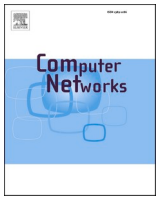
Computer Networks 273 (2025) 111787

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet



Spectral reinforcement learning based dynamic routing for unmanned aerial vehicle (UAV) networks

Saif ullah [a], Khalid Hussain [b,c], Muhammad Faheem [d,e,*], Nisar Ahmed Memon [f]

[a] Future Research Institute, Shenzhen Kaihong Digital Industry Development Co., Ltd., Shenzhen, China
[b] Department of Artificial Intelligence and MMG, Aror University, Sukkur, Pakistan
[c] Department of Electrical Engineering, Technical University of Eindhoven (TU/e), Eindhoven, Netherlands
[d] VTT Technical Research Centre of Finland, Maarintie 3, 02150, Espoo, Finland
[e] School of Technology and Innovations, University of Vaasa, Vaasa, Finland
[f] Faculty of Engineering and Technology, University of Sindh, Jamshoro, Pakistan

ARTICLE INFO

Keywords:
Unmanned aerial vehicle
Reinforcement learning
Spectral clustering
Routing
Optimal path

ABSTRACT

Unmanned Aerial Vehicles (UAVs) have received a lot of interest for their prospective uses in various types of disciplines, including communication, disaster management, surveillance, and military applications. UAV ad-hoc networks enable UAVs to interact wirelessly without a permanent infrastructure, making them suited for many circumstances. Conventional methods require predefining the number of clusters, which can lead to inaccurate results, and existing schemes focus on distance as the key parameter while neglecting UAV connectivity; additionally, traditional algorithms struggle with complex UAV network structures due to varying distances, obstacles, and dynamic configurations, making them unable to adapt to frequent changes in connectivity, signal strength, and network topology. This study proposes a framework that integrates spectral clustering and reinforcement learning to optimize the performance of UAV ad hoc networks. Spectral clustering groups UAVs with similar communication characteristics, such as signal strength and geographic location. Reinforcement learning is then used to optimize the path UAVs take within each clustered group, leading to further improvements in network performance. Our approach effectively adapts to changes in network topology and communication patterns, allowing for optimal performance even in dynamic environments. Experimental results demonstrate the effectiveness of our strategy, achieving a Packet Delivery Ratio (PDR) improvement of approximately 18.42% over k-means routing at high mobility scenarios, with an end-to-end delay reduction of around 40% compared to traditional methods. Additionally, the Network Routing Load (NRL) of our proposed scheme remains consistently below 18%, indicating enhanced efficiency compared to existing protocols, which can reach NRL values of up to 35%. Our approach optimizes the communication efficiency of UAV ad-hoc networks by adopting an optimal route policy, resulting in reduced end-to-end delay and improved packet delivery ratio. The proposed framework offers several advantages over existing methods, including adaptability to changes in network topology and communication patterns, efficient communication, and optimal routing decisions.

1. Introduction

AVs have gained significant achievements in recent years because of wide practical deployment in various fields, including cellular communication, surveillance, military applications, and rescue missions in disasters [1]. Wireless Networks or Mobile Ad hoc Networks (MANETs), in the case of UAV ad-hoc networks, incorporate technologies that allow the UAVs to communicate directly with each other without the use of wired structures. For this reason, they can be utilized in numerous events, including rescue searches in inaccessible regions, traffic monitoring in countless cities, and other communication services in areas affected by disasters [2,3]. Nevertheless, the UAV Ad hoc network has the following challenges that complicate its operation. For example, the UAVs ad-hoc networks have unstable communication links and limited battery power; thus, ensuring reliable communication is challenging. In addition, the network's topology is constantly changing, primarily due to the movement of the UAVs, which creates significant disruptions in network management and resource distribution, as proposed in [4,5].

* Corresponding author.
E-mail address: muhammad.faheem@vtt.fi (M. Faheem).

https://doi.org/10.1016/j.comnet.2025.111787
Received 10 June 2025; Received in revised form 22 September 2025; Accepted 13 October 2025
Available online 14 October 2025

The above challenges pose a challenge to developing an algorithm that can improve the performance of UAV ad hoc networks.

Researchers have used topology-based routing protocols, including Dynamic Source Routing (DSR), Ad-hoc On-demand Distance Vector (AODV), and Hybrid Routing Protocol (HRP) for UAV-based networks to tackle these issues [6,7]. Further, HRP protocol combines the advantages of reactive and proactive protocols to enhance the network's functionality. However, HRP faces challenges in 3D UAV-based networks due to the complexity and dynamic environment involved in forming and maintaining the network's topology [8–15]. Additionally, topology-based routing protocols rely on exchanging topology information with neighbours and updating routing tables, resulting in significant overhead and operation latencies in the network [16]. This is incredibly complicated in UAV networks, where energy and computational sources may constrain the nodes. To solve these problems, position-based routing protocols have been proposed, where UAVs transmit packets to neighbouring UAVs based on their geographical coordinates, without storing routing table information [11]. Position-based routing protocols [12] require each node to have a unique position in the network, which is typically obtained by using GPS or another localisation technology. Nodes utilise this position information to determine their own position and the positions of other nodes in the network, as described in [13]. With this information, nodes can establish routes that they can take to reach other nodes, depending on their proximity. However, Position-based routing has many benefits for UAV ad hoc networks; the networks are scalable, efficient, adaptable, and robust [14]. Nonetheless, there are also disadvantages to relying on localisation, movement, and dynamics sensitivity technology, including susceptibility to attacks, scalability issues, and limited coverage and adaptability [15,16].

Despite their considerable potential, ad hoc Unmanned Aerial Vehicle (UAV) networks face significant challenges in optimising routing paths and ensuring effective communication. Traditional routing techniques often rely on fixed or heuristic-based methodologies, which require predefining the number of clusters, potentially leading to inaccurate results. Moreover, these static approaches lack the necessary flexibility to adapt to the rapidly changing topologies and communication patterns characteristic of UAV environments. Such rigidity frequently results in suboptimal performance, particularly in dynamic scenarios where UAVs frequently change positions or communication roles. Additionally, traditional methods do not adequately account for the heterogeneity in UAV capabilities, communication ranges, and varying traffic demands, resulting in inefficient resource utilisation, increased network congestion, and compromised Quality of Service (QoS). Therefore, there is a growing need for an intelligent and adaptive routing framework that can effectively respond to the evolving topology and diverse communication requirements in UANETs.

This article adopts spectral clustering as the foundation of the proposed routing framework and combines it with reinforcement learning to counter the above-mentioned challenges of UAV ad hoc networks (UAVNETs). Spectral clustering has gained attention as a robust pattern recognition tool across various domains, including computer vision, natural language processing, and social network analysis. It operates by analyzing the eigenvalues and eigenvectors of a graph-based similarity matrix to divide nodes into groups with closely related properties. In the context of UAV ad hoc networks, spectral clustering can be applied to dynamically group UAVs based on shared communication characteristics such as signal strength, relative position, or link stability. This clustering enables localized and efficient routing decisions, reducing complexity and improving adaptability in highly mobile environments. Meanwhile, reinforcement learning has shown promising results in wireless networks due to its ability to learn optimal routing policies through reward-based adaptation in complex and dynamic scenarios.

Moreover, reinforcement learning is employed to determine optimal routing paths within these dynamically formed clusters, allowing UAVs to make intelligent forwarding decisions based on their local state and the evolving network conditions. Based on this design, we introduce a framework that integrates spectral clustering and reinforcement learning to enhance communication efficiency and adaptability in UAV ad hoc networks. We conduct both quantitative and qualitative evaluations to demonstrate the effectiveness of the proposed methodology, showing improvements in routing efficiency and reductions in network load. In this way, UAVs can achieve enhanced communication performance with lower overhead and improved scalability. The spectral clustering component helps reduce routing complexity by grouping UAVs, while the reinforcement learning agent dynamically learns the best paths within and across these clusters.

This paper's primary contributions are summarized as follows:

- Integration of Spectral Clustering**:** We introduce spectral clustering to group UAVs with similar communication characteristics, such as signal strength and proximity. This clustering reduces routing complexity, enables localized communication, and minimizes network load. It offers a more adaptive alternative to heuristic or fixed routing approaches.
- Reinforcement Learning-Based Routing: Within each cluster, we employ reinforcement learning to learn optimal routing paths. The RL agent dynamically adapts its routing strategy based on network state observations, leading to more efficient and reliable data delivery compared to static schemes that do not account for UAV diversity.
- Adaptability to Dynamic Environments: Our framework is specifically designed to handle the highly dynamic nature of UAV ad hoc networks. It adapts in real time to changes in network topology and traffic patterns, ensuring robust routing performance under varying conditions, including scenarios generated by different mobility models.
- Enhanced Communication Efficiency: By combining spectral clustering with reinforcement learning, the proposed protocol reduces routing path lengths and communication overhead. This results in improved throughput, lower congestion, and better scalability compared to conventional routing protocols.

The rest of the paper is structured as follows: Section 2 provides an overview of spectral clustering and its applications in various fields. Section 3 describes the UAV ad hoc network model, along with the calculation of the similarity matrix. Section 4 presents the experimental results and analysis. Finally, Section 5 concludes the paper and outlines potential directions for future research.

2. Related work

This section presents a literature review of existing UAV routing schemes for ad hoc–based UAV networks. Unmanned Aerial Vehicle Ad-hoc Networks (UAVANs) are a type of wireless communication network in which unmanned aerial vehicles (UAVs) form a network without the need for infrastructure [17]. Routing schemes play a crucial role in UAVANs in maintaining communication for data forwarding. There has been growing interest in developing efficient routing algorithms for UAVANs [18,19]. Different methods have been suggested to improve network performance, reduce network overhead, and enhance reliability.

To achieve efficient routing from source to destination, researchers initially proposed IP-based routing protocols [20–22]. These protocols can be broadly categorized as either proactive or reactive based on their operational principles [23,24]. In proactive routing, as introduced in [25], routes are continuously maintained by monitoring changes in network topology, enabling rapid data forwarding. For example, a swarm control-based proactive routing scheme was proposed to adaptively manage the movements of UAVs. Another approach, intelligent cluster-based routing [26], selects the cluster head based on UAV position and residual energy, aiming to enhance routing decisions. While

proactive protocols offer fast forwarding, they often incur high control overhead due to frequent updates.

Reactive routing protocols were proposed to resolve the network overhead issues. A performance-aware routing algorithm introduced in [27] proposes a self-adaptive network that mitigates broadcast storms by circumventing route loops. A trajectory-based 3D transformation algorithm is another technique that is described in [28]. This algorithm combines reactive and proactive tactics in order to construct and maintain pathways in a dynamic manner. Despite these advancements, reactive and hybrid protocols still face challenges in high-mobility environments where maintaining up-to-date routes becomes difficult.

To further optimize routing, position-based protocols were proposed [29–31]. These schemes eliminate the need for maintaining routing tables by forwarding packets based on the geographical positions of neighboring UAVs. Although position-based routing reduces overhead, it faces several challenges, including suboptimal routing decisions due to greedy forwarding, routing voids in sparse regions, localisation errors leading to incorrect forwarding, scalability issues as the number of UAVs increases, and increased energy consumption resulting from frequent position updates. Researchers have proposed enhancements, including alternative forwarding strategies and the use of auxiliary information, to address these challenges.

