Zhang, Z. Cai, J. Zhao, and K. Wu, "Multi-UAV coordination control by chaotic grey wolf optimization based distributed MPC with event-triggered strategy," *Chin. J. Aeronaut.*, vol. 33, no. 11, pp. 2877–2897, Nov. 2020, doi: 10.1016/j.cja.2020.04.028.

[19] M. Fahad, F. Aadil, K. Muhammad, Zahoor-ur-Rehman, S. Khan, P. A. Shah, J. Lloret, H. Wang, J. W. Lee, and I. Mehmood, "Grey wolf optimization based clustering algorithm for vehicular ad-hoc networks," *Comput. Elect. Eng.*, vol. 70, pp. 853–870, Aug. 2018, doi: 10.1016/j.compeleceng.2018.01.002.

[20] L. Merino, F. Caballero, J. R. Martínez-de-Dios, I. Maza, and A. Ollero, "An unmanned aircraft system for automatic forest fire monitoring and measurement," *J. Intell. Robot. Syst.*, vol. 65, nos. 1–4, pp. 533–548, Jan. 2012, doi: 10.1007/s10846-011-9560-x.

[21] A. Trotta, L. Montecchiari, M. D. Felice, and L. Bononi, "A GPS-free flocking model for aerial mesh deployments in disaster-recovery scenarios," *IEEE Access*, vol. 8, pp. 91558–91573, 2020, doi: 10.1109/access.2020.2994466.

[22] J. S. Russell, M. Ye, B. D. O. Anderson, H. Hmam, and P. Sarunic, "Cooperative localization of a GPS-denied UAV using direction-of-arrival measurements," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 56, no. 3, pp. 1966–1978, Jun. 2020, doi: 10.1109/taes.2019.2942704.

[23] F. She, Y. Zhang, D. Shi, H. Zhou, X. Ren, and T. Xu, "Enhanced relative localization based on persistent excitation for multi-UAVs in GPS-denied environments," *IEEE Access*, vol. 8, pp. 148136–148148, 2020, doi: 10.1109/access.2020.3015593.

[24] A. Eldosouky, A. Ferdowsi, and W. Saad, "Drones in distress: A game-theoretic countermeasure for protecting UAVs against GPS spoofing," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 2840–2854, Apr. 2020, doi: 10.1109/jiot.2019.2963337.

[25] D. Niculescu, "DV based positioning in ad hoc networks," *Telecommun. Syst.*, vol. 22, nos. 1–4, pp. 267–280, 2003, doi: 10.1023/a:1023403323460.

[26] R. Rajakumar, J. Amudhavel, P. Dhavachelvan, and T. Vengattaraman, "GWO-LPWSN: Grey wolf optimization algorithm for node localization problem in wireless sensor networks," *J. Comput. Netw. Commun.*, vol. 2017, pp. 1–10, Mar. 2017, doi: 10.1155/2017/7348141.

[27] G. Sharma and A. Kumar, "Improved range-free localization for three-dimensional wireless sensor networks using genetic algorithm," *Comput. Electr. Eng.*, vol. 72, pp. 808–827, Nov. 2018, doi: 10.1016/j.compeleceng.2017.12.036.

[28] L. Cui, C. Xu, G. Li, Z. Ming, Y. Feng, and N. Lu, "A high accurate localization algorithm with DV-hop and differential evolution for wireless sensor network," *Appl. Soft Comput.*, vol. 68, pp. 39–52, Jul. 2018, doi: 10.1016/j.asoc.2018.03.036.

[29] P. Raguraman, M. Ramasundaram, and V. Balakrishnan, "Localization in wireless sensor networks: A dimension based pruning approach in 3D environments," *Appl. Soft Comput.*, vol. 68, pp. 219–232, Jul. 2018, doi: 10.1016/j.asoc.2018.03.039.

[30] M. Y. Arafat and S. Moh, "Localization and clustering based on swarm intelligence in UAV networks for emergency communications," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8958–8976, Oct. 2019, doi: 10.1109/jiot.2019.2925567.

[31] B. Jiang, B. D. O. Anderson, and H. Hmam, "3-D relative localization of mobile systems using distance-only measurements via semidefinite optimization," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 56, no. 3, pp. 1903–1916, Jun. 2020, doi: 10.1109/taes.2019.2935926.

[32] Y. Liu, Y. Wang, J. Wang, and Y. Shen, "Distributed 3D relative localization of UAVs," *IEEE Trans. Veh. Technol.*, vol. 69, no. 10, pp. 11756–11770, Oct. 2020, doi: 10.1109/tvt.2020.3017162.

[33] K. Guo, X. Li, and L. Xie, "Ultra-wideband and odometry-based cooperative relative localization with application to multi-UAV formation control," *IEEE Trans. Cybern.*, vol. 50, no. 6, pp. 2590–2603, Jun. 2020, doi: 10.1109/tcyb.2019.2905570.

