Proceedings of CONF-SEML 2026 Symposium: Computational Analysis and Modeling in Complex Intelligent Systems
DOI: 10.54254/2755-2721/2026.GU35238

H-FedSpace: Hierarchical Federated Learning for Uplink-Constrained LEO Satellite Networks

Zhihang Fu

Beijing-Dublin International College, Beijing University of Technology, Beijing, China
azazk007@163.com

Abstract. Low Earth orbit (LEO) satellite federated learning is constrained by short ground-station contacts, intermittent inter-satellite links (ISLs), and stale updates. Under these conditions, the central issue is not whether communication can be reduced in general, but whether uplink traffic to the ground can be reduced without materially weakening learning quality. This paper compares FedSpace and H-FedSpace under one shared implementation, one cached feature benchmark, and one common communication accounting scheme. The results show that H-FedSpace cuts logged uplink traffic by 80.0%, from 133.03 MB to 26.61 MB, while preserving final accuracy at 1.0000. It also achieves a much lower final loss, decreasing from 0.0838 to 0.0014, and records zero logged staleness at evaluation checkpoints. At the same time, these gains are accompanied by 133.03 MB of ISL traffic, a 20.0% increase in total logged communication, and longer runtime. The evidence therefore supports H-FedSpace in the narrower operating regime where satellite-to-ground uplink is the binding resource. The interpretation remains cautious because, in the current implementation, H-FedSpace performs local training on all 12 satellites in each round, whereas FedSpace trains only a randomly selected satellite per round.

Keywords: federated learning, LEO satellite networks, hierarchical aggregation, inter-satellite links, uplink-constrained learning

1. Introduction

LEO constellations support Earth observation, sensing, and delay-sensitive data services, but their communication environment is highly uneven. Ground-station contact is sparse, link availability changes over time, and satellites must often continue training under intermittent connectivity and delayed synchronization. These conditions make communication-efficient federated learning attractive as decentralized training can reduce repeated raw-data transfer while still allowing a shared model to evolve [1].

In this setting, hierarchical aggregation is a natural system-level response: sparse ground-station contact motivates in-orbit partial aggregation, which naturally leads to a hierarchical structure [2, 3]. Satellite studies further show that ISLs can support relaying and partial aggregation in orbit, while intermittent ground access remains a key bottleneck for global synchronization [4, 5].

This paper evaluates the bottleneck directly by comparing FedSpace within a single shared LEO-style experimental stack. The study focuses on whether the hierarchical method reduces satellite-to-

DOI: 10.54254/2755-2721/2026.GU35238

ground uplink while preserving predictive performance, and on how much of that reduction is achieved by shifting traffic to ISLs rather than eliminating communication altogether.

The contribution of this paper is threefold. First, it provides a controlled comparison of FedSpace and H-FedSpace under shared data, shared configuration, and a unified traffic ledger. Second, it separates uplink reduction from ISL growth and total traffic instead of treating all transmissions as one aggregate number. Third, it interprets the observed loss and staleness improvements alongside an important implementation caveat: the two methods do not use the same local training volume per round in the current codebase.

2. Literature review

2.1. Federated learning and hierarchical aggregation

Federated learning research establishes the general problem setting in which model training is distributed across multiple clients and communication becomes a primary systems constraint rather than a secondary engineering detail [1]. Building on that foundation, hierarchical federated learning studies show that introducing intermediate aggregation layers can reduce pressure on the global server and improve communication efficiency under constrained networks [2]. Related work on communication-aware optimization also emphasizes that communication cost is multidimensional: the timing, direction, and medium of model transfer all affect system performance [3].

However, the existing literature does not determine whether the hierarchical method is valuable primarily under conditions of uplink scarcity, rather than serving as a universal communication minimizer. Most of them do not distinguish the practical trade-off between saving satellite-to-ground uplink and increasing in-orbit ISL traffic. As a result, they explain why hierarchy may help, but they do not by themselves show whether hierarchy is beneficial in an uplink-constrained satellite environment.