To overcome these limitations, clustering approaches have been introduced to improve the scalability and energy efficiency of UAV ad hoc networks. An early example is the Energy-Efficient Clustering Algorithm (EECA) [32], which organises nodes based on their remaining energy, with the UAV holding the highest energy level serving as the cluster head. EECA aims to reduce overall energy consumption while maintaining a fair distribution of energy use within the network. Another approach, the Load-Balanced Clustering Algorithm (LBCA) [33], employs a centralised strategy in which cluster heads are selected based on both residual energy and communication demand, thereby enhancing resource utilisation and network performance.

A hierarchical clustering algorithm (HCA) was proposed in [34] for UAVANs, where local clusters are first formed and then aggregated into global clusters. This two-level clustering structure reduces communication overhead and improves scalability. Additionally, a Swarm Intelligence-Based Clustering Algorithm (SICA) [35] utilises particle swarm optimisation (PSO) techniques to form clusters, optimising for residual energy, communication load, and distance parameters. SICA aims to enhance network performance while reducing energy consumption. Lastly, a Multi-Objective Clustering Algorithm (MOCA) [36] was introduced to simultaneously optimize multiple factors, including energy consumption, communication overhead, and network lifetime, using a weighted multi-objective optimization approach. Recent investigations have examined routing protocols in highly dynamic UAV networks [37], demonstrating that modified OLSR effectiveness in search-and-rescue scenarios, while mobility model analyses [38,39] highlight trajectory impacts on routing performance. Beyond routing path optimization, Reinforcement Learning (RL) has been extensively applied to address other critical challenges in UAV networks. Studies highlight its role in improving connectivity and reducing network fragmentation by supporting smarter UAV placement and coordinated movement [40,41].

Furthermore, RL frameworks are increasingly used for collision avoidance in dense swarm operations, ensuring safe navigation in complex 3D airspace [42]. To address the stringent energy constraints of UAVs, RL-based strategies have been developed for optimizing flight trajectories, transmission power, and computation offloading, significantly improving network lifetime and operational efficiency [43,44]. These works highlight RL's versatility as a powerful tool for holistic UAV network management beyond the routing layer [45].

Existing protocols have made notable progress in UAVAN routing; however, they continue to encounter difficulties in adapting to dynamic topologies and varying communication characteristics. The identified limitations underscore the need for more adaptable and sophisticated routing frameworks, as exemplified by the spectral clustering and reinforcement learning-based approach introduced in this study. A comparison of the key technical features and optimization methods of the discussed UAV routing protocols is summarized in Table 1.

3. System model

In this study, UAVs are assumed to operate in an outdoor environment, resulting in a three-dimensional (3D) scenario. Communication is established directly between UAVs without reliance on a central server. Within this 3D framework, data exchange occurs among UAVs, and inter-UAV distances are computed using the Euclidean distance formula at each time interval t.

$$UAV_i = \sqrt{(x_i - x_0) + (y_i - y_0) - (z_i - z_j)} \tag{1}$$

where, x_0, y_0, z_0 denote the initial position coordinates of UAV_i while x_i, y_i, z_i represent the final positions. Since UAVs send packets within a three-dimensional space, losses occur due to wireless channel effects. To capture these effects, a three-dimensional logarithmic propagation path loss model is considered. This model needs three distance positions: near-field, mid-field, and far-field exponents of UAVs, respectively. Eq (2) shows a 3D logarithm propagation path loss model Fig. 1.

$$UAV_L = UAV_0 + 1 - 10 \times n_0 \log_{10}\left(\frac{d_0}{d_1}\right) \tag{2}$$

To represent UAV communication ver different ranges, three reference fields with point *(d_0)* are combined and ordered as $d_0, d_1, d_2, d_3, \ldots, d_i$. The resulting piecewise formulation of the path-loss model is given by Eq (3):

$$UAV_R = \begin{cases} 0 \; d < d_0 \\ UAV_L + 10 \times n_0 \log_{10}\left(\frac{d_0}{d_1}\right) d_0 < d < d_1 \\ UAV_L + 10 \times n_0 \log_{10}\left(\frac{d_0}{d_1}\right) + UAV_L + 10 \times n_1 \log_{10}\left(\frac{d_1}{d_2}\right) \; d_1 < d < d_2 \end{cases} \tag{3}$$

3.1. Spectral clustering

Let the entire network N be divided into sub-networks or regions, denoted as $N = \{N1, N2, N3, N4, N5\}$. Each sub-region is partitioned in such a way that clusters are formed to separate the network into distinct areas. Spectral clustering is applied because it offers two main advantages over traditional clustering approaches: (i) there is no restriction on the shape of the clusters, and (ii) it can effectively handle complex, intertwined structures.

Table 1
Comparative summary of existing UAV routing techniques.

Ref	Routing Scheme	Routing Type	Mobility Management	Energy Efficiency	Overhead Reduction	Clustering	Optimization/Key Method
[25]	Topology-Aware Inter-UAV Routing Optimization	Proactive	✓	–	✓	–	Topology and Swarm Coordination
[26]	Intelligent Cluster Routing Scheme	Clustering	✓	–	✓	✓	Moth-Flame Optimization
[27]	Performance-Aware Routing Mechanism	Proactive	✓	–	✓	–	Link Quality Estimation
[28]	3D Transformative Routing	Hybrid	✓	–	✓	–	Skeleton-Guided GPS-Free Transformations
[29]	Comparative Study on Geographic Routing	Geographic	–	–	✓	–	Greedy Forwarding (Comparison Study)
[30]	GeoUAVs Geocast Routing Protocol	Geographic	–	–	✓	–	Geocast Routing, Distance Metric
[31]	GeoSaW Waypoint-Based Routing	DTN/Geographic	✓	–	✓	–	Waypoint Prediction and Time-To-Intercept
[32]	EECA (Energy Efficient Clustering Algorithm)	Clustering	–	✓	✓	✓	Machine Learning-based Clustering
[33]	LBCA (Load Balanced Clustering Algorithm)	Clustering	–	✓	✓	✓	Hybrid Metaheuristic Technique
[34]	Hierarchical Clustering (Dendrogram-Based)	Clustering	–	✓	✓	✓	Hierarchical Energy-aware Clustering
[35]	Dynamic Routing and Cluster Coordination	Clustering	✓	–	✓	✓	Swarm Intelligence - Cauchy PSO
[36]	Integrated Host-and Content-Centric Routing	Hybrid	✓	–	✓	–	Host and Content-Aware Routing
–	Proposed work	Clustering + RL-Based	✓	✓	✓	✓	Spectral Clustering + Q-learning Path Optimization

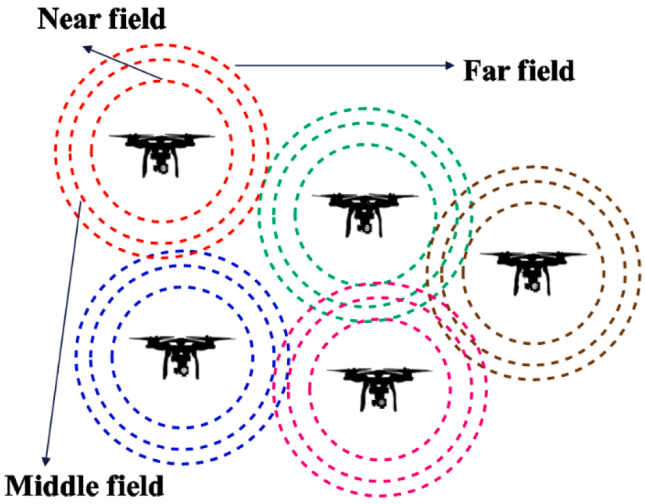


Fig. 1. 3D distance model.

1st step: Form a graph network consisting of vertices and edges (vertices show the connections and edge shows the number of UAVs).
2nd step: Apply spectral embedding. In order to have proper knowledge about network connections and strong ties with low dimensions. We find the eigenvector values of each edge node (UAV) to form a Laplacian graph network.
3rd step: To form clusters, apply k-means clustering on the Laplacian graph network.

First, to create a UAV network. Let's suppose each UAV has an adjacency matrix on which save the weight connection of other UAVs connected to it as mentioned below.

$$A_{UAV} = \begin{Bmatrix} w_{i,j} \text{ is wight of } i.j \text{ position} \\ 0 \text{ otherwise no connection} \end{Bmatrix} \quad (4)$$

To calculate the weights of the edges in the adjacency matrix, we use the Gaussian similarity function in Eq (5):

$$w_{i,j} = exp\left(\frac{d_{i,j}^2}{2\sigma^2}\right) \quad (5)$$

where w_{ij} represents the weight of the edge connecting UAVs i and j, while d_{ij} denotes their Euclidean distance. The parameters σ serves as a scaling parameter that determines the effective neighborhood size. The Gaussian similarity function decreases with distance and ranges between zero and one, which means that it assigns higher weights to closer UAVs and lower weights to farther UAVs. The parameter σ is typically chosen with respect to the average inter-UAV distance, ensuring that the model effectively captures local similarity relationships. A smaller value of σ means that only very close UAVs are considered similar, while a larger value of σ means that more UAVs that are distant are also considered similar.

The Laplacian matrix is then defined as:

$$L_{UAV} = D_{UAV} - A_{UAV} \quad (6)$$

where D_{UAV} is a diagonal matrix, which contains the degree of each UAV, respectively, such as $d_{i,j}$ as described below is sum of weights of a specific row of a diagonal matrix.

$$d_{i,j} = \sum_{\{j|(i,j)\in E} w_{i,j} \quad (7)$$

The eigenvectors and eigenvalues of the Laplacian matrix play a crucial role in graph theory, as they offer insights into the structural properties and connectivity of UAV networks. Because the Laplacian matrix is symmetric, its eigenvectors are mutually orthogonal. Within a UAV communication graph, these eigenvectors capture the connectivity of nodes. The eigenvalues also carry valuable information. For example, the total sum of all eigenvalues equals the sum of the degrees of the graph's vertices. The second-smallest eigenvalue, also referred to as algebraic connectivity, indicates how well connected the overall network is and is closely related to the graph's expansion properties.

$$Av = \lambda v \quad (8)$$

where v is an eigenvector of Adjacency matrix and λ is the Eigenvalue. For UAV networks, each UAV has an adjacency matrix (We have described this in our previous paper OGR) having n number of Eigen

values $UAV_{\mu} = \{UAV_{\mu 1}, UAV_{\mu 2}, \ldots\ldots UAV_{\mu n}\}$. Eq (9) presents the spectrum of the Laplacian matrix.

$$0 = \lambda_0 > \lambda_1 > \lambda_2 \tag{9}$$

A detailed mathematical discussion of the Laplacian matrix can be found in [46] and [47]. Instead, the study applies its eigenvectors for UAV ad hoc routing. Once the eigenvector values of the UAV network are obtained, the network is divided into smaller subnetworks using the K-means clustering algorithm.

