Uplink-Aware Federated Learning Based on Model Pruning in Satellite Networks

Chenyu Xu [1] **Yijie Mao** [1] **Xiong Wang** [2] **Jingjing Zhang** [2] **Yuanming Shi** [1]

Abstract

Satellite federated learning (SFL) allows satellites to collaboratively train models without sharing raw data, enhancing privacy and reducing communication costs. Traditional SFL requires a ground station (GS) to upload models to satellites, under the premise of adequate ground-satellite uplink (GSUL) resources. However, this assumption does not hold in dense LEO constellations, where frequent command interaction or parameter delivery make the bandwidth-constrained uplink a bottleneck. This work proposes satellite federated learning with uplink scheduling and model pruning (FedLSMP). The key idea behind this is jointly optimizing the GSUL bandwidth allocation plan and model compression ratio to maximize the approximated loss reduction, adhering to bandwidth constraints. Finally, numerical results demonstrate that FedLSMP improves convergence rates while reducing GSUL bandwidth usage, achieving higher overall effectiveness compared with conventional SFL approaches.

1. Introduction

Low earth orbit (LEO) satellites have emerged as a promising approach for providing worldwide communication and remote sensing service (Shayea et al., 2024). However, as satellite constellations grow denser and image resolution increases, transmitting images collected at the LEO satellites back to the ground station (GS) becomes increasingly challenging, especially for advanced applications such as real-time disaster monitoring, autonomous navigation, and military surveillance, which require instantaneous data processing and decision-making (Tao et al., 2024a). To address this issue, federated learning (FL) has been introduced in satellite networks, known as satellite federated learning (SFL) (Chen et al., 2022; Tao et al., 2024b), to enable onboard neural network training for image classification. Instead of transmitting massive raw data, SFL only conveys model updates between the satellites and the GS, thereby enhancing data privacy and mitigating communication overhead.

However, SFL in LEO differs significantly from traditional terrestrial FL due to several unique challenges. Unlike stable, well-connected terrestrial networks, LEO satellite networks exhibit dynamic topologies with intermittent connectivity caused by continuous orbital movement (Tao et al., 2023). Additionally, energy constraints are more pronounced in satellites due to their reliance on limited onboard power sources, making efficient computation and communication crucial (Wu et al., 2022). Moreover, the ever-changing space environment, characterized by fluctuating link availability and transmission delays, further complicates the stability and efficiency of FL in satellite networks. One of the most significant bottlenecks in SFL is the **ground-satellite uplink (GSUL)** (Lin et al., 2025), which is responsible for crucial model uploading from the GS to satellites. Two main factors contribute to the growing challenges posed by GSUL:

- **Denser constellations**: The increasing number of satellites in the constellation intensifies competition for available GSUL bandwidth, while the total transmission rate of GSUL is approximately 10 times lower than the ground-satellite downlink (GSDL) (Mohan et al., 2024). Inefficient GSUL resource allocation can result in delayed or even failed model transmissions, severely affecting SFL performance. Moreover, GSUL resources are constrained by the satellite's visibility to the ground station, which is very sparse and irregular in LEO satellite constellations (Tao et al., 2023).
- **Increasing network demands**: As data volume and precision requirements increase, larger neural network models are necessary to handle the processing tasks. Effectively transmitting these models requires more substantial GSUL resources. Additionally, GSUL bandwidth is also consumed by other essential tasks, such as data transmission and control signal transmission, further complicating the scheduling and allocation of GSUL bandwidth. As constellation density and model size grow, bandwidth limitations in GSUL

[1]School of Information Science and Technology, ShanghaiTech University, Shanghai, China. [2]Fudan University, Shanghai, China. Correspondence to: Yijie Mao <maoyj@shanghaitech.edu,cn>.

ICML2025 Workshop on Machine Learning for Wireless Communication and Networks, Vancouver, Canada. 2025. Copyright 2025 by the author(s).

become increasingly problematic, posing significant challenges to the scalability of SFL.

