ATLAS-HFL: LEARNING-AWARE ANCHOR GRAPHS FOR CLOSED-LOOP FEDERATED LEARNING IN DYNAMIC UAV SWARMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Dynamic UAV swarms must support federated task learning under changing topology, limited resources, heterogeneous data, and intermittent cluster-head failures. Existing UAV clustering methods usually treat the cluster structure as a communication route or aggregation scaffold. However, in dynamic UAV federated learning, a useful organization structure should encode not only physical feasibility and resource sustainability, but also task relations that guide future model cooperation. The observations suggest that swarm organization should be maintained as an evolving learning-aware state rather than a one-time clustering result. Following this reasoning, a hierarchical federated learning framework called ATLAS-HFL is developed through a sparse anchor graph. ATLAS-HFL represents UAVs with physical states, resource states, and compact local learning states without constructing a full pairwise UAV graph. The learned UAV-to-anchor associations determine dynamic clusters, support primary-backup cluster-head election, and produce task-aware inter-cluster cooperation weights for cluster-specific initialization in the next control period. Updated local training states then refine the anchor graph, forming a closed loop between federated learning and swarm organization. Experiments across heterogeneous federated learning settings, cluster-head failure scenarios, scalability tests, and network simulations show that ATLAS-HFL improves task performance, preserves service availability under cluster-head failures, and enhances communication reliability while maintaining comparable network lifetime.

1 INTRODUCTION

UAV swarms support environmental monitoring, disaster response, and public safety. Their ad hoc networks provide wide sensing coverage, but mobility, intermittent links, limited energy, and uneven computing resources make stable collaboration difficult (Mrad et al., 2021; Bekmezci et al., 2013; Alam et al., 2022). Federated learning allows each UAV to train on local mission data without uploading raw samples (McMahan et al., 2017). However, different sensing regions also produce non-IID data. Dynamic UAV federated learning must therefore organize communication and learning at the same time.

Clustered and hierarchical federated learning reduce communication load and limit harmful aggregation across unrelated clients (Sattler et al., 2021; Hoang et al., 2023). UAV methods also combine data similarity with position, energy, and routing states (Lan et al., 2025). However, the cluster structure usually serves as an aggregation route or scheduling result. It does not remain an explicit learning state that controls knowledge transfer in the next control period. Physical-only clustering can group UAVs with different task distributions. Learning-only clustering can produce unstable links or select resource-poor CHs. A useful cluster structure must reflect both constraints and must affect later model training. Figure 1 summarizes the motivation. Physical/resource-based clustering and learning-only clustering both lead to one-way organization. ATLAS-HFL instead couples swarm organization with later task learning.

ATLAS-HFL introduces a sparse multi-view anchor graph for this purpose. Each UAV produces a compact learning summary from its local loss, gradient representation, feature prototype, and

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

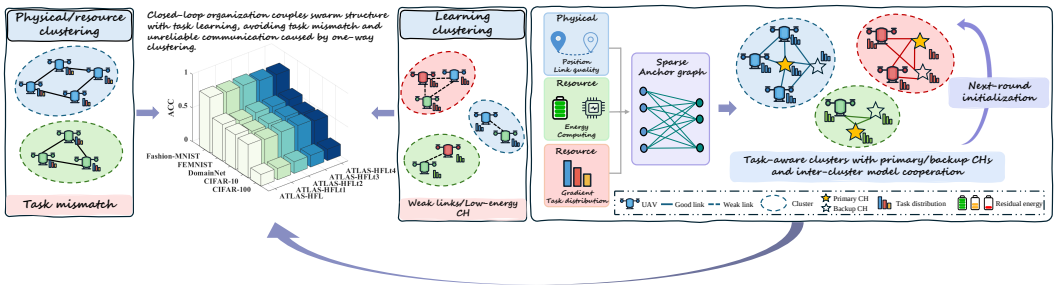


Figure 1: **Motivation of ATLAS-HFL.** ATLAS-HFL couples task learning with UAV organization. Physical/resource-based clustering may cause task mismatch, while learning-only clustering may cause unreliable links or low-energy CHs. ATLAS-HFL uses a sparse anchor graph to update task-aware clusters, primary-backup CHs, and next-round model cooperation.

model update. The summary is fused with physical and resource states. A small set of anchors then represents the joint swarm structure. The UAV-to-anchor associations determine dynamic clusters, and anchor representativeness supports primary-backup CH election. More importantly, task similarity between anchors determines inter-cluster cooperation weights. The weights produce cluster-specific initial models for the next period. Updated local states then refine the graph. The mechanism forms a closed loop between task learning and swarm organization.

The main contributions are:

- A learning-aware UAV organization model is introduced. It represents physical, resource, and private local training states with a sparse multi-view anchor graph and avoids a full pairwise UAV graph.
- The learned structure jointly controls clustering, primary-backup CH election, and task-aware model initialization. The last component turns clustering from a routing result into an input to the next federated training period.
- Experiments evaluate task accuracy, statistical heterogeneity, component effects, CH failure, scalability, and NS-3 network performance. The results also identify the IID setting as a failure boundary, in which the single global model of FedAvg remains better and the gain of learning-aware organization disappears.

2 Related Work

Federated learning enables distributed model training without exchanging raw task data, with FedAvg serving as a standard baseline for communication-efficient aggregation (McMahan et al., 2017). In UAV swarms, federated learning must also handle mobility, intermittent links, limited energy, unreliable participation, and non-IID sensing data. Prior UAV-FL studies have considered power allocation and scheduling, class imbalance, energy constraints, reliability-aware participation, and hierarchical aggregation (Zeng et al., 2020; Mrad et al., 2021; Cheriguene et al., 2024; Hoang et al., 2023). The works clarify the system constraints of UAV-FL, but they usually treat swarm organization as an aggregation structure rather than a state that participates in later model cooperation.

Clustered and personalized federated learning reduce harmful aggregation among heterogeneous clients by learning client groups or client-specific model combinations (Sattler et al., 2021; Kairouz & McMahan, 2021; Li et al., 2020). UAV clustering and FANET routing methods instead emphasize connectivity, mobility, residual energy, neighborhood relations, and link quality (Arafat & Moh, 2018; Abdulhae et al., 2022). Recent cross-layer UAV-FL methods combine learning similarity with network and resource states. For example, X-CFL integrates application-layer similarity with location, energy, and routing states for UAV swarm organization (Lan et al., 2025). The methods suggest that physical feasibility and task similarity should be considered jointly. ATLAS-HFL follows the same view, but further uses the learned organization to initialize next-period cluster models instead of only supporting current-round aggregation.