2.2. LEO satellite federated learning

In satellite federated learning research, Razmi et al. proposed an on-board FL scheme for dense LEO constellations that uses intra-orbit ISLs and partial aggregation [4]. Their method offers an early clustered in-orbit coordination perspective for hierarchical satellite federated learning.. So et al. proposed FedSpace, which schedules model aggregation under deterministic and time-varying satellite connectivity by balancing staleness and idleness. It serves as the key asynchronous baseline for H-FedSpace [5]. Lin et al. proposed FedSN for heterogeneous LEO satellite networks, which combines sub-structure local training with pseudo-synchronous aggregation. This work highlights how resource heterogeneity can influence FL behavior in orbit [6]. Shi et al. proposed FedMega, which performs ISL-based intra-orbit aggregation before global aggregation. This provides a direct hierarchical aggregation reference for H-FedSpace [7].

These studies provide useful scheduling, heterogeneity, and hierarchical aggregation perspectives for H-FedSpace, but they also show a common limitation that is directly related to the present study. Existing work usually does not clearly separate satellite-to-ground uplink reduction from the extra ISL cost introduced by in-orbit coordination, and this trade-off is rarely quantified in a directly comparable way [8]. For H-FedSpace, this is the key gap, because the main question is not whether communication changes in general, but whether uplink can be reduced without hiding the added cost that is shifted to ISLs.

DOI: 10.54254/2755-2721/2026.GU35238

3. Methodology

3.1. Experimental design

This study uses a repeated-run quantitative experiment in which FedSpace and H-FedSpace are evaluated under the same simulated constellation, data split, optimizer settings, and logging schedule. Both methods are run for 200 communication rounds with five random seeds: 42, 123, 2024, 7, and 314. Metrics are recorded every 10 rounds, including the final round, resulting in 21 checkpoints per seed.

An important implementation condition is stated upfront because it affects interpretation of results. In the current codebase, FedSpace performs local training on one randomly selected satellite in each round, whereas H-FedSpace performs local training on all 12 satellites in each round before performing cluster aggregation.

With that design boundary established, the following subsections clarify the two compared methods and the shared parameter setting under which they are evaluated.

3.2. Compared methods and key settings

FedSpace is treated as the baseline asynchronous method. After local training, a single satellite's upload is aggregated into the global model using a staleness-decay weight of $\alpha(\tau)=0.85^{\tau}$, where τ is the recorded version gap. H-FedSpace uses the same local model and optimizer, but first groups active satellites by current ISL connectivity and then averages their locally trained models within each cluster. A representative cluster head then performs global upload at an interval of five rounds in the current configuration. Table 1 summarizes the common experimental settings that define this comparison.

Table 1. Common experimental settings

Item	Value
Constellation size	12 satellites, 3 ground stations
Simulation horizon	24 hours, 10-minute time step
Federated rounds	200
Random seeds	42, 123, 2024, 7, 314
Feature dimension / classes	512 features, 10 classes
Dataset size	12,000 training samples / 2,000 test samples
Data split	Dirichlet non-IID, $\alpha=0.5$
Model	$512 \rightarrow 256 \rightarrow 128 \rightarrow 10$ multilayer perceptron
Optimizer	SGD, learning rate 0.05, momentum 0.9, weight decay 10^{-4}
Local training	Batch size 64, 5 local steps
FedSpace local participation	1 satellite per round
H-FedSpace local participation	12 satellites per round
H-FedSpace upload interval	5 rounds

Using this shared setup, Figure 1 illustrates how the experimental pipeline branches into the baseline and hierarchical procedures. Both methods start from the same constellation setting, the

DOI: 10.54254/2755-2721/2026.GU35238

same non-IID data partition, the same local model, and the same evaluation schedule, so the main difference lies in the aggregation pattern rather than the input conditions.

In the baseline branch, FedSpace trains one selected satellite in each round and then merges its local update into the global model through the following staleness-aware asynchronous aggregation:

$$\alpha(\tau)=0.85^{\tau},\ W_{t+1}=(1-\alpha(\tau))W_t+\alpha(\tau)W_t^{(local)} \tag{1}$$

where τ denotes the recorded version gap. This branch represents the direct satellite-to-ground update pattern.

