

An adaptive, balanced and energy efficient clustering mechanism for UAV networks

İHA ağları için uyarlanabilir, dengeli ve enerji verimli bir kümeleme mekanizması

Sedat GORMUS^{1*} , Harun Emre KIRAN² 

¹Department of Computer Engineering, Faculty of Engineering, Karadeniz Technical University, Trabzon, Turkey.
sedatgormus@ktu.edu.tr

²Department of Computer Engineering, Faculty of Engineering, Hitit University, Corum, Turkey.
harunemrekiran@hitit.edu.tr, harunemrekiran@gmail.com

Received/Geliş Tarihi: 14.04.2020
Accepted/Kabul Tarihi: 11.08.2020

Revision/Düzeltilme Tarihi: 29.06.2020

doi: 10.5505/pajes.2020.53059
Research Article/Araştırma Makalesi

Abstract

Unmanned aerial vehicles (UAVs) are widely used in many fields, both civilian and military. Mostly mini UAVs are used in civilian applications which are preferred both in terms of affordability and availability. However, these vehicles are insufficient for some applications when they are used alone. This inadequacy is often observed in mini-UAVs having limited capacity in terms of energy storage and payloads. As a solution to this challenge, swarms of networked mini UAVs have been proposed. Thus, the UAVs in such networks are assigned with different tasks to accomplish the overall mission. While such UAV networks have many advantages, they also come with challenges. These challenges include limited on-board energy storage, low power and lossy wireless communication interface, and limited useful payload carrying capability. In this study, a new adaptive, balanced and energy efficient clustering mechanism has been proposed for such UAV networks.

Keywords: Unmanned aerial vehicles network, Cluster head selection, Wireless sensor networks, Energy efficiency, Topology stability.

Öz

İnsansız hava araçları (İHA), hem sivil hem de askeri olmak üzere birçok alanda yaygın olarak kullanılmaktadır. Özellikle sivil uygulama alanlarında hem ekonomiklik hem de kolay temin edilebilirlikleri sayesinde küçük ölçekli İHA'lar tercih edilmektedir. Ancak bu araçlar bazı uygulamalarda tek başına kullanıldığında yetersiz kalmaktadır. Mini İHA'lardaki bu yetersizlik çoğu durumda kısıtlı enerji ve taşıma kapasitesi olarak karşımıza çıkmaktadır. Bu soruna çözüm olarak birden fazla İHA'ların birleşmesiyle oluşturulan sürü İHA ağları önerilmiştir. Böylelikle bu ağlarda bulunan İHA'lara farklı görevler verilerek bu yetersizliğe çözüm önerisi sunulmuştur. Bu ağların birçok avantajları olmasıyla birlikte zorlukları da bulunmaktadır. Bu zorluklar sırasıyla enerji kısıtı, düşük güçlü ve kayıplı kablosuz haberleşme arayüzü ve düşük faydalı yük taşıma kapasitesi olarak karşımıza çıkar. Bu çalışmada, insansız hava aracı sürüleri için uyarlanabilir, dengeli ve enerji verimli yeni bir kümeleme mekanizması önerilmiştir.

Anahtar kelimeler: İnsansız hava araçları ağı, Küme başı seçimi, Kablosuz algılayıcı ağlar, Enerji verimliliği, Topoloji kararlılığı.

1 Introduction

Ad-hoc networks consist of mobile wireless nodes without a fixed infrastructure [1]. Furthermore, these networks are interconnected according to the application use cases [2]. Unmanned air vehicle (UAV) swarms are ad-hoc networks where nodes communicate with each other to coordinate and control the network and perform pre-determined tasks [3]. Recently, UAV swarms have found widespread applications in many areas ranging from military to search and rescue [4]. In particular, the defense industry actively carries out research to come up with novel applications involving UAV swarms [5],[6]. Furthermore, civilian UAV swarm applications such as search and rescue [7], surveillance [8] and disaster management [9] have recently become popular.

Large-scale UAVs require a very high budget and are not suitable for certain scenarios. Thanks to UAV swarms, specific tasks are easily accomplished. However, these UAVs can't be spread over a large area due to their limited communication coverage [10]. In addition, communication between the UAVs in a swarm tends to be intermittent since UAVs in such networks

move instantaneously. Furthermore, UAVs in a swarm typically have a limited radio range and are constrained in terms of energy storage which renders such small UAVs with a limited air endurance [11]. Thus, the UAV swarm architectures should address these challenges to make them applicable to the highlighted applications.

In this study, a new adaptive, balanced and energy efficient clustering mechanism has been developed for UAV swarms. With the proposed mechanism, the network is divided into sub-networks where the main clustering parameter is the traffic density within the UAV swarm. Here, the UAV swarm is formed by the nodes with different roles within the network. The roles are central cluster head (CCH), and the sub cluster-heads (SCH) respectively. The CCH determines the sub-area borders in the UAV network and elects the SCHs. The SCHs enable the exchange of data between UAVs in the sub-area. In addition, the SCH can in some cases exchange data with both UAVs in other sub-areas and the control center. In this study a *virtual center axis (VCA) based clustering* approach is proposed for creating the cluster boundaries independent of UAV density. The coverage area of each cluster is shaped according to the density of data traffic between the sub-areas. Furthermore, the

*Corresponding author/Yazışılan Yazar

proposed mechanism selects SCHs by taking the geographical location of the nodes into account where nodes around the centre of the cluster are prioritized.

This article is structured as follows; Section 2 presents the relevant literature, Section 3 presents the details of the proposed clustering mechanism, Section 4 outlines the simulation results and discussions, Section 5 concludes the paper and proposes future works.