Unfortunately, these GSUL issues have not received sufficient attention in existing SFL schemes. In (Razmi et al., 2022b), the impact of GSUL on SFL was first mentioned but not studied in-depth. Razmi et al. (2022a) proposed a predictable scheduling method, but it assumes the same transmission rate on GSUL and GSDL, which is not practical, as mentioned earlier (Mohan et al., 2024). Other studies like FedSN (Lin et al., 2025) treat GSUL simply as a part of the budget without considering how to schedule it more effectively. Razmi et al. (2024) and Elmahallawy et al. (2024) respectively introduce inter-satellite links (ISLs) and high-altitude platforms (HAPs) to alleviate the data transmission pressure on GSUL, which may raise additional concerns regarding technological challenges, such as laser tracking and hardware consumption. Furthermore, FedMega proposed by Shi et al. (2024) considers SFL with multiple GS, yet it still assumes the GSUL and GSDL as a symmetric process.

To address the limitations of existing research and the challenges posed by GSUL, this work seeks to answer two fundamental questions: "*Is it necessary to upload the entire model to the satellites in SFL?*" and "*How should GSUL bandwidth be scheduled?*".

Our main contributions are summarized as follows:

1. The primary novelty of this work lies in a novel asynchronous SFL framework, called satellite federated learning with link scheduling and model pruning (FedLSMP). This framework enables bandwidth allocation and neural network model pruning (hereafter referred to as model pruning) to address the aforementioned GSUL problems in LEO satellite constellations. Specifically, we co-optimize the bandwidth allocation plan and model compression ratio to maximize the approximated loss function reduction, subject to bandwidth and link budget constraints. To the best of our knowledge, this is the first work to combine bandwidth allocation with model pruning in SFL.

2. To further alleviate communication overhead while mitigating performance degradation, we propose a novel pruning method with a novel approach to evaluate the importance of parameters in neural network model. This approach is based on the first-order Taylor expansion of the loss function, revealing that parameters with smaller gradient norm have less ability to reduce the loss and should be pruned with higher priority and vice versa. The importance model directs the pruning process and is updated in each communication round based on the updates from the satellites.

3. We evaluate FedLSMP based on the real-world GSUL resource distribution of Starlink (Mohan et al., 2024). Through extensive experiments using both independently and identically distributed (IID) and non-IID data, the results demonstrate that FedLSMP can save over 70% uplink budget while maintaining comparable performance and achieving higher overall effectiveness and robustness than conventional SFL approaches.

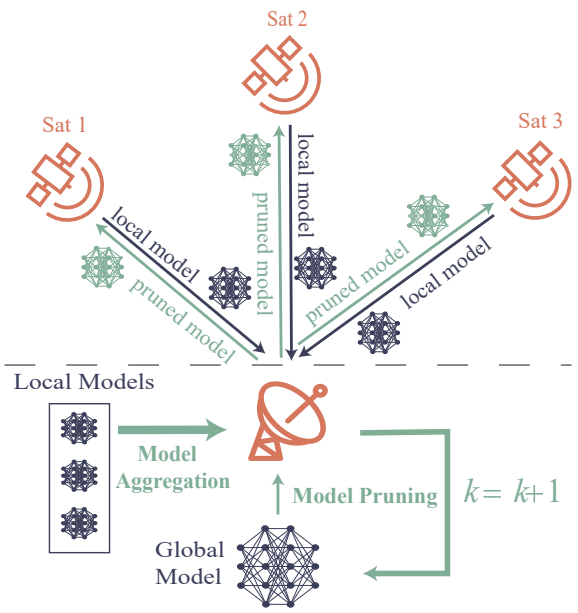


Figure 1. The proposed framework of FedLSMP.

2. System Model

2.1. Workflow of Asynchronous SFL

As shown in Figure 1, we consider a LEO satellite constellation of a single GS and N satellites, indexed by $\mathcal{S} := \{1, 2, ..., N\}$. We assume the satellites are evenly distributed following a Walker Delta constellation pattern as in (Razmi et al., 2024). Additionally, we make the following assumptions at the GS:

1. **Bandwidth allocation**: The GS communicates with multiple LEO satellites simultaneously based on frequency division multiple access (FDMA) (Shi et al., 2024; Hua et al., 2024). This approach enables bandwidth allocation to partition the total available bandwidth among the satellites.

