

SafeLearning: Mobility-Aware Client Selection in Federated Edge Learning over UAV Networks

Ahmed Chaouch Ilies Akram
Department of Computer Science
Université Amar Telidji
Laghouat, Algeria
i.ahmedchaouch.mi@lagh-univ.dz

Mohammed Khaled Ali Oudenani
Department of Computer Science
Université Amar Telidji
Laghouat, Algeria
m.oudenani.mi@lagh-univ.dz

Youssra Cheriguene
Department of Computer Science
Université Amar Telidji
Laghouat, Algeria
y.cheriguene@lagh-univ.dz

Mohamed Chahine Ghanem
Cybersecurity Institute & Cyber Security Research Centre
University of Liverpool & London Metropolitan University
Liverpool & London, UK
mohamed.chahine.ghanem@liverpool.ac.uk

Abstract—One promising way to enable distributed intelligence at the edge while protecting data privacy is to integrate Federated Learning (FL) with Unmanned Aerial Vehicles (UAV) networks. Using FL enables each UAV to cooperatively train a global model without sharing raw data, especially in UAV swarms used for surveillance, monitoring, or emergency response missions. But choosing the best clients (UAVs) for every training cycle is made extremely difficult by the dynamic and diverse character of UAV environments. These difficulties are brought on by things like fluctuating connectivity, shifting patterns of movement, and energy limitations. In this work, we investigate the problem of client selection for Federated Edge Learning in UAV networks. We first present a taxonomy of existing selection strategies, considering criteria such as model performance and UAV mobility. Then, we propose SafeLearning, an adaptive client selection framework that integrates both mobility-awareness and distance with speed to enhance learning efficiency and model accuracy. Simulations demonstrate that our method significantly improves convergence speed and reduces client dropout, while maintaining high model performance in dynamic UAV scenarios.

Index Terms—FL, UAV Networks, Client Selection, Mobility-Aware Selection, Edge Intelligence.

I. INTRODUCTION

Multi-access Edge Computing (MEC) supported by Unmanned Aerial Vehicles (UAVs) has gained significant attention as a flexible and scalable solution for meeting the needs of latency-sensitive and computation-intensive applications, particularly in remote, rural, or disaster-stricken environments [1]. UAVs provide unique advantages such as agile 3D mobility, line-of-sight communication, rapid deployment, and improved onboard computing capabilities. These features enable UAVs to serve as mobile edge nodes that can offer on-demand computation to ground users or act as data collectors in large-scale sensing and monitoring tasks [2], [3].

With the increasing volume and sensitivity of collected data, Federated Learning (FL) has emerged as a privacy-preserving alternative to centralized learning by enabling decentralized model training directly on UAVs [4], [5]. However, most existing FL implementations in UAV networks apply simplistic

client selection techniques, typically random sampling, that fail to account for the dynamic nature of UAVs. Challenges such as frequent topology changes, mobility-induced disconnections, limited battery life, and fluctuating link quality contribute to unstable participation and high dropout rates, ultimately degrading training performance [6], [7].

Motivated by these limitations, this work focuses on improving the reliability and efficiency of FL in UAV-assisted MEC systems by introducing mobility-aware client selection strategies. The key contributions include:

- The development of a novel client selection mechanism that leverages UAV-specific metrics such as speed, displacement, and positional consistency to predict and minimize dropout events.
- Demonstrating through empirical evaluation that the proposed method reduces both the average and variance of client dropout ratios.
- Showing that this improved stability leads to faster model convergence and enhanced global accuracy, thereby improving the overall efficiency and robustness of federated learning in UAV networks.

The remainder of this paper is organized as follows. In the next section we provide an overview of related works on this topic, discussing the client selection strategies. Section III details our contribution in this paper. Then, section IV presents the main findings of this paper. Finally, in section VI, the paper is finalized with a conclusion, along with references to future work.

II. RELATED WORK

Client selection plays a crucial role in enhancing the efficiency, robustness, and fairness of FL systems. A wide range of strategies has been proposed in the literature to address various challenges such as data heterogeneity, communication overhead, limited resources, security risks, and client mobility especially in dynamic environments like UAV networks or

vehicular systems. There are many client selection methods adapted in the literature, which can be broadly classified into four main categories: (i) Data-based selection, (ii) Resource-based selection, (iii) Security-based selection, and (iv) Mobility-based selection. The following summarizes key selection strategies, along with their benefits and limitations.