2 Literature review

Recently, there is an increased interest toward topology modeling of UAV swarms. The main driving factor in this increase is the new and novel applications for these swarms. Such UAV swarms [12] have many challenges to be addressed for successful future application. In the study of [13], authors propose two different approaches to solve the challenge of high mobility, one of the most challenging aspects of UAV swarms. Here, authors make UAV networks more stable with the proposed methods. However, packet sharing of UAVs is not considered in this study. In addition, the battery levels of the UAVs tend to decrease rapidly as the topology of the fast-moving UAVs needs to be updated more frequently. In [14], authors propose a clustering mechanism to increase the stability of UAV swarms. In this mechanism, selection of the cluster heads is shaped according to the movements of the nodes. Each node also has information about two-hop neighboring nodes. Although this study provides solutions to important challenges of UAV swarms, traffic density of the networks is not considered where different parts of the UAV network may generate data packets at different rates. Authors in [15] propose a consensus-based mechanism that selects a cluster based on any failure in the network. However, this study is only recommended for homogeneous UAV swarms. In another study [16], authors propose dynamic angle adjustment and omnidirectional and directional communication schemes. The authors propose a solution that takes advantage of both unicast and geocast hash routing using location and trajectory information. The results show a higher packet delivery performance since the routes are kept stable for longer durations. However, this study does not also consider heterogeneous UAV traffic patterns. Furthermore, delay during the packet transmission is ignored. In [17] authors propose a clustering mechanism in which each cluster is made up of a fixed number of UAVs. Here, UAVs with the best local connectivity are selected to form a cluster. Thus, this proposed mechanism reduces communication costs and increases network performance. However, in this study, effect of wireless connectivity on UAV stability is ignored.

Several works have been proposed that create clusters based on meta-heuristic algorithms inspired by nature. Authors of [18] propose a mixed mechanism using nature-inspired krill herd (KH) and glowworm swarm optimization (GSO) algorithms. One of these algorithms is energy-aware clustering formation and cluster selection with GSO algorithm. Efficient cluster management is provided by KH algorithm. In this study, routing is created by using residual energy of nodes, neighbor numbers and distance between nodes. However, clusters are not updated according to the tendency of movements here. In another study [19], which uses yet another nature inspired algorithm, authors use the Bee Colony Algorithm, which is borrowed from the beehive working principle. However, serious delays in packet sharing may occur for the UAV swarms proposed in this study. Together with these nature-inspired

studies, there are several other studies for the UAV network [20],[21]. However, in all these studies, authors ignore tasks of the UAVs and pay attention only to the network connection between UAVs [22].

Many authors have proposed algorithms specific to the UAV networks. In addition, several studies have integrated algorithms for Mobile Ad Hoc Network (MANET) and Vehicular Ad Hoc Network (VANET) into the UAV networks [23],[24]. However, almost all studies for MANET and VANET do not address challenges specific to UAV swarms.

3 Proposed clustering mechanism

Network topologies in UAV swarms change rapidly as compared to other types of mobile vehicle networks. In addition, UAV topologies do not generally have a dedicated central node where the nodes in the topology are distributed randomly [25]. In addition, UAVs in a swarm tend to have a limited capacity in terms of data processing and storage. Since these UAVs are battery powered, their flight times are generally limited. In view of these constraints and conditions, a new clustering mechanism has been proposed for UAV swarms consisting of many low power low cost UAVs. The proposed mechanism in this study aims to balance the data traffic in each cluster by dynamically changing the cluster size. In this case, data traffic intensity in a section of the UAV swarm dictates the size of the cluster where the cluster size of a high data intensity part of the network is kept small. In addition, since the altitude of mini-UAVs in the UAV network are not very high, it is assumed that UAVs are at the same or close altitude. We believe that this mechanism will be able to operate in 3D environments as we consider the obstacles in 3D for this study.

In this study, there are two types of cluster heads namely Central Cluster Head (CCH) and Sub-Cluster Head (SCH). The role of the CCH node is to maintain the virtual center axis position and define the borders between the sub-clusters. On the other hand, SCH nodes perform routing functions on behalf of the nodes within the cluster. Furthermore, boundaries of sub-areas from the virtual center are calculated. The central location of the network, through which the sub-area border lines are formed, is called the Virtual Central Axis (VCA). Due to rapid mobility of UAVs, boundaries will vary greatly. Therefore, it is preferred to create a fixed virtual axis that changes over time. Location of this axis will be explained in detail in the following subsections. An example view of this mechanism is shown in Figure 1.

The proposed clustering mechanism is composed of three distinct steps. In the first step, VCA and CCH selection is done. In the second step, selection of SCHs in the sub-areas is done. In the third step, angular values of the sub-areas are updated to provide a balanced data traffic density in these sub-areas. Cluster heads are selected by the CCH during the initial part of the solution. In this round, the network is divided into sub-areas with equal angles, while the first SCHs are selected from nodes that are close to the center of these sub-areas. After the initial partitioning, the sub-areas are updated in three steps. These steps are described separately in detail here.

3.1 VCA adjustment and CCH selection

Purpose of this first stage in the proposed clustering mechanism is to calculate the center of the topology and select the most suitable node as CCH. The center point of this topology needs to be updated constantly due to the high mobility of the UAV swarm.

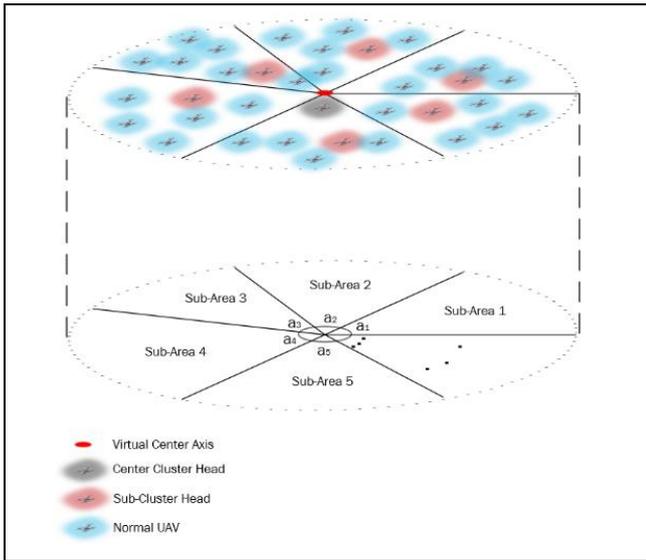


Figure 1. Example clustering mechanism proposed in UAV network.