Anchor graphs approximate large similarity graphs through a small set of representative anchors, avoiding exhaustive pairwise graph construction (Liu et al., 2010; Wang et al., 2016). Multi-view

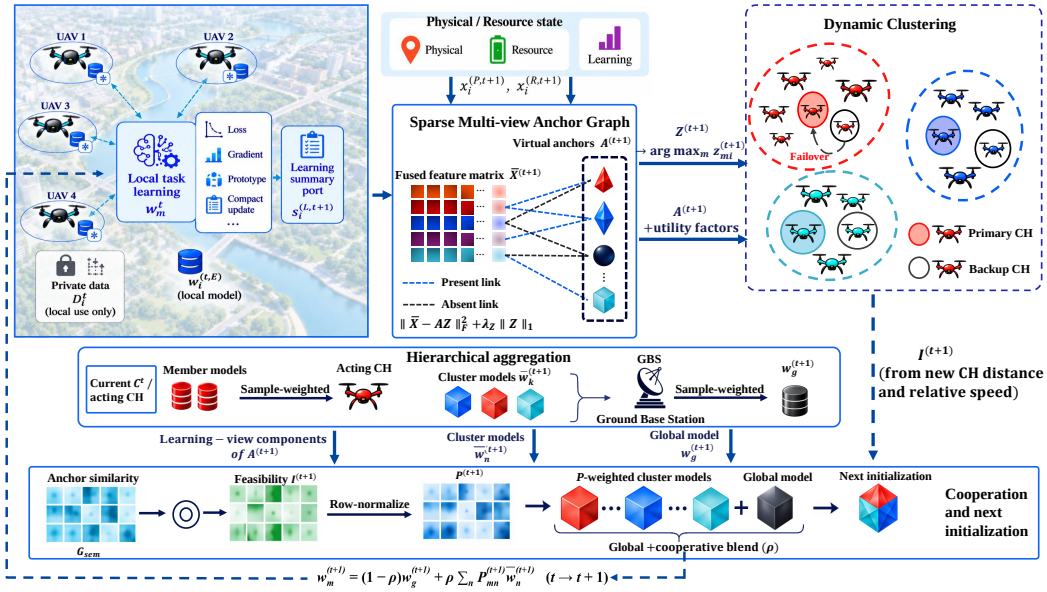


Figure 2: ATLAS-HFL couples local task learning with dynamic UAV organization. Physical, resource, and learning views update a sparse anchor graph. The graph determines clusters, primary-backup CHs, and task-aware model cooperation for the next control period.

anchor graph methods further combine heterogeneous representations for scalable clustering (Zhang et al., 2023; Guo et al., 2026). Existing anchor graph methods mainly target centralized clustering, where the learned graph is used to partition samples. ATLAS-HFL embeds a sparse anchor graph into dynamic hierarchical federated learning, where the graph supports clustering, primary-backup CH election, and task-aware cooperation weights for cluster-specific initialization. The design positions ATLAS-HFL closer to a closed-loop use of anchor graphs in UAV federated learning than to anchor optimization alone.

3 Method

3.1 Overview and Problem Setting

ATLAS-HFL considers a dynamic UAV swarm with N UAVs and one ground base station. The swarm operates over control periods $t = 0, \ldots, T-1$. At each period, the UAVs are organized into M clusters, where $M \ll N$. Each cluster contains one primary cluster head (CH) and one backup CH. The primary CH performs intra-cluster aggregation and uplink communication. The backup CH takes over the CH role if the primary CH becomes unavailable. Each UAV keeps its task data local. UAV i holds a private dataset. The objective is to form clusters that remain communication-feasible, resource-sustainable, and task-consistent. The cluster structure should also support later model training instead of serving only as a current-round communication route.

ATLAS-HFL uses a sparse anchor graph as a reusable organization state. As shown in Figure 2, local training produces compact learning states, which are fused with physical and resource states to update the anchor graph. The graph determines dynamic clusters and primary-backup CHs, and anchor-level task similarity produces cluster-specific initial models for the next control period. New local training states then update the graph again, forming a closed loop between federated learning and swarm organization.

3.2 Learning-Aware UAV State

At control period t, UAV i receives the cluster-specific model $\boldsymbol{w}_m^t$ if $i \in C_m^t$. The local empirical loss is:

$$F_i^t(\boldsymbol{w}) = \frac{1}{n_i} \sum_{j=1}^{n_i} \ell\big(f_{\boldsymbol{w}}(\boldsymbol{x}_{ij}^t), y_{ij}^t\big)\,, \tag{1}$$

where $f_{\boldsymbol{w}}(\cdot)$ denotes the task model and $\ell(\cdot,\cdot)$ denotes the loss function. The local model is initialized by

$$\boldsymbol{w}_i^{t,0} = \boldsymbol{w}_m^t. \tag{2}$$

Each UAV performs E local optimization steps:

$$\boldsymbol{w}_i^{t,e+1} = \boldsymbol{w}_i^{t,e} - \eta \nabla F_i^t(\boldsymbol{w}_i^{t,e}), \quad e = 0, \ldots, E-1. \tag{3}$$

The local model update is

$$\Delta \boldsymbol{w}_i^t = \boldsymbol{w}_i^{t,E} - \boldsymbol{w}_m^t. \tag{4}$$

After local training, UAV i forms a compact learning summary:

$$\boldsymbol{s}_i^{L,t} = \left[\ell_i^t, \phi_g(\boldsymbol{g}_i^t), \phi_p(\boldsymbol{p}_i^t), \phi_w(\Delta \boldsymbol{w}_i^t)\right]^\top. \tag{5}$$

Here, ℓ_i^t is the final local loss. The vector $\boldsymbol{g}_i^t$ denotes a gradient representation. The vector $\boldsymbol{p}_i^t$ denotes a feature prototype. The vector $\Delta \boldsymbol{w}_i^t$ denotes the model update. The operators $\boldsymbol{\phi}_g(\cdot)$, $\boldsymbol{\phi}_p(\cdot)$, and $\boldsymbol{\phi}_w(\cdot)$ produce compact representations. Raw task samples are not exchanged.

ATLAS-HFL represents UAV i through physical, resource, and learning views:

$$\boldsymbol{x}_i^t = \left[(\boldsymbol{x}_i^{P,t})^\top, (\boldsymbol{x}_i^{R,t})^\top, (\boldsymbol{x}_i^{L,t})^\top\right]^\top, \quad \boldsymbol{x}_i^{L,t} = \boldsymbol{s}_i^{L,t}. \tag{6}$$

The physical view contains mobility and link-related states. The resource view contains energy, computation, and communication resources. The learning view contains the compact local training state. After view-wise normalization, the fused vector is denoted by $\widetilde{\boldsymbol{x}}_i^t$. The fused feature matrix is:

$$\widetilde{\boldsymbol{X}}^t = \left[\widetilde{\boldsymbol{x}}_1^t, \widetilde{\boldsymbol{x}}_2^t, \ldots, \widetilde{\boldsymbol{x}}_N^t\right] \in \mathbb{R}^{d \times N}. \tag{7}$$

The fused state allows the organization module to consider network feasibility, resource capacity, and task relations in one representation.