A. Data-Based Client Selection

Data-based approaches prioritize clients based on the diversity, quality, or representativeness of their local datasets. In UAV networks, this helps improve global model convergence and generalization [8]. By selecting UAVs with complementary data or those aligned with training goals, unnecessary redundancy is reduced [9], [10]. However, these methods often rely on UAVs transmitting metadata such as data size or distribution, introducing communication overhead and energy consumption [11]. Additionally, data relevance may change rapidly due to UAV mobility, making consistent selection challenging [11].

B. Security-Based Client Selection

To address security vulnerabilities in FL, such as label-flipping attacks or poisoned updates, researchers have proposed selection strategies that exclude unreliable or adversarial clients [12]. These strategies improve model robustness by prioritizing clients with high reliability scores and incorporating secure mechanisms like blockchain and homomorphic encryption [13]. While this improves model integrity and privacy, it also reduces the client pool, potentially excluding UAVs with valuable data but weaker security features, thereby limiting model generalization.

C. Resource-Based Client Selection

Resource-aware strategies select clients based on their computational capacity, energy availability, and network bandwidth. In UAV settings, these approaches prevent system overloads and reduce dropout rates caused by limited resources [14]. Energy-efficient techniques like model compression and workload adaptation further support low-resource UAVs [15]. Despite improved efficiency, these strategies may exclude resource-limited but data-rich UAVs, introducing bias. Additionally, real-time monitoring of device status adds overhead to the selection process [16], [17].

D. Mobility-Based Client Selection

Mobility-aware selection leverages the dynamic movement of UAVs to improve data diversity and model generalization. Cluster-based FL approaches, such as CFL, group mobile clients with similar patterns for local aggregation, reducing communication loads and improving accuracy in non-IID scenarios [18]. Hierarchical FL methods have shown that mobility can enhance convergence by promoting data mixing across edge servers [19]. However, client scheduling in such environments is complex, and most existing models are limited to 2D vehicular mobility or fixed UAV trajectories, limiting their effectiveness in real-world UAV networks. Systems like

the one proposed in [9] introduce edge-assisted FL frameworks to mitigate issues such as label noise and data heterogeneity through quality-aware aggregation and data augmentation. While these models perform well in controlled conditions, they often overlook critical factors such as UAV mobility, node dropout, and real-time communication disruptions, which limits their adaptability in dynamic UAV environments.

E. Discussion and Limitations

Despite the progress made, a gap remains in effectively addressing the combined impact of UAV mobility and energy constraints on FL model convergence and reliability. Most existing works neglect the variability of UAV speed and its influence on training performance, as well as the energy-efficiency challenges unique to UAVs. To address these limitations, the next section introduces a novel client selection approach designed specifically for UAV-assisted MEC systems that accounts for both mobility dynamics and energy constraints.

III. SAFELEARNING: MOBILITY-AWARE CLIENT SELECTION STRATEGY

To address the challenges posed by the dynamic nature of UAV networks, we propose a mobility-aware client selection algorithm that strategically selects a subset of UAVs for each communication or training round. This method prioritizes UAVs that exhibit favorable mobility characteristics namely, stability, reachability, and communication reliability ensuring their sustained participation in latency-sensitive and collaborative FL tasks. SafeLearning client selection process is described in the following sub-sections. Moreover, Algorithm .2 and Algorithm .1 summarize our contribution.

1) *Inputs and Output*: Our proposed client selection framework operates based on three key inputs: the total number of UAVs in the network V_n , the total number of communication or training rounds R_n , and a selection fraction f_p , which specifies the proportion of UAVs to be selected in each round. Using these inputs, the algorithm generates a single output, which is a structured list that records the subset of UAVs chosen for each round based on their mobility characteristics.

2) *Utility Calculation Process*: In each communication or training round, the algorithm initiates the selection process by resetting a list designated to store the utility scores of candidate UAVs. It then iterates over the set of available UAVs, evaluating each one based on a combination of eligibility criteria and performance indicators. Specifically, a UAV must satisfy the following minimum conditions: (1) its battery level must exceed a predefined threshold $\beta_{\text{threshold}}$ (e.g. 15%) to ensure task completion without interruption, and (2) it must lie within the maximum allowable communication range from the base station to guarantee reliable connectivity. Once these prerequisites are met, the UAV's mobility behavior is assessed through a set of dynamic features, including its current distance from a reference center, instantaneous speed, change in distance since the previous round, and variation in speed. These mobility attributes are then fed into a scoring

function that computes a composite mobility score. Only UAVs that receive a valid (non-null or above-threshold) score are included in the candidate pool for potential selection in the given round.

