Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.039363

Article

Submarine Hunter: Efficient and Secure Multi-Type Unmanned Vehicles

Halah Hasan Mahmoud[1], Marwan Kadhim Mohammed Al-Shammari[1], Gehad Abdullah Amran[2,3,*], Elsayed Tag eldin[4,*], Ala R. Alareqi[5], Nivin A. Ghamry[6], Ehaa ALnajjar[7] and Esmail Almosharea[8]

[1]Computer Center, University of Baghdad, Baghdad, 6751, Iraq
[2]Department of Management Science and Engineering, Dalian University of Technology, Liaoning, Dalian, 116024, China
[3]Department of Information Technology Faculty of Computer Sciences and Information Technology, AL Razi University, Sana'a, Yemen
[4]Faculty of Engineering and Technology, Future University in Egypt, New Cairo, 11835, Egypt
[5]Faculty of Engineering Mechanics, Dalian University of Technology, Liaoning, Dalian, 116024, China
[6]Faculty of Computers and Artificial Intelligence, Cairo University, Giza, Egypt
[7]Department of Business Administration, Dalian University of Technology, Liaoning, Dalian, 116024, China
[8]College of Software Engineering, Dalian University of Technology, China

*Corresponding Authors: Gehad Abdullah Amran. Email: jehad.westran@gmail.com; Elsayed Tag eldin. Email: elsayed.tageldin@fue.edu.eg

Received: 25 January 2023; Accepted: 14 April 2023; Published: 09 June 2023

Abstract: Utilizing artificial intelligence (AI) to protect smart coastal cities has become a novel vision for scientific and industrial institutions. One of these AI technologies is using efficient and secure multi-environment Unmanned Vehicles (UVs) for anti-submarine attacks. This study's contribution is the early detection of a submarine assault employing hybrid environment UVs that are controlled using swarm optimization and secure the information in between UVs using a decentralized cybersecurity strategy. The Dragonfly Algorithm is used for the orientation and clustering of the UVs in the optimization approach, and the Re-fragmentation strategy is used in the Network layer of the TCP/IP protocol as a cybersecurity solution. The research's noteworthy findings demonstrate UVs' logistical capability to promptly detect the target and address the problem while securely keeping the drone's geographical information. The results suggest that detecting the submarine early increases the likelihood of averting a collision. The dragonfly strategy of sensing the position of the submersible and aggregating around it demonstrates the reliability of swarm intelligence in increasing access efficiency. Securing communication between Unmanned Aerial Vehicles (UAVs) improves the level of secrecy necessary for the task. The swarm navigation is based on a peer-to-peer system, which allows each UAV to access information from its peers. This, in turn, helps the UAVs to determine the best route to take and to avoid collisions with other UAVs. The dragonfly strategy also increases the speed of the mission by minimizing the time spent finding the target.

Keywords: Unmanned vehicles; unmanned aerial vehicles; unmanned underwater vehicles; high altitude unmanned aerial vehicles; anti-submarine warfare; re-fragmentation; dragonfly algorithm

1 Introduction

The anti-submarine (submarine hunter) is an efficient and secure unmanned vehicle network based on the dragonfly strategy. The Dragonfly strategy is a novel approach to the development of anti-submarine warfare (ASW) systems. It is based on the use of many autonomous unmanned vehicles (UVs) as the main offensive platform. The UVs are organized in a distributed network and intelligently coordinate their operations. Each UV is equipped with an advanced autonomous navigation system, acoustic sensors, and weapons. The UVs can be deployed in swarms to search for and track submarines while avoiding detection and engagement by friendly forces. The UVs can also be used to launch offensive strikes against detected targets. The Dragonfly strategy provides a high degree of flexibility and efficiency in the deployment and operation of anti-submarine forces. This UV network has different types of drones that can operate with autonomous behavior, communicating via the wireless network. The goal of this network is to protect coastal cities from submarines or enemy boats and conduct intelligence missions. The main components of the system are a control center represented by HA-UAV and several drones (UUV and UAV) equipped with cameras and other sensors. The control center is responsible for the management and operation of the entire system. It communicates with drones and receives data from them. It also allows the user to control the system, such as setting flight paths, monitoring data, and controlling the drones. The drones are the main components of the system. They are equipped with cameras, sensors, and other equipment that allows them to gather data and information. The data is then sent to the control center. The drones are also capable of autonomous flight and can be programmed with specific flight paths and missions. The system also includes other components, such as communication systems, navigation systems, and computer processors. These components help the control center and drones communicate and interact with each other, as well as help the drones to navigate and complete their missions. These drones travel above and under the water surface and can communicate with each other using their network layer security scheme that guarantees a secure and efficient communication mechanism. This autonomous system can scan the surrounding area and identify potential threats constantly. The control center analyzes the data sent by the drones and issues commands to them to neutralize the threat. If a threat is detected, the commander can issue an order to the drone fleet to launch an attack against the target vessel. This system has the capability of tracking multiple targets simultaneously and it can monitor a 360-degree field of view. Furthermore, the system is also able to identify smaller vessels operating in the area, such as small boats and floating debris. It can use this information to identify potential sources of danger. In addition, the device can be used to track marine mammals such as whales and use its sonar capabilities to detect other threats that may be present in the water. Consider the threat of having a fast Submarine close to a city capable of performing multi duties such as intelligence observation, troop support, target capacity evaluation, and battle damage evaluation. Submarines used in military strikes and reconnaissance present a significant threat to surface targets due to their capacity to carry massive and guided weapons, maneuvers, and concealment. The main target of the study is to establish an early alerting system with flexibility and variety in the reconnaissance movement, with the duty of alerting and confronting the submarine as the first line of defense. There are many studies about anti-submarine warfare (ASW). This study concentrated on prior studies that employ Unmanned Aerial Vehicles (UAV) and Unmanned Underwater Vehicles (UUV) to determine logistical targets and swarm strategies for clustering around this target. Some researchers have employed the Particle Swarm