The CCH node should be close to the center of the topology since it will provide best possible connectivity within the network. In addition, SCH can with high reliability connect with CCH when the CCH is in the center of the topology. This in turn will enable a low latency communication between the nodes located in different sub clusters. Here, it needs to be pointed out that each node within the sub-cluster forwards its data to the SCH, and SCH node forwards the data to the CCH which then forwards it to the SCH where the destination node is located. Here, the CCH obtains the locations of the SCH nodes periodically ensuring a balanced and low delay topology.

The CCH manages initial steps of the mechanism and acquires SCH locations. Sub-area information packet obtained from each SCH is given in Figure 2. As shown in the figure, the data packet contains label of *Sub-Area Sequence*. This label is sequence numbers of each sub-area. The sequence of this sub-area starts with *Sub-Area 1*. *Sub-Area Counter* label is incremented if a new SCH is selected for the sub-cluster. *Cluster Head Node ID* field is the identity of sub-cluster node in the network, while *Location* field holds location information for that node. *Total Amount of Data Sent* field is the data that a SCH receives from the nodes in its sub-area per unit time. Finally, the *Total Number of Nodes* field indicates total number of nodes in the sub-area.

Sub-Area Sequence	Sub-Area Counter	Cluster Head Node ID	Location
4 byte	4 byte	32 byte	24 byte
Total Amount of Data Sent		Total Number of Nodes	
8 byte		4 byte	

Figure 2. SCH information packet sent to CCH.

The CCH uses information packets received from the SCHs to calculate the total data traffic in each SCH area. The CCH obtains variance value from the collected SCH data for each sub-area. If the calculated variance value is more than a threshold β specified as a configuration parameter, the VCA location will be moved to a new location with the aim of reducing the calculated variance. Centre point of sub-area with highest data traffic

density is calculated so that the VCA direction is determined. Number of nodes in the selected sub-area and its neighboring areas determine the shift in VCA location. To calculate this shift value, the following formula is used.

$$axisChange = \left(\frac{d}{totalNodes} \right) * maxSpeed \quad (1)$$

The VCA location moves in the direction determined by the distance obtained from this equation. The *maxSpeed* in this equation represents maximum speed of UAVs per unit time in the network, while the *totalNodes* value represents the number of nodes in the swarm. When the new location of the VCA is determined, existing CCH transmits new location information to all nodes in the network.

Having calculated the new location of the VCA, the new CCH is determined. At least one of the two of the following conditions must be met in order to select a new CCH node.

- Distance of the existing CCH from the calculated VCA is greater than the maximum distance threshold,
- The existing CCH has exceeded the maximum time that it can remain as the CCH.

When at least one of these two conditions are met, the existing CCH broadcasts the packet shown in Figure 3 to nodes within specified range of VCA for initiating the selection of a new CCH and waits for a certain period. The CCH Sequence field in this packet is incremented by the node selected as the new CCH. With this value, other nodes realize that the new CCH is selected. This packet also contains identification information of the existing CCH and location information of VCA. Distance is the desired maximum range of the CCH to the VCA to be selected. Timestamp field indicates the instance of the broadcasted packet by the existing CCH. Candidate nodes are initially required to be within a quarter of the range of the VCH. The candidates then wait for a period inversely proportional to remaining energy in their batteries.

Center Cluster Head Sequence		Center Cluster Head ID	
4 byte		32 byte	
Virtual Center Axis	Distance	Timestamp	
24 byte	4 byte	8 byte	

Figure 3. Existing CCH packet.

The node whose timer expires first increases the CCH sequence field and broadcasts a CCH advertisement packet informing nearby nodes of the outcome of CCH process. If two or more center cluster candidates broadcast the advertisement packets declaring themselves as the CCHs, then the nodes receiving these packets elects the node with the most remaining energy as the CCH for the UAV swarm. The rest of the central cluster head candidates revert to normal operations after receiving the CCH advertisement packet from the elected CCH node. Format of CCH information message is given in Figure 4.

This packet contains 6 fields. The CCH Sequence field represents the current CCH sequence value. CCH ID field contains new center cluster identity information. Remaining Battery field contains remaining battery percentage of this UAV. In addition, Location field contains location information of

the broadcasting node. One-Hop Neighbor Nodes field contains information of its neighbors to which it can directly connect.

Center Cluster Head Sequence		Center Cluster Head ID	Remaining Battery
4 byte		32 byte	8 byte
Location	One-Hop Neighbour Nodes		Timestamp
24 byte	Array		8 byte

Figure 4. Candidate UAV central cluster head packet.

If no suitable CCH can be found, the existing CCH increases the specified range so that a new CCH may be selected. This step is repeated until a suitable CCH is elected. When the CCH election process is finalised, the existing CCH continue its operation as a standard UAV swarm node.

The CCH selection process fails when the selection criteria are not met. In this case, existing CCH broadcasts packet shown in Figure 4 without updating *CCH Sequence* value. Algorithm for this stage is given in Algorithm 1.

Algorithm 1 VCA adjustment and CCH selection algorithm

```

1: if (UAVState = CCH)
2:   CalculateUAVDensityOfEachSubArea()
3:   CalculateUAVDensityVariance()
4:   if VarianceValue > β
5:     AdjustVirtualCenterAxis()
6:   end
7:   CalculateDistanceToVirtualCenterAxis()
8:   if (DistanceToVirtualCenterAxis > MAX || ClusterTime > MAXTIME)
9:     BroadcastForNewCenterClusterHeadSelection()
10:  end
11:  end

```

3.2 SCH selection

The SCH nodes in each sub-area can act as relays in inter-area communication and can communicate directly with the control center on the ground if within range. This creates a hierarchical network topology which lends itself to energy efficient communication.