We begin by defining the Euclidean distance function, which measures spatial separation between two points:

$$\text{EuclideanDistance}(x, y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + (x_3 - y_3)^2} \quad (1)$$

where $x = (x_1, x_2, x_3)$ and $y = (y_1, y_2, y_3)$ denote the 3D coordinates of two entities typically a UAV and the base station, where each component corresponds to the x , y , and z axes, respectively.

Using this function, we calculate the current distance between UAV v at time t and the base station as:

$$\text{dist}_v^{(t)} = \text{EuclideanDistance}(v^{(t)}.coords, \text{BS}.coords), \quad (2)$$

where $v^{(t)}.coords$ denotes the coordinates of UAV v in round t , and $\text{BS}.coords$ denotes the position of the base station.

To quantify the positional change of UAV v between the current and previous rounds, we compute the displacement:

$$\Delta d_v^{(t)} = \text{EuclideanDistance}(v^{(t)}.coords, v^{(t-1)}.coords). \quad (3)$$

The variation in speed between rounds t and $t - 1$ is computed as:

$$\Delta v_v^{(t)} = |v_{\text{speed}}^{(t)} - v_{\text{speed}}^{(t-1)}|. \quad (4)$$

A UAV is only considered further if it satisfies the range constraint:

$$\text{dist}_v^{(t)} < \text{max_dist}. \quad (5)$$

We then compute a set of normalized feature-specific scores to assess UAV suitability.

The proximity score favors UAVs closer to the base station:

$$\text{score}_{\text{dist}}^{(t)} = 1 - \frac{\text{dist}_v^{(t)}}{\text{max_dist}} \quad (6)$$

The speed score rewards UAVs with lower velocities:

$$\text{score}_{\text{speed}}^{(t)} = 1 - \frac{v_{\text{speed}}^{(t)}}{\text{max_speed}}. \quad (7)$$

The displacement score captures spatial stability:

$$\text{score}_{\Delta d}^{(t)} = 1 - \frac{\Delta d_v^{(t)}}{\text{max}(\Delta d)}. \quad (8)$$

The speed variation score measures temporal consistency:

$$\text{score}_{\Delta v}^{(t)} = 1 - \frac{\Delta v_v^{(t)}}{\text{max}(\Delta v)}. \quad (9)$$

Finally, the overall utility score is a weighted combination of the individual feature scores:

$$\text{final_score}_v^{(t)} = a \cdot \text{score}_{\text{dist}}^{(t)} + b \cdot \text{score}_{\text{speed}}^{(t)} + c \cdot \text{score}_{\Delta d}^{(t)} + d \cdot \text{score}_{\Delta v}^{(t)}. \quad (10)$$

Here, a , b , c , and d are scalar weights (typically between 0 and 1) indicating the relative importance of each feature.

3) Scoring and Ranking UAVs: UAVs that remain close to the base station, exhibit lower speeds, demonstrate minimal displacement, and maintain consistent speed over time are assigned higher scores. These characteristics are indicative of UAVs that are reliable. By assigning greater weight to such mobility attributes, the scoring function ensures that UAVs with the most desirable behavioral patterns are prioritized.

Following the scoring process, all UAVs are ranked in descending order according to their final scores. A fraction f_p of the highest-ranked UAVs representing the target selection rate for each round is then selected. These UAVs are added to the selected UAVs list, which stores the identifiers of UAVs selected in each round.

Upon completion of all communication or training rounds, the algorithm returns the selected UAVs list as its output. This list serves as a complete schedule of UAV participation across rounds, based on the calculated trade-offs between mobility, availability, and energy constraints.