In the hierarchical branch, H-FedSpace first allows satellites connected by active ISLs to complete local training within a cluster and then performs intra-cluster aggregation:

$$\bar{W}_c=\frac{1}{|C|}\sum_{i\in C}W_i \tag{2}$$

The aggregated cluster model is then uploaded periodically by a representative cluster head for global synchronization. This branch therefore inserts a constellation-side aggregation stage before global upload, which is the core workflow difference from FedSpace.

After both branches run under the same seeds and logging schedule, the study records accuracy, loss, staleness, uplink traffic, ISL traffic, and total traffic using one shared accounting rule. The workflow therefore clarifies that the study evaluates not only predictive performance but also the location of the communication burden. This leads directly to the communication-accounting strategy described next.

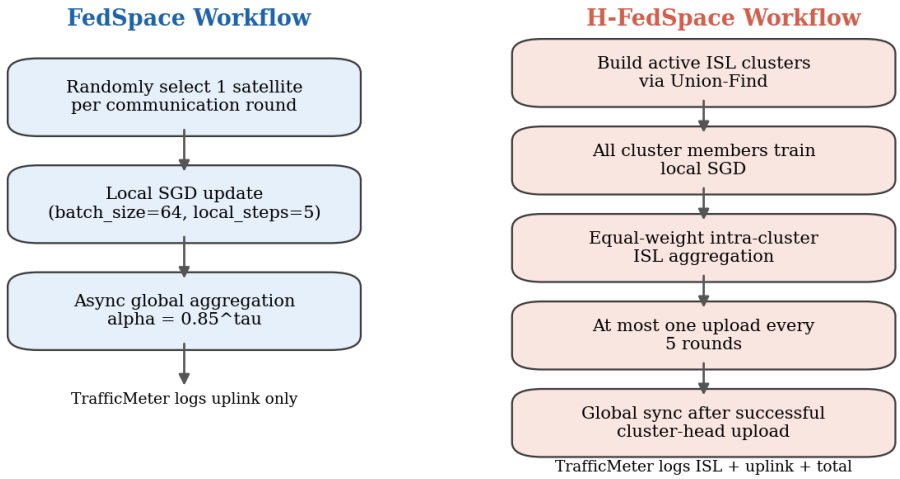

Figure 1. Experimental workflow for the shared FedSpace and H-FedSpace comparison

3.3. Communication accounting and analysis strategy

All communication results are computed from logged model payloads rather than from heuristic bandwidth estimates. The classifier used in both methods contains 166,282 trainable parameters, so each transmitted model corresponds to 665,128 bytes under float32 precision. Uplink traffic records satellite-to-ground model uploads, ISL traffic records member-to-head model transfers within a cluster, and total traffic is the sum of all logged transmissions.

DOI: 10.54254/2755-2721/2026.GU35238

The analysis evaluates three hypotheses. The first is that H-FedSpace preserves final predictive accuracy while reducing uplink traffic. The second is that H-FedSpace lowers loss and recorded staleness relative to FedSpace. The third is that these gains are achieved by shifting coordination load to ISLs, not by minimizing total communication. Runtime is reported as local wall-clock time on a macOS ARM64 machine with 10 CPU cores and is used only as an implementation-level cost indicator.

4. Results

4.1. Final outcome comparison

Table 2. Final-round comparison across five seeds

Metric	FedSpace	H-FedSpace
Final accuracy	1.0000 ± 0.0000	1.0000 ± 0.0000
Final loss	0.0838 ± 0.0295	0.0014 ± 0.0001
Final uplink traffic (MB)	133.03 ± 0.00	26.61 ± 0.00
Final ISL traffic (MB)	0.00 ± 0.00	133.03 ± 0.00
Final total traffic (MB)	133.03 ± 0.00	159.63 ± 0.00
Final wall time (s)	0.5864 ± 0.0238	5.9466 ± 0.2253
Mean recorded staleness (rounds)	90.75	0.00

According to the table 2, the third hypothesis is supported by the communication breakdown analysis. H-FedSpace reduces uplink by replacing frequent direct uploads with in-orbit coordination, but it does not reduce communication in aggregate. The hierarchical method adds 133.03 MB of ISL traffic and increases total logged traffic to 159.63 MB.