3.3 Sparse Anchor Graph for Dynamic Organization

A full pairwise UAV graph is expensive and sensitive to mobility. ATLAS-HFL uses M anchors to represent the swarm structure. The anchor matrix is:

$$\boldsymbol{A}^t = [\boldsymbol{a}_1^t, \boldsymbol{a}_2^t, \ldots, \boldsymbol{a}_M^t] \in \mathbb{R}^{d \times M}. \tag{8}$$

Each anchor represents a latent cluster prototype in the fused feature space. The UAV-to-anchor association matrix is:

$$\boldsymbol{Z}^t = [\boldsymbol{z}_1^t, \boldsymbol{z}_2^t, \ldots, \boldsymbol{z}_N^t] \in \mathbb{R}^{M \times N}. \tag{9}$$

The sparse anchor graph is learned by:

$$\min_{\boldsymbol{A}^t, \boldsymbol{Z}^t \geq 0} \left\| \widetilde{\boldsymbol{X}}^t - \boldsymbol{A}^t \boldsymbol{Z}^t \right\|_F^2 + \lambda_z \left\| \boldsymbol{Z}^t \right\|_1. \tag{10}$$

The reconstruction term preserves the fused UAV state. The sparse term suppresses weak or redundant anchor associations. UAV i is assigned to the anchor with the largest association:

$$c_i^t = \arg \max_{m \in \{1, \ldots, M\}} z_{mi}^t. \tag{11}$$

The cluster set is:

$$C_m^t = \{i \in \mathcal{U} \mid c_i^t = m\}. \tag{12}$$

The anchor graph acts as a compact organization state. The association matrix $\boldsymbol{Z}^t$ gives cluster membership. The anchor matrix $\boldsymbol{A}^t$ summarizes the physical, resource, and learning characteristics of latent clusters. The learned state is reused by CH election and next-period model cooperation.

The anchor matrix is warm-started from the previous control period. The optimization alternates between UAV-to-anchor association estimation and anchor refinement. Warm-starting encourages temporal consistency across control periods. The graph can then evolve gradually with mobility, resource variation, and local training states.

3.4 Anchor-Assisted Primary-Backup CH Election

The anchor graph also supports CH election. For UAV $i \in C_m^t$, anchor representativeness is measured by:

$$r_{im}^t = -\left\|\widetilde{\boldsymbol{x}}_i^t - \boldsymbol{a}_m^t\right\|_2. \tag{13}$$

A larger value means that UAV i is closer to the anchor of cluster m. The normalized CH utility is:

$$u_{im}^t = \omega_r \widehat{r}_{im}^t + \omega_E \widehat{E}_i^t + \omega_R \widehat{R}_i^t - \omega_v \widehat{v}_i^t, \tag{14}$$

with

$$\omega_r + \omega_E + \omega_R + \omega_v = 1. \tag{15}$$

Here, $\widehat{E}_i^t$ is normalized residual energy. The term $\widehat{R}_i^t$ is normalized computation and communication resource capacity. The term $\widehat{v}_i^t$ is normalized mobility. The term $\hat{r}_{im}^t$ denotes the normalized value of r_{im}^t. High energy and high resource capacity increase the CH score. High mobility decreases the CH score.

The primary CH and backup CH are selected by

$$\begin{aligned} h_m^t &= \arg\max_{i \in C_m^t} u_{im}^t, \\ b_m^t &= \arg\max_{i \in C_m^t \setminus \{h_m^t\}} u_{im}^t. \end{aligned} \tag{16}$$

If the primary CH fails, the backup CH becomes the acting CH:

$$\widehat{h}_m^t = \begin{cases} h_m^t, & \text{if } h_m^t \text{ is available,} \\ b_m^t, & \text{otherwise.} \end{cases} \tag{17}$$

The failover step keeps current cluster membership. It avoids immediate re-clustering after a primary CH failure. CH roles are updated again after the next anchor-graph update.

3.5 Task-Aware Hierarchical Aggregation and Initialization

The acting CH first aggregates local models inside each cluster:

$$\bar{\boldsymbol{w}}_m^{t+1} = \sum_{i \in C_m^t} \frac{n_i}{n_m} \boldsymbol{w}_i^{t,E}, \quad n_m = \sum_{i \in C_m^t} n_i. \tag{18}$$

The ground base station then aggregates cluster-level models:

$$\boldsymbol{w}_g^{t+1} = \sum_{m=1}^{M} \frac{n_m}{n} \bar{\boldsymbol{w}}_m^{t+1}, \quad n = \sum_{m=1}^{M} n_m. \tag{19}$$

Standard hierarchical aggregation sends the same global model to all clusters. ATLAS-HFL instead uses anchor-level task similarity to produce cluster-specific initial models. Let $\boldsymbol{a}_{m,L}^{t+1}$ denote the learning-view component of anchor $\boldsymbol{a}_m^{t+1}$. The communication feasibility indicator matrix $\boldsymbol{I}^{t+1}$ has entries:

$$I_{mn}^{t+1} = \begin{cases} 1, & m = n, \\ 1, & d(\widehat{h}_m^{t+1}, \widehat{h}_n^{t+1}) \le D_{\text{coop}} \text{ and } \left\|\boldsymbol{v}_{\widehat{h}_m}^{t+1} - \boldsymbol{v}_{\widehat{h}_n}^{t+1}\right\|_2 \le V_{\text{coop}}, \\ 0, & \text{otherwise.} \end{cases} \tag{20}$$

The diagonal value is always one. Each cluster can retain its own model even if direct cooperation with other clusters is infeasible.

The task-aware cooperation matrix $\boldsymbol{P}^{t+1}$ has entries:

$$P_{mn}^{t+1} = \frac{I_{mn}^{t+1} \exp\left(-\left\|\boldsymbol{a}_{m,L}^{t+1} - \boldsymbol{a}_{n,L}^{t+1}\right\|_2^2 / \tau\right)}{\sum_{q=1}^{M} I_{mq}^{t+1} \exp\left(-\left\|\boldsymbol{a}_{m,L}^{t+1} - \boldsymbol{a}_{q,L}^{t+1}\right\|_2^2 / \tau\right)}. \tag{21}$$

The matrix $\boldsymbol{P}^{t+1}$ is row-normalized. Similar and reachable clusters receive larger weights. The collaborative cluster model is:

$$\boldsymbol{w}_{m,\mathrm{sim}}^{t+1} = \sum_{n=1}^{M} P_{mn}^{t+1} \bar{\boldsymbol{w}}_{n}^{t+1}. \tag{22}$$

The next initial model for cluster m is:

$$\boldsymbol{w}_{m}^{t+1} = (1-\rho)\boldsymbol{w}_{g}^{t+1} + \rho \boldsymbol{w}_{m,\mathrm{sim}}^{t+1}, \quad 0 \le \rho \le 1. \tag{23}$$

The global term prevents complete cluster isolation. The cooperation term transfers useful knowledge from reachable clusters with similar anchor-level task states. The resulting model $\boldsymbol{w}_{m}^{t+1}$ initializes local training in the next control period.