Algorithm .1: SafeLearning Client Selection Strategy

Input: Number of UAVs V_n , rounds R_n , selection factor f_p

Output: Selected UAVs per round:

```

    selected_UAVs_per_round
1  selected_UAVs_per_round ← [] // Storage
   for selections
2  foreach round  $r \in R_n$  do
3       $U \leftarrow \emptyset$  // Reset utility list
4      foreach UAV  $v \in V_n$  do
5          if  $v_{\text{battery}} > \beta_{\text{threshold}}$  then
6               $\text{dist} \leftarrow \text{EuclideanDistance}(v.coords, \text{base\_station}.coords)$ 
7               $\Delta d \leftarrow \text{EuclideanDistance}(v.coords, v.previous\_coords)$ 
8               $\Delta v \leftarrow |v.speed - v.previous\_speed|$ 
9               $\text{score} \leftarrow \text{ComputeMobilityScore}(\text{dist}, \text{MAX\_COMMUNICATION\_RANGE},$ 
10                  $v.speed, \text{MAX\_SPEED}, \Delta d, \text{MAX\_}\Delta d, \Delta v, \text{MAX\_}\Delta v)$ 
11              if  $\text{score} \neq \text{None}$  then
12                   $U \leftarrow U \cup \{(\text{score}, v)\}$ 
13          end if
14      end foreach
15       $U_{\text{sorted}} \leftarrow \text{Sort } U \text{ by score descending}$ 
16       $\text{num} \leftarrow \max(0, \lfloor f_p \times |V_n| \rfloor)$ 
17       $\text{selected\_UAVs} \leftarrow \text{First num UAVs in } U_{\text{sorted}}$ 
18       $\text{selected\_UAVs\_per\_round} \leftarrow \text{selected\_UAVs\_per\_round} \cup \{\text{selected\_UAVs}\}$ 
19  return selected_UAVs_per_round

```

Algorithm .2: Computation of UAV Mobility Score

```
1 Function ComputeMobilityScore (dist, max_dist,  
   speed, max_speed,  $\Delta d$ , max_Δd,  $\Delta v$ , max_Δv):  
2   if dist > max_dist then  
3     return 0  
4   score_dist  $\leftarrow 1 - \frac{\text{dist}}{\text{max\_dist}}$   
5   score_speed  $\leftarrow 1 - \frac{\text{speed}}{\text{max\_speed}}$   
6   score_delta_d  $\leftarrow 1 - \frac{|\Delta d|}{\text{max\_}\Delta d}$   
7   score_delta_v  $\leftarrow 1 - \frac{|\Delta v|}{\text{max\_}\Delta v}$   
8   score  $\leftarrow a \cdot \text{score\_dist} + b \cdot \text{score\_speed} + c \cdot$   
   score_delta_d + d · score_delta_v  
9   return score
```

IV. EXPERIMENTAL RESULTS AND DISCUSSION

A. Experimental Setup

The FL system consists of a single base station that integrates edge computing capabilities and 20 UAVs operating within its coverage area. Each UAV is assumed to have a fixed processing frequency of 2.5 GHz and a maximum communication range of 250 meters. Each UAV executes one epoch of training per communication round using a batch size of 32 samples. Data heterogeneity is explicitly incorporated by enforcing a non-IID distribution across the UAVs, and client selection is limited to 10% of the total population per round.

For model optimization, we use SGD with a learning rate of 0.001. Each local update consists of a single iteration per round. The model size transmitted by each UAV is 1 MB, with each parameter occupying 32 bytes. Communication is modeled with an uplink bandwidth of 1 MHz and a fixed UAV transmission power of 0.1 Watts. The channel environment includes background noise, modeled with a noise power of 10^{-9} Watts, consistent with typical wireless settings. To reflect realistic deployment scenarios where clients often possess non-uniform data, we simulate a non- IID setting using a $k\%$ random sampling strategy. In our configuration, 70% of the dataset is randomly assigned across the participating UAVs, while the remaining 30% is distributed in a biased fashion, resulting in highly imbalanced and skewed local datasets.

In terms of coordination overhead, model aggregation at the BS is assumed to take 0.5 seconds per round. Additionally, UAV mobility is dynamically tracked and updated at 0.1 second intervals throughout the training process.

To evaluate the performance and robustness of the framework, simulations are conducted over 100 FL rounds. During this process, we test the system's resilience under various network dynamics, including fluctuating connectivity, diverse data distributions, and communication delays. The utility function used to score UAVs is parameterized with weights set to $a = 0.3$, $b = 0.3$, $c = 0.2$, and $d = 0.2$, assigning relative importance to proximity, speed, displacement, and speed variation, respectively.

B. Benchmarks

To contextualize the effectiveness of SafeLearning, we compare it against two baseline methods: random selection and

distance-based filtering. Each method represents a different strategy for selecting clients to participate in FL rounds. In order to highlight the difference between them, we define the benchmarks as follows:

- **Random Selection Strategy** In the random selection approach, UAVs are chosen arbitrarily without considering their location, speed, or movement stability. The advantages of this method are that it is easy to understand and implement and does not take into account any element or value, but the disadvantages are that it does not choose the best drones to participate in the rounds.
- **Distance-based Filtering Strategy** This strategy selects UAVs that are geographically closest to the edge server. By reducing communication delays and the risk of disconnections, this method improves performance compared to random selection. It takes into account the distance between the center and each UAV participating in the training.