Optimization (PSO) technique to plan the route of Unmanned Aerial Vehicles (UAVs) in a hostile, turbulent environment with unknown hazards. One of the research projects employed the Voronoi and Dijkstra technique [1] to calculate the beginning route of the PSO algorithm. Another study utilized the chaos-based logistic map [2] on PSO. The third research improved Membership Functions (MFs) using the sophisticated Adaptive Neuron Fuzzy Inference System (ANFIS) [3]. This research did not address the vulnerability in data transmission and the limits of terrain and weather. Some research used a 3D path-planning algorithm based on PSO for a more flexible environment for UAVs [4,5]. In addition to the localization and clustering of UAVs based on the PSO that has meant to cover far regions [6]. In addition, the combination of Hybrid PSO and Genetic Algorithm PSO produces a new vision of swarm optimization, but all methods above still use fitness functions (V_{best}, G_{best}) to find the right direction [7]. Depending on its behavior, the Dragonfly Algorithm (DA) employs more parts to adjust the swarm's trajectory (Collision, Avoidance, Pairing, Aggregation, and Foraging). Some researchers focused on the DA from the aspect of directing UAVs, with some of these studies employing the Dragonfly Algorithm in offensive guidance [8]. While prior research has merged DA and Machine Learning (ML) to improve Fly Ad-Hoc Network (FANET) management to cover huge regions with fewer isolated UAVs, this work has introduced additional complexity to the FANETs' mobility structure [9]. Some research has also developed countermeasures against UAVs in the military environment inspired by DA to confound the opponent and avoid hazards [10]. In one study, Blockchain preserves and distributes crowd data in UAVs used for crowd surveillance in anomalous actions [11]. While several researchers examined the idea of accumulating data in a secure hierarchical manner by proposing a non-fixed scheduling mechanism to manage network nodes, they depended on the DA [12]. There is also a discussion of employing optimization theories and wireless millimeter waves with a stationary base station to send sensitive data between UAVs [13]. Some researchers utilize heterogeneous networks (HetNets) with UAVs in a military scenario for reconnaissance in the studies can be summarized as enemy territory [14,15] or with the aid of Reconfigurable Intelligent surfaces (RISs) [16]. Other research has proposed leveraging ground-based mobility platforms as MVs to secure data communication using a Genetic Algorithm (GA) between Vehicles Delay-Carry Networks (VDTNs) and UAVs [17]. A method based on dynamic Bayesian network modeling was proposed to evaluate the UUV in an underwater threat situation [18]. To obtain better endurance performance of unmanned underwater vehicles under limited resources [19] propose a three-level optimization strategy for the fuel cell hybrid system [20]. Introduce a 30-day operational UUV fuel cell-battery system under development. The algorithm proposed provides a new way for autonomous path planning of underwater vehicles [21,22] presents a three-dimensional control algorithm using reinforcement learning to guide an attacking hunter drone capable of performing a global navigation satellite systems (GNSS) repeater attack on the GNSS receiver of a target invader drone. Reference [23] Deal with the Unmanned Aircraft Vehicle (UAV) flight phases dynamics simulation. A novel bullet shapes Trans-Domain Amphibious Vehicle (TDAV) is proposed which achieves free trans-domain motion and has the advantages of small size, high maneuverability, and high reliability for both rotary-wing UAV and Autonomous Underwater Vehicle (AUV) operation [24]. An electromagnetic immune Free Space Optical Communication (FSOC) system for an Unmanned Aerial Vehicle (UAV) command and control link is introduced [25]. Other influential work includes [26,27]. In some research, two hybrid models of Ant Colony Optimization were compared concerning convergence time, i.e., the Max-Min Ant Colony Optimization approach in conjunction with the Differential Evolution and Cauchy mutation operators [28]. In some previous scenarios, boosted wireless signals in remote areas would require a mobile signal booster, and the data secured by a secure database [29]. Regarding the stability of drones, some works of literature present an intelligent control design for the helical trajectory tracking of an under-actuated quadrotor [30]. The prior studies can be summarized in Table 1. The significance of this work emerges from anti-submarine alert systems employing UAVs, UUVs, and High-Attitude UAVs, which are considered two main factors.

1. Optimize the efficiency of Unmanned Vehicles (UAVs, UUVs) in terms of localization and clustering to achieve the best defensive formation in the least amount of time. The dragonfly Algorithm finds a submarine and treats it as a military target.
2. Secure the information shared between Unmanned Vehicles UVs and High-Attitude UAVs to prevent the enemy from knowing their whereabouts and formations. To secure data between UVs and HA-UAVs cybersecurity approach has been suggested Re-fragmentation achieved cybersecurity understanding in this paper by changing the fragment ID index in the packet Trailer.

Table 1: Prior studies analysis

References	Methods	Limitations
[4,5]	3D path planning algorithm based on PSO	Flexibility for UAVs
[6]	Localization and clustering of UAVs based on the PSO	Coverage area limitation
[7]	Hybrid PSO and Genetic Algorithm	A new vision of swarm optimization with fitness accuracy
[8]	DA to directing UAVs	Offensive guidance limitation
[9,10]	DA and ML to improve FANET and anti UAVs	Complexity
[11–13]	Blockchain, and DA to secure UAVs data	Crowd data
[14–16]	HetNets and RISs with UAVs in a military scenario	Reconnaissance in enemy territory
[17]	Proposed leveraging ground-based mobility platforms as MVs to secure data communication using a Genetic Algorithm (GA) between vehicles delay-carry networks (VDTNs) and UAVs	Crowd data

This study hypothesized that the dragonfly approach should achieve four improvements in the system:

- Improved drones' ability to evasive and change location.
- Improving the ability of drones to stay as long as possible in the air by regulating the energy.
- Quickly discover the enemy using the Five Dragonfly Factors.
- Securing the data transmitted over the network.

Indeed, many factors need to study. The battery timing of a UAV is a factor in determining the amount of time it can stay in the air and it is important to consider the various dynamic factors that can affect the battery life, such as temperature, wind speed, and air turbulence. Additionally, the type of UAV and its design also play a role in how long its battery can last. To improve the ability of drones to stay in the air for a longer period, it is necessary to consider all of these factors, as well as the power management and optimization of the UAV itself.

The remainder of the paper's structure is as follows. Section 2 presents the theoretical, then Material, and Methods related to the proposed algorithm. The Dragonfly Algorithm for Optimization and Network Layer Security cooperation to optimize the location and clustering of UVs introduced in Section 3 as the method's implementation, and the results and discussion are presented in Section 4. Finally, Section 5 has established the conclusion. In addition, Table 2 shows all abbreviations with descriptions.