Let $U = \{u_1, u_2, \ldots\ldots u_n\}$ be the set of UAVs in network, and assume that it is divided into C clusters. Because UAVs are mobile, clustering is carried out iteratively, beginning with index $\varepsilon = 1$, and continuing until the maximum iteration limit ε_{max} is reached, at which point all UAVs are grouped into clusters. The centroid of each cluster $C = \{c_1, .c_2, \ldots.c_n\}$ is determined by computing the Euclidean distance between each UAV and its corresponding centroid.

$$C_{UAVi} = \frac{\sum_{x=1}^{x=1} x}{\sum UAV_n}, \frac{\sum_{y=1}^{y=1} y}{\sum UAV_n}, \frac{\sum_{z=1}^{z=1} z}{\sum UAV_n} \tag{10}$$

Eq. (10) represents the geometric center of a group of UAVs in three-dimensional space. This centroid is a vector composed of the average x, y, and z coordinates of all UAVs belonging to the cluster. The numerator in each term, $\sum x$, $\sum y$, and $\sum x$, signifies the sum of all respective coordinates from every UAV within the cluster. The denominator $\sum UAV_n$, is the total number of UAVs in that cluster. To form the cluster and ensure that UAVs are within it, we have applied the sum of the minimum distances.

$$SMD = \sum_{C=1}^{C=N} \left[\sum_{UAV_i \in C_{UAVi}}^{U_n} d(UAV_i, C_{UAVi})^2 \right] \tag{11}$$

Eq. (11) defines the Sum of Minimum Distances (SMD), which quantifies total cluster compactness. Here, N is the total number of clusters, UAV_i represents an individual UAV,$C_{UAVi}$is the centroid of a cluster, and $d(UAV_i, C_{UAVi})$ is the Euclidean distance between a UAV and its centroid. The inner sum calculates the sum of squared distances of all UAVs within a cluster to their centroid, while the outer sum aggregates this value across all clusters. Minimizing the SMD ensures that UAVs are assigned to optimally compact clusters. This process repeats the iteration until it reaches the maximum number of iterations, as shown in Algorithm 1 below.

Fig. 2 shows the clustering formation process, where LM is a Laplacian matrix that contains the Eigenvector values e, as shown in the connections between UAVs. The UAVs with strong e values are considered in one cluster and the weak e values separate the one UAV cluster from the other. We have used the NJ Jordan algorithm above for graph separation. The whole network is divided into N sub-networks$N = \{n_1, n_2, n_3, n_4, n_5$ as shown in Fig. 2.

Algorithm 1
Spectral clustering for UAV networks.

Line	Algorithm Step
1	Input: N ← Number of UAVs, k ← Number of neighbors to consider, NG ← Number of subnetworks to create, clusters ← Number of clusters to form
2	Output:Clustered UAV groups
3	Initialize neighborList ← [], subnetworkList ← [], clusterList ← []
4	// Step 1: Find k-nearest neighbors
5	for $i = 1$ to N do
6	Find k nearest neighbors of UAV i
7	Store neighbors in neighborList[i]
8	end for
9	// Step 2: Compute Laplacian matrix
10	Construct adjacency matrix A from neighborList
11	Compute degree matrix D
12	Compute Laplacian matrix $L = D$ – A (Eq 0.6)
13	// Step 3: Divide into subnetworks
14	Apply NG-Jordan-Weiss algorithm on L
15	Store resulting subnetworks in subnetworkList
16	// Step 4: Perform clustering
17	for each subnetwork in subnetworkList do
18	Apply k-means clustering on the subnetwork (Eq.11)
19	Store resulting clusters in clusterList
20	end for
21	return clusterList

Algorithm 1 commences with the crucial task of defining key parameters: the number of UAVs N the number of neighboring UAVs to consider$k$, the desired number of subnetworks, and the targeted number of clusters. The initial step involves iterating over each UAV to identify its$k$ nearest neighbors. This process is efficiently conducted through a for loop, where each UAV's immediate environment is meticulously examined. The identified neighbours are stored in a structured array, neighborList, thereby ensuring a comprehensive mapping of each UAV's local neighbourhood, which is vital for precise network analysis. The second step involves constructing an adjacency matrix $A$from the neighborList, which encapsulates the connectivity architecture among the UAVs. Following the establishment of this matrix, the degree matrix $D$is computed, detailing each UAV's direct connections. Subsequently, the Laplacian matrix L is derived through$L = D - A$. This matrix is instrumental in elucidating the network's structural framework and is pivotal in subsequent steps to segregate the network into discrete subnetworks.

In the third step, the NG-Jordan-Weiss algorithm is employed on the Laplacian matrix, effectively partitioning the network into several subnetworks. This segmentation captures the intrinsic connectivity patterns among UAVs, allowing these subnetworks to be cataloged into an array known as subnetworkList. This organization not only facilitates the efficient management of smaller network sections but also enhances analytical accuracy. The final step involves applying k-means clustering to each subnetwork within the subnetwork list. This method classifies the UAVs into cohesive clusters based on intrinsic characteristics. The resultant clusters are systematically stored in an array, clusterList, thereby organizing the UAVs into logical groupings that accurately reflect spatial proximity and communication patterns. This comprehensive approach optimally positions the network for advanced operational efficacy and enhanced communication reliability, ensuring that UAVs operate within intelligent, dynamically structured clusters.. The algorithm then returns the array of clusters as the output. The overall time complexity of Algorithm 1 is $O(N^3)$ in theory due to the eigen-decomposition of the Laplacian matrix; however it reduces to approximately $O(N^2)$ in practice because the UAV network graph is sparse and efficient eigen-solvers are applied.

3.2. Optimal path reinforcement learning

To estimate the optimal path from the source to the destination for the UAV, a reinforcement learning-based reward path is adopted in this paper. Each UAV in the network is represented as a state space, where i is the 3D coordinate of state i. The Action state is set by considering the Action space, which means the change in 3D position and speed for action j. We have formulated transition T, which counts the transition probability from a state to the state after taking an action.

$$T(s_i, a_j, s_k) = P(S_k | s_i, a_j) \tag{12}$$

Where s_i the current state (3D position) of the UAV is, $a_j$is the action representing a change in position and speed, ands_k is the resulting next state. The shortest path from the source UAV to the destination is optimised by using a reward function that counts the distance (d) travelled from the source to the destination, the energy consumed during data forwarding, the connectivity value c, and the weights as described in the Equation.

$$r(s_i, a_j, s_k) = w_1 d_{ij} + w_2 d_{ij}(v) + w_4 c \tag{13}$$

In Eq. (13), term d_{ij} is distance, c is connectivity metric and w1, w2,

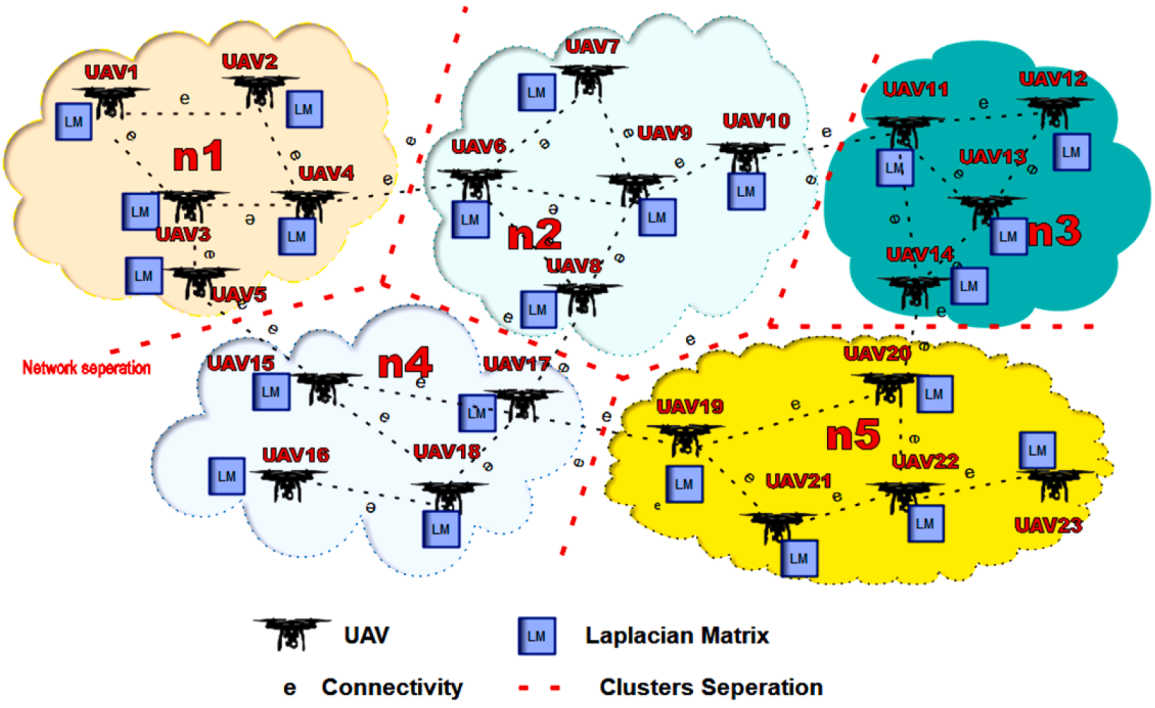


Fig. 2. Spectral reinforcement learning UAV network.

and w3 are weights, respectively. The connectivity metric is defined as the negative logarithm of the probability of connectivity between states. The energy metric is defined as the energy consumed to move from the current states_i to next states_k at speed v. By optimizing the path using this system model, the UAV ad hoc network can achieve efficient and reliable communication. The energy value can be calculated from the routing metric and connectivity metric values, which are calculated using Eqs. (14) and (15).

$$c(s_i, a_j, s_k) = -\log(1 - P_c(s_k)), where\ P_c(s_k) r(s_i, a_j, s_k) = d_{ij}(v) \tag{14}$$

Where $P_c(s_k)$ denotes the probability of maintaining a stable connection with s_k and $c(s_i, a_j, s_k)$ is the resulting connectivity cost, computed using the negative logarithm transformation. This logarithmic formulation ensures that when $P_c(s_k)$ is close to 1 (i.e., the link is highly reliable), the penalty$c(s_i, a_j, s_k)$ approaches zero, resulting in a minimal cost. Conversely, as $P_c(s_k)$approaches 0 (i.e., the link is unstable or likely to break), the value of $-\log(1 - P_c(s_k))$,increases sharply toward infinity, imposing a heavy penalty on the routing decision. This strongly discourages the reinforcement learning agent from selecting paths through poorly connected UAVs.

$$d_{i,j}(v) = \alpha \times Distance + \beta \times \left(\frac{1}{UAV_{speed}}\right) + \gamma \times Link_{Quality} + e \tag{15}$$

The term $d_{i,j}$in Eq. (15) represents the routing cost metric, which models the energy and efficiency implications of forwarding data from the current UAV to the next hop at a given speed UAV_{speed} . It considers the physical distance between UAVs, the inverse of the UAV's speed (to account for energy-efficient flight), the quality of the wireless link (reflecting signal strength and potential retransmissions), and energy e.