C. Evaluation Metrics

To evaluate the performance and robustness of the proposed client selection strategy in UAV-based FL, we use two primary categories of metrics: model performance and client participation stability.

The first metric assesses the global model performance, while the second group captures the dropout behavior of UAV clients across training rounds.

- **Global Test Accuracy:**

This metric reflects the classification accuracy of the aggregated global model after each FL round. It is evaluated on a centralized test set and indicates how well the model generalizes after incorporating updates from selected UAVs. Higher accuracy values correspond to more effective client contributions and improved learning convergence.

- **Mean Dropout Ratio:**

The mean dropout ratio in FL represents the average percentage of UAVs that fail to complete local training and upload their model updates during each round. A lower mean dropout ratio implies higher reliability and faster convergence of the global model. It is computed as:

$$\text{Mean Dropout Ratio} = \frac{\sum_{i=1}^N D_i}{N},$$

where D_i is the number of dropout events in the i -th round and N is the total number of UAVs.

- **Standard Deviation of Dropout Ratio:**

This metric quantifies the variability in UAV participation across training rounds. A low standard deviation implies stable client availability, minimizing disruption to model updates. It is calculated using:

$$\text{Standard Deviation} = \sqrt{\frac{\sum_{i=1}^N (D_i - \mu)^2}{N}},$$

where μ is the mean dropout ratio. We note that a low mean and standard deviation in dropout ratios indicate an efficient client selection strategy.

V. ANALYSIS OF THE OBTAINED RESULTS

In this section, we analyze the performance of the proposed client selection strategy by examining its effect on model accuracy and dropout behavior under two distinct mobility scenarios. The primary objective is to evaluate how varying UAV speeds impact the global model's convergence rate and system stability.

To this end, we define two experimental scenarios that differ in terms of UAV mobility patterns:

- **Scenario 1: Low-Speed Mobility** In contrast, this configuration assumes UAV speeds between 2 and 15 meters per second, reflecting slower, more stable flight conditions typical of persistent monitoring or coordinated sensing operations.
- **Scenario 2: High-Speed Mobility** In this setting, UAVs move at speeds ranging from 30 to 40 meters per second. This scenario simulates highly dynamic environments.

The graphs (Fig. 3) and (Fig. 1) compares three different strategies for selecting drones to participate in each round of FL. The goal is to evaluate the impact of each selection strategy on the model's accuracy over 100 training rounds. As for Figures (Fig. 4) and (Fig. 2) for the bar chart, it contains 3 parts. The first part is for the average dropout for each method, while for Part 2 it represents the standard deviation values, while Part 3 represents the total number of dropout (number of UAVs).

A. Scenario 1:

1) *Random Selection Strategy*: From the curve (Fig. 1) we notice that when the speed decreases, the acceleration of accuracy increases and becomes better , and the final value is around 95% . As for the dropout values, we notice from Fig. 2 that they are a total of 8 out of 200, meaning that by selecting 200 drones to participate in the FL tasks, there are 8 drones that were out of range, as for the mean dropout (AVG), the value was 0.04 . This indicates that random selection is the worst of the three methods.

2) *Distance-based Filtering Strategy*:: From the curve (1) we notice a slight advantage in acceleration for the distance filter, and it reaches 80% around round 38, and the final value is 95.14%, which is a little better than random selection. As for the dropout values (2), they are a total of 7 out of 200, and for the mean dropout AVG, the value was 0.035; this is a little better than random and contributes to better accuracy values. However, since it does not account for the drones' motion dynamics (e.g., velocity or trajectory changes), it may still select unstable UAVs, resulting in fluctuations in model accuracy during later rounds.

3) *SafeLearning Strategy*: From (1), we notice that the acceleration and final accuracy value have become clearly better than other methods, as the final value is 95.63%. As for the dropout, from (2), we notice that it has become 4

out of 200, and for the mean dropout AVG, The value was 0.02, which is much better than the other strategies, and this contributed to the rapid acceleration of the accuracy value.

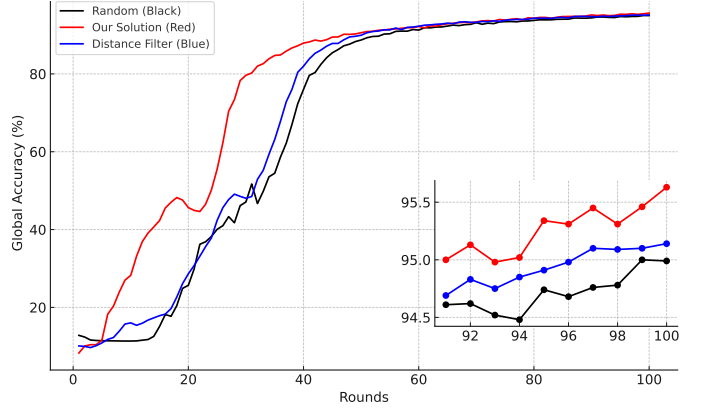


Fig. 1. Global test accuracy comparison of participant selection methods vs. number of FL rounds in Scenario 1.