Table 2: Abbreviation description

Abbreviations	Description
AI	Artificial intelligence
UVs	Unmanned vehicles
UAVs	Unmanned aerial vehicles
ASW	Anti-submarine warfare
UUVs	Unmanned underwater vehicles
PSO	Particle swarm optimization
MFs	Membership functions
ANFIS	Adaptive neuro-fuzzy inference system
DA	Dragonfly algorithm
ML	Machine learning
FANET	Fly Ad-Hoc network
HetNets	Heterogeneous networks
RISs	Reconfigurable intelligent surfaces
MVs	Mobile vehicles
GA	Genetic Algorithm
VDTNs	Vehicles delay-carry networks
HA-UAVs	High attitude unmanned aerial vehicles

2 Material and Methods

This portion of the study describes the methodology and the preliminary related to the suggested scheme. The section focused on two matters:

1. Illustrated the process of achieving a quick localization and clustering of autonomous vehicles (UVs) toward the anti-submarine utilizing the dragonfly Algorithm.
    - The UVs in the vicinity of the anti-submarine need to be identified. This can be done by using sensors to detect the presence of the anti-submarine.
    - Once identified, the UVs will need to be clustered based on their nearest distance to the anti-submarine. This can be done with the dragonfly algorithm, which divides the area around the anti-submarine into a set of circular sectors.
    - Each UV is allocated to a sector based on its proximity to the anti-submarine. This will result in a quick localization of the UVs.
    - The UVs in each sector can then be clustered by using the dragonfly algorithm. This will help to reduce the number of UVs in each sector and improve the overall efficiency of the system.
    - The UVs that are clustered can then be sent to the anti-submarine to begin the search and destroy mission.

    - The UVs can also communicate with each other to share information and coordinate their movements. This ensures that they are working as a unified force and not just as individual units.

2. Illustrated the data-security mechanism to secure the communication of a swarm of autonomous vehicles.
    - Encryption: Encryption is a key data security mechanism for securing communication between autonomous vehicles. Encryption ensures that the data transmitted between vehicles is secure and cannot be accessed or modified by outsiders. This is done by encrypting the data being transmitted with a secret key.
    - Authentication: Authentication is another important data security mechanism for autonomous vehicles. Authentication ensures that only authorized vehicles can access the data being transmitted. This is done by verifying the identity of the vehicle before allowing access.
    - Access Control: Access control is a data security mechanism used to regulate access to data. Access control ensures that only authorized vehicles are allowed to access the data being transmitted. This is done by defining rules that regulate access to the data based on the identity of the vehicle.
    - Firewalls: Firewalls serve as a barrier between vehicles and the outside world. Firewalls are used to prevent unauthorized access to the data being transmitted between vehicles.
    - Intrusion Detection: Intrusion detection is a data security mechanism used to detect and respond to malicious activity. Intrusion detection systems are used to detect any attempts to access the data without authorization. When an intrusion is detected, the system will respond by taking appropriate actions.

2.1 Dragonfly Algorithm for Optimization

The dragonfly algorithm (DA) is one of the algorithms that utilize the swarm intelligence principle to imitate the behavior of a dragonfly colony on UVs. Fig. 1 illustrates the procedure of UVs that use DA. From the Unified Modeling Language (UML) flowchart, there are two operations after finding the Submarine by one of any UVs [31]. The algorithm is divided into two steps as follows:

1. Localization: The UVs congregate in small groups, allowing them to investigate various environments for a specific target. Position and velocity data are kept and updated on a semi-permanent and completely secured database. This step is used to explore the environment and search for the submarine. The UVs, with the help of DA, use the exploration behavior of dragonflies. In the exploration procedure, each UVs explores the environment, and if it finds a better solution than it is previous one, it updates its position and the best new solution.
2. Clusterization: Whenever one of the small groups of UVs reaches the target, the leader UVs immediately transmit the position and velocity data to the High Attitude Un-manned Aerial Vehicles (HA-UAV). The HA-UAV's job is to broadcast the leader group's data to the remainder of the UVs colony and allow the Swarm to move immediately to the leader group to support them. Fitness is the quickest way to achieve this scenario. After finding the submarine, the UVs enter the exploitation procedure. In this step, the UVs will start to search for the best path to reach the submarine. The exploitation behavior is inspired by the hunting behavior of dragonflies. The UVs will use their current position and the position of the submarine to generate a path with the lowest energy consumption.

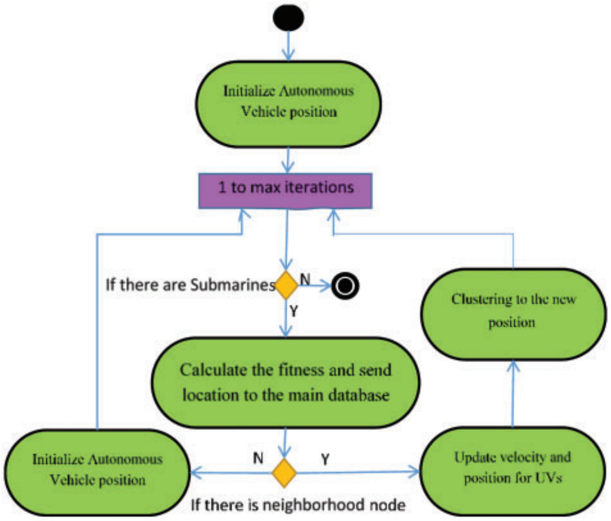

Initialize Autonomous Vehicle position
1 to max iterations
If there are Submarines
N
Y
Calculate the fitness and send location to the main database
Clustering to the new position
Initialize Autonomous Vehicle position
N
Y
Update velocity and position for UVs
If there is neighborhood node

Figure 1: The UML flowchart of the dragonfly algorithm for UV

The major advantage of DA is its scalability and robustness, which makes it suitable for large-scale tasks. Additionally, DA is a self-organizing algorithm, meaning that it does not require any prior knowledge or information about the environment. Furthermore, it can adapt to different environments and optimize the search process [32].