The new local updates produce new losses, gradients, prototypes, and model changes. Such training states update the fused UAV representation and the anchor graph. The graph then reshapes clustering, CH election, and next-period model cooperation. The mechanism turns UAV organization from a communication scaffold into an active component of federated training.

4 Experiments

4.1 Datasets and Experimental Setup

ATLAS-HFL is evaluated on Fashion-MNIST, CIFAR-10, CIFAR-100, FEMNIST, and DomainNet. The datasets cover grayscale images, natural images, fine-grained classes, writer-dependent samples, and domain-shifted visual data. Each dataset is partitioned across 20 UAVs. Unless otherwise specified, the swarm contains one ground base station and four clusters. Training runs for 50 communication rounds, organized as 10 control periods with 5 rounds per period.

The baselines include FedAvg (McMahan et al., 2017), CFL (Sattler et al., 2021), FedCH (Hoang et al., 2023), FedProx (Li et al., 2020), MOON (Li et al., 2021a), Ditto (Li et al., 2021b), and FedBN (Li et al., 2021c). The baselines cover global averaging, clustered learning, resource-aware hierarchical aggregation, proximal regularization, contrastive learning, personalization, and feature normalization. Label-skew partitions are used for heterogeneity analysis. Additional experiments vary the heterogeneity level, primary-CH failure probability, and UAV population size.

Accuracy and loss measure task performance. Service availability measures CH robustness under primary-CH failures. NS-3 reports packet delivery ratio, throughput, delay, residual energy, and network lifetime. FND, HND, and LND denote depletion times for the first UAV, half of the UAVs, and the last UAV.

Table 1: Mean and standard deviation of the test accuracy of ATLAS-HFL and the baselines on five image classification benchmarks across 20 UAVs.

Models	Dataset				
	Fashion-MNIST	CIFAR-10	CIFAR-100	FEMNIST	DomainNet
FedAvg	0.769±0.007	0.429±0.026	0.131±0.003	0.424±0.037	0.446±0.017
CFL	0.834±0.020	0.492±0.031	0.144±0.001	0.439±0.044	0.473±0.010
FedCH	0.808±0.009	0.464±0.011	0.150±0.008	0.422±0.035	0.429±0.025
FedProx	0.768±0.007	0.428±0.025	0.131±0.003	0.425±0.037	0.446±0.017
MOON	0.765±0.006	0.426±0.021	0.127±0.003	0.427±0.028	0.425±0.022
Ditto	**0.875±0.003**	**0.611±0.016**	**0.222±0.014**	0.497±0.050	0.448±0.008
FedBN	0.770±0.009	0.453±0.025	0.133±0.004	0.455±0.062	0.451±0.009
ATLAS-HFL	0.823±0.027	0.495±0.075	0.173±0.036	**0.526±0.011**	**0.546±0.023**

4.2 Main Results across Datasets

Table 1 reports the main task performance on five datasets. ATLAS-HFL achieves the best accuracy on FEMNIST and DomainNet, and obtains the second-best accuracy on CIFAR-10 and CIFAR-100.

Table 2: Ablation results across datasets.

Model	Dataset				
	Fashion-MNIST	DomainNet	FEMNIST	CIFAR-10	CIFAR-100
ATLAS-HFL	**0.823±0.027**	**0.546±0.023**	**0.526±0.011**	**0.495±0.075**	**0.173±0.036**
ATLAS-HFLt1	0.791±0.022	0.506±0.016	0.474±0.111	0.476±0.034	0.168±0.004
ATLAS-HFLt2	0.814±0.019	0.531±0.003	0.490±0.023	0.494±0.003	0.172±0.018
ATLAS-HFLt3	0.809±0.025	0.525±0.010	0.490±0.149	0.494±0.086	0.173±0.034
ATLAS-HFLt4	0.786±0.014	0.534±0.003	0.457±0.155	0.467±0.066	0.169±0.012

It also improves over FedAvg, FedProx, MOON, FedCH, and FedBN on all five datasets. Compared with CFL, ATLAS-HFL obtains higher accuracy on four of the five datasets. The gains are more visible on CIFAR-100, FEMNIST, and DomainNet, where fine-grained classes, user-dependent samples, or domain shifts create stronger local heterogeneity. The results support the role of learning-aware UAV organization under heterogeneous federated tasks.

Ditto obtains the highest accuracy on Fashion-MNIST, CIFAR-10, and CIFAR-100. The result is expected because Ditto directly optimizes personalized client models. ATLAS-HFL studies a different axis. Local learning states are used to update UAV organization, and the learned organization is then reused for clustering, CH election, and next-period model initialization. The competitive results with a simple hierarchical aggregation backbone indicate that the closed-loop organization provides complementary gains.

4.3 Ablation Study

Table 2 reports the ablation results. Four degraded variants are used to isolate the main links of ATLAS-HFL. ATLAS-HFLt1 removes the learning view by setting the learning-view representation to zero, so the anchor graph is built only from physical and resource states. ATLAS-HFLt2 removes sparse regularization. ATLAS-HFLt3 removes inter-cluster model cooperation, so each cluster uses only its own aggregated model for the next control period. ATLAS-HFLt4 fixes the anchor matrix and the anchor graph after initialization, so the organization structure is not updated over time.

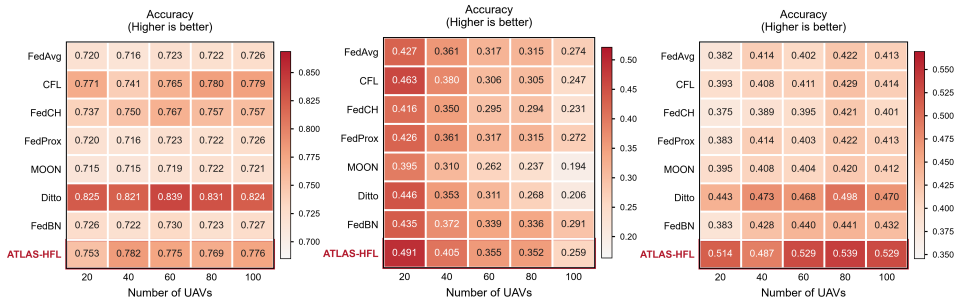
The full ATLAS-HFL model achieves the best accuracy on four datasets and a tied-best result on CIFAR-100. Removing the learning view causes clear drops on Fashion-MNIST, DomainNet, and FEMNIST, which shows that local learning states provide useful task information beyond physical and resource states. Removing model cooperation also lowers accuracy on most datasets, showing that anchor-level task similarity helps build better cluster-specific initialization. Fixing the anchor graph causes large degradation, especially on Fashion-MNIST, FEMNIST, and CIFAR-10, which confirms the need for dynamic organization updates in UAV swarms. Removing sparse regularization gives a smaller but consistent drop.