For the estimation of the highest reward function in a particular state, the reward value is calculated with the help of the value function. The value function is represented as $V(s_i)$, which shows the expected cumulative reward that an agent can get starting from state s_i and following an optimal policy. The optimal policy determines the expected cumulative reward, which is derived by adding the value of the immediate reward to the discounted expected cumulative reward of the ultimate state. The discount factor (γ) weighs current gains against future ones. The value function is recursively defined since it is dependent on the value function of the resulting state(s_k).

$$V(s_i) = max_{a \in A}\left\{r(s_i, a_j, s_k) + \gamma \sum kP(s_k|s_i, a_j) V(s_k)\right\} \tag{16}$$

Based on the value function optimal policy, we have formed an optimal policy to select the shortest path. Optimal policy explores new paths, starts a fraction ε of episodes from each state's ∈ S, followingπ^* from there, otherwise selects aε-greedy action selection with probability ε, and selects a random action$a_j \in A$. The optimal policy is described in Eq. (17) below.

$$\pi \times (s_i) = argmax_{a \in A}\left\{\sum kP(s_k|s_i, a_j) V(s_k) + \gamma(s_k)\right\} \tag{17}$$

The Q-learning method updates the Q-value for each state-action pair, $Q(s_i, a_j)$, based on the Bellman equation, which calculates the greatest predicted total of rewards over all potential following states and actions.

$$Q(s_i, a_j) = r(s_i, a_j, s_k) + \gamma max a' \epsilon A Q(s_k, a') \tag{18}$$

Here, in Eq. (18) $s_i$represents the current state of the UAV (e.g., its position and connectivity), a_j is the chosen action (e.g., which neighbor to forward a packet to), and s_k is the resulting new state. The immediate reward received for this transition is $r(s_i, a_j, s_k)$, which quantifies the direct benefit (e.g., reduced distance, high signal strength). The discount factor γγ determines the importance of future rewards versus immediate ones. Finally,$\gamma max a' \epsilon A Q(s_k, a')$ represents the maximum estimated future reward achievable from the new state $s_k$by taking the best possible subsequent action a', ensuring the decision is optimal for the entire path.

$$\pi \times (s_i) = argmax_{a \in A} Q(s_i, a_j) \tag{19}$$

Equation (19)determines the policy is a function that dictates the best possible action a_j to take when the agent is in any given states_i. The$argmax$operator specifies that the chosen action is the one that maximizes the corresponding Q-value$Q(s_i, a_j)$.Further, learning rate, α, determines the rate at which the Q-values are updated. The connectivity

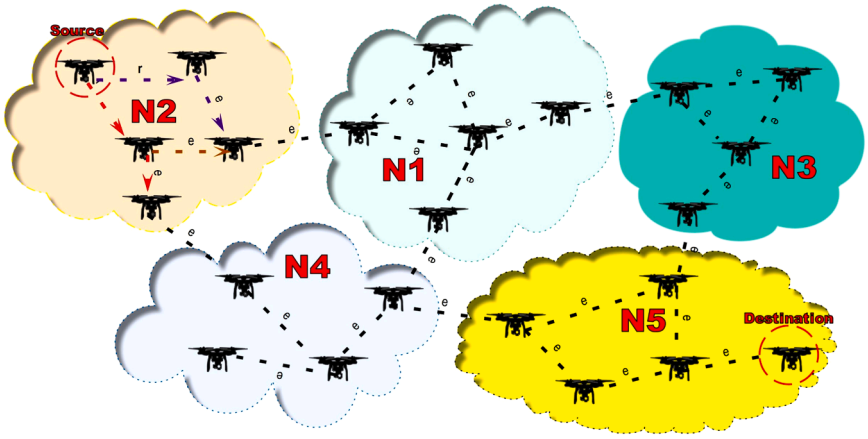


Fig. 3. Route Inquiry Phase.

metric, $c(s_i,a_j,s_k)$, is defined as the negative logarithm of the probability of connectivity between a_j state s_i and s_k as shown in Eq. (20).

$$\alpha : Q(s_i,a_j) = Q(s_i,a_j) + \alpha\{r(s_i,a_j,s_k) + \gamma max\alpha' \epsilon AQ(s_k,a') - Q(s_i,a_j)\ \} \tag{20}$$

In Eq. (20), $Q(s_i,a_j)$indicates the current estimation of the expected cumulative reward for acting a_j in state s_i, whereas s_k represents the next state resulting from action a_j. The reward $r(s_i,a_j,s_k)$ refers to the immediate reward obtained by executing an action in a state s_i and then transitioning to a state s_k. The learning rate (α) influences the rate at which Q-values are modified. A high learning rate results in rapid updates, whereas a low learning rate produces more gradual changes. The discount factor (γ) weighs current gains against future ones. The term $\gamma max\alpha' \in AQ(sk,a')$ refers to the maximum expected cumulative reward for performing any action a' in the following state sk. The temporal difference error is calculated by subtracting the current estimate $Q(s_i,a_j)$ from this value. It displays the discrepancy between the projected and actual reward results Fig. 3.

3.3. Working principle

The proposed scheme operates in two phases. In the first phase, it forms the clusters by adopting spectral clusters. The second phase described the optimal path selection by utilizing reinforcement learning, which counts the optimal policy based on the reward function. In the paragraph below, we have described the optimal path selection process.

Step 1: Source UAV Sends Inquiry. Initially, the source UAV sends a route inquiry request to nearby UAVs (within the cluster) to find a path to the destination UAV.

The route inquiry request contains the location of the source UAV, cluster number, and destination UAV. When the nearby UAVs receive the inquiry, they use their state space S and action space A to calculate the transition probability $T(s_i,a_j,s_k)$ from their location to the cluster in which the destination UAV belongs. Based on the transition probability, the nearby UAVs send a response to the source UAV with a list of possible paths to reach the destination UAV (usually, the UAVs near those cluster paths are counted). As shown in Fig. 4. The source UAV sends an inquiry request message to the neighbouring UAVs within the cluster, and then the neighbouring UAVs estimate the paths that lead to

P1	UAV2	UAV4	UAV8	UAV17	UAV19	UAV20	UAV22	UAV23
P2	UAV3	UAV5	UAV15	UAV17	UAV19	UAV21	UAV22	UAV23
P3	UAV3	UAV4	UAV6	UAV9	UAV10	UAV14	UAV20	UAV22	UAV23
P4	UAV3	UAV4	UAV6	UAV9	UAV10	UAV8	UAV17	UAV19	UAV20	UAV22	UAV23

Fig. 5. List of Pat.

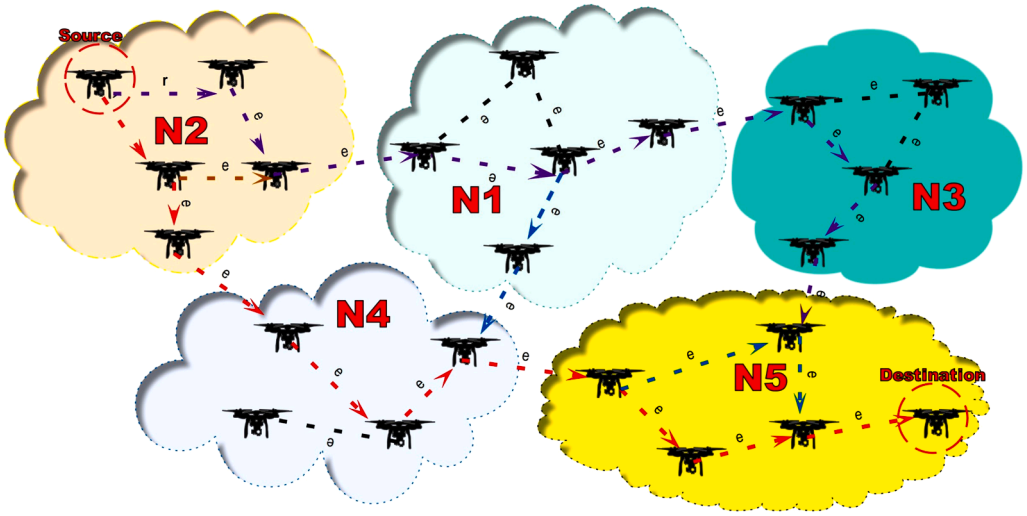


Fig. 4. Shortest Paths.

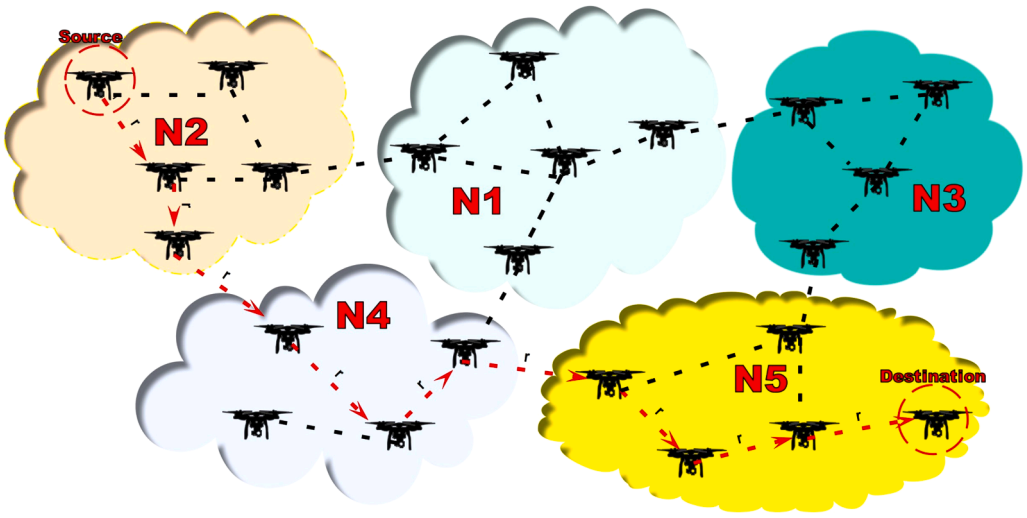


Fig. 6. Data Forwarding.

the destination UAV.

Step 2: List of Paths: The source UAV receives the responses from the nearby UAVs and creates a list of possible paths to reach the destination UAV. The list of paths is based on the connectivity metric $c(s_i, a_j, s_k)$, routing metric $e(s_i, a_j, s_k)$, and distance metric $d(s_i, a_j, s_k)$ of each path.

The source UAV calculates the reward function $r(s_i, a_j, s_k)$ using the weights w1, w2, and w3 for each path in the list. Fig. 4 illustrates that neighbouring UAVs estimate the path by utilising reinforcement learning, which involves counting the reward function for each path. After estimating the path, the neighbouring UAVs send back the path information to the source UAV.

These paths can be observed through arrows of different colors, which indicate the routes leading to the destination UAV. Figs. 5, 6 shows the four optimal paths from the source UAV to the destination UAV. Each path takes a different route to reach the destination.

Step 3: Find the Optimal Path: The source UAV uses the reinforcement learning technique Q-learning to update its Q-values for each state-action pair, $Q(s_i, a_j)$, based on the expected cumulative reward.

The Q-values are updated using the Bellman equation, which calculates the maximum expected sum of rewards over all possible next states and actions. The source UAV uses the Q-values to determine the optimal policy $\pi \times (s_i)$that maximizes the expected cumulative reward. The optimal policy is the one that results in the shortest and most efficient path for the UAV to reach its destination, taking into account factors such as connectivity, routing decisions, and distance. The path with the highest reward is selected as the optimal route, and data will be forwarded through that route.

Algorithm 2 initiates its process by defining the state and action spaces, establishing the state space (S) as a set of 3D coordinates and the action space (A) as a set of changes in position and speed represented by $(\Delta x, \Delta y, \Delta z)$ and Δv. In the subsequent steps, the algorithm formalizes key functions necessary for the learning process. The transition function $T(s_i, a_j, s_k)$ is defined to elucidate the dynamics of state transitions based on selected actions, while the reward function $r(s_i, a_j, s_k)$ is established to quantify the rewards associated with transitions between states. Additionally, the algorithm introduces the value function $V(s_i)$, which assigns a value to each state, and the optimal policy $\pi^*(s_i)$, which serves as a guiding framework for making decisions within the environment.To facilitate the exploration of various actions, the algorithm

Algorithm 2
Initialize State, Action Spaces, and Q-Learning.