Prior to SCH selection, each node in the network first needs to learn in which sub-area it is currently located. Once location of new VCA is determined, CCH reports sub-areas to every node within the network. According to this reported location information and with angular values already present in nodes, nodes learn which sub-area they are in. The nodes are then ready for new SCH selection.

Nodes that learn their sub-areas first calculate their stability values (SV). The SV indicates the relative distance of the node to the center of the sub-area. In turn, a high SV also indicates a high probability of finding connections to other nodes. SCHs are preferred to be in areas near the center of the sub-area to maximize the probability successful communication. The SVs of these nodes are calculated with

$$K_i = C_i * \sum_{i \in I_k \cap M_i} L_{ij} \quad (2)$$

where C_i is the proximity of the node i to the center of its sub-area. This factor ranges from 0 to 1. If this node is in the center of the sub-area, the value becomes 1. If the node is close to sub-area boundary, this value approaches to 0. C_i factor is calculated by

$$C_i = \begin{cases} 1, & a = b \\ \frac{2 * a}{a + b}, & a < b \\ \frac{2 * b}{a + b}, & a > b \end{cases} \quad (3)$$

In this equation value a indicates shortest distance to lower boundary of the sub-area where node i is located, while value b represents shortest distance to upper boundary of this sub-area (Figure 5). Here, nodes can easily calculate the closest distance to the border with their location because they know the axis of sub-area where they are located and angular values between the borders. Another important value in (2) is L_{ij} , which is the probability that node i can find a connection with a neighboring node j . In this equation, I_k also represents nodes of the sub-area k where node i is located, while M_i represents nodes neighbour to node i . That is, after node i receives location information from neighboring nodes, it calculates probability that if these neighboring nodes are in sub-area where they are located, they can find connections with each other. These calculated probabilities are summed and multiplied by the factor B_i and SV of the node i is calculated. The value of L_{ij} is calculated by the equation $L(d_0, t)$ in [26].

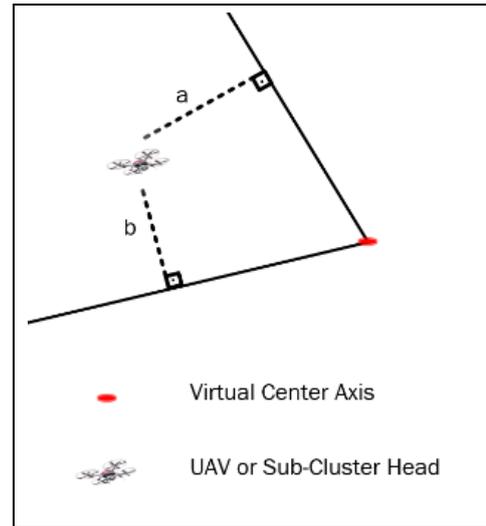


Figure 5. Proximity of a node to the center of its sub-area.

During the SCH selection, UAV node checks whether SCH of the sub-area in which it resides moves to another sub-area after obtaining SV. If the SCH does not move to another sub-area, the UAV node first compares its SV with the SV of the SCH in the sub-area. The UAV node then checks whether the SV value of the SCH differs from its own SV more than a threshold value Ω . If this is the case, this UAV node declares itself Potential SCH node and waits for a random time. If there are other Potential SCH nodes in the sub-area, node that broadcasts information packet shown in Figure 6 earlier is chosen as the new SCH. Value in 'Sub-Cluster Head Sequence' label in this information packet is incremented by the UAV node that declares itself as the new SCH. In addition, this packet contains identity of sub-area of the SCH, its identity and location information. At the same time, in order to find location of the SCH more easily this packet contains, this packet contains array labeled One-Hop Neighbor Nodes. Timestamp label is used to eliminate SCH selection collisions that occurs when there are two or more nodes declaring itself as the SCH.

Sub-Cluster Head Sequence		Sub-Area ID	Sub-Cluster Head ID
4 byte		32 byte	32 byte
Location of Sub-Cluster Head	One-Hop Neighbour Nodes		Timestamp
24 byte	Array		8 byte

Figure 6. Packet that declares a node as the new SCH.

If SCH of a sub-area moves to another sub-area, all the suitable UAV nodes within the area declare themselves as *Potential* SCH nodes. These candidate UAVs then wait for a certain time inversely proportional to their SV and broadcast SCH information packet. The information packet is shown in Figure 6. If there are more than one UAV that declares itself as SCH, the UAV, which has a higher SV, is selected as SCH. The other UAV nodes revert to standard UAV after receiving the information packet with a high SV.

For UAVs to be SCH, they must be neighbors with at least two UAVs in their sub-area. One of important reasons for this is that although some UAVs are within boundary of a sub-area, all their neighboring nodes may be in another sub-area. UAVs with such a condition update their sub-area after receiving information of their neighbor UAVs. An example of such a condition is shown in Figure 7. As can be seen in this figure, UAV 1 is geographically located in Sub-area A initially. After discovering its neighbors, UAV 1 learns that it has only UAV 2 as its neighbor and it is in Sub-Area B. Thus, UAV 1 migrates to Sub-Area B.

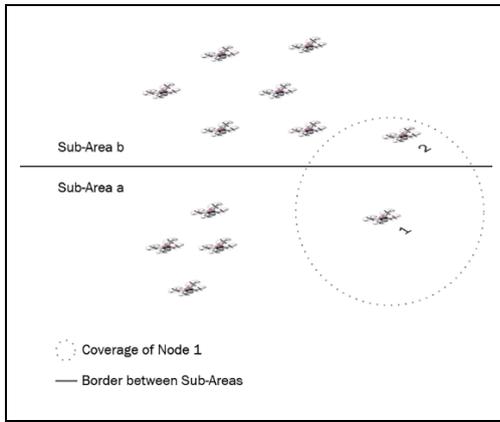


Figure 7. Node 1 changes its sub-area.