4.2. Accuracy, loss, and traffic dynamics

Figure 2 plots final accuracy against cumulative uplink traffic. H-FedSpace reaches 1.0000 accuracy, and does so with a much smaller logged uplink budget. This supports the central claim that the hierarchical method is valuable primarily under uplink scarcity, rather than as a universal communication minimizer.

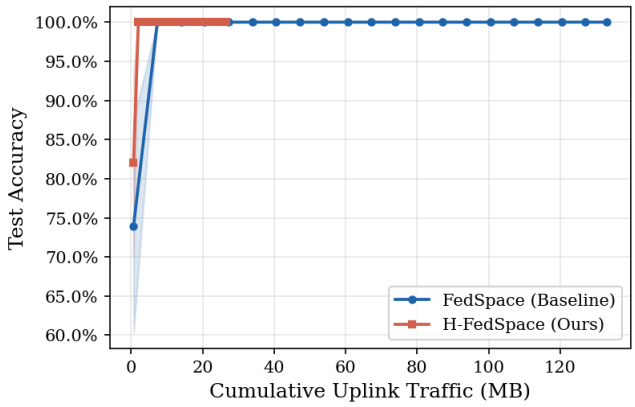

Figure 2. Accuracy versus cumulative uplink traffic (MB)

DOI: 10.54254/2755-2721/2026.GU35238

Figure 3 shows the loss trajectory over communication rounds. The gap opens very early: by round 10, mean loss is 0.5272 for FedSpace and 0.0050 for H-FedSpace. This result is consistent with stronger optimization progress under the hierarchical pipeline, but it should again be read together with the higher per-round training volume of H-FedSpace.

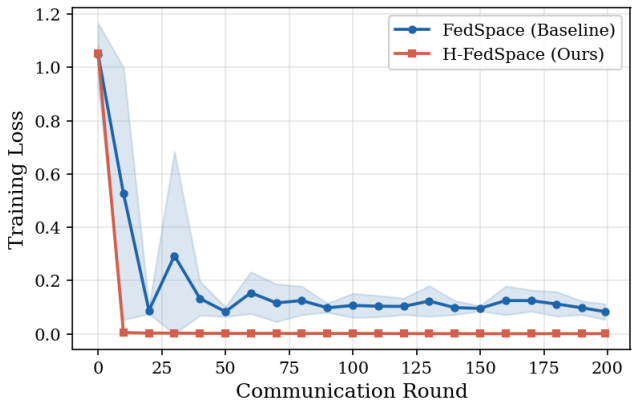

Figure 3. Loss versus communication round

The hierarchical method reduces loss much faster and remains more stable after the early rounds. Figure 4 explains where the saved uplink traffic goes. FedSpace accumulates uplink almost linearly, whereas H-FedSpace keeps uplink growth modest by moving a large share of coordination to ISLs. The figure therefore gives direct visual support to the interpretation adopted for the third hypothesis.

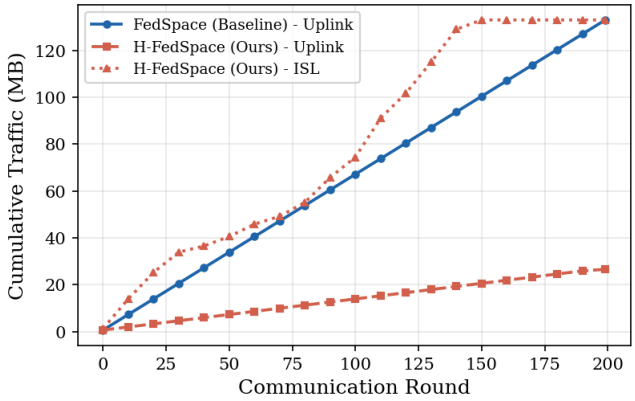

Figure 4. Traffic decomposition by method

H-FedSpace reduces logged uplink by shifting a large share of coordination to ISL aggregation.

4.3. Staleness, robustness, and connectivity context

Figure 5 shows a clear separation in recorded staleness: FedSpace exhibits high recorded staleness, while H-FedSpace remains at zero at evaluation checkpoints. Even so, FedSpace still reaches final accuracy 1.0000. This indicates that the present feature benchmark and model are highly tolerant to stale updates in terms of final accuracy,although loss remains worse than in H-FedSpace.