2.2 *Dragonfly Optimization*

In this section, the Dragonfly Method is utilized to show how UVs detect submarine locations and cluster around them. Where the first step in the Dragonfly Method is to deploy an array of UVs that can detect the presence of a submarine. The UVs can be deployed in a grid-like pattern or a circular formation, depending on the situation. After the UVs have been deployed, they need to be programmed to scan the surrounding environment for any sign of a submarine. This can be done using sonar or other acoustic methods. Once a submarine has been detected, the UVs will cluster around the submarine to better detect its location and movements. This clustering behavior is a key part of the Dragonfly Method. The UVs will then use their clustering behavior to track the submarine's movements and use this information to relay back to a command center [33,34].

2.2.1 UV Localization and Clustering Procedure

UVs are scattered at random locations and form scattered groups. Then the target is detected by some UVs using sensors on each UV. Initialization of parameters for the Dragonfly algorithm begins. It depends on the energy and distance to find the closest UV to the target. The localization process starts when the Dragonfly algorithm is applied to select the effective UV-Leader for each group. The filtering process continues repeated to reach the appropriate leader-UV. Then the clustering process starts whereby the candidate's Leader-UV information send to all network members. The HA-UAV collects data from the UVs and forwards it to the Leader-UV.

2.2.2 UV Mathematical Components

The behavior of the Dragonfly swarm uses to optimize UVs networks. The Dragonfly colonies fly in small swarms in different directions, then form larger swarms and fly in the same direction. The optimization of UVs networks is accomplished by Dragonfly swarms using a combination of cooperative behaviors, such as collective decision-making, collective intelligence, and distributed problem-solving. The swarms search for the best solution to a problem and communicate with each other to make decisions. The swarms also use a local search algorithm to identify the most optimal route and optimize the UVs network. This optimization process helps reduce energy consumption and improve the overall efficiency of the network. The Dragonfly uses Separation as in Eq. (1), Alignment as in Eq. (2), Cohesion as in Eq. (3), Attraction to food as in Eq. (4), and Distraction from the enemy as in Eq. (5), strategies to update its position as shown in Eq. (6). The next position calculation to UV comes from Eq. (7), where (t) is the current iteration. Using random positions, the Dragonfly finds the adjacent position using Eq. (8).

$$S_i = \sum_{i-1}^{N} X - X_i \tag{1}$$

$$A_i = \frac{\sum_{i-1}^{N} V_i}{N} \tag{2}$$

$$C_i = \frac{\sum_{i-1}^{N} X_u}{N} - X \tag{3}$$

$$F_i = X^{+} - X \tag{4}$$

$$E_i = X^{-} - X \tag{5}$$

$$\Delta X_{t+1} = (s\ S_i + a\ A_i + c\ C_i + f\ F_i + e\ E_i) + w \Delta X_t \tag{6}$$

$$X_{t+1} = X_t + \Delta X_{t+1} \tag{7}$$

$$X_{t+1} = X_t + (d) *\ X_t \tag{8}$$

where:

i; dragonfly population

X; position of the current individual

Xi; position of the i^{th} neighboring

N; number of neighboring

Vi; velocity of the individual

X^{+}; position of the food

X^{-}; position of the enemy

s; separation weight

a; alignment weight

c; cohesion weight

f; food factor

e; enemy factor

w; individual weight

t; iteration counter

d; position vector

2.2.3 UV Testing Benchmarks

The fitness of a leader-UV is computed by computing the sum of the energy and distance values between the UV and the leader-UV. The energy value is calculated by measuring the number of communication hops between the UV and the leader UV. The distance value is computed by measuring the physical distance between the UV and the leader UV. The fitness of a leader-UV is then computed by summing the energy and distance values. The leader UV with the lowest fitness value is then selected as the leader. The fitness is computed by using the distance and energy values. Leader-UVs were selected by obtaining the minimum fitness as shown in Eq. (9). Eq. (10) calculates the energy value between UV and Leader-UV and Eq. (11) calculates the distance.

$$f = a f_1 + (1 - a)\ f_2 \tag{9}$$

$$f_1 = \frac{\sum_{i=1}^{M} E\left(n_i\right)}{\sum_{j=1}^{N} E\left(Leader - UV_j\right)} \tag{10}$$

$$f_2 = min \left\{ \sum_{kC_j} d\left(n_i,\ Leader - UV_j\right) \middle| \ C_j \right\} \tag{11}$$

where:

f; fitness function

f_1; energy between UV and Leader-UV

f_2; distance between UV and Leader-UV

Leader-UV; best position

2.3 Network Layer Security

One of the methods of cybersecurity where the operation takes place at the network layer, also known as the secure socket layer, is the method of re-fragmenting (assembling) packets and reassembling them between the sending and receiving point-to-point ends in a predetermined order. This method is known as packet reassembly, and it helps to protect the integrity of the data being transmitted by ensuring that all the packets are received in the correct order. This helps to prevent attackers from intercepting the data and using it for malicious purposes. Additionally, packet reassembly can provide an additional layer of security by allowing the sender to include a code in the packet that can be used to verify that it originated from the intended sender. The security mechanism depends on the sequence of the receiving segments formed by the Transport layer in the TCP/IP protocol, as shown in Fig. 2. The security mechanism uses the sequence of receiving segments to ensure the integrity of data. The receiving segments are marked with sequence numbers. When a segment is received, the sequence number is compared to the previously received segment to verify that the data has not been corrupted. If the sequence number matches the previous segment, the data is assumed to be intact and accepted, otherwise it is discarded. The sequence of receiving segments also enables the receiver to detect and discard duplicate segments. If a segment is received with the same sequence number as a previously received segment, it is assumed to be a duplicate and discarded. This security mechanism is used to help protect against malicious attacks such as man-in-the-middle, replay, and packet injection attacks. By verifying the sequence numbers, the receiver can detect and discard any maliciously modified or replayed segments. This technique is a successful defense against potential assaults that aim to access the UVs' data to determine or change their positions. On the network, the Re-fragmentation represents an end-to-end access point. Using a network middleware that includes a software library capable of changing the structure of transmission in the network, a transmission sequence for the segments is first agreed upon and identified by a Fragment ID so that only the sender and receiver are aware of the transmission sequence. Re-fragmentation is the most successful method to protect wireless networks, although wireless networks are less dependable than wired networks [35].