2. **Link estimation**: The GS is capable of predicting the signal-to-noise ratio (SNR) information over a brief future time interval based on the link quality model. The link quality model utilizes the weather information and the predictable feature of LEO satellite constellation

to give an SNR estimation on the ground-satellite link (Tao et al., 2023; Vasisht et al., 2021). This serves as the foundation for bandwidth scheduling.

In SFL, the GS collaborates with LEO satellites to process a FL task aimed at obtaining a global model $\boldsymbol{w}$ that minimizes the global loss function $f(\cdot)$ defined as

$$\min_{\boldsymbol{w}} f(\boldsymbol{w}) := \sum_{i=1}^{N} \rho_i f_i(\boldsymbol{w}), \tag{1}$$

where $\rho_i = \frac{|\mathcal{D}_i|}{|\mathcal{D}|}$ is the weight of satellite i, $|\mathcal{D}_i|$ represents the size of the local dataset at satellite i, $|\mathcal{D}| = \sum_{i=1}^{N} |\mathcal{D}_i|$ represents the size of the global dataset. The local loss function $f_i(\cdot)$ is defined as

$$f_i(\boldsymbol{w}) := \frac{1}{|\mathcal{D}_i|} \sum_{(\boldsymbol{x}_{i,j}, y_{i,j}) \in \mathcal{D}_i} l(\boldsymbol{w}; \boldsymbol{x}_{i,j}, y_{i,j}),$$

where $l(\boldsymbol{w}; \boldsymbol{x}_{i,j}, y_{i,j})$ is the loss function (e.g., cross-entropy) for $\boldsymbol{w}$ evaluated at the j-th data point of satellite i. Here, $\boldsymbol{x}_{i,j}$ denotes the input data and $y_{i,j}$ denotes the corresponding label.

In this work, we adapt asynchronous SFL (ASFL) to obtain the target global model. ASFL is achieved by iterating through the Communication Round (CR). Typically, an arbitrary CR k consists of the following four steps:

Step 1 model upload: The GS uploads the global model $\boldsymbol{w}^k$ through bandwidth allocation to satellites once visible.

Step 2 local training: After receiving model from the GS, each satellite i trains the global model $\boldsymbol{w}^k$ using its local dataset $\mathcal{D}_i$ to obtain a well-trained local model $\boldsymbol{w}_i^k$. Specifically, denote the local gradient computed by satellite i using stochastic gradient descent (SGD) as $\boldsymbol{g}_i(\boldsymbol{w}^k)$, and the local learning rate as η_l. The training process is formulated as

$$\boldsymbol{w}_i^k = \boldsymbol{w}^k - \eta_l \boldsymbol{g}_i(\boldsymbol{w}^k). \tag{2}$$

Step 3 model download: After local training, the satellites download the well-trained model $\boldsymbol{w}_i^k$ back to the GS. However, in ASFL, the global model uploaded to satellite i in CR k is typically not downloaded back in the same CR k, but may be downloaded in a later CR $k + \Delta k_i$ (detailed in Section 2.2). The latency Δk_i is known as *staleness* in asynchronous federated learning (AFL) (Xu et al., 2023), arises from the time misalignment inherent in AFL and is more pronounced in the sparse ASFL system, which means, although model upload, local training, and model download are consecutive stages for satellite i, they are widely separated in time and thus lead to high *staleness* in ASFL. Modified by the *staleness*, a more explicit form of (2) is

$$\boldsymbol{w}_i^{k+\Delta k_i} = \boldsymbol{w}^k - \eta_l \boldsymbol{g}_i(\boldsymbol{w}^k). \tag{3}$$