DOI: 10.54254/2755-2721/2026.GU35238

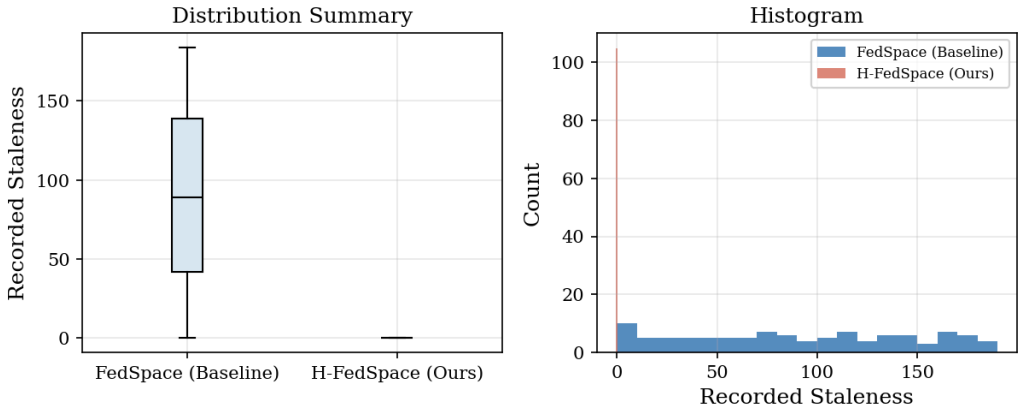

Figure 5. Recorded staleness by method

In the current implementation, the logged staleness variable remains at zero for H-FedSpace checkpoints under the interval-controlled upload schedule.

Figure 6 provides the seed-level comparison. All five seeds end with lower final loss and lower uplink under H-FedSpace. The directional consistency across seeds renders the communication trade-off robust within the present implementation, though it does not eliminate the training-budget caveat discussed earlier.

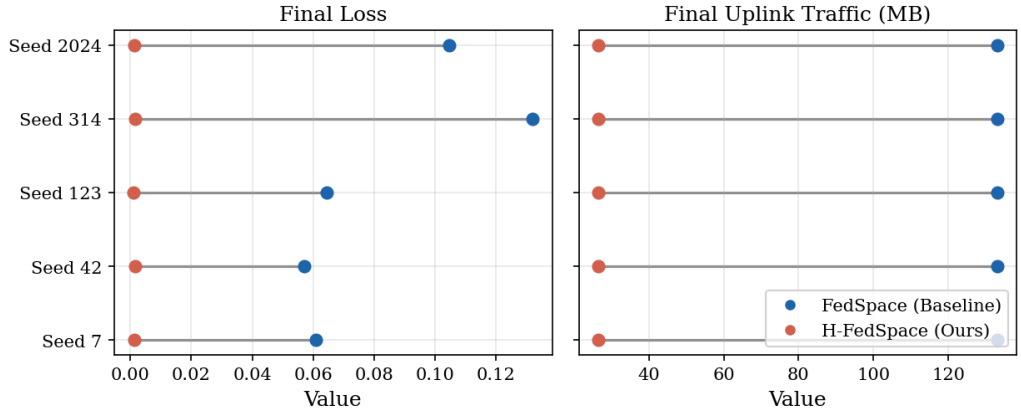

Figure 6. Per-seed final outcomes

Each dumbbell connects the baseline and hierarchical result for the same seed.

Figure 7 summarizes the connectivity context used in the experiment. Ground-station windows are sparse and uneven, whereas several ISL pairs remain repeatedly available over time. Under these caching conditions, 33 of the 40 logged H-FedSpace upload rounds used the interval-driven fallback path rather than a contemporaneous ground-station contact. The empirical interpretation should therefore remain implementation-bound, and the observed loss advantage may still be influenced by the larger amount of local training performed in H-FedSpace.