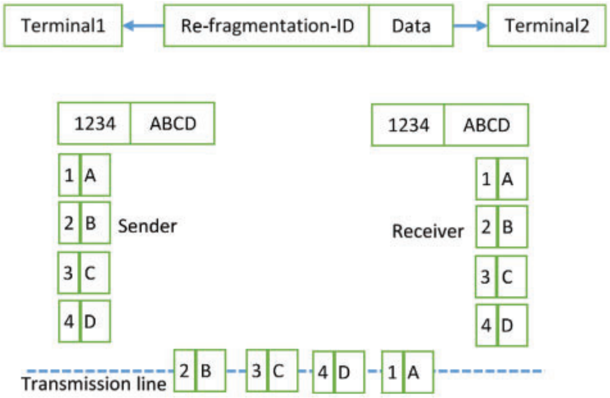


Figure 2: **A** simple block diagram to illustrate packet re-fragmentation depends on the handshaking agreement between sender and receiver

3 Implementation

3.1 Anti-Submarine Proposed System

The localization and clustering methodology enhances the UVs networks' performance in rapidly locating threatening submarines. Hence, the Leader-UV was selected for each group after the UVs divided into small groups using the dragonfly approach. Data transmission control and coordination are the principal functions of the Leader-UV. To manage the data of the surrounding group of UUVs or UAVs, the Leader-UV needs more power. Periodically, each UV sends to HA-UAV the current position. The HA-UAV must compute the nearest distance to the enemy among the UVs. The Leader UV, which represents the best location to attack the enemy submarine since the other UVs directed toward it, will be selected. During the steady state phase, the UVs gather their data and frequently put them on the HA-UAV. The UVs redirected toward the enemy submarine if one of the UVs detects an enemy submarine, and the Dragonfly algorithm uses to address the flaws. Fig. 3 shows the dragonfly protection approach.

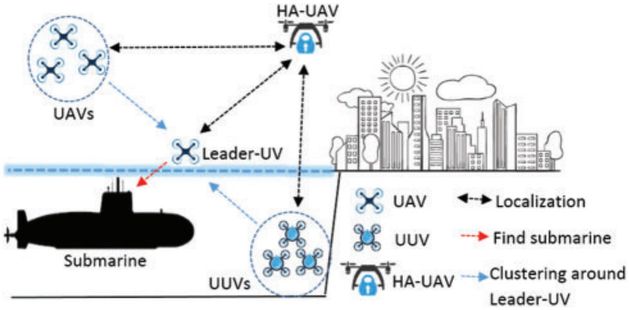


Figure 3: Submarine hunter system utilizing dragonfly approach

3.2 UV Middleware Procedure

A dragonfly is an insect with chaotic motions. A dragonfly's flight pattern is obtained from its natural behavior and consists of five main features: alignment, separation, cohesion, attraction to food, and distraction from the enemy. The current position for UV, which represent the Dragonfly in this article indicated by Xi and Xi + 1 is the next step position, while "n" represents the number of iterations. The fitness factor in the middleware represents the main factor in deciding which UV will be the leader UV, as illustrated in Table 3, which represents the procedure steps of the middleware. This is also to say that there is a subroutine to protect UVs information using a specifically agreed fragmentation index, which is added to the middleware using a (Scapy) software library. Where the procedure steps are stated with random positions in the initialization step. Through the normal case, the UVs update their positions and send them to the HA-UAV, whether if a submarine enemy is coming the nearest UV get the control line the become Leader UV, and all together UVs send their information to the HA-UAV. HA-UAV then sends an update position signal to make UVs follow Leader-UV for support.

Table 3: UVs anti-submarine middleware steps code based on dragonfly algorithm and re-fragmentation approach

```
1.    Initialize UVs (position, energy, i, and t ... n) with random values
2.    Get the position of UVs, calculate the next step position $X_i$ (i−1, ..., n)
3.      **While** a submarine is detected **do**
4.      Calculate the objective values of all UUVs, and UAVs
           Sort E edges by increasing the weight.
         T = {}
         for (i = 0; i < edgelist.length; i++)
           if adding e = edgelist[i] does not form a cycle
             add e to T
           else ignore e
5.      **Re-fragmentation on specifically agreed end-to-end index**
6.      Update HA-UAV frequently
```

(Continued)

Table 3: Continued

```
7.      Calculate fitness Eqs. (1) to (6)
          T = {s}
          enqueue edges connected to s in PQ (by inc weight)
          while (!PQ.isEmpty)
            if (vertex v linked with e = PQ.remove ∉ T)
              T = T ∪ {v, e}, enqueue edges connected to v
            else ignore e
8.      **De-fragmentation on specifically agreed end-to-end index**
9.      Update neighbors' UVs position
10.     Update neighbors' UVs energy
11.   **end**
```

The Dragon Network is made up of various components, as seen in Fig. 4.

1. Cluster head, represented by nodes (1, and 6). Cluster 1 is led by Node 1, and Cluster 2 is led by Node 6. Each node is responsible for directing the group of nodes to the most likely sink node. The titles of all nodes vary depending on their location in the concurrent scenario.
2. Cluster sink, represented by nodes (7). The submarine was discovered by node 7 for the first time. The sink node should be supported by all its neighbors.
3. Node member: which is represented by nodes (2, 3, 4, and 5). All nodes are typical nodes that follow the cluster head closest to them.
4. Backup data node: This is represented by a node (8). Ultraviolet location information is collected and updated constantly by node 8.
5. Enemy node: A node that represents an enemy (0). The target node is represented by node 0.

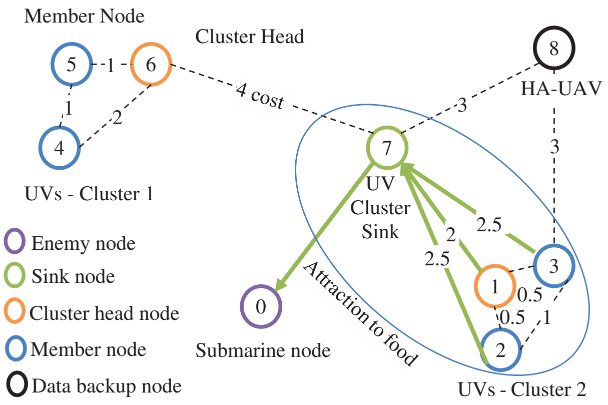


Figure 4: Dragonfly nodes network procedure

The method begins with sink node (7) when notice submarine (node (0)) delivers position information to node (8), which is HA-UAV. Next, depending on the cost, node (8) sends a support message to cluster 2 as the nearest support group. The message is sent to the cluster head (node (1)) to lead nodes (2 and 3) to support node 7.