Overall, the results support the closed-loop design of ATLAS-HFL, where learning states update swarm organization and the learned organization further guides model cooperation.

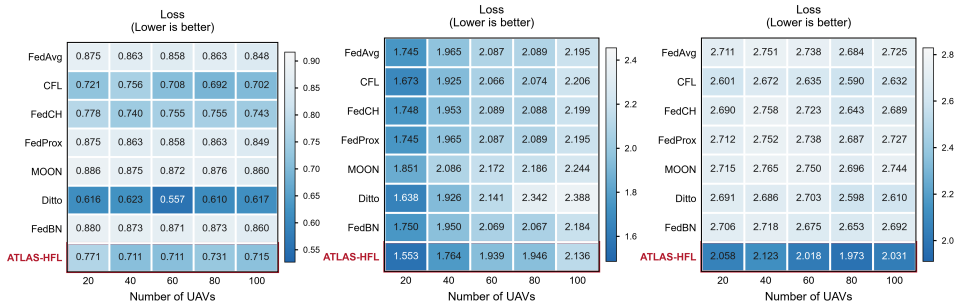
4.4 Heterogeneity and Scalability Analysis

4.4.1 UAV Population Scalability

Figure 3 shows accuracy and loss under different UAV population sizes. The scalability study uses a separate set of random seeds from the main comparison in Table 1, so the 20-UAV entries are used for trend analysis rather than direct numerical matching. ATLAS-HFL keeps stable performance across swarm sizes. On FEMNIST, it achieves the best accuracy and the lowest loss for all sizes. On DomainNet and Fashion-MNIST, it remains competitive with strong baselines and keeps low loss in most settings. The result shows that the sparse anchor graph supports scalable UAV organization. The anchor-based representation avoids full pairwise UAV relations and keeps clusters aligned with current physical, resource, and learning states.

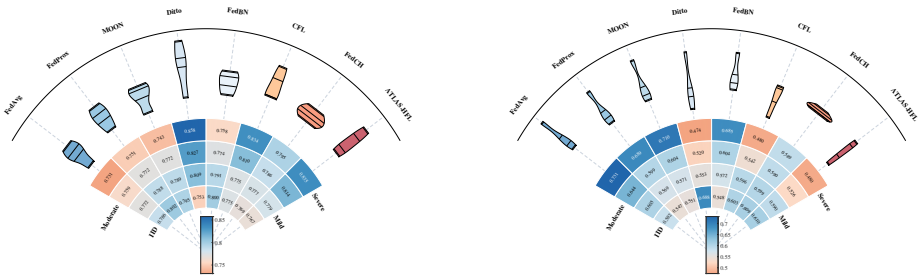


(a) Fashion-MNIST Acc. (b) DomainNet Acc. (c) FEMNIST Acc.



(d) Fashion-MNIST Loss (e) DomainNet Loss (f) FEMNIST Loss

Figure 3: Scalability under different UAV population sizes.



(a) ACC (b) Loss

Figure 4: Accuracy and loss under different non-IID strengths.

4.4.2 Non-IID Robustness Analysis

Figure 4 reports accuracy and loss under different non-IID strengths on Fashion-MNIST. The heatmap gives the mean value under IID, mild, moderate, and severe label skew. The distribution above each method summarizes the mean and variance across heterogeneity levels.

ATLAS-HFL keeps strong accuracy and low loss as data heterogeneity increases. Under severe non-IID, the methods split into two groups. FedAvg, FedProx, MOON, and FedBN all degrade, whereas Ditto, CFL, and ATLAS-HFL all improve. A clear pattern appears. Methods that rely more strongly on a shared global model tend to degrade. Personalized or cluster-specific methods benefit because stronger label skew makes each local task more consistent. Ditto benefits from personalized client models, while ATLAS-HFL benefits from task-aware clustering and cluster-specific initialization.

Under the IID setting, the trend reverses. FedAvg reaches 0.786, against 0.767 for ATLAS-HFL. Overall, the gain of learning-aware organization grows with the level of label skew and disappears under IID data.

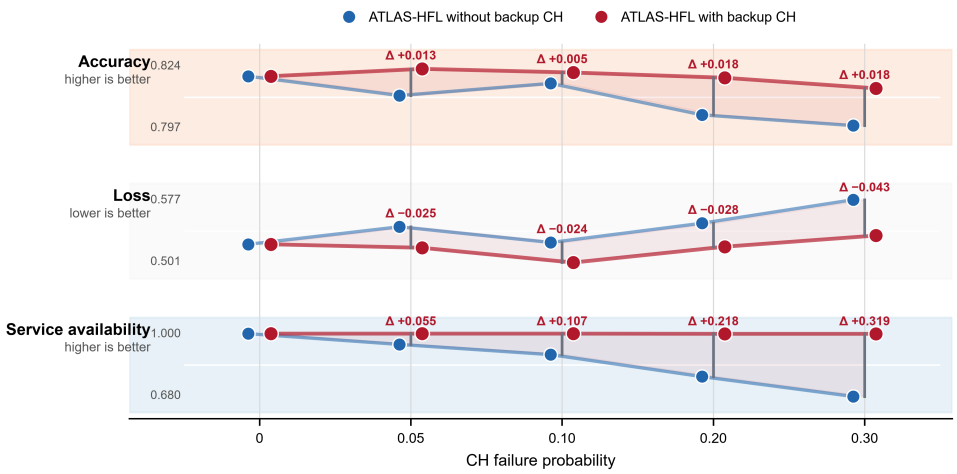
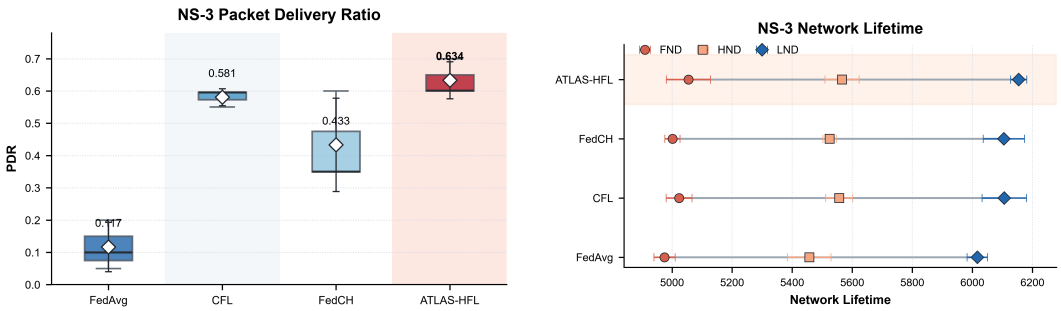


Figure 5: Effect of primary-CH failures. The backup CH preserves aggregation and uplink service as the failure probability increases.