DOI: 10.54254/2755-2721/2026.GU35238

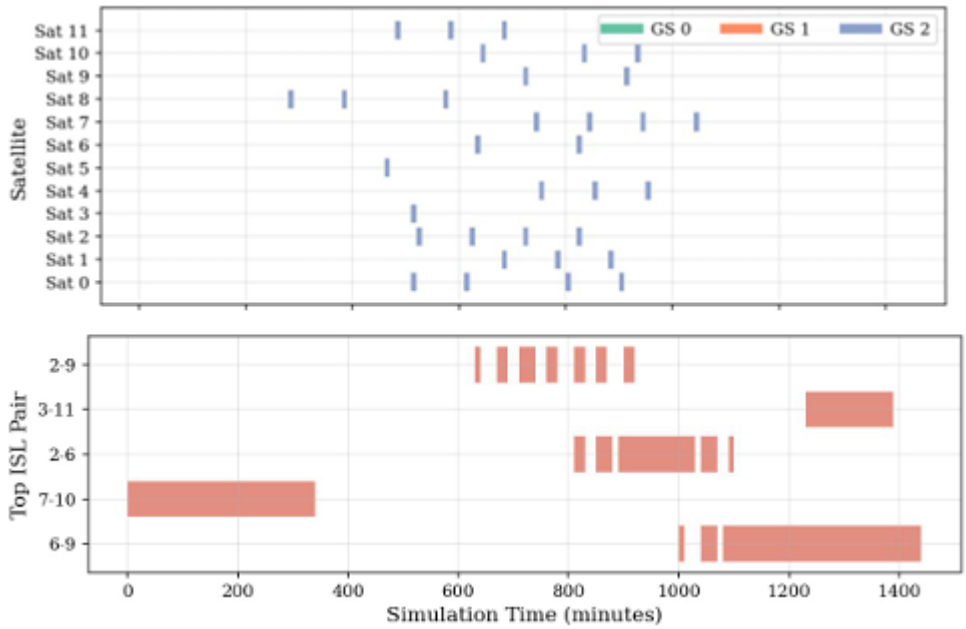

Figure 7. (a)(b). Cached connectivity context used by the experiments

The upper panel shows ground-station visibility intervals per satellite, and the lower panel shows the most active ISL pairs over simulation time.

5. Discussion

5.1. Deeper interpretation of the results

The results point to a specific communication trade-off rather than a general win for hierarchy. In the present implementation, H-FedSpace substantially reduces satellite-to-ground uplink and recorded staleness, but it does so by increasing in-orbit coordination and by training many more local models per round. For that reason, the final loss improvement from 0.0838 to 0.0014 should not be interpreted as evidence that hierarchy alone is responsible for better optimization. A more defensible interpretation is that the current H-FedSpace pipeline combines two advantages: reduced reliance on repeated uplink transmissions and substantially greater per-round training exposure.

The staleness result also deserves a narrow reading. FedSpace reaches a mean recorded staleness of 90.75 rounds yet still attains final accuracy 1.0000. This suggests that the benchmark used here is not highly sensitive to staleness at the level of final classification accuracy. In this workload, loss, traffic decomposition, and staleness are therefore more informative than final accuracy alone.

5.2. Practical implications

The practical implication is conditional. H-FedSpace is appealing in deployments where ground contact is scarce, uplink is the dominant bottleneck, and ISL capacity is relatively easy to allocate. In that operating regime, replacing frequent uploads with cluster-level coordination is sensible even if total communication increases. By contrast, if ISL bandwidth, energy, or intra-constellation coordination time becomes the dominant constraint, the attractiveness of H-FedSpace would narrow considerably because the method shifts rather than removes communication cost.

DOI: 10.54254/2755-2721/2026.GU35238

5.3. Limitations

This study remains bounded by three main limitations. First, the experiments use a cached feature benchmark rather than raw satellite imagery, so the absolute performance levels should not be generalized to more difficult perception tasks. Second, the constellation contains only 12 satellites, which is sufficient to reveal the uplink-ISL trade-off but not enough to represent larger orbital systems. Third, the runtime values are local implementation measurements on one machine and should not be read as deployment latency in orbit. In addition, the unequal local training volume between FedSpace and H-FedSpace limits the extent to which the loss comparison can be attributed to aggregation design alone.