3.3 Simulation Setup

The proposed anti-submarine scheme was implemented using NS.3 simulator. The UVs distributed in a random position. Every UV has initial energy. HA-UAV, UUV, and UAV are positioned at specific points. The UVs far from the Submarine get low energy than the UVs near the Submarine. The HA-UAVs should monitor and control the behavior of the UVs for the whole network. Table 4 presents the simulation setup and parameters.

Table 4: Simulation setup parameters

Parameter	value
Environment size	$1000 \times 1000\,\mathrm{m}^2$
Number of UUV	500
Number of UAV	500
Number of HA-UAV	10
UUV location	50, 160
UAV location	30, 175
HA-UAV location	20,150
Simulation time	3600 s
Initial energy	1–5 J
Transmission range	1750 m
Packet size	500 bytes
Send/receive energy	50 nJ/bit
Energy for data aggregation	5 nJ/bit
Energy for dissipation	10 pJ/bit

4 Results and Discussion

The proposed algorithm is simulated using NS 3.0, where the efficiency of UVs in dealing with enemy submarines is compared. In the suggested scenario, it assumed that an enemy submarine was detected at the time of start. Fig. 5 shows the flight route of the UV maneuvering and attacking a submarine using the Dragonfly approach. In the figure, the straight line represents the UV's itinerary, while the points on the line represent the UV's detection and attack points for the submarine. In contrast, the dotted straight line represents the submarine's itinerary. The UV's movements show an irregular flight pattern as it is difficult to predict the locations and assemblies of the UVs compared to the regular flight pattern. Accordingly, it is proven that the proposed algorithm increased the ability to evasive enemy submarines and their weapons. Based on the performance measurement, the average distance between the UVs and the submarine is 27.55 m, while the standard deviation is 16.2 m. That indicates the extent of the possible evasiveness of the UAV under the use of the dragonfly approach. Nonetheless, the energy consumption increases with the flight path increase and the risk of data security, which is what the HA-UAV handles. Fig. 6a shows a comparison of the energy consumption between UAVs in the hypothetical case and in the case of using the dragonfly approach. In a typical case, the energy of UV is active for 370.5 s, while it reaches 673 s after using the dragonfly approach. The proposed approach increases the efficiency of energy conservation for a longer period. Fig. 6b shows the amount of energy consumed during the simulation period. The potential of the Dragonfly algorithm as a low-power solution for data processing is promising. Future studies can explore more about this algorithm and its implementation in different scenarios. Also, the performance of the algorithm needs to be evaluated further in different applications. There is still a need for further study to identify the best approach for optimizing the power consumption of the algorithm. Furthermore, the scalability of the algorithm should be studied to determine its potential in large-scale distributed systems. Finally, the security and privacy of data should be considered while designing and implementing the algorithm.

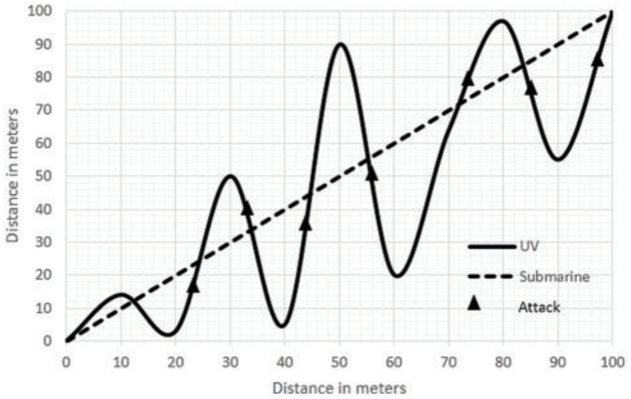
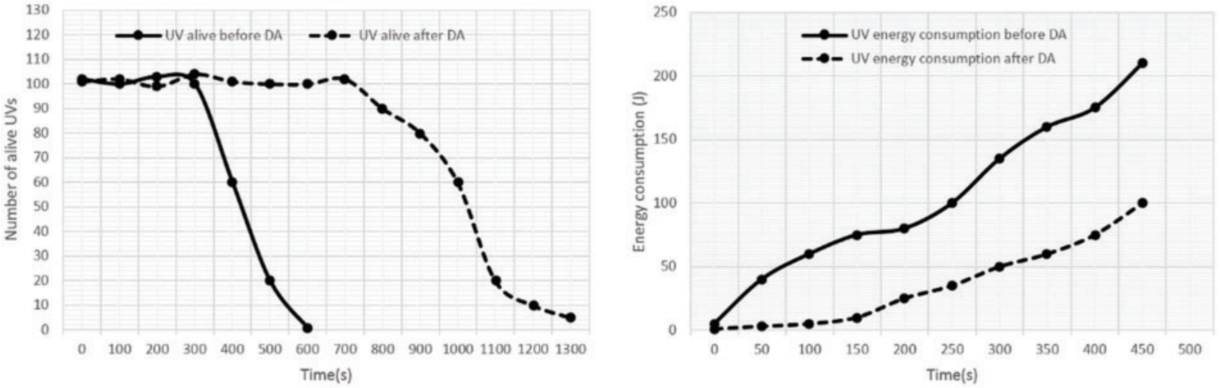


Figure 5: Flight pattern comparison between linear submarine movement and dragonfly base UV movement



(a) (b)

Figure 6: (a) Number of alive UVs concerning time, (b) energy consumption

The graph below illustrates the efficiency of the three optimization methods for UVs management optimization. The x-axis represents the interaction for each optimization method, the y-axis represents the efficiency. As shown in Fig. 7, the Dragonfly Optimization (DO) method is the most efficient, followed by the Ant Colony Optimization (ACO) and Bee Colony Optimization (BCO) methods. The DO method achieved an efficiency of 97%, while the ACO and BCO methods achieved an efficiency of 95% and 80%, respectively.