[34] R. Chen, B. Yang, and W. Zhang, "Distributed and collaborative localization for swarming UAVs," *IEEE Internet Things J.*, early access, Nov. 10, 2020, doi: 10.1109/jiot.2020.3037192.

[35] J.-S. Lee and C.-L. Teng, "An enhanced hierarchical clustering approach for mobile sensor networks using fuzzy inference systems," *IEEE Internet Things J.*, vol. 4, no. 4, pp. 1095–1103, Aug. 2017, doi: 10.1109/jiot.2017.2711248.

[36] H. El Alami and A. Najid, "Energy-efficient fuzzy logic cluster head selection in wireless sensor networks," in *Proc. Int. Conf. Inf. Technol. Organizations Develop. (IT4OD)*, Fez, Morocco, Mar. 2016, pp. 1–7, doi: 10.1109/it4od.2016.7479300.

[37] H. El Alami and A. Najid, "ECH: An enhanced clustering hierarchy approach to maximize lifetime of wireless sensor networks," *IEEE Access*, vol. 7, pp. 107142–107153, 2019, doi: 10.1109/access.2019.2933052.

[38] C. Lin, G. Han, X. Qi, J. Du, T. Xu, and M. Martinez-Garcia, "Energy-optimal data collection for UAV-aided industrial WSN-based agricultural monitoring system: A clustering compressed sampling approach," *IEEE Trans. Ind. Informat.*, early access, Sep. 30, 2020, doi: 10.1109/tii.2020.3027840.

[39] J. Liu, P. Tong, X. Wang, B. Bai, and H. Dai, "UAV-aided data collection for information freshness in wireless sensor networks," *IEEE Trans. Wireless Commun.*, early access, Dec. 8, 2020, doi: 10.1109/twc.2020.3041750.

[40] M. Y. Arafat and S. Moh, "A survey on cluster-based routing protocols for unmanned aerial vehicle networks," *IEEE Access*, vol. 7, pp. 498–516, 2019, doi: 10.1109/access.2018.2885539.

[41] N. Shi and X. Luo, "A novel cluster-based location-aided routing protocol for UAV fleet networks," *Int. J. Digit. Content Technol. Appl.*, vol. 6, no. 18, pp. 376–383, Oct. 2012, doi: 10.4156/jdcta.vol6.issue18.45.

[42] C. Zang and S. Zang, "Mobility prediction clustering algorithm for UAV networking," in *Proc. IEEE GLOBECOM Workshops (GC Wkshps)*, Dec. 2011, pp. 1158–1161, doi: 10.1109/glocomw.2011.6162360.

[43] F. Aadil, A. Raza, M. Khan, M. Maqsood, I. Mehmood, and S. Rho, "Energy aware cluster-based routing in flying ad-hoc networks," *Sensors*, vol. 18, no. 5, p. 1413, May 2018, doi: 10.3390/s18051413.

[44] A. Khan, F. Aftab, and Z. Zhang, "BICSF: Bio-inspired clustering scheme for FANETs," *IEEE Access*, vol. 7, pp. 31446–31456, 2019, doi: 10.1109/access.2019.2902940.

[45] A. Khan, F. Aftab, and Z. Zhang, "Self-organization based clustering scheme for FANETs using glowworm swarm optimization," *Phys. Commun.*, vol. 36, Oct. 2019, Art. no. 100769, doi: 10.1016/j.phycom.2019.100769.

[46] S. Bhandari, X. Wang, and R. Lee, "Mobility and location-aware stable clustering scheme for UAV networks," *IEEE Access*, vol. 8, pp. 106364–106372, 2020, doi: 10.1109/access.2020.3000222.

[47] Y. Jin, Z. Qian, and W. Yang, "UAV cluster-based video surveillance system optimization in heterogeneous communication of smart cities," *IEEE Access*, vol. 8, pp. 55654–55664, 2020, doi: 10.1109/access.2020.2981647.

[48] W. You, C. Dong, X. Cheng, X. Zhu, Q. Wu, and G. Chen, "Joint optimization of area coverage and mobile-edge computing with clustering for FANETs," *IEEE Internet Things J.*, vol. 8, no. 2, pp. 695–707, Jan. 2021, doi: 10.1109/jiot.2020.3006891.

[49] T. Duan, W. Wang, T. Wang, X. Chen, and X. Li, "Dynamic tasks scheduling model of UAV cluster based on flexible network architecture," *IEEE Access*, vol. 8, pp. 115448–115460, 2020, doi: 10.1109/access.2020.3002594.

[50] F. Zhou and R. Q. Hu, "Computation efficiency maximization in wireless-powered mobile edge computing networks," *IEEE Trans. Wireless Commun.*, vol. 19, no. 5, pp. 3170–3184, May 2020, doi: 10.1109/twc.2020.2970920.