4.5 CH FAILURE AND NETWORK PERFORMANCE

Figure 5 evaluates the primary-backup CH design under failure probabilities from 0 to 0.30. At zero failure probability, the two settings perform similarly. At 0.30, ATLAS-HFL with backup CH maintains 0.999±0.002 service availability, while the variant without backup CH drops to 0.680±0.024. Accuracy also remains higher, 0.815±0.023 versus 0.797±0.026, and loss is reduced by 0.043. The result supports the failover mechanism in dynamic UAV swarms.

Figure 6 shows the NS-3 results with FedAvg, CFL, and FedCH because the baselines define communication or clustering behavior. Other FL baselines mainly change local training or personalization and do not specify UAV clustering, CH forwarding, backup service, or multi-hop communication. ATLAS-HFL reaches the highest mean PDR of 0.634, compared with 0.581 for CFL, 0.433 for FedCH, and 0.117 for FedAvg. Network lifetime remains close to CFL and FedCH, showing stronger communication reliability with comparable energy consumption.



(a) Packet delivery ratio (b) Network lifetime

Figure 6: NS-3 system results. ATLAS-HFL improves packet delivery and retains a network lifetime comparable to hierarchical baselines.

5 CONCLUSION

ATLAS-HFL presents a learning-aware anchor graph for hierarchical federated learning in dynamic UAV swarms. Local learning signals update swarm organization, and the learned organization guides clustering, primary-backup CH election, and next-period model cooperation. Under label-skew non-IID partitions, ATLAS-HFL outperforms FedAvg, FedProx, MOON, FedCH, and FedBN on all five benchmarks, exceeds CFL on four of five, and achieves the best accuracy on FEMNIST and DomainNet. Ditto remains stronger on some datasets because per-device personalization and swarm-level organization pursue different goals. CH failure experiments and NS-3 simulations show higher service availability and packet delivery with comparable network lifetime.

AI USE STATEMENT

Generative AI tools (*ChatGPT*) were used to assist with language editing, LaTeX formatting, literature organization, as well as in discussions on experimental design and result interpretation. All research ideas, methodological decisions, experimental settings, implementations, and conclusions are the authors' own work. All reported numbers, tables, and experimental results were produced exclusively from the authors' experiments and were not generated by any AI tool, and the reference list was compiled and verified by the authors. All AI-assisted text, code, and analyses were manually checked by the authors for correctness and consistency with the reported experiments. The authors take full responsibility for the content of this work.

REPRODUCIBILITY STATEMENT

The code and instructions for reproducing the experiments are available at `https://anonymous.4open.science/r/ATLAS-HFL-4D7F/`.

REFERENCES

Omer T. Abdulhae, Jit Singh Mandeep, and Mt Islam. Cluster-based routing protocols for flying ad hoc networks (fanets). *IEEE Access*, 10:32981–33004, 2022. doi: 10.1109/ACCESS.2022.3161446.

Muhammad Morshed Alam, Muhammad Yeasir Arafat, Sangman Moh, and Jian Shen. Topology control algorithms in multi-unmanned aerial vehicle networks: An extensive survey. *Journal of Network and Computer Applications*, 207:103495, 2022.

Muhammad Yeasir Arafat and Sangman Moh. A survey on cluster-based routing protocols for unmanned aerial vehicle networks. *IEEE Access*, 7:498–516, 2018.

Ilker Bekmezci, Ozgur Koray Sahingoz, and Şamil Temel. Flying ad-hoc networks (fanets): A survey. *Ad Hoc Networks*, 11(3):1254–1270, 2013.

Youssra Cheriguene, Wael Jaafar, Halim Yanikomeroglu, and Chaker Abdelaziz Kerrache. Towards reliable participation in uav-enabled federated edge learning on non-iid data. *IEEE Open Journal of Vehicular Technology*, 5:125–141, 2024. doi: 10.1109/OJVT.2023.3341304.

Wei Guo, Zhe Wang, and Wei Shao. Structure anchor graph learning for multi-view clustering. *Pattern Recognition*, 170:111880, 2026. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2025.111880. URL `https://www.sciencedirect.com/science/article/pii/S0031320325005400`.

Duc NM Hoang, Vu Tuan Truong, Hung Duy Le, and Long Bao Le. Clustered and scalable federated learning framework for uav swarms. TechRxiv preprint, 2023.

Peter Kairouz and H Brendan McMahan. Advances and open problems in federated learning. *Foundations and Trends in Machine Learning*, 14(1-2):1–210, 2021.

Jiangyu Lan, Xiaoting Ma, Shuai Gao, Weiting Zhang, and Xindi Hou. X-cfl: Enabling cross-layer clustered federated learning in uav swarms. In *GLOBECOM 2025-2025 IEEE Global Communications Conference*, pp. 4403–4408. IEEE, 2025.

Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10708–10717. IEEE, 2021a.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems*, 2:429–450, 2020.

Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. Ditto: Fair and robust federated learning through personalization. In *International conference on machine learning*, pp. 6357–6368. PMLR, 2021b.

Xiaoxiao Li, Meirui Jiang, Xiaofei Zhang, Michael Kamp, and Qi Dou. Fedbn: Federated learning on non-iid features via local batch normalization. *arXiv preprint arXiv:2102.07623*, 2021c.

Wei Liu, Junfeng He, and Shih-Fu Chang. Large graph construction for scalable semi-supervised learning. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pp. 679–686, 2010.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pp. 1273–1282. PMLR, 2017. URL https://proceedings.mlr.press/v54/mcmahan17a.html.

Ilyes Mrad, Lutfi Samara, Alaa Awad Abdellatif, Abubakr Al-Abbasi, Ridha Hamila, and Aiman Erbad. Federated learning for uav swarms under class imbalance and power consumption constraints. In *2021 IEEE Global Communications Conference (GLOBECOM)*, pp. 01–06. IEEE, 2021.