The pseudo code for SCH selection process is given in Algorithm 2.

Algorithm 2 SCH algorithm

```

1:BroadcastLocationInformation()
2:CalculateNodeSV
3:if Is SCH?
4:  if NodeSV < SCH-SV
5:    return;
6:  end
7:  if nodeStatus = 'Normal'
8:    nodeStatus = 'Potential'
9:    WaitRandomTime;
10: else if nodeStatus = 'Potential'
11:   BroadcastSCHInfo()
12: end
13: end

```

3.3 Border value calculation

Purpose of the border adjustment is to update cluster size of the network according to the traffic density of the UAV network. This stage is managed by CCH. SVs reported by SCH contain information about the traffic density of each cluster. This information is used by the CCH to update the boundaries of cluster areas.

In order to update border values in the third stage of the mechanism, SCHs first report to CCH SVs obtained in the second stage. CCH obtains variance ($var(K_i)$) values from SVs. If node densities between sub-clusters are very irregular, $var(K_i)$ value becomes high. $var(K_i)$ value is then compared to a predetermined α threshold value representing upper limit in irregularities in each sub-cluster. If $var(K_i)$ value is greater than this upper limit, new boundary values are calculated. The changes in boundaries are inversely proportional to SVs of SCHs ($newAng(r_1, r_2, \dots, r_{p-1}, r_p)$). However, these new boundaries are not directly applied as they will create very sharp changes in the topology.

$$\Delta K_i = \frac{newAng - oldAng}{h} \quad (4)$$

As seen in the equation, the h value is used to normalise the angular changes in the boundaries where h prevents sharp changes of the boundaries. Then the new angular boundaries are calculated using

$$NewSCHAngle = oldCHAngle + \Delta K_i \quad (5)$$

These angular values obtained from the equation are broadcasted by the CCH to the SCHs. The SCHs then report these values to UAVs in their sub-areas. Thus, the partitioning adjustment is completed. After the new partition adjustment is carried out, UAVs do not update their own areas according to new angular values immediately. This update will be done when the boundary adjustment exchanges between the CCH and SCH is completed. Pseudo code of this process is given in Algorithm 3.

Algorithm 3 Updating angle values between borders

```

1:if UAVState = CCH
2:  ObtainSVsFromSub-Areas ()
3:  CalculateVariance()
4:  if VarianceValue >  $\alpha$ 
5:    CalculateNewSCHAngle ()
6:     $\Delta K_i = (newSCHAngle - oldSCHAngle)/h$ 
7:    SCHAngle = oldSCHAngle +  $\Delta K_i$ 
8:    BroadcastSCHAngles ()
9:  else
10:   ReportExistingSCHAngles()
11: end
12: end

```

4 Results

In this section, the performance results of the UAV network are presented for different scenarios with several predetermined thresholds. The proposed clustering mechanism are tested and compared with 3 different previously proposed mechanisms. These tests are done in MATLAB based simulation environment. The random walking mobility model is currently one of preferred models for FANETs [14] and is used in the simulations carried out in this work. Parameters used in the simulation environment are given briefly in Table 1.

Table 1. Parameters used in simulation.

Parameters	Values
Area Radius	2500 meters
Transmission Range	1000 meters
Number of UAVs	100
Mobility Model	Random Walk (RW)
Maximum Speed	30 m/second
Minimum Speed	20 m/second
Node Data Capacity	8 kilobyte/second
Data Packet	6400 bit
Energy Used for Transmission	$50 \cdot 10^{-9}$ J / bit
Energy Used for Reception	$50 \cdot 10^{-9}$ J / bit
Transmission Amplifier	$10 \cdot 10^{-8}$ J

4.1 Performance comparisons for different threshold

4 different threshold values are tested in this section which are thresholds β in the 1st stage of the mechanism, Ω in the 2nd stage, α and h in the 3rd stage. These thresholds are evaluated according to average energy values consumed in the network, average number of packets dropped and the duration of cluster heads of UAVs per cluster.

β threshold is used in the first stage of the mechanism. The threshold values here are set to send 1 packet per round of UAVs in the network. Each SCH acquires total amount of data they receive from normal UAVs in their own sub-area in a round. SCHs then transmit this data to the CCH. Finally, variance value is obtained from the transmitted data packets.

In Table 2, the test results are given for three different scenarios with increasing values of β . Initially, the range for the optimal β value is kept large. Consequently, after trying these range of values, the best range is determined to be between 0.01 and 0.1. These ranges are optimized for the data flow rates of the simulated UAV network. For other networks with different data flow rate requirements, we expect to find different β value ranges. Here, the average energy consumption values change slightly with changing values of β . On the other hand, significant changes are observed in the average dropped number of packets and the duration of remaining as CH. The best β threshold for the average dropped packet numbers is at 0.05. In this case, boundaries separating the sub-clusters in the UAV network do not change very often. If the chosen β value is high, the data traffic densities of the SCHs do not match the rapidly changing UAV network. As seen in the table, there is a high amount of packet loss in this case. If a lower β value is selected, the VCA will be constantly updated and UAVs will change their sub-areas very often. This leads to high packet losses.

Table 2. Impact of β value on the network performance.

β value	Average Consumed Energy (J)	Average Dropped Packets	Continuous CH Time (Iteration)
0.01	0.0093	13.51	5.92
0.02	0.0093	13.06	6.17
0.03	0.0094	12.84	6.47
0.04	0.0092	10.97	7.13
0.05	0.0092	10.21	7.21
0.06	0.0091	10.44	7.30
0.07	0.0090	10.53	7.36
0.08	0.0090	10.89	7.28
0.09	0.0089	11.12	7.27
0.1	0.0089	11.42	7.26

Tests are carried out for three different scenarios with changing Ω thresholds. Here, if the SV value of a UAV node is greater than the Ω threshold value of the SCH in a sub-area, this UAV is chosen as the new SCH candidate. Results indicates a significant increase in SCH duration with increasing Ω value as indicated in Table 3. The reason for this is that a large Ω value allows an UAV to stay as SCH even if there is a significant variation in SVs of SCHs.