Line	Algorithm Step
1	Input: State space: S = { (x, y, z) } (3D coordinates), Action space: A = { Δx, Δy, Δz, Δv } (position & speed changes)
2	Output: Trained Q-table representing optimal policy
3	Define transition function T(s_i, a_j, s_k)(Eq.12)
4	Define reward function r(s_i, a_j, s_k)
5	Define value function V(s_i)
6	Define optimal policy π*(s_i)
7	Initialize exploration strategy: Exploring starts.
8	Initialize action selection strategy: Epsilon-greedy selection
9	Initialize Q-table Q for state-action values (Eq.18)
10	Set learning rate α
11	Set discount factor γ
12	for each episode do
13	Start from the initial state s_i
14	repeat until terminal state reached do
15	Select action aj using epsilon-greedy policy.
16	Take action a_j, observe next state s_k and reward.
17	Update Q(s_i, a_j) using Bellman equation (Eq.20)
18	Update current state $s_i \leftarrow s_k$
19	end repeat
20	end for

Algorithm 3
Bellman equation for value function and optimal policy.

Line	Algorithm Step
1	Input: State space S, Action space A, Weights w1, w2, w3
2	Output: Optimal value function V(si) and policy π*(si)
3	Define transition function T(si, aj, sk):
4	Compute the probability of transitioning from si to sk after taking action aj
5	Return probability
6	Define reward function r(si, aj, sk):
7	Compute the distance traveled closer to the destination.
8	Compute routing value
9	Compute connectivity
10	Compute reward using weights w1, w2, w3
11	Return reward
12	Define value function V(si):
13	For each action aj in A do
14	Compute the value using Bellman equation
15	End for
16	Return the maximum value
17	Define optimal policy π*(si):
18	For each action aj in A do
19	Compute the value using Bellman equation
20	End for
21	Return action that maximizes value
22	Define epsilon_greedy(si):
23	With probability epsilon: select random action aj in A
24	Else: select action that maximizes value from π*(si)
25	Return selected action

implements an exploration strategy, initializing an epsilon-greedy action selection method that effectively balances the trade-off between exploration of new actions and exploitation of known rewarding actions. It sets up a Q-table (Q) to store the estimated state-action values, along with defining the learning rate (α) and the discount factor (γ) to ensure effective learning.

As the algorithm progresses through each episode, the agent begins from an initial state (s_i) and continues the iterative process until it reaches a terminal state. During each iteration, the agent applies the epsilon-greedy policy to choose an action a_j and then records the next state s_k and the corresponding reward. The Q-table is updated using the Bellman equation to refine the state-action value$\left(Q(s_i, a_j)\right)$. The agent then transitions to the newly observed state (s_k), and this iterative learning process continues until the agent completes the designated episodes. Through repeated interaction with the environment, the agent accumulates experiences that enhance its decision-making capabilities, enabling it to converge on an optimal policy that maximises cumulative rewards.

Algorithm 3 implements a systematic framework to derive an optimal value function and policy through reinforcement learning principles. The process begins with the computation of the transition function (T(s_i, a_j, s_k))$T(s_i, a_j, s_k)$, which quantifies the probability of transitioning from a current state (s_i) to a subsequent state (s_k) upon executing action (a_j). This probabilistic assessment is essential for the agent's comprehension of state transitions within the environment. Next, the reward function $r(s_i, a_j, s_k)$ quantifies the reward associated with each state transition. It synthesises key factors, including progress toward the destination, routing efficiency, and connectivity. Each of these components is assigned specific weights w_1, w_2 and w_3 , allowing the function to produce a comprehensive reward value that reflects the multidimensional performance metrics critical for enhancing the agent's learning trajectory. The value function $V(s_i)$ evaluates the utility of being in a certain state (s_i) at a given moment. Utilizing the Bellman equation, this function assesses all possible actions within the action space A and returns the maximum value, thereby indicating the most advantageous path in terms of long-term benefits.

In parallel, the optimal policy function $\pi^*(s_i)$,is designed to evaluate the potential value of each action in the action space using the Bellman equation. From this evaluation, the agent selects the action that maximises the expected value, thereby guiding it toward the most beneficial decisions.

Finally, the epsilon-greedy functionencompasses a strategy for balancing exploration and exploitation in action selection. With a probability epsilon, the agent randomly chooses an action from the action space to explore new opportunities. Conversely, it selects the action that maximizes the value derived from the optimal policy $\pi^*(s_i)$,with a probability of (1 - epsilon). This sophisticated approach enables the agent to navigate its environment effectively, facilitating improved learning and optimising decision-making processes.

4. Experiment setup

In this section, we compare the proposed Spectral Reinforcement Learning (SRL) scheme with several benchmark routing protocols: AODV [48], OLSR [49], CSPO [50], K-Means [51], and OGR [52,53]. Three different mobility scenarios are generated to evaluate the effectiveness of the proposed method under dynamic network conditions. To assess performance, we consider key Quality of Service (QoS) metrics including packet delivery ratio (PDR %), end-to-end delay(s), and normalized routing load (NRL %). The simulation environment comprises three different mobility scenarios with varying numbers of UAV nodes, enabling the creation of an Ad-hoc network environment with 3D coverage. Each UAV uses the 5 G version of the Wi-FiWi-Fi standard, IEEE 802.11ac, and communicates with each other using 20 dBm transmission power. Each simulation runs for a total duration of 120 s.

Table 2
Simulation and training parameters.

Simulation Parameter	Value
NS-3 Version	3.35[54]
Computer Hardware	Ubuntu 22.04 LTS in Core i5 12 G 16GB RRAM, GPU 3050Ti
Number of nodes	10, 20, 30, 40 and 50 UAV nodes
Simulation area dimensions	2000 × 2000 × 80 (m^3)
Wireless Network Type	IEEE 802.11ac Ad hoc mode (5G-WiFi)
Central frequency	5 GHz
Link Speed/Data rate	6.5 Mbps
Channel width	20 MHz
Number of spatial streams (NSS)	2
Source: destination UAVs	4:4
Offered data rate per source	20 Mbps
Propagation Delay	Constant Speed Propagation Delay Model
Propagation Loss	Friis Propagation Loss Model
Transmission Power	20 dBm
Simulation Duration	120 s
Routing Protocol	SRL, AODV, OLSR, CSPO, K-MEANs, and OGR
UAV average speed	20,40, and 60 m/s
UDP Packet Size	1472 Bytes
Mobility Model	Gauss-Markov[53]
Number of Simulations	270
Training Parameters	**Value**
Number of Episodes	12,000
Learning Rate	0.0008 to0.0001
Discount Factor	0.99
Epsilon (start)	1.0
Epsilon (final)	0.1–0.2
Batch Size	64

Table 2 shows the simulation parameters used during the experimental work.

5. Results and discussions

This section presents the simulation results of the proposed SRL routing scheme, comparing it with existing protocols under various UAV mobility scenarios and different traffic loads.

5.1. Packet delivery ratio (PDR)

The Packet Delivery Ratio (PDR) is one of the key parameters for measuring the efficiency of routing protocols, representing the ratio of the total number of packets generated at the source UAV node to those successfully received at the destination UAV node. We have evaluated all routing schemes under different mobility scenarios as shown in Fig. 7 (a), (b), and (c). From Fig. 7(a), it can be observed that the topology-based schemes AODV and OLSR achieve lower PDR values as the number of UAVs increases; although AODV initially starts at 78 %, its PDR declines rapidly with network size and mobility due to its reliance on predetermined source-to-destination paths, which become unstable in highly mobile UAV environments. This instability worsens at higher mobility rates, as seen in Fig. 7(b) for 40 m/s and Fig. 7(c) for 60 m/s, where link breakages and route re-computations cause sharp drops in delivery performance. In contrast, our proposed SRL scheme consistently achieves the highest PDR across all mobility conditions, maintaining above 95 % for smaller UAV counts and still achieving 82–89 % at the highest density and mobility. This shows improvement of approximately 12–18 % over K-Means, 25–35 % over OGR, and 30–40 % over topology-based protocols such as AODV and OLSR in the most challenging high-mobility scenarios. This superior performance stems from the integration of spectral clustering, which divides the network into manageable sub-regions, and reinforcement learning, which adaptively selects optimal routes using a reward function that accounts for connectivity, routing optimality, and inter-UAV distance, enabling shorter and more resilient paths even under rapid topology changes. The

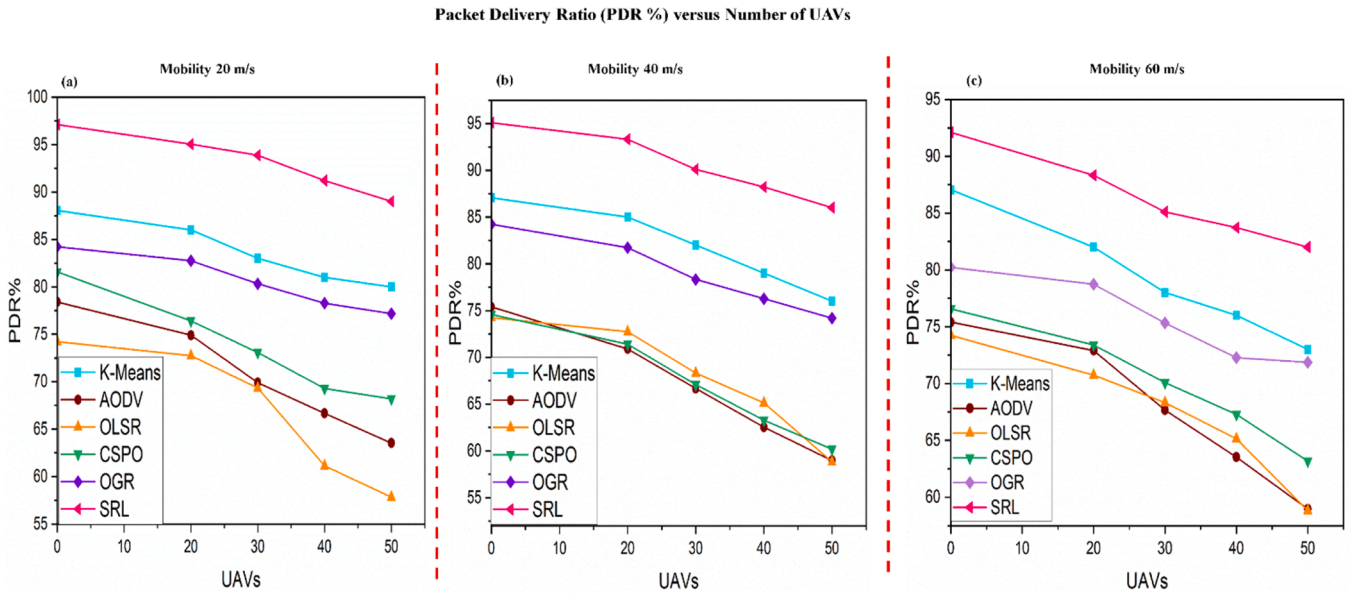


Fig. 7. PDR % verus Number of UAVs (a) Mobility 20 m\s, (b) Mobility 40 m\s, (c) Mobility 60 m\s.