B. Scenario 2:

1) *Random Selection Strategy* : As the graph shows in Fig. 3, this method leads to the slowest convergence and the lowest overall performance. Accuracy remains below 60% until about round 38 and shows significant fluctuation even in later rounds , It reaches 80% only in round 50 , and this is worse than the first scenario . As for the dropout values, we notice from Fig. 4 that they are a total of 13 out of 200, meaning that by selecting 200 drones to participate in the FL tasks, there are 13 drones that were out of range, as for the mean dropout (AVG), the value was 0.065 , and this is a large numbers compared to SafeLearning and the distance-filter, and this contributes to reducing the accuracy value.

2) *Distance-based Filtering Strategy* :: From the curve (3) we notice the value of accuracy reaches 80% around round 35 , which is much better than random selection, and continues to climb steadily. As for the dropout values (4) that they are a total of 8 out of 200, and for the mean dropout AVG, the value was 0.04, and this is much better than random and contributes to better accuracy values. But these values remain worse than the first scenario.

3) *SafeLearning Strategy* :: From the curve (3), we notice that this strategy clearly outperforms the others. It achieves over 80% accuracy by round 27 and continues to improve rapidly. By the final round, it ended in 95.50% accuracy. This indicates that our method maintains its efficiency even at higher speeds. As for the dropout values (4), they are a total of 4 out of 200, and for the mean dropout (AVG), the value was 0.02, therefore, outperforming both random and distance-filter. This demonstrates the effectiveness of selecting UAVs that are not only near the server but also predictable and reliable in their mobility.

Scenario 1 VS Scenario 2 :

We observe that as the UAV speed increases, the performance of benchmark methods such as random selection

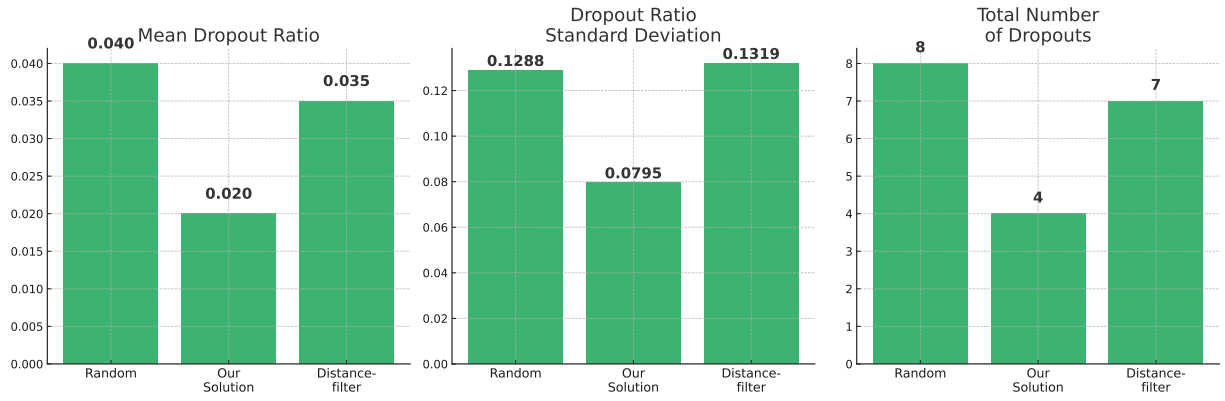


Fig. 2. Dropout Metrics Across Selection Strategies in Scenario 1.