[51] X. Zhang, Y. Zhong, P. Liu, F. Zhou, and Y. Wang, "Resource allocation for a UAV-enabled mobile-edge computing system: Computation efficiency maximization," *IEEE Access*, vol. 7, pp. 113345–113354, 2019, doi: 10.1109/access.2019.2935217.

[52] J. Zhang, L. Zhou, F. Zhou, B.-C. Seet, H. Zhang, Z. Cai, and J. Wei, "Computation-efficient offloading and trajectory scheduling for multi-UAV assisted mobile edge computing," *IEEE Trans. Veh. Technol.*, vol. 69, no. 2, pp. 2114–2125, Feb. 2020, doi: 10.1109/tvt.2019.2960103.

[53] Y. Sun, D. Xu, D. W. K. Ng, L. Dai, and R. Schober, "Optimal 3D-trajectory design and resource allocation for solar-powered UAV communication systems," *IEEE Trans. Commun.*, vol. 67, no. 6, pp. 4281–4298, Jun. 2019, doi: 10.1109/tcomm.2019.2900630.

[54] Y. Cai, Z. Wei, R. Li, D. W. K. Ng, and J. Yuan, "Joint trajectory and resource allocation design for energy-efficient secure UAV communication systems," *IEEE Trans. Commun.*, vol. 68, no. 7, pp. 4536–4553, Jul. 2020, doi: 10.1109/tcomm.2020.2982152.

[55] D. W. Casbeer, D. B. Kingston, R. W. Beard, and T. W. McLain, "Cooperative forest fire surveillance using a team of small unmanned air vehicles," *Int. J. Syst. Sci.*, vol. 37, no. 6, pp. 351–360, May 2006, doi: 10.1080/00207720500438480.

[56] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014, doi: 10.1016/j.advengsoft.2013.12.007.

[57] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Trans. Wireless Commun.*, vol. 1, no. 4, pp. 660–670, Oct. 2002, doi: 10.1109/twc.2002.804190.

[58] H. X. Pham, H. M. La, D. Feil-Seifer, and M. C. Deans, "A distributed control framework of multiple unmanned aerial vehicles for dynamic wildfire tracking," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 50, no. 4, pp. 1537–1548, Apr. 2020, doi: 10.1109/tsmc.2018.2815988.

[59] H. Duan, Q. Luo, Y. Shi, and G. Ma, "Hybrid particle swarm optimization and genetic algorithm for multi-UAV formation reconfiguration," *IEEE Comput. Intell. Mag.*, vol. 8, no. 3, pp. 16–27, Aug. 2013, doi: 10.1109/mci.2013.2264577.

[60] N. Xing, Q. Zong, L. Dou, B. Tian, and Q. Wang, "A game theoretic approach for mobility prediction clustering in unmanned aerial vehicle networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 10, pp. 9963–9973, Oct. 2019, doi: 10.1109/tvt.2019.2936894.

[61] (2019). *MathWorks—Makers of MATLAB and Simulink*. Accessed: Jan. 11, 2019. [Online]. Available: https://www.mathworks.com

MUHAMMAD YEASIR ARAFAT received the B.Sc. degree in electronics and telecommunication engineering and the M.Sc. degree in computer networks and communication from Independent University, Bangladesh, in 2011 and 2014, respectively, and the Ph.D. degree in computer engineering from Chosun University, South Korea, in 2020. From 2011 to 2016, he has worked as the Systems Manager with Amber IT Ltd., Bangladesh. Since 2017, he has been a member with the Mobile Computing Laboratory, Chosun University. Since November 2020, he has been an Assistant Professor with the Department of Computer Engineering, Chosun University. His current research interests include ad hoc networks, unmanned aerial vehicle networks, and cognitive radio sensor networks, with a focus on network architectures and protocols. He was the Korean Government Scholarship Program (KGSP) Grantee and received the Minister Award from the Ministry of Education, South Korea, for his excellent study.

SANGMAN MOH (Member, IEEE) received the M.S. degree in computer science from Yonsei University, South Korea, in 1991, and the Ph.D. degree in computer engineering from the Korea Advanced Institute of Science and Technology (KAIST), South Korea, in 2002. Since late 2002, he has been a Professor with the Department of Computer Engineering, Chosun University, South Korea. From 2006 to 2007, he was on leave at Cleveland State University, USA. Until 2002, he was with the Electronics and Telecommunications Research Institute (ETRI), South Korea, where he has been serving as the Project Leader, since 1991. His research interests include mobile computing and networking, ad hoc and sensor networks, cognitive radio networks, and parallel and distributed computing systems. He is a member of ACM, IEICE, KIISE, IEIE, KIPS, KICS, KMMS, IEMEK, KISM, and KPEA.