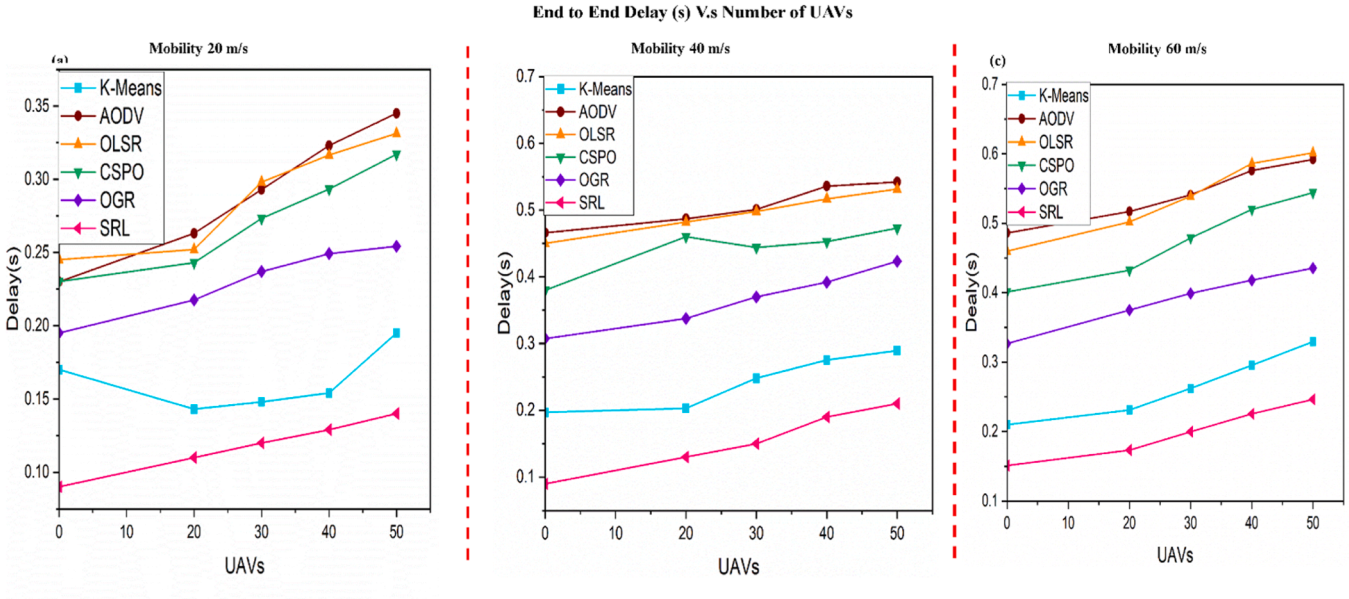


Fig. 8. End to End Delay (s) versus Number of UAVs (a) Mobility 20 m\s, (b) Mobility 40 m\s, (c) Mobility 60 m\s.

second-best results are produced by K-Means, a machine learning–based method, which reaches 87 % at 20 m/s mobility and starts from 85.6 % at 60 m/s before decreasing to 76 % as UAV count increases; however, K-Means suffers when UAVs move randomly and do not form circular clusters, leading to irregular cluster shapes and overlapping membership, especially at high mobility, which degrades its stability. The third-best results are achieved by OGR, which, although not a machine learning protocol, benefits from high connectivity and next-hop selection based on routing decisions, allowing it to outperform CSPO, AODV, and OLSR, but it still lacks adaptive learning to sustain optimal performance in fast-changing topologies.

5.2. End-to-end delay

The end-to-end delay is the time taken for packets to successfully travel from the source UAV node to the destination UAV node, and it serves as a critical measure of protocol efficiency the lower the delay, the more effective the routing protocol. Fig. 8(a), (b), and (c) show the delay performance of each scheme under different mobility conditions. At a mobility of 20 m/s (Fig. 8(a)), the topology-based protocols AODV and OLSR experience delays of around 23 % and increase to 34 % as the number of UAVs grows, primarily due to frequent route recalculations and unstable links in mobile environments. In contrast, the machine learning–based K-Means protocol, which offers the second-best results overall, achieves lower delays at low mobility because it can form stable clusters efficiently; however, as mobility increases to 40 m/s (Fig. 8(b)), its delay rises to about 27 %, and at 60 m/s (Fig. 8(c)) it reaches approximately 30 %, reflecting the negative impact of irregular cluster shapes and overlapping membership under high mobility. The third-best performer, OGR, starts with a delay below 23 % at 20 m/s but climbs to about 36 % at 60 m/s, as its connectivity-driven routing decisions are not adaptive enough to maintain low latency in rapidly changing topologies. CSPO performs better than purely topology-based protocols because its particle swarm optimisation features divide the network into

manageable groups; however, its delay still increases noticeably with mobility. From all three subfigures, it is evident that our proposed SRL scheme maintains the lowest delays across all mobility levels — starting at just 10.5 % at 20 m/s and remaining around 18 % even at 60 m/s. This represents a reduction of approximately 33–40 % compared to K-Means, 50–55 % compared to OGR, and over 60 % compared to AODV and OLSR in the highest-mobility scenarios. SRL achieves this by utilising spectral clustering, which accounts for eigenvector values, node degree, and Laplacian matrix properties to form optimal clusters, regardless of the UAV's movement patterns. At the same time, reinforcement learning selects the shortest and most reliable paths based on connectivity, route optimality, and hop weights. As a result, SRL consistently delivers much lower delays than all other schemes, ensuring suitability for real-time UAV applications where latency must remain minimal.

5.3. Normalized routing load

NRL describes the number of packets generated divided by the number of packets delivered to the destination. As it represents the traffic load generated across the network, the routing protocol with less NRL is considered a more efficient scheme. Fig. 9(a), (b), and (c) show the values of the NRL of different mobility's. Fig. 9 shows that the proposed scheme maintains stable performance, with NRL values of 12 % at mobility 20, 15.8 % at mobility 40, and remaining below 18 % at mobility 60, as illustrated in Fig. 9(a)–(c). In comparison, the other schemes experience a higher normalised load as UAV mobility increases, because greater speed increases the chances of link breakage. Initially, the k-means routing scheme NRL value is low due to the low speed of UAVs, which can form clusters and have stable paths through the cluster head. However, as the mobility of UAVs increases, it causes unstable paths and overlapping clusters, thereby raising the issue of link breakage. As a result, the cluster heads generate more traffic to forward data; therefore, the NRL value increases.

On the other hand, topology-based routing schemes (AODV and OLSR) have the highest NRL because they operate on predetermined paths. Since UAVs move fast, there is no stable environment to form paths. These schemes generate extra packets, which causes high NRL. The other two schemes, OGR and CSPO, offer NRL at under 40 % under different scenarios. These schemes also suffer due to the absence of stable links between the UAVs.

5.4. Training performance metrics

The comparative training analysis, illustrated in Fig. 10 and its subfigures (a), (b), (c), and (d), provides comprehensive empirical validation for the superior performance and learning efficiency of the proposed Spectral Reinforcement Learning (SRL) framework over the course of the training episodes. As shown in subfigure 10(a), the Average Episode Reward for SRL not only converges to a significantly higher value but also achieves a much more stable plateau compared to the benchmark schemes after approximately 4000 episodes. This performance is a direct consequence of the foundational two-stage architecture of SRL. The initial application of spectral clustering, which leverages the eigenvector decomposition of the graph Laplacian (Eqs. (6), (9)), dynamically partitions the network into topologically coherent clusters. This preprocessing creates a simplified and stable state space, allowing the subsequent reinforcement learning agent to learn a more effective policy (Eq. (17)) that expertly maximizes the multi-objective reward function (Eq. (13)). In contrast, the K-means and simple RL rewards exhibit higher variance and slower, less stable convergence throughout the 12,000 episodes. This efficient architectural design directly facilitates a more optimal Exploration vs. Exploitation strategy, as depicted in subfigure (b). The SRL curve shows a rapid and monotonic decline in its exploration rate, indicating that the agent, operating within well-defined clusters, can efficiently identify high-quality paths and swiftly transition to exploiting this knowledge after just 2000 episodes. This stands in stark contrast to the K-means and simple RL approaches, which are burdened by a noisy and high-dimensional state space, forcing them to maintain a higher rate of exploration for a significantly longer duration. The rapid and stable convergence of the SRL agent is further evidenced in subfigure (c), the Learning Rate Schedule. The SRL curve decays the fastest, signifying that the algorithm's Q-value estimates (updated via the Bellman optimality principle in Eq. (18) and Eq. 21) stabilize quickly as a result of learning from a consistent and predictable environmental model underpinned by robust clusters. The other schemes, by comparison, require a higher effective learning rate for longer to adapt to their more volatile learning environments. Finally, this entire process culminates in the Training Loss profile shown in subfigure 10 (d). SRL demonstrates swift and smooth convergence to a minimal loss value after roughly 6000 training steps, reflecting highly accurate value function approximations (Eq. (16)) and low temporal difference errors. Conversely, the K-means-based

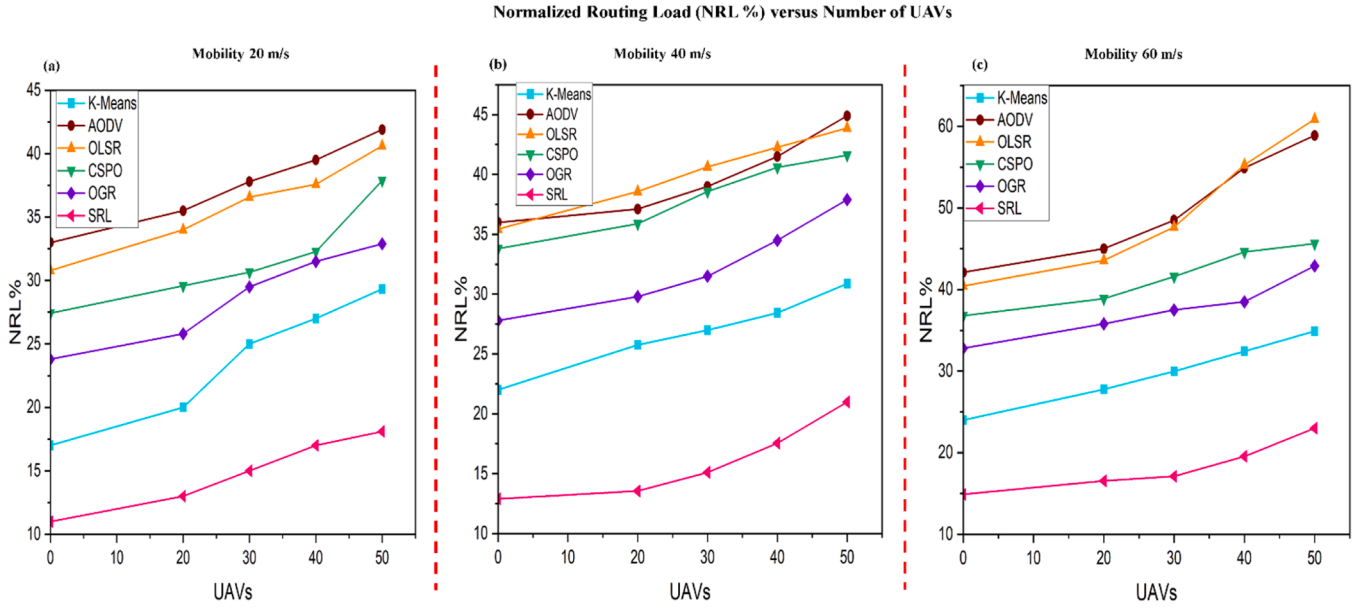


Fig. 9. NRL % versus Number of UAVs (a) Mobility 20 m\s, (b) Mobility 40 m\s, (c) Mobility 60 m\s.