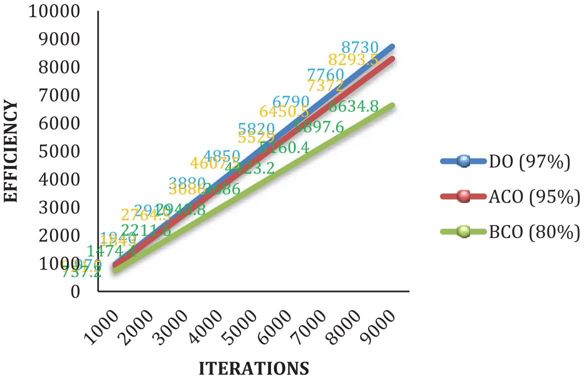


Figure 7: Efficiency comparison among dragonfly optimization (DO), ant colony optimization (ACO), and bee colony optimization (ACO)

5 Conclusion

This study proposes a defensive and evasive algorithm against hostile submarines using UVs based on the natural behavior of dragonflies. The proposed algorithm allows UVs to counter enemy submarine threats by moving in an unexpectedly defensive manner. That is through the benefits of the dragonfly in evasion, which combines the strategy of separation, cohesion, alignment, and maneuvering, thus distracting the enemy's attention and attacking the submarine. The defensive process begins with determining the Leader-UV, which is responsible for driving the rest of the UVs, with the presence of the HA-UAV to manage and protect the data in the proposed Dragonfly network. The simulation results in the proposed approach show that the displacement value is 27.55 meters with a standard deviation of 16.2, which means it the difficulty to predict the accumulation of UVs in the evasive and attack phase. In addition, the work-energy between the UVs distributed uniformly between the UVs and depending on the HA-UAV. The HA-UAV organizes the data security and directs the proposed squadron in an orderly and fast formation. Despite the fulfillment of the hypotheses mentioned above, the study can be expanded by applying the methodology in a less complex computational very in the future. There is an urgent need to develop effective defensive and evasive algorithms to counteract the presence of submarines and avoid attacking them. This paper proposes an algorithm using the natural behavior of dragonflies to neutralize their threat. The method is based on the principles of interlacing aggressiveness of the dragonfly for prevention from predators and the behavior avoidance movement of stingrays through the repelling action of its tail fin to keep away its predator's attacks. The study begins with a theoretical part that describes the model and its components. The second part describes the development and evaluation of the proposed model using a simulation environment. The third part presents the conclusions resulting from the analysis conducted in this study. The maneuverability of UVs can be characterized by the speed attained by the general speed value of the system. This parameter plays an important role in many critical survival situations. It can change the behavior of an UVs swarm when it encounters a new obstacle or when it intends to avoid an attack by an enemy organism. Therefore, it is essential to study the factors that affect this parameter to determine the most effective strategies to increase their speed in a hostile environment.

Funding Statement: This work was funded by the research center of the Future University in Egypt, in 2023.

Availability of Data and Materials: The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] J. J. Shin and H. Bang, "UAV path planning under dynamic threats using an improved PSO algorithm," *International Journal of Aerospace Engineering*, vol. 2020, pp. 1–17, 2020.

[2] S. Shao, Y. Peng, C. He and Y. Du, "Efficient path planning for UAV formation via comprehensively improved particle swarm optimization," *International Society of Automation Transactions*, vol. 97, pp. 415–430, 2020.

[3] B. Selma, S. Chouraqui and H. Abouaïssa, "Fuzzy swarm trajectory tracking control of unmanned aerial vehicle," *Journal of Computational Design and Engineering*, vol. 7, no. 4, pp. 435–447, 2020.

[4] A. Mirshamsi, S. Godio, A. Nobakhti, S. Primatesta, F. Dovis *et al.,* "A 3D path planning algorithm based on PSO for autonomous UAVs navigation," in *Int. Conf. on Bioinspired Methods and Their Applications*, Brussels, Belgium, pp. 268–280, 2020.

[5] S. Konatowski and P. Pawlowski, "PSO algorithm for UAV autonomous path planning with threat and energy cost optimization," in *XII Conf. on Reconnaissance and Electronic Warfare Systems*, Oltarzew, Poland, vol. 11055, pp. 251–260, 2019.

[6] M. Y. Arafat and S. Moh, "Localization and clustering based on swarm intelligence in UAV networks for emergency communications," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8958–8976, 2019.

[7] B. Abhishek, S. Ranjit, T. Shankar, G. Eappen, P. Sivasankar *et al.,* "Hybrid PSO-HSA and PSO-GA algorithm for 3D path planning in autonomous UAVs," *Springer Nature Applied Sciences*, vol. 2, no. 11, pp. 1–16, 2020.

[8] Y. Zhang, G. Wang, X. Huang, J. Xi, Y. Dang *et al.,* "Research on task assignment of cruise ammunition cooperative attack based on dragonfly algorithm," in *Int. Conf. on Algorithms, High Performance Computing, and Artificial Intelligence (AHPCAI 2021)*, Sanya, China, vol. 12156, pp. 80–89, 2021.

[9] S. Hameed, Q. A. Minhas, S. Ahmad, F. Ullah, A. Khan *et al.,* "Connectivity of drones in FANETs using biologically inspired dragonfly algorithm (DA) through machine learning," *Wireless Communications and Mobile Computing*, vol. 2022, pp. 1–11, 2022.

[10] S. L. K. C. Gudi, B. Kim, S. Silvirianti, S. Y. Shin and S. Chae, "Bio-inspired evasive movement of UAVs based on dragonfly algorithm in military environment," *Journal of Information and Communication Convergence Engineering*, vol. 17, no. 1, pp. 84–90, 2019.

[11] W. Xiao, M. Li, B. Alzahrani, R. Alotaibi, A. Barnawi *et al.,* "A blockchain-based secure crowd monitoring system using UAV swarm," *IEEE Network*, vol. 35, no. 1, pp. 108–115, 2021.

[12] E. Yousefpoor, H. Barati and A. Barati, "A hierarchical secure data aggregation method using the dragonfly algorithm in wireless sensor networks," *Peer-to-Peer Networking and Applications*, vol. 14, no. 4, pp. 1917–1942, 2021.