6. Conclusion

This paper examines whether hierarchical coordination is beneficial in an uplink-constrained LEO federated learning setting by comparing FedSpace and H-FedSpace under a shared implementation, a common dataset, and a unified traffic accounting scheme. The evidence shows a clear and consistent communication trade-off. H-FedSpace preserves final accuracy at 1.0000 while reducing logged uplink traffic by 80.0%, from 133.03 MB to 26.61 MB. It also lowers final loss from 0.0838 to 0.0014 and records zero staleness at evaluation checkpoints. At the same time, these gains are not free: the method adds 133.03 MB of ISL traffic, raises total logged communication to 159.63 MB, and increases local runtime.

The central conclusion is therefore conditional rather than absolute. H-FedSpace is supported by the present results when satellite-to-ground uplink is the primary system bottleneck and added ISL coordination is acceptable.The method is not supported as a universal communication-saving strategy, because total communication increases, and part of the optimization advantage may be explained by the fact that H-FedSpace trains all 12 satellites in each round, whereas FedSpace trains only one.

Several limitations also affect the scope of the findings. The benchmark uses pre-extracted features rather than raw imagery, the constellation scale is modest, and wall-clock runtime on a local machine does not represent in-orbit deployment delay. Future work should therefore equalize local training budget across methods, test stricter upload rules that require active ground-station contact, and extend the evaluation to larger constellations and more realistic sensing datasets. Under those conditions, the present study still provides a useful conclusion: hierarchical coordination is most persuasive here as a targeted uplink-saving strategy, rather than as a blanket improvement over all other communication objectives.

References

[1] Peter Kairouz et al. Advances and Open Problems in Federated Learning. Foundations and Trends in Machine Learning, 14(1-2): 1-210, 2021. DOI: https: //doi.org/10.1561/2200000083.

[2] Lumin Liu, Jun Zhang, Shenghui Song, and Khaled B. Letaief. Hierarchical Federated Learning with Quantization: Convergence Analysis and System Design. IEEE Transactions on Wireless Communications, 23(1): 920-935, 2024. DOI: https: //doi.org/10.1109/TWC.2023.3288145.

[3] Jingyang Zhu, Yuanming Shi, Yong Zhou, Chunxiao Jiang, Wei Chen, and Khaled B. Letaief. Over-the-Air Federated Learning and Optimization. IEEE Communications Surveys & Tutorials, 26(2): 1384-1413, 2024. DOI: https: //doi.org/10.1109/COMST.2024.3352875.

[4] Nasrin Razmi, Bho Matthiesen, Armin Dekorsy, and Petar Popovski. On-Board Federated Learning for Dense LEO Constellations. In IEEE Global Communications Conference (GLOBECOM), pages 4084-4089, 2022. DOI: https: //doi.org/10.1109/GLOBECOM48099.2022.10001661.

DOI: 10.54254/2755-2721/2026.GU35238

[5] Jinhyun So, Kevin Hsieh, Behnaz Arzani, Shadi A. Noghabi, Salman Avestimehr, and Ranveer Chandra. FedSpace: An Efficient Federated Learning Framework at Satellites and Ground Stations. CoRR, abs/2202.01267, 2022. DOI: https: //doi.org/10.48550/arXiv.2202.01267.

[6] Zheng Lin, Zhe Chen, Zihan Fang, Xianhao Chen, Xiong Wang, and Yue Gao. FedSN: A Federated Learning Framework over Heterogeneous LEO Satellite Networks. IEEE Transactions on Mobile Computing, early access, 2024. DOI: https: //doi.org/10.1109/TMC.2024.3487878.

[7] Yuanming Shi, Li Zeng, Jingyang Zhu, Yong Zhou, Chunxiao Jiang, and Khaled B. Letaief. Satellite Federated Edge Learning: Architecture Design and Convergence Analysis. IEEE Transactions on Wireless Communications, 23(10): 15212-15229, 2024. DOI: https: //doi.org/10.1109/TWC.2024.3427377.

[8] Mohammad Reza Jabbarpour, Bahman Javadi, P. H. W. Leong, Rodrigo N. Calheiros, and David Boland. FedOrbit: Energy Efficient Federated Learning for Orbital Edge Computing Using Block Minifloat Arithmetic. IEEE Transactions on Services Computing, 17(6): 3657-3671, 2024. DOI: https: //doi.org/10.1109/TSC.2024.3478768.