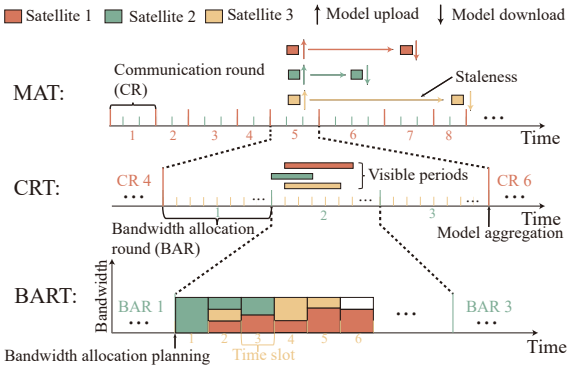


Figure 2. Timescale of ASFL. The model aggregation timeline (MAT) represents the overall timescale of multiple communication round timelines (CRTs), while CRT illustrates a specific CR within the MAT, which is divided by model aggregation. A CRT can be further decomposed into multiple bandwidth allocation round timelines (BARTs) segments that emphasize the bandwidth allocation process in each time slot. The numerical values beneath the time axis correspond to the indices of their respective timelines.

Step 4 model aggregation: Once satellites have downloaded their respective local models, the GS aggregates these well-trained local models to obtain the updated global model $\boldsymbol{w}^{k+1}$. Specifically, we denote the index cluster of the downloaded local models in step 3 as the aggregation pool $\mathcal{Z}^k$. The aggregation timing is determined by the number of elements in the aggregation pool, denoted as $|\mathcal{Z}^k|$, and a pre-defined threshold Z. When $|\mathcal{Z}^k| \geq Z$, the GS will carry out the model aggregation. Using η_g as the global learning rate and $\bar{\rho}_i = \frac{\rho_i}{\sum_{i \in \mathcal{Z}^k} \rho_i}$ as the normalized weight of satellite i, the model aggregation process can be formulated as

$$\boldsymbol{w}^{k+1} = \boldsymbol{w}^k - \eta_g \sum_{i \in \mathcal{Z}^k} \bar{\rho}_i (\boldsymbol{w}^k - \boldsymbol{w}_i^k). \tag{4}$$

Model aggregation signifies the update of CR and the global model. After aggregation, the GS uploads the updated model $\boldsymbol{w}^{k+1}$ to the satellites, and the aforementioned steps are iterated until the global model achieves convergence.

2.2. Time Scale of ASFL

To further clarify the process of ASFL, we can break down CR into the following time scales, as illustrated in Figure 2:

1. **Time slot**: A time slot represents the minimum time unit for the GSUL bandwidth allocation. During each time slot, the bandwidth proportion assigned to each satellite is fixed.

2. **Visible period**: A visible period refers to the time slots during which a LEO satellite can communicate with the GS. Each satellite possesses multiple visible periods, which can be predicted in advance by the GS using two-line elements (Vasisht et al., 2021). Model upload occurs during the relatively short visible period (usually several time slots), while

model training occurs during the much longer non-visible period (typically spanning multiple CRs and thus adequate for conduct local training), and model download occurs in the next visible period. For instance, as illustrated in the MAT part of Figure 2, satellite 3 receives the model from the GS in CR 5, but the next visible period occurs in CR 8. Therefore, the *staleness* of satellite 3 is $\Delta k_3 = 8 - 5 = 3$. Given that GSDL resources are significantly larger than those of GSUL and are usually adequate for model downloading, the time occupied by the GSDL is omitted. Note that, as shown in the CRT part of Figure 2, the visible periods of different satellites are highly likely to overlap, particularly in a dense constellation. When such overlap occurs, the GS must determine the bandwidth allocation for each satellite. However, making this decision is challenging without information about link quality. As previously highlighted, the GS is assumed to predict near-future link quality, with the time frame for the near future defined as a bandwidth allocation round (BAR).

3. **Bandwidth allocation round (BAR)**: A BAR refers to the time interval during which the GS can assess link quality. At the start of each BAR, according to the estimated link quality, the GS generates a bandwidth allocation plan (BAP), which specifies the bandwidth allocation in all time slots of the BAR. Here, we denote the BAP of the t-th BAR using a matrix $\mathbf{\Lambda}_t$, and we further use $\mathbf{\Lambda}_t(i,j)$ to denote the bandwidth proportion allocated to satellite i in the j-th time slot. As illustrated in the BART part of Figure 2, the colored block represents the bandwidth proportion to the dedicated satellite. For example, the bandwidth of CR 5, BAR 2, time slot 3, is distributed evenly to satellites 1 and 2, i.e., $\mathbf{\Lambda}_2(1,3) = \mathbf{\Lambda}_2(2,3) = 0.5$. Due to irregular visibility of the LEO satellites, we assume non-overlapping bandwidth allocation among the satellites in each time slot to prevent inter-satellite interference, leading to the following bandwidth allocation constraints:

$$0 \leq \mathbf{\Lambda}_t(i,j) \leq 1, \forall i \in \mathcal{S}_v^t, j \in \mathcal{L}^t, \tag{5}$$

$$\sum_{i \in \mathcal{S}_v^t} \mathbf{\Lambda}_t(i,j) \leq 1, \forall j \in \mathcal{L}^t, \tag{6}$$

where $\mathcal{S}_v^t$ represents the set of all visible satellites during the t-th BAR and $\mathcal{L}^t = \{1, ..., L^t\}$ denotes the set of all time slots within the t-th BAR. A satellite is visible in a BAR if one of its visible periods lies in the BAR. Note that the BAP is determined solely by the link quality of the specific BAR and is uncorrelated with other BARs. In the subsequent section, we will focus on the BAP design for each individual BAR. For simplicity, the BAR index t in $\mathbf{\Lambda}_t$ and $\mathcal{S}_v^t$ will be omitted in the following discussions.

Once the BAP is determined at the beginning of each BAR, the upload capacity of satellite i in a time slot within that BAR can be calculated as

$$\boldsymbol{C}_{i,j}^{\text{GSUL}} = B\mathbf{\Lambda}(i,j)\log_2(1 + \text{SNR}_{i,j}), \tag{7}$$

where B is the total available GSUL bandwidth at GS and $\text{SNR}_{i,j}$ denotes the SNR of satellite i in j-th time slot.

In the ASFL system, the BAP $\mathbf{\Lambda}$ determines the GSUL bandwidth allocation, which in turn dictates model upload process. Local training and model download step are solely influenced by the characteristics of the LEO satellite constellation and cannot be altered by the GS. Therefore, the primary goal of our proposed FedLSMP is to identify the optimal BAP that maximizes loss reduction in each BAR. Since the BARs are independent, maximizing loss reduction for each BAR ultimately minimizes the overall global loss and accomplishes the SFL task defined in (1).

3. FedLSMP

In this section, we will first quantify mathematically the relationship between model upload and loss reduction. Then, we will introduce a pruning method to reduce the model size with minimal loss reduction cost, and formulate the overall resource allocation problem. Finally, we introduce quadratic fitting to solve the problem.

3.1. Loss Reduction

We begin with a simple scenario that in an arbitrary BAR within an arbitrary CR k, the GS uploads the global model $\boldsymbol{w}^k$ to a single satellite i without any pruning. The well-trained model $\boldsymbol{w}_i^{k+\Delta k_i}$ will be sent back to the GS in CR $k+\Delta k_i$. Δk_i is the *staleness* defined in Section 2. The global loss changed by aggregating $\boldsymbol{w}_i^{k+\Delta k_i}$ can be approximated using a first-order Taylor expansion commonly employed in the literature (Jiang et al., 2022; Lee et al., 2019) as

$$f(\boldsymbol{w}^{k+\Delta k_i+1}) \approx f(\boldsymbol{w}^{k+\Delta k_i}) + \left\langle \nabla f(\boldsymbol{w}^{k+\Delta k_i}), \Delta \boldsymbol{w} \right\rangle, \tag{8}$$

where $\langle \cdot, \cdot \rangle$ denotes inner product and $\Delta \boldsymbol{w} = \boldsymbol{w}^{k+\Delta k_i+1} - \boldsymbol{w}^{k+\Delta k_i}$ refers to the model difference caused by aggregation. Substituting $k + \Delta k_i$ into (4) we have

$$\Delta \boldsymbol{w} = -\eta_g \bar{\rho}_i (\boldsymbol{w}^{k+\Delta k_i} - \boldsymbol{w}_i^{k+\Delta k_i}) \tag{9}$$