Table 3. Impact of Ω value on the network performance.

Ω value	Average Consumed Energy (J)	Average Dropped Packets	Continuous CH Time (Iteration)
0.1	0.0092	11.80	7.22
0.2	0.0092	11.51	7.94
0.3	0.0093	10.89	8.81
0.4	0.0093	11.15	9.58
0.5	0.0093	11.18	10.69
0.6	0.0094	11.26	11.39
0.7	0.0095	11.49	11.92
0.8	0.0095	12.11	13.07
0.9	0.0094	12.04	13.58
1	0.0095	11.05	15.68

The α threshold in the third stage of the mechanism is the lowest angular limit specified for a new boundary adjustment. Here, the CCH obtains the SVs from the SCHs. If the variance value obtained from SVs is more than α , a new angular value is calculated to adjust cluster borders. Performance results for different values of this threshold are given in Table 4. The results indicate that α has a very little impact on the analyzed performance metrics. The reason for this is that UAVs close to borders usually balance each other.

Table 4. Impact of α value on the network performance.

α value	Average Consumed Energy (J)	Average Dropped Packets	Continuous CH Time (Iteration)
0.3	0.0092	11.28	7.14
0.6	0.0092	11.19	7.00
0.9	0.0092	11.48	7.01
1.2	0.0092	11.31	7.14
1.5	0.0093	11.21	6.87
1.8	0.0092	11.33	7.12
2.1	0.0093	10.96	7.07
2.4	0.0092	11.05	6.90
2.7	0.0093	11.38	6.75
3	0.0092	11.54	7.28

Variable h in the third stage of the mechanism is used to prevent sharp changes after new angular values are obtained. Performance test results obtained according to different values of this variable are given in the graphics in Table 5. Looking at these graphics, the most important changes were observed in the dropped packet amounts. It is seen that the most appropriate value of the dropped package amount is 4. In values greater than this value, there is a continuous increase in number of packets dropped. The reason for this is to decrease flexibility of angular values between borders and prevent more appropriate sub-area selection of UAVs. In addition, it was observed that in the condition where the variable h is 2, more packet drop compared to condition where it is 4. The reason for this is that sharp change of angular values between borders affects UAV's sub-area changes unbalanced.

Table 5. Impact of h value on the network performance.

h value	Average Consumed Energy (J)	Average Dropped Packets	Continuous CH Time (Iteration)
2	0.0093	10.54	10.54
4	0.0092	9.93	9.93
6	0.0092	10.46	10.46
8	0.0092	11.07	11.07
10	0.0092	11.30	11.30
12	0.0092	11.55	11.55
14	0.0092	12.70	12.70
16	0.0093	13.18	13.18
18	0.0093	12.91	12.91
20	0.0094	13.79	13.79

4.2 Performance comparisons for different threshold

In this section, the performance of the proposed mechanism is compared with 3 different previously proposed mechanisms. The analyzed mechanisms are proposed in [27] (Lowest-ID), [28] (CONID) mechanisms for MANET and [14] (DGWS) for UAV networks. These mechanisms are evaluated using two different approaches. Firstly, the evaluated mechanisms are compared against each other taking average amount of energy consumption into account. Afterwards, these mechanisms are compared for packets drops. In addition, the performance results are given for 3 different traffic patterns as shown in Figure 8. As seen in Figure 8a, each UAV sends 1 packet in one round. In Figure 8b, the UAVs on the left side of the UAV network send 1.5 packets in one round, while the UAVs on the right send 0.5 packets. Finally, in Figure 8c, the UAVs send data with a random distribution, half send 0.5 packet per round and the other half in 1.5 packets in a round. In addition, these comparisons are made for 5 different node movement patterns.

First, energy consumption of the proposed clustering mechanism is compared against that of the previously proposed mechanisms. The energy consumption is obtained by summing the energy consumed by the data packet received from SCHs in one round. Here, if there is a direct connection between SCH and UAV node, packet sharing is provided without using any relay. If there is no direct connection, the UAV communicates by finding the best relay to the SCH.

In Table 6, the energy consumed by one UAV in one round for 4 different mechanisms is given in Joule (J). These calculations

are obtained using equations in [29]. The energy consumption values of 5 different movements are averaged. The results indicate that the proposed clustering mechanism consumes less energy for all the analyzed scenarios as compared to previously proposed mechanisms. In addition, we see that with the proposed mechanism, UAV consumed almost the same average energy compared for 3 different packet sharing approaches. The reason for this is the fact that the sub-areas and VCA are dynamically shaped according to the density of packet sharing of UAVs. As a result, if the packet sharing is more intense in one area of the UAV network, the VCA shifts to that area. In addition, the SCHs are smaller in those areas than other areas.

In Table 7, the number of packet-drops in a round is given for the analyzed mechanisms. The results in this table are based on the sum of the dropped packets at each SCH in one round. The purpose here is to measure the impact of high intensity traffic on the UAV network. Thus, it is of interest to find out whether the packets received from the cluster heads are balanced. The average dropped packets for 5 different mobility models are given in Table 7. It is observed from the results that the number of dropped packets in the network to which the proposed mechanism is applied is less than that of previously proposed mechanisms. Furthermore, this result indicates that the network traffic is more balanced when the proposed mechanism is used.

5 Conclusion and future goals

In this article, a new clustering mechanism is proposed for UAV networks. Since the UAV network has a dynamic structure, this mechanism is repeated continuously and UAVs can access both their own and neighbor's locations with the packet sharing used in these mechanisms. The purpose of this mechanism is to provide energy efficiency, balance the network and minimize number of dropped packets. As a result of the detailed simulation studies, it has been observed that the proposed mechanism performs very well compared to the previously proposed mechanisms. In the future, heterogeneous packet sharing approaches will be analyzed. And the complexity analysis of the proposed system will be carried out. In addition, studies on security aspects of the proposed mechanism will be carried out.