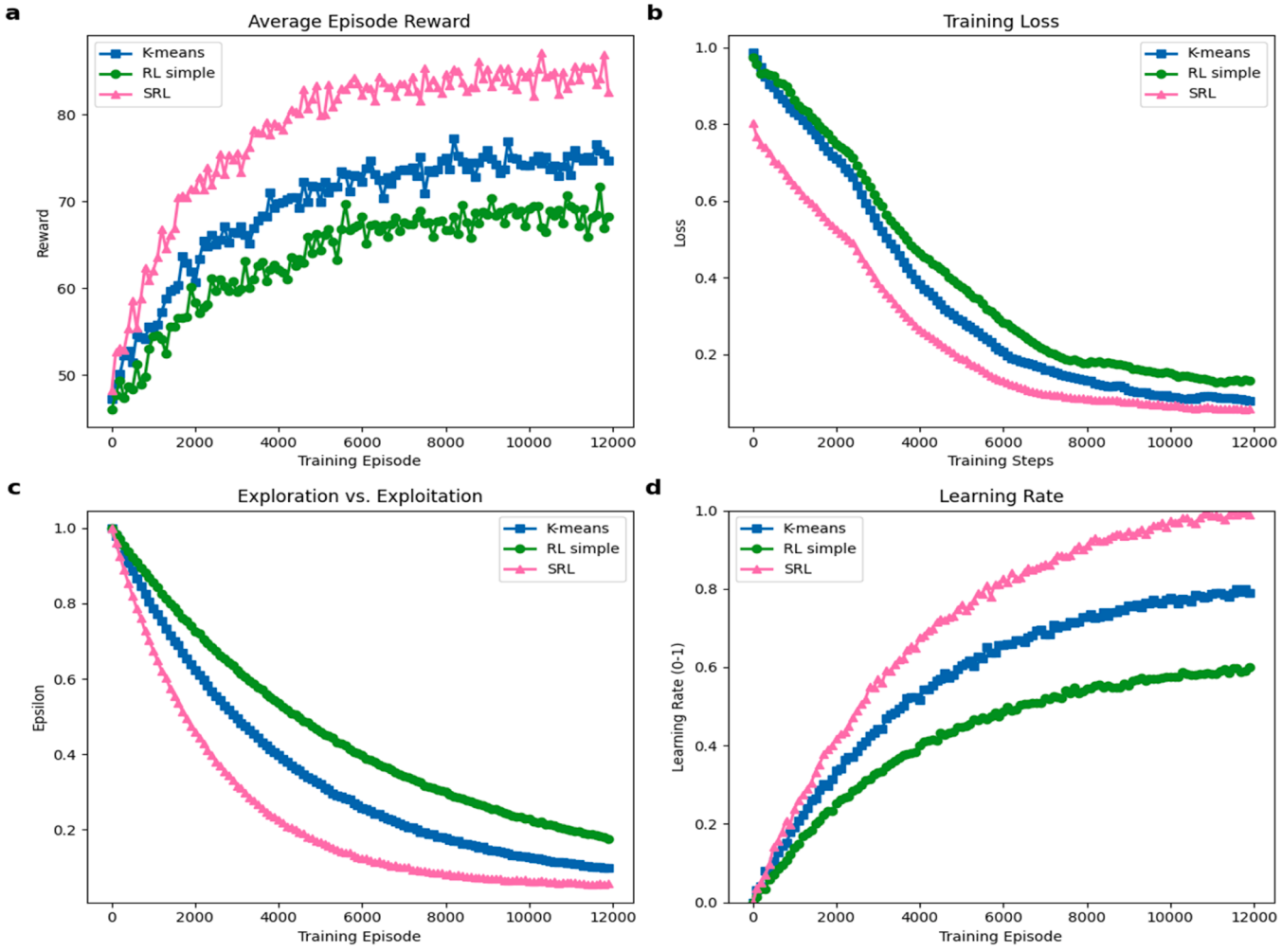


Fig. 10. Training Performance Metrics.

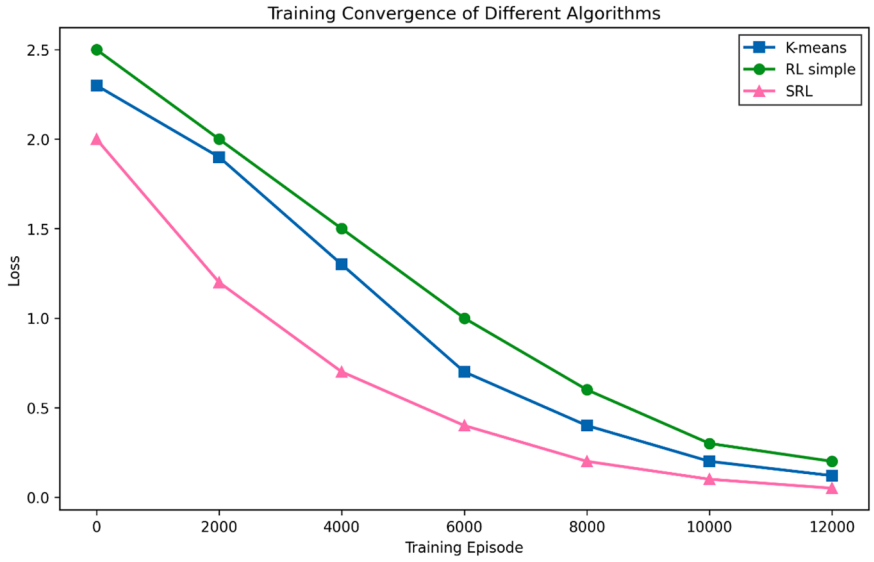


Fig. 11. Convergence metric.

approach suffers from its inherent limitation in forming optimal clusters, injecting significant noise and instability into the learning process, resulting in a higher and more erratic loss curve. The simple RL agent, lacking any clustering mechanism, is fundamentally overwhelmed by the intractably large state space of the entire network, exhibiting the highest and most volatile loss throughout the training.

5.5. Convergence metrics

The convergence graph in Fig. 11 shows that the proposed scheme SRL demonstrates superior performance, evidenced by its steeper loss reduction and lower asymptotic error compared to K-means and the RL simple. This accelerated convergence is a direct result of SRL's integration of spectral clustering, which dynamically groups UAVs based on eigenvector-derived similarity metrics (e.g., signal strength, proximity) into non-linear, optimal clusters without predefining their shape or number. This structural adaptability minimizes routing complexity and prevents cluster overlap issues that plague K-means in dynamic topologies. Furthermore, the embedded reinforcement learning component continuously optimizes routing paths within clusters by leveraging a multi-objective reward function (Eq. (13)) that balances distance, connectivity, and energy metrics. This enables SRL to rapidly learn and exploit efficient paths, reducing packet loss and end-to-end delay, as confirmed by the experimental results.

6. Conclusion

In this study, we introduce a novel methodology for optimizing the performance of UAV ad hoc networks by combining spectral clustering with reinforcement learning. By utilising machine learning approaches, the issues that UAV networks encounter, including dynamic topology, limited battery life, and restricted communication range, can be overcome. The proposed approach enables efficient management of communication resources and improves overall network performance. Moreover, by using reinforcement learning for each cluster, the most efficient routing paths were identified in response to the network's condition and the UAV's position, further enhancing the network's effectiveness. The result of the experimental evaluation showed the usefulness of the proposed framework. There was an improvement in communication efficiency, as evidenced by the reduced number and frequency of packets transmitted, decreased end-to-end delay, and increased packet delivery ratio. By integrating spectral clustering and reinforcement learning, the routing decisions adapt fully to real network conditions and communication patterns due to their flexibility. This flexibility represents the best approach in a dynamic environment where the conventional routing algorithms fail to operate optimally. Therefore, It can be concluded that this work's contribution goes beyond previous work. Hence, this paper proposed a solution that leverages the features of UAV ad hoc networks by integrating both spectral clustering and reinforcement learning methods, thereby enhancing the communication and routing capabilities of the network.

In future work, this framework can be extended by evaluating its performance in heterogeneous and large-scale UAV network scenarios, including integration with terrestrial, satellite, and IoT systems. Real-world trials and hardware-in-the-loop simulations can be conducted to assess the practical effectiveness and address implementation challenges such as latency under varying weather conditions, energy optimisation, and security threats. Furthermore, integrating advanced deep reinforcement learning models could enhance scalability and adaptability in ultra-dense, high-mobility environments. Finally, expanding the performance analysis to include metrics such as jitter, packet loss under interference, and computational overhead will provide a more comprehensive validation of the proposed approach.

CRediT authorship contribution statement

Saif ullah: Writing – original draft, Software, Methodology, Formal analysis, Conceptualization. **Khalid Hussain:** Writing – review & editing, Supervision, Software, Resources, Project administration, Investigation. **Muhammad Faheem:** Writing – review & editing, Validation, Supervision, Software, Methodology, Formal analysis, Data curation. **Nisar Ahmed Memon:** Visualization, Validation, Project administration, Formal analysis.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was funded by the VTT Technical Research Centre of Finland and Shenzhen Kaihong Digital Industry Development Co., Ltd., Shenzhen China.

Data availability

Data will be made available on request.