Felix Sattler, Klaus-Robert Müller, and Wojciech Samek. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. *IEEE Transactions on Neural Networks and Learning Systems*, 32(8):3710–3722, 2021. doi: 10.1109/TNNLS.2020.3015958. URL https://doi.org/10.1109/TNNLS.2020.3015958.

Meng Wang, Weijie Fu, Shijie Hao, Dacheng Tao, and Xindong Wu. Scalable semi-supervised learning by efficient anchor graph regularization. *IEEE Transactions on Knowledge and Data Engineering*, 28(7):1864–1877, 2016. doi: 10.1109/TKDE.2016.2535367.

Tengchan Zeng, Omid Semiari, Mohammad Mozaffari, Mingzhe Chen, Walid Saad, and Mehdi Bennis. Federated learning in the sky: Joint power allocation and scheduling with uav swarms. In *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, pp. 1–6, 2020. doi: 10.1109/ICC40277.2020.9148776.

Pei Zhang, Siwei Wang, Liang Li, Changwang Zhang, Xinwang Liu, En Zhu, Zhe Liu, Lu Zhou, and Lei Luo. Let the data choose: Flexible and diverse anchor graph fusion for scalable multi-view clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 11262–11269, 2023.

A Anchor-Graph Optimization and Implementation Details

Relation to the main paper. The appendix complements the main paper with material that is not stated there, namely the cold-start initialization of the learning view, the complete optimization procedure of the sparse multi-view anchor graph in Algorithm 1, the failover rules of the primary-backup cluster-head (CH) mechanism, and the outer-loop procedure of ATLAS-HFL in Algorithm 2. All other formulation details are given in Sections 3.2–3.5 of the main paper and are not repeated here.

A.1 Sparse Multi-view Anchor Graph Optimization

Section 3.3 defines the sparse anchor-graph learning objective in Eq. (10). The corresponding optimization procedure is presented below. Let $\tilde{\boldsymbol{X}}^t \in \mathbb{R}^{d\times N}$ denote the fused feature matrix at control period t, where $\tilde{\boldsymbol{x}}_i^t$ is the multi-view representation of UAV i. To reduce the influence of unequal feature magnitudes on reconstruction error, the feature matrix is normalized column-wise:

$$\bar{\boldsymbol{X}}^t = \left[\frac{\tilde{\boldsymbol{x}}_1^t}{\|\tilde{\boldsymbol{x}}_1^t\|_2+\varepsilon}, \ldots, \frac{\tilde{\boldsymbol{x}}_N^t}{\|\tilde{\boldsymbol{x}}_N^t\|_2+\varepsilon}\right], \tag{24}$$

all subsequent updates are performed on $\bar{\boldsymbol{X}}^t$.

Cold start of the learning view. At the first control period, no loss, gradient summary, or class prototype is available. The learning view is initialized as:

$$\boldsymbol{x}_i^{L,0} = \mathbf{0}. \tag{25}$$

The initial anchor graph is therefore driven mainly by the physical-network and resource views. As federated training proceeds, learning representations gradually participate in anchor-graph updating, allowing clustering to jointly reflect communication conditions, resource states, and task relatedness.

A.1.1 ANCHOR INITIALIZATION AND TEMPORAL WARM START

At the first control period, anchors are initialized from UAVs satisfying the CH energy condition. The initialization priority of UAV i is defined as:

$$q_i = 0.55E_i + 0.30R_i - 0.15L_i. \tag{26}$$

The first anchor is selected from the eligible UAV with the highest priority. The remaining anchors are selected by jointly considering resource priority and representation coverage:

$$i^\star = \arg\max_{i \notin \mathcal{S}} \left[q_i + 0.20 \min_{j \in \mathcal{S}} \left\| \bar{\boldsymbol{x}}_i^t - \bar{\boldsymbol{x}}_j^t \right\|_2 \right]. \tag{27}$$

Here, $\mathcal{S}$ denotes the index set of already selected anchors. The strategy improves both resource feasibility and representation diversity.

For subsequent control periods, the anchor matrix is warm-started by the result from the previous period:

$$\boldsymbol{A}^{t,(0)} = \boldsymbol{A}^{t-1}. \tag{28}$$

The warm start allows the anchor graph to evolve gradually with changes in UAV positions, links, resources, and learning states.

A.1.2 THE $\boldsymbol{Z}$ SUBPROBLEM

With $\boldsymbol{A}^{(r)}$ fixed at outer iteration r, Eq. (10) in Section 3.3 is decomposed into N independent non-negative sparse-coding problems:

$$\min_{\boldsymbol{z}_i \geq \boldsymbol{0}} \left\| \bar{\boldsymbol{x}}_i - \boldsymbol{A}^{(r)} \boldsymbol{z}_i \right\|_2^2 + \lambda_z \left\| \boldsymbol{z}_i \right\|_1, \quad i = 1, \ldots, N. \tag{29}$$

The gradient of the smooth reconstruction term is:

$$\nabla_{\boldsymbol{z}_i} = 2\boldsymbol{A}^{(r)\top} \left(\boldsymbol{A}^{(r)} \boldsymbol{z}_i - \bar{\boldsymbol{x}}_i \right). \tag{30}$$

A non-negative proximal-gradient update is applied:

$$\boldsymbol{z}_i^{(r,s+1)} = \left[\boldsymbol{z}_i^{(r,s)} - \eta \nabla_{\boldsymbol{z}_i} - \eta \lambda_z \boldsymbol{1} \right]_+, \tag{31}$$

where $[\,\cdot\,]_+$ denotes element-wise non-negative projection and s is the inner iteration index. The step size is defined as:

$$\eta = \min \left(\eta_{\max}, \frac{1}{2 \left\| \boldsymbol{A}^{(r)} \right\|_2^2 + \varepsilon} \right). \tag{32}$$

The ℓ_1 penalty in Eq. (31) progressively shrinks weak anchor associations towards zero. Therefore, sparsity in $\boldsymbol{Z}^t$ is formed automatically without prescribing that each UAV must connect to exactly one anchor.

A.1.3 THE $\boldsymbol{A}$ SUBPROBLEM

With $\boldsymbol{Z}^{(r+1)}$ fixed, the anchor-update subproblem of Eq. (10) becomes:

$$\min_{\boldsymbol{A}} \left\| \bar{\boldsymbol{X}} - \boldsymbol{A} \boldsymbol{Z}^{(r+1)} \right\|_F^2. \tag{33}$$

With a numerical stabilization term $\epsilon_A \boldsymbol{I}$, the anchor matrix is updated by

$$\boldsymbol{A}^{(r+1)} = \bar{\boldsymbol{X}} \boldsymbol{Z}^{(r+1)\top} \left(\boldsymbol{Z}^{(r+1)} \boldsymbol{Z}^{(r+1)\top} + \epsilon_A \boldsymbol{I} \right)^\dagger, \tag{34}$$

where $(\cdot)^\dagger$ denotes the Moore-Penrose pseudoinverse. Each anchor is then normalized as:

$$\boldsymbol{a}_m^{(r+1)} \leftarrow \frac{\boldsymbol{a}_m^{(r+1)}}{\left\|\boldsymbol{a}_m^{(r+1)}\right\|_2 + \varepsilon}, \quad m = 1, \ldots, M. \tag{35}$$

The normalization removes the scale ambiguity between $\boldsymbol{A}$ and $\boldsymbol{Z}$.