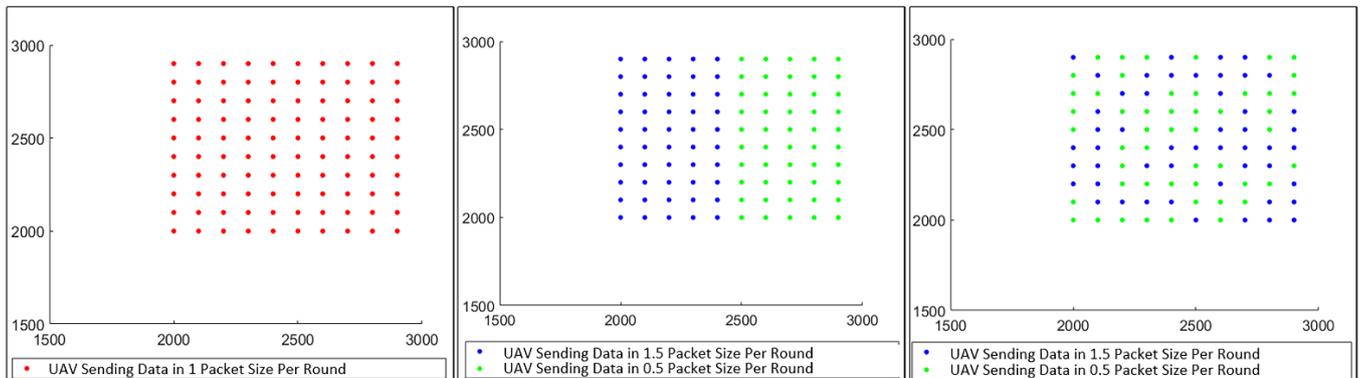


Figure 8. Packet shares used in simulation. (a): Homogeneous packet sharing. (b): Heterogeneous packet sharing-1. (c): Heterogeneous packet sharing.

Table 6. Average consumed energy.

		Average Consumed Energy (Joule)			
	Node Movements	Proposed Mechanism	DGWS [14]	Lowest-ID [27]	CONID [28]
Homogeneous Packet Sharing	1	0.0088	0.0102	0.0158	0.0158
	2	0.0088	0.0108	0.0167	0.0157
	3	0.0090	0.0095	0.0151	0.0166
	4	0.0101	0.0117	0.0165	0.0164
	5	0.0096	0.0118	0.0160	0.0161
	Average	0.0092	0.0108	0.0161	0.0161
Heterogeneous Packet Sharing-1	1	0.0090	0.0096	0.0156	0.0152
	2	0.0084	0.0101	0.0163	0.0157
	3	0.0097	0.0086	0.0149	0.0160
	4	0.0091	0.0108	0.0161	0.0168
	5	0.0103	0.0110	0.0159	0.0156
	Average	0.0093	0.0101	0.0158	0.0159
Heterogeneous Packet Sharing-2	1	0.0089	0.0206	0.0312	0.0313
	2	0.0084	0.0204	0.0328	0.0313
	3	0.0088	0.0182	0.0303	0.0326
	4	0.0092	0.0222	0.0329	0.0327
	5	0.0102	0.0230	0.0321	0.0319
	Average	0.0092	0.0209	0.0319	0.0320

Table 7. Average number of dropped packets.

		Average Dropped Packets per Round			
	Node Movements	Proposed Mechanism	DGWS [14]	Lowest-ID [27]	CONID [28]
Homogeneous Packet Sharing	1	10.831	22.940	51.855	59.558
	2	9.909	22.398	55.577	61.546
	3	9.436	21.366	49.531	59.994
	4	9.863	25.747	48.027	61.312
	5	9.738	31.577	43.546	55.403
	Average	9.955	24.806	49.707	59.562
Heterogeneous Packet Sharing-1	1	14.034	37.899	50.542	61.334
	2	12.825	30.697	54.605	61.407
	3	12.994	23.890	48.446	62.638
	4	11.757	18.923	47.337	63.149
	5	14.610	30.701	41.959	55.756
	Average	13.244	28.422	48.578	60.857
Heterogeneous Packet Sharing-2	1	12.029	40.397	52.494	60.096
	2	11.105	33.435	56.363	62.583
	3	10.182	23.878	49.748	60.417
	4	11.092	20.736	48.674	61.949
	5	10.695	32.805	44.110	55.764
	Average	11.020	30.250	50.278	60.161

6 Author contribution statements

In the scope of this study, Sedat Görmüş contributed to UAV formation, clustering mechanisms, routing approaches, literature review and editing of the paper; Harun Emre Kirans contributions include UAV formation, clustering mechanisms, algorithm development and implementation, literature review and editing of the paper.

7 Ethics committee approval and conflict of interest statement

There is no need to obtain permission from the ethics committee for the article prepared.

There is no conflict of interest with any person / institution in the article prepared.

8 References

- [1] Mcdonald AB. A Mobility-Based Framework for Adaptive Dynamic Cluster-Based Hybrid Routing in Wireless Ad Hoc Networks. Ph.D. Thesis, University of Pittsburgh, USA, 2000.
- [2] Ramanathan R, Martha S. "Hierarchically-organized, multihop mobile wireless networks for quality-of-service support". *Mobile Networks and Applications*, 3(1), 101-119, 1998.
- [3] Sahingoz OK. "Mobile networking with UAVs: Opportunities and challenges". *2013 International Conference on Unmanned Aircraft Systems (ICUAS)*, Atlanta, GA, 28-31 May 2013.
- [4] Li X, Zhang T, Li J. "A particle swarm mobility model for flying ad hoc networks". *2017 IEEE Global Communications Conference*, Singapore, 4-8 January 2017.