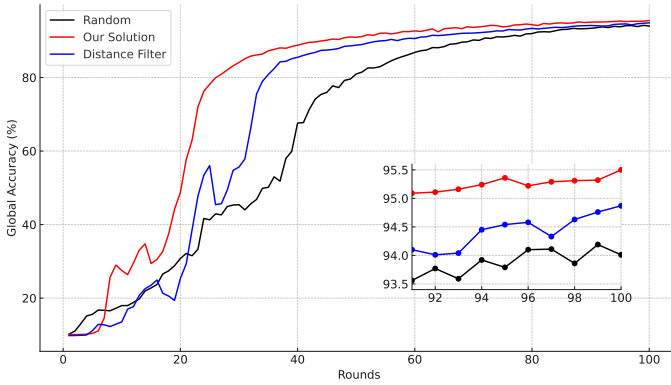


Fig. 3. Global test accuracy comparison of participant selection methods vs. number of FL rounds in Scenario 2.

and distance-based filtering significantly degrades, exhibiting higher dropout rates and greater variability. In contrast, our proposed solution maintains robust performance under high-mobility conditions. This resilience stems from the fact that, unlike the distance-based approach which relies solely on proximity, our method incorporates multiple mobility features including speed and its variation into the client selection process. As a result, it consistently identifies UAVs that are not only within communication range but also exhibit stable and predictable mobility patterns, thereby ensuring more reliable participation and efficient learning convergence.

VI. CONCLUSION AND FUTURE WORK

In this paper, we proposed SafeLearning, a mobility-aware client selection strategy tailored for federated learning (FL) in UAV-assisted edge computing environments. Unlike traditional approaches that rely on random sampling or static metrics such as distance, SafeLearning integrates multiple UAV mobility attributes—namely speed, displacement, positional consistency, and proximity to the base station—to identify the most reliable clients for each training round. This multidimensional scoring approach allows the system to dynamically adapt to the fluctuating conditions inherent in UAV networks, ensuring more stable participation and improving learning efficiency.

Our simulation results under both low-speed and high-speed UAV scenarios clearly demonstrate the superiority of SafeLearning compared to benchmark strategies. Specifically, our approach achieved faster convergence, higher final accuracy (up to 95.63%), and significantly reduced client dropout rates. These results validate the importance of mobility-aware client selection and highlight the role of behavioral patterns in improving the robustness and scalability of FL in highly dynamic environments. Moreover, by maintaining stability across varying mobility conditions, SafeLearning proves to be well-suited for real-world UAV applications such as disaster response, aerial surveillance, and remote sensing.

The impact of this work extends beyond UAV systems, providing a generalizable framework for intelligent client coordination in mobile edge networks. It lays the groundwork for future research into context-aware federated learning mechanisms that can adapt to both network conditions and client behaviors. The SafeLearning model offers a step forward in bridging the gap between theoretical FL frameworks and their practical deployment in mission-critical, latency-sensitive scenarios.

As future work, we plan to enhance SafeLearning by incorporating data utility metrics—such as label diversity and data quality—into the scoring function to better support non-IID data environments. We also aim to extend the framework with energy-awareness to dynamically balance mobility and energy constraints, especially critical for power-limited UAVs. Integrating trust and anomaly detection mechanisms will strengthen security against adversarial clients, while real-world testing using UAV testbeds and sim2real transfer will ensure practical viability. Finally, adopting a multi-objective optimization approach will help harmonize trade-offs between model accuracy, resource consumption, fairness, and stability.

REFERENCES

- [1] H. Kheddar, Y. Habchi, M. C. Ghanem, M. Hemis, and D. Niyato, “Recent advances in transformer and large language models for uav applications,” *arXiv preprint arXiv:2508.11834*, 2025.

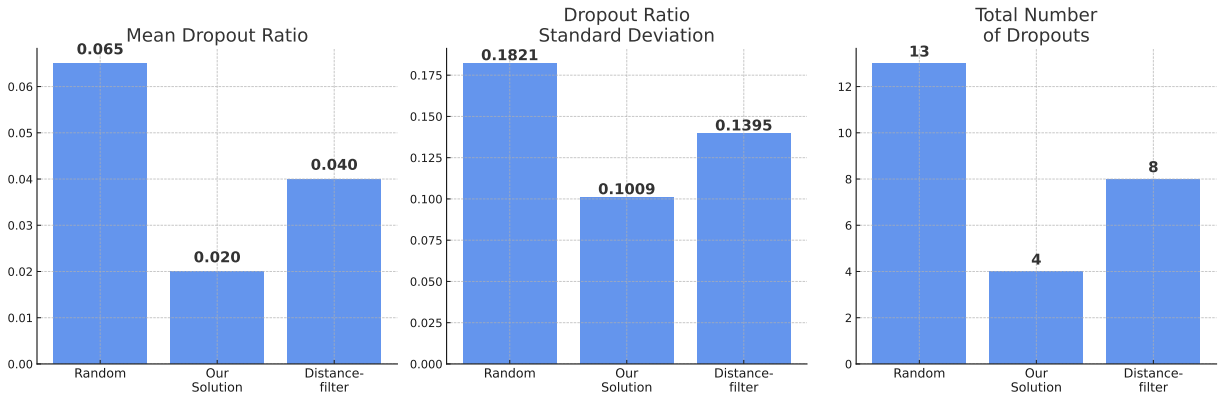


Fig. 4. Dropout Metrics Across Selection Strategies in Scenario 2.