If the sum of an association vector is below a threshold after the prescribed alternating updates, nearest-anchor fallback assignment is applied:

$$m_i^\star = \arg\max_m \left(\boldsymbol{a}_m{}^\top \bar{\boldsymbol{x}}_i\right), \qquad z_{m_i^\star, i} = 1. \tag{36}$$

Cluster labels and memberships are then obtained using Eqs. (11) and (12) in Section 3.3.

The major complexity of association updating is $O(R_Z NdM)$, where R_Z is the number of inner sparse-coding iterations. The approximate complexity of anchor updating is $O\left(NdM + NM^2 + M^3\right)$. Since $M \ll N$, the proposed procedure avoids the construction of a full $N \times N$ UAV similarity graph.

Algorithm 1 shows the whole update rules for Sparse Multi-view Anchor Graph Optimization.

Algorithm 1: Algorithm for Sparse Anchor-Graph Update Procedure

Input: Fused multi-view feature matrix $\tilde{\boldsymbol{X}}^t$, UAV state $\mathcal{S}^t$, the number of anchors M, sparsity coefficient λ_z, maximum outer iterations R_A, maximum inner iterations R_Z, and previous anchor matrix $\boldsymbol{A}^{t-1}$.

Output: Anchor matrix $\boldsymbol{A}^t$, sparse association matrix $\boldsymbol{Z}^t$, and cluster labels $\{c_i^t\}_{i=1}^N$.

1. Normalize $\tilde{\boldsymbol{X}}^t$ column-wise to obtain $\bar{\boldsymbol{X}}^t$ by Eq. (24).
2. **If** $t = 0$ **then**
3. Initialize $\boldsymbol{A}^{(0)}$ by Eqs. (26)–(27);
4. **else**
5. Set $\boldsymbol{A}^{(0)} = \boldsymbol{A}^{t-1}$ by Eq. (28).
6. **end**
7. Initialize $\boldsymbol{Z}^{(0)} = \boldsymbol{0}$ and normalize $\boldsymbol{A}^{(0)}$.
8. **For** $r = 0$ to $R_A - 1$
9. Calculate step size η by Eq. (32).
10. **For** $s = 0$ to $R_Z - 1$
11. Update $\boldsymbol{z}_i^{(s+1)}$ by Eq. (31).
12. **end for**.
13. Assemble all association vectors to obtain $\boldsymbol{Z}^{(r+1)}$.
14. Update $\boldsymbol{A}^{(r+1)}$ by Eq. (34) and normalize each anchor by Eq. (35).
15. **end for**.
16. Perform fallback assignment for near-zero association vectors by Eq. (36).
17. Obtain cluster labels and memberships by Eqs. (11) and (12).

A.2 Failover Rules of the Primary-Backup CH Mechanism

Cluster formation and primary-backup CH election follow Eqs. (11)–(12) in Section 3.3 and Eqs. (13)–(17) in Section 3.4 of the main paper, and are not repeated here. The following implementation rules are used in all experiments and are not stated in the main text.

Rule 1. Backup-unavailable fallback. If the primary CH is unavailable, its backup CH is activated according to Eq. (17). If the backup CH is also unavailable, the system searches for an energy-feasible replacement within the same cluster. If no such UAV exists, the cluster is marked as temporarily unavailable during the current control period.

Rule 2. Role continuity after a primary-CH failure. A primary CH service failure does not force the corresponding UAV to leave local task training. The UAV retains its local data and model state as a regular cluster member, and only the cluster-level aggregation and communication-coordination role changes. CH roles are updated again after the next anchor-graph update.

Algorithm 2: Algorithm for solving ATLAS-HFL

Input: Initial global model w_g^0, UAV state $\mathcal{S}^0$, the number of control periods T, the number of anchors M, sparsity coefficient λ_z, model cooperation coefficient ρ, and communication parameters.

Output: Task models, cluster structures, primary and backup CHs, cooperation matrices, failover records, and performance metrics.

1. Initialize the learning view by $\boldsymbol{X}^{L,0} = \mathbf{0}$ and construct the initial fused feature matrix $\tilde{\boldsymbol{X}}^0$ by Eqs. (6)–(7).
2. Apply Algorithm 1 to obtain the initial cluster structure by Eqs. (11)–(12), and select primary and backup CHs by Eqs. (13)–(16).
3. Initialize each cluster-specific model as $w_m^0 = w_g^0$, $m = 1, \ldots, M$.
4. **For** $t = 0$ to $T - 1$
5. Detect CH failures and activate backup CHs according to Eq. (17).
6. Perform local task training and generate learning summaries by Eqs. (2)–(5), and construct $\tilde{\boldsymbol{X}}^{t+1}$ by Eqs. (6)–(7).
7. Perform intra-cluster aggregation and global aggregation by Eqs. (18)–(19).
8. Apply Algorithm 1 to update the cluster structure by Eqs. (11)–(12).
9. Select primary and backup CHs for the next control period by Eqs. (13)–(16).
10. Construct $\boldsymbol{I}^{t+1}$ by Eq. (20), and obtain $\boldsymbol{P}^{t+1}$ by Eq. (21).
11. Initialize the next cluster-specific models by Eqs. (22)–(23).
12. **end for**
13. Return the final global model, cluster-specific models, cluster structures, primary and backup CHs, cooperation matrices, failover records, and performance metrics.

A.3 Scope of the Feasibility Indicator

The feasibility indicator matrix $\boldsymbol{I}^t$ in Eq. (20) constrains only direct inter-cluster model cooperation and does not block global aggregation through the ground base station in Eqs. (18)–(19). Hence, a temporarily isolated cluster still receives the global model and, through the diagonal entries of $\boldsymbol{I}^t$, retains its own model. After determining $\boldsymbol{I}^t$, the system constructs the anchor-task-similarity-based cooperation matrix $\boldsymbol{P}^t$ by Eq. (21), obtains the cooperative model by Eq. (22), and initializes the next cluster-specific model by Eq. (23). Through the above path, anchor-graph updating changes not only UAV memberships and CH organization but also the initial task model of the next control period.

Algorithm 2 shows the whole framework of ATLAS-HFL.