[13] X. Sun, W. Yang, Y. Cai, Z. Xiang and X. Tang, "Secure transmissions in millimeter wave SWIPT UAV-based relay networks," *IEEE Wireless Communications Letters*, vol. 8, no. 3, pp. 785–788, 2019.

[14] A. Rashid, D. Sharma, T. A. Lone, S. Gupta and S. K. Gupta, "Secure communication in UAV assisted HetNets: A proposed model," in *Int. Conf. on Security, Privacy and Anonymity in Computation*, Communication and Storage, Jammu & Kashmir, India, pp. 427–440, 2019.

[15] Y. Li, R. Zhang, J. Zhang and L. Yang, "Cooperative jamming via spectrum sharing for secure UAV communications," *IEEE Wireless Communications Letters*, vol. 9, no. 3, pp. 326–330, 2019.

[16] H. Long, M. Chen, Z. Yang, B. Wang, Z. Li *et al.,* "Reflections in the sky: Joint trajectory and passive beamforming design for secure UAV networks with reconfigurable intelligent surface," arXiv preprint arXiv:2005.10559, 2020.

[17] R. Liu, A. Liu, Z. Qu and N. N. Xiong, "An UAV-enabled intelligent connected transportation system with 6G communications for internet of vehicles," *Annals of Telecommunications*, IEEE, vol. 24, no. 2, pp. 2045–2059, 2023.

[18] H. Yao, H. Wang, Y. Li, Y. Wang and C. Han, "Research on unmanned underwater vehicle threat assessment," *IEEE Access*, vol. 7, pp. 11387–11396, 2019.

[19] C. Yangyang, H. Hui, W. Xixiu and L. Weibo, "Endurance optimization of unmanned underwater vehicle based on three-level optimization strategy," in *2020 IEEE/IAS Industrial and Commercial Power System Asia (I&CPS Asia)*, Weihai, China, pp. 82–87, 2020.

[20] T. -R. Park, K. Kim and J. -H. Cho, "Operating point optimization of fuel cell-battery power system for unmanned underwater vehicle," in *2020 20th Int. Conf. on Control, Automation and Systems (ICCAS)*, Busan, Korea, pp. 928–932, 2020.

[21] Y. Ma, Z. Mao, T. Wang, J. Qin, W. Ding *et al.,* "Obstacle avoidance path planning of unmanned submarine vehicle in ocean current environment based on improved firework-ant colony algorithm," *Computers & Electrical Engineering*, vol. 87, pp. 106773, 2020.

[22] D. L. da Silva, F. Antreich, O. L. Coutinho and R. Machado, "Q-learning applied to soft-kill countermeasures for unmanned aerial vehicles (UAVs)," in *2020 IEEE/ION Position, Location and Navigation Symp. (PLANS)*, Portland, OR, USA, pp. 91–99, 2020.

[23] A. N. Sedláčková, P. Kurdel and J. Labun, "Simulation of unmanned aircraft vehicle flight precision," *Transportation Research Procedia*, vol. 44, pp. 313–320, 2020.

[24] Y. Gao, H. Zhang, H. Yang, S. Tan, T. A. Gulliver *et al.,* "Trans-domain amphibious unmanned platform based on coaxial counter-propellers: Design and experimental validation," *IEEE Access*, vol. 9, pp. 149433–149446, 2021.

[25] Y. Zhang, Y. Wang, Y. Deng, A. Du and J. Liu, "Design of a free space optical communication system for an unmanned aerial vehicle command and control link," *Photonics*, vol. 8, no. 5, pp. 163, 2021.

[26] J. Su, H. Zhang and Y. Yao, "Design of anti-submarine warfare module for unmanned surface vehicle," in *2020 3rd Int. Conf. on Unmanned Systems (ICUS)*, Harbin, China, pp. 342–345, 2020.

[27] B. Jiang, G. Huang, T. Wang, J. Gui and X. Zhu, "Trust based energy efficient data collection with unmanned aerial vehicle in edge network," *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 6, pp. e3942, 2022.

[28] M. Shafiq, Z. A. Ali, A. Israr, E. H. Alkhammash, M. Hadjouni *et al.,* "Convergence analysis of path planning of multi-UAVs using max-min ant colony optimization approach," *Sensors Journal*, vol. 22, no. 14, pp. 5395, 2022.

[29] G. A. Amran, S. Wang, M. A. A. Al-Qaness, S. A. H. Mohsan, R. Abbas *et al.,* "Efficient and secure Wi-Fi signal booster via unmanned aerial vehicles Wi-Fi repeater based on intelligence-based localization swarm and blockchain," *Micromachines Journal*, vol. 13, no. 11, pp. 1924, 2022.

[30] G. E. M. Abro, S. A. B. M. Zulkifli and V. S. Asirvadam, "Dual-loop single dimension fuzzy-based sliding mode control design for robust tracking of an underactuated quadrotor craft," *Asian Journal of Control*, vol. 25, no. 1, pp. 144–169, 2023.

[31] Ş. Gülcü, "Training of the feed forward artificial neural networks using dragonfly algorithm," *Applied Soft Computing*, vol. 124, pp. 109023, 2022.

[32] M. Alshinwan, L. Abualigah, M. Shehab, M. A. Elaziz, A. M. Khasawneh *et al.,* "Dragonfly algorithm: A comprehensive survey of its results, variants, and applications," *Multimedia Tools and Applications*, vol. 80, no. 10, pp. 14979–15016, 2021.

[33] F. Aadil, W. Ahsan, Z. U. Rehman, P. A. Shah, S. Rho *et al.,* "Clustering algorithm for internet of vehicles (IoV) based on dragonfly optimizer (CAVDO)," *The Journal of Supercomputing*, vol. 74, no. 9, pp. 4542–4567, 2018.

[34] B. Pitchaimanickam, "Dragonfly algorithm for hierarchical clustering in wireless sensor networks," in *2021 5th Int. Conf. on Intelligent Computing and Control Systems (ICICCS)*, Madurai, India, pp. 192–197, 2021.

[35] H. H. Mahmoud, A. S. Alghawli, M. K. M. Al-shammari, G. A. Amran, K. H. Mutmbak *et al.,* "IoT-based motorbike ambulance: Secure and efficient transportation," *Electronics*, vol. 11, no. 18, pp. 2878, 2022.