- [5] Newcome LR. *Unmanned Aviation: A Brief History of Unmanned Aerial Vehicles*. 1st ed. Reston, VA, USA, American Institute of Aeronautics and Astronautics Inc., 2004.
- [6] Valavanis KP, Vachtsevanos GJ. *Handbook of Unmanned Aerial Vehicles*. 1st ed. Netherlands, Springer, 2015.
- [7] Scherer J, Yahyanejad S, Hayat S, Yanmaz E, Vukadinovic V, Andre T, Bettstetter C, Rinner B, Khan A, Hellwagner H. "An autonomous multi-UAV system for search and rescue". *2015 Workshop on Micro Aerial Vehicle Networks, Systems, and Applications for Civilian Use*, Florence, Italy, 18 May 2015.
- [8] Saska M, Chudoba J, Přeučil L, Thomas J, Loianno G, Třešňák A, Vonásek V, Kumar V. "Autonomous deployment of swarms of micro-aerial vehicles in cooperative surveillance". *2014 International Conference on Unmanned Aircraft Systems*, Orlando, FL, USA, 27-30 May 2014.
- [9] Saha HN, Das NK, Pal SK, Basu S, Auddy S, Dey R, Nandy A, Pal D, Roy N, Mitra D, Biswas S, Maity T. "A cloud based autonomous multipurpose system with self-communicating bots and swarm of drones". *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*, Las Vegas, NV, USA, 8-10 January 2018.
- [10] Ruan L, Wang J, Chen J, Xu Y, Yang Y, Jiang H, Zhang Y, Xu Y. "Energy-efficient multi-UAV coverage deployment in UAV networks: A game-theoretic framework". *China Communications*, 15(10), 194-209, 2018.
- [11] Palan NG, Barbadekar BV, Patil S. "Low energy adaptive clustering hierarchy (LEACH) protocol: A retrospective analysis". *2017 International Conference On Inventive Systems and Control*, Coimbatore, India, 19-20 January 2017.
- [12] Sahingoz OK. "Networking models in flying ad-hoc networks (FANETs): Concepts and challenges". *Journal of Intelligent & Robotic Systems*, 74(1-2), 513-527, 2014.
- [13] Zang C, Zang S. "Mobility prediction clustering algorithm for UAV networking". *2011 IEEE GLOBECOM Workshops*, Houston, TX, USA, 5-9 December 2011.
- [14] Luo F, Jiang C, Du J, Yuan J, Ren Y, Yu S, Guizani M. "A distributed gateway selection algorithm for UAV networks". *IEEE Transactions on Emerging Topics in Computing*, 3(1), 22-33, 2014.
- [15] Feng T, Fan W, Tang J, Zeng W. "Consensus-based robust clustering and leader election algorithm for homogeneous UAV clusters". *Journal of Physics: Conference Series, IOP Publishing*, 1168(3), 1-10, 2019.
- [16] Gankhuyag G, Shrestha AP, Yoo SJ. "Robust and reliable predictive routing strategy for flying ad-hoc networks". *IEEE Access*, 5, 643-654, 2017.
- [17] Zafar W, Khan BM. "A reliable, delay bounded and less complex communication protocol for multicluster FANETs". *Digital Communications and Networks*, 3(1), 30-38, 2017.
- [18] Khan A, Aftab F, Zhang Z. "BICSF: Bio-inspired clustering scheme for FANETs". *IEEE Access*, (7), 31446-31456, 2019.
- [19] Leonov AV. "Application of bee colony algorithm for FANET routing". *2016 17th International Conference of Young Specialists on Micro/Nanotechnologies and Electron Devices*, Erlagol, Russia, 30 June-4 July 2016.
- [20] Arafat MY, Moh S. "Localization and clustering based on swarm intelligence in UAV networks for emergency communications". *IEEE Internet of Things Journal*, 6(5), 8958-8976, 2019.
- [21] Maistrenko VA, Alexey LV, Danil VA. "Experimental estimate of using the ant colony optimization algorithm to solve the routing problem in FANET". *2016 International Siberian Conference on Control and Communications*, Moscow, Russia, 12-14 May 2016.
- [22] Xing N, Zong Q, Tian B, Dou L, Wang Q. "Network Formation Game for Routing in Unmanned Aerial Vehicle Networks". *2018 IEEE/CIC International Conference on Communications in China*, Beijing, China, 16-18 August 2018.
- [23] Singh K, Verma AK. "Applying OLSR routing in FANETs". *2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies*, Ramanathapuram, India, 8-10 May 2014.
- [24] Singh K, Verma AK. "Experimental analysis of AODV, DSDV and OLSR routing protocol for flying adhoc networks (FANETs)". *2015 IEEE International Conference on Electrical, Computer and Communication Technologies*, Coimbatore, India, 5-7 March 2015.
- [25] Zeng Y, Zhang R, Lim TJ. "Wireless communications with unmanned aerial vehicles: Opportunities and challenges". *IEEE Communications Magazine*, 54(5), 36-42, 2016.
- [26] Gong L, Bai Y, Chen M, Qian D. "Link availability prediction in ad hoc networks". *2008 14th IEEE International Conference on Parallel and Distributed Systems*, Melbourne, VIC, Australia, 8-10 December 2008.
- [27] Ephremides A, Wieselthier JE, Baker DJ. "A design concept for reliable mobile radio networks with frequency hopping signaling". *Proceedings of the IEEE*, 75(1), 56-73, 1987.
- [28] Chen G, Nocetti FG, Gonzalez JS, Stojmenovic I. "Connectivity based k-hop clustering in wireless networks". The *35th Annual Hawaii International Conference on System Sciences*, Big Island, HI, USA, 10-10 January 2002.
- [29] Heinzelman WB, Chandrakasan AP, Balakrishnan H. "An application-specific protocol architecture for wireless microsensor networks". *IEEE Transactions on wireless communications*, 1(4), 660-670, 2002.