- [2] Y. Gao, Z. Wang, Y. Zhang, W. Lu, J. Tang, N. Zhao, and F. Gao, "Multi-IRS-aided secure communication in UAV-MEC networks," *IEEE Transactions on Vehicular Technology*, vol. 74, no. 5, pp. 7327–7338, 2025.
- [3] A. M. Hadjkouider, C. A. Kerrache, Y. Sahraoui, A. Adnane, and C. T. Calafate, "A machine learning-assisted game theoretical service selection strategy in uav networks," in *Global Congress on Emerging Technologies (GCET-2024)*. IEEE, 2024, pp. 58–64.
- [4] A. S. S. Al Farsi, A. Khan, M. R. Mughal, and M. M. Bait-Suwailam, "Privacy and security challenges in federated learning for UAV systems: A systematic review," *IEEE Access*, vol. 13, pp. 86 599–86 615, 2025.
- [5] G. Rathee, C. A. Kerrache, and C. T. Calafate, "A sustainable and trusted solution for iot-based federated learning using feedback behavior," *Internet of Things*, vol. 26, p. 101177, 2024.
- [6] Y. Cheriguene, W. Jaafar, H. Yanikomeroglu, and C. A. Kerrache, "Towards reliable participation in uav-enabled federated edge learning on non-iid data," *IEEE Open Journal of Vehicular Technology*, vol. 5, pp. 125–141, 2023.
- [7] S. Dahmane, M. B. Yagoubi, B. Brik, C. A. Kerrache, C. T. Calafate, and P. Lorenz, "Multi-constrained and edge-enabled selection of uav participants in federated learning process," *Electronics*, vol. 11, no. 14, p. 2119, 2022.
- [8] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings *et al.*, "Advances and open problems in federated learning," *Foundations and Trends in Machine Learning*, vol. 14, no. 1–2, pp. 1–210, 2021.
- [9] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civan, and V. Chandra, "Federated learning with non-iid data," *arXiv preprint arXiv:1806.00582*, 2018.
- [10] T. Nishio and R. Yonetani, "Client selection for federated learning with heterogeneous resources in mobile edge," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. IEEE, 2019, pp. 1–7.
- [11] C. Meng, Y. Li, H. Wang, and W. Zhang, "Federated learning in uav-enabled networks: Applications, challenges, and opportunities," *IEEE Communications Magazine*, vol. 60, no. 4, pp. 62–68, 2022.
- [12] S. Yu, J. Shen, S. Xu, J. Wang, Z. Wang, and Q. Xuan, "Label-flipping attacks in gnn-based federated learning," *IEEE Transactions on Network Science and Engineering*, pp. 1–12, 2025.
- [13] R. Shen, H. Zhang, B. Chai, W. Wang, G. Wang, B. Yan, and J. Yu, "BAFL-SVM: A blockchain-assisted federated learning-driven svm framework for smart agriculture," *High-Confidence Computing*, vol. 5, no. 1, p. 100243, 2025.
- [14] Y. Zhao, M. Li, L. Lai, and H. V. Poor, "Energy-efficient federated learning for edge intelligence in 6g networks," *IEEE Internet of Things Journal*, vol. 8, no. 11, pp. 8852–8862, 2021.
- [15] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, and K. Chan, "Adaptive federated learning in resource-constrained edge computing systems," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 6, pp. 1246–1259, 2020.
- [16] H. Kheddar, Y. Habchi, M. C. Ghanem, M. Hemis, and D. Niyato, *arXiv preprint arXiv:2508.11834*, 2025.
- [17] J. Wang, X. Liu, and J. Liang, "Optimizing federated learning in resource-constrained environments," *IEEE Transactions on Neural Networks*, vol. 31, no. 11, pp. 2003–2015, 2020.
- [18] A. Taïk, Z. Mlika, and S. Cherkaoui, "Clustered vehicular federated learning: Process and optimization," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 25 371–25 383, 2022.
- [19] T. Chen, J. Yan, Y. Sun, S. Zhou, D. Gündüz, and Z. Niu, "Mobility accelerates learning: Convergence analysis on hierarchical federated learning in vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 74, no. 1, pp. 1657–1673, 2025.