References

[1] H. Wang, H. Zhao, J. Zhang, D. Ma, J. Li, J. Wei, Survey on unmanned aerial vehicle networks: a cyber physical system perspective, IEEE Commun. Surv. Tutor. 22 (2) (2020) 1027–1070.
[2] R. Nagasawa, E. Mas, L. Moya, et al., Model-based analysis of multi-UAV path planning for surveying post-disaster building damage, Sci. Rep. 11 (2021) 18588.
[3] Ren Saifullah, Khalid Hussain Zhi, Muhammad Faheem, K-means online-learning routing protocol (K-MORP) for unmanned aerial vehicles (UAV) adhoc networks, Ad. Hoc. Netw. 154 (2024) 103354.
[4] Q. Sang, H. Wu, L. Xing, H. Ma, P. Xie, An Energy-Efficient Opportunistic Routing Protocol Based on Trajectory Prediction for FANETs, IEEe Access. 8 (2020) 192009–192020, https://doi.org/10.1109/ACCESS.2020.3032956.
[5] S. Ullah, K.H. Mohammadani, M.A. Khan, Z. Ren, R. Alkanhel, A. Muthanna, U. Tariq, Position-monitoring-based hybrid routing protocol for 3D UAV-based networks, Drones 6 (11) (2022) 327.
[6] L. Zhang, F. Hu, Z. Chu, E. Bentley, and S. Kumar, "3D transformative routing for UAV swarming networks: a skeleton-guided, GPS free approach," IEEe Trans. Veh. Technol., vol. 70,no. 4, pp. 3685.
[7] N. Zhao, Z. Ye, Y. Pei, Y.-C. Liang, D. Niyato, Multi-agent deep reinforcement learning for task offloading in UAV-assisted mobile edge computing, IEEe Trans. Wirel. Commun. 2022 (2021) 3701.
[8] B. Li, S. Zhao, R. Miao, R. Zhang, A survey on unmanned aerial vehicle relaying networks, IET Commun. 15 (10) (2021) 1262–1272.
[9] F. Zhou, R.Q. Hu, Z. Li, Y. Wang, Mobile Edge Computing in Unmanned Aerial Vehicle Networks, IEEe Wirel. Commun. 27 (1) (February 2020) 140–146, https://doi.org/10.1109/MWC.001.1800594.
[10] J. Jiang, G. Han, Routing protocols for unmanned aerial vehicles, IEEe Commun. Mag. 56 (2018) 58–63.
[11] H. Touati, A. Chriki, H. Snoussi, F. Kamoun, Cognitive Radio and Dynamic TDMA for efficient UAVs swarm communications, Comput. Netw. 196 (2021) 108264.
[12] C. Jiang, Z. Hu, Z.P. Mourelatos, D. Gorsich, P. Jayakumar, Y. Fu, M. Majcher, R2-RRT*: reliability-based robust mission planning of off-road autonomous ground vehicle under uncertain terrain environment, IEEe Trans. Autom. Sci. Eng. 19 (2021) 1030–1046.
[13] L. Hong, H. Guo, J. Liu, Y. Zhang, Toward swarm coordination: topology-aware inter-UAV routing optimization, IEEe Trans. Veh. Technol. 69 (2020) 10177–10187.
[14] Shanthy, R.; Padma, T. A zone routing protocol incorporated with sleep scheduling for MANETs. J. Ambient. Intell. Humaniz. Comput. 2021, 12, 4181–4191.
[15] B. Zheng, K. Zhuo, H. Zhang, and H.-X. Wu, "A novel airborne greedy geographic routing protocol for flying ad hoc networks," Wireless Networks, pp. 1–15, 2022.
[16] Y. Cui, Q. Zhang, Z. Feng, Z. Wei, C. Shi, H. Yang, Topology-aware resilient routing protocol for fanets: an adaptive Q-learning approach, IEEe Internet. Things. J. 9 (19) (2022) 18632–18649.
[17] Maryam Bavaghar, Amin Mohajer, Sarah Taghavi Motlagh, Energy efficient clustering algorithm for wireless sensor networks, J. Inform. Syst. Telecommun. (JIST) 4 (2020) 28, 238M. Y. Arafat and S. Moh, "A Q-learning-based topology-aware routing protocol for flying ad hoc networks," IEEE Internet of Things Journal, vol. 9, no. 3, pp. 1985–2000, 2021.

[18] Q. Wu, et al., Routing protocol for heterogeneous FANETs with mobility prediction, China Commun. 19 (1) (2022) 186–201, https://doi.org/10.23919/JCC.2022.01.014. Jan.

[19] Konstantinos Kanistras, Goncalo Martins, Matthew J. Rutherford, Kimon P. Valavanis, A survey of unmanned aerial vehicles (UAVs) for traffic monitoring, in: 2013 International Conference on Unmanned Aircraft Systems (ICUAS), IEEE, 2013, pp. 221–234.

[20] Jacob. Chakareski, UAV-IoT for next generation virtual reality, IEEE Trans. Image Process. 28 (12) (2019) 5977–5990.

[21] M.A. Khan, N. Kumar, S.A.H. Mohsan, W.U. Khan, M.M. Nasralla, M.H. Alsharif, J. Ywioek, Ullah, I. Swarm of UAVs for Network Management in 6G: a Technical Review, IEEE Trans. Netw. Serv. Manag. (2022).

[22] Awadhesh Dixit, Sunil Kumar Singh, BMUDF: hybrid Bio-inspired Model for fault-aware UAV routing using Destination-aware Fan-shaped clustering, Internet Things 22 (2023) 100790.

[23] Y. Liu, H.N. Dai, Q. Wang, M.K. Shukla, M. Imran, Unmanned aerial vehicle for internet of everything: opportunities and challenges, Comput. Commun. 155 (1) (2020) 66–83.

[24] Q. Sang, H. Wu, L. Xing, P. Xie, Review and comparison of emerging routing protocols in flying ad hoc networks, Symmetry. (Basel) 12 (6) (2020) 1–24, https://doi.org/10.3390/sym12060971.

[25] L. Hong, H. Guo, J. Liu, Y. Zhang, Toward swarm coordination: topology-aware inter-UAV routing optimization, IEEe Trans. Veh. Technol. 69 (9) (2020) 10177–10187.

[26] A. Khan, S. Khan, A.S. Fazal, Z. Zhang, A.O. Abuassba, Intelligent cluster routing scheme for flying ad hoc networks, Sci. China Inform. Sci. 64 (8) (2021) 182305.

[27] H. Ali, S.u. Islam, H. Song, K. Munir, A performance-aware routing mechanism for flying ad hoc networks, Trans. Emerg. Telecommun. Technol. 32 (1) (2021) e4192.

[28] L. Zhang, F. Hu, Z. Chu, E. Bentley, S. Kumar, 3D transformative routing for UAV swarming networks: a skeleton-guided, GPSfree approach, IEEe Trans. Veh. Technol. 70 (4) (2021) 3685–3701.

[29] Juhi Agrawal, Monit Kapoor, A comparative study on geographic-based routing algorithms for flying ad-hoc networks, Concurr. Comput.: Pract. Exper. 33 (16) (2021) e6253.

[30] Fatima Zohra Bousbaa, Chaker Abdelaziz Kerrache, Zohra Mahi, Abdou El Karim Tahari, Nasreddine Lagraa, Mohamed Bachir Yagoubi, GeoUAVs: a new geocast routing protocol for fleet of UAVs, Comput. Commun. 149 (2020) 259–269.

[31] A. Bujari, C.T. Calafate, J.-C. Cano, P. Manzoni, C.E. Palazzi, D. Ronzani, A Location-aware Waypoint-based Routing Protocol for Airborne DTNs in SAR, Sensors 18 (11) (Nov 2018) 3758.

[32] Kumar Debasis, Lakhan Dev Sharma, Vijay Bohat, Robin Singh Bhadoria, An Energy-Efficient Clustering Algorithm For Maximizing Lifetime of Wireless Sensor Networks Using Machine Learning, Mobile Networks and Applications, 2023, pp. 1–15.

[33] Govind P. Gupta, Binit Saha, Load balanced clustering scheme using hybrid metaheuristic technique for mobile sink based wireless sensor networks, J. Ambient. Intell. Humaniz. Comput. (2020) 1–12.

[34] Arun Kumar Sangaiah, Amir Javadpour, Forough Ja'fari, Weizhe Zhang, Shadi Mahmoodi Khaniabadi, Hierarchical clustering based on dendrogram in sustainable transportation systems, IEEE Trans. Intell. Transp. Syst. (2022).

[35] Ali Abbas, Bhawani Shankar Chowdhry, Muhammad Saqib, Vishal Dattana, Dynamic routing and coordination of cluster for unmanned aerial vehicle (UAV) swarms, Math. Probl. Eng. 2021 (2021) 1–11.

[36] Xiaohan Qiu, Shan Zhang, Zhiyuan Wang, Hongbin Luo, Integrated Host-and Content-Centric Routing for Efficient and Scalable Networking of UAV Swarm, IEEe Trans. Mob. Comput. (2023).

[37] A.H. Wheeb, R. Nordin, A. Samah, D. Kanellopoulos, Performance Evaluation of Standard and Modified OLSR Protocols for Uncoordinated UAV Ad-Hoc Networks in Search and Rescue Environments, Electronics. (Basel) 12 (6) (2023) 1–23.

[38] A.H. Wheeb, Flying Ad hoc Networks (FANET): performance Evaluation of Topology Based Routing Protocols, Int. J. Interact. Mob. Technol. 16 (4) (2022) 137–149.

[39] M.T. Naser, A.H. Wheeb, Implementation of RWP and Gauss Markov Mobility Model for Multi-UAV Networks in Search and Rescue Environment, iJIM 16 (23) (2022) 125–137.

[40] Xueyuan Wang, M.Cenk Gursoy, Tugba Erpek, Yalin E. Sagduyu, Learning-based UAV path planning for data collection with integrated collision avoidance, IEEe Internet. Things. J. 9 (17) (2022) 16663–16676.

[41] Abdul Mannan, Mohammad S. Obaidat, Khalid Mahmood, Aftab Ahmad, Rodina Ahmad, Classical versus reinforcement learning algorithms for unmanned aerial vehicle network communication and coverage path planning: a systematic literature review, Int. J. Commun. Syst. 36 (5) (2023) e5423.

[42] Babatunji Omoniwa, Boris Galkin, Ivana Dusparic, Communication-enabled deep reinforcement learning to optimise energy-efficiency in UAV-assisted networks, Vehic. Commun. 43 (2023) 100640.

[43] Hongpeng Wang, Shangyuan Song, Qianghui Guo, Dian Xu, Xiaoyang Zhang, Peizhao Wang, Cooperative motion planning for persistent 3d visual coverage with multiple quadrotor uavs, IEEE Trans. Automat. Sci. Eng. 21 (3) (2023) 3374–3383.

[44] Martin Jacquet, Max Kivits, Hemjyoti Das, Antonio Franchi, Motor-level N-MPC for cooperative active perception with multiple heterogeneous UAVs, IEEe Robot. Autom. Lett. 7 (2) (2022) 2063–2070.

[45] Wendi Sun, Mingrui Hao, A survey of cooperative path planning for multiple UAVs, in: International Conference on Autonomous Unmanned Systems, Springer Singapore, Singapore, 2021, pp. 189–196.

[46] Hye Jin Lee, Hyeon-Woo Na, PooGyeon Park, Graph Network Centralization via Asymmetric Edge Weight Allocation: laplacian Conditioning and Multi-UAV System Application, IEEe Trans. Netw. Sci. Eng. (2025).

[47] Ulrike. Von Luxburg, A tutorial on spectral clustering, Stat. Comput. 17 (4) (2007) 395–416.

[48] H.S. MANSOUR, M.H. MUTAR, I.A. AZIZ, et al., Cross-Layer and Energy-Aware AODV routing protocol for flying Ad-hoc networks[J], Sustain. MDPI 14 (15) (2022) 8980.

[49] D. DING, F. BU, Z. HOU, Study of Improved OLSR Routing Protocol in UAV, in: Swarm;[C]Proceedings of the Proceedings of the 7th International Conference on Cyber Security and Information Engineering, IEEE, 2022, p. 7.

[50] Y.Volkan Pehlivanoglu, Perihan Pehlivanoglu, An enhanced genetic algorithm for path planning of autonomous UAV in target coverage problems, Appl. Soft. Comput. 112 (2021) 107796.

[51] A. Raza, M.F. Khan, M. Maqsood, B. Haider, F. Aadil, Adaptive k-means clustering for Flying Ad-hoc Networks, KSII Trans. Internet Inf. Syst. 14 (6) (2020) 2670–2685.

[52] Saifullah, Z. Ren, K.H. Mohammadani and W. Riaz, "Optimal Game Routing for UAV Adhoc Networks in Smart City," 2023 6th World Conference on Computing and Communication Technologies (WCCCT), Chengdu, China, 2023, pp. 34–38.

[53] M.F. Khan, I.N.D.R.A.N.I. Das, Critical analysis of modified gauss markov mobility model using varying values of parameters to check the impact of QoS In MANET, J. Eng. Sci. Technol. 17 (5) (2022) 3393–3409. Available online, https://www.nsnam.org/releases/ns-3-35/ (accessed on 14 August 2024).