$$= -\eta_g \bar{\rho}_i (\boldsymbol{w}^{k+\Delta k_i} - \boldsymbol{w}^k + \eta_l \boldsymbol{g}_i(\boldsymbol{w}^k)), \tag{10}$$

where (10) is obtained by substituting (3) into (9). Then, substituting (10) back to (8), we obtain the loss reduction caused by aggregating the model $\boldsymbol{w}_i^{k+\Delta k_i}$ from satellite i as

$$\begin{aligned}\Delta f_i =& f(\boldsymbol{w}^{k+\Delta k_i}) - f(\boldsymbol{w}^{k+\Delta k_i+1}) \\ =& \bar{\rho}_i (\eta_g \eta_l \langle \nabla f(\boldsymbol{w}^k), \boldsymbol{g}_i(\boldsymbol{w}^k) \rangle + n^i),\end{aligned} \tag{11}$$

where $n^i = n_1^i + n_2^i$ is the *staleness* noise defined by

$$n_1^i = \eta_g \langle \nabla f(\boldsymbol{w}^{k+\Delta k_i}), \boldsymbol{w}^{k+\Delta k_i} - \boldsymbol{w}^k \rangle, \tag{12}$$

$$n_2^i = \eta_g \eta_l \langle \nabla f(\boldsymbol{w}^{k+\Delta k_i}) - \nabla f(\boldsymbol{w}^k), \boldsymbol{g}_i(\boldsymbol{w}^k) \rangle. \tag{13}$$

As illustrated in (12) and (13), the primary source of *staleness* noise n^i in (11) can be attributed to the *staleness* Δk_i. Moreover, due to the lack of knowledge for future model $\boldsymbol{w}^{k+\Delta k_i}$, the effect of the *staleness* noise is unpredictable and uncontrollable. However, in AFL, an empirical consensus was that larger *staleness* usually introduces larger noise and leads to a worse performance (Xu et al., 2023). Here, we provide a novel theoretical perspective on this consensus in ASFL. (12) and (13) quantitatively characterize the impact of *staleness* on loss reduction. Notably, when $\Delta k_i = 0$ (which represents synchronous federated learning), both (12) and (13) reduce to zero, meaning that $n_1^i = n_2^i = 0$. In contrast, a larger value of Δk_i usually results in a more significant shift in $\nabla f(\boldsymbol{w}^{k+\Delta k_i})$ and $\boldsymbol{w}^{k+\Delta k_i}$, thereby creating a higher noise on (11).

To alleviate the negative effect of *staleness*, we modify the weight in (4) to reduce the reliance of model update on those satellites with high *staleness* as

$$\rho_i' = \frac{1}{1+\Delta k_i}\rho_i, \quad \bar{\rho}_i' = \frac{\rho_i'}{\sum_{i\in\mathcal{Z}^k}\rho_i'}.$$

This design is commonly used in terrestrial AFL (Xu et al., 2023). The key difference between terrestrial AFL and FedLSMP is that in FedLSMP, ρ_i' can be predicted in the model upload step in CR k, thanks to the predictable nature of LEO satellite orbits. This enables us to schedule the SFL from the model upload step, whereas in terrestrial scenarios, *staleness* can only be determined in CR $k+\Delta k_i$ and thus only modified in the model aggregation step.

Next, we consider a more complex scenario, where the GS sends the global model to all visible satellites in $\mathcal{S}_v$ in the BAR, following the derivation in (11) for a single satellite case, we can derive the total loss reduction in this BAR as

$$\Delta f = \sum_{i\in\mathcal{S}_v} \rho_i' \left(\eta_g\eta_l \left\langle \nabla f(\boldsymbol{w}^k), \boldsymbol{g}_i(\boldsymbol{w}^k)\right\rangle + n_i\right). \tag{14}$$

By using stochastic gradient to approximate the actual gradient (Jiang et al., 2022), i.e., $\mathbb{E}[\boldsymbol{g}_i(\boldsymbol{w}^k)] \approx \nabla f(\boldsymbol{w}^k)$, we can approximate the loss reduction as

$$\Delta f \approx \sum_{i\in\mathcal{S}_v} (\eta_g\eta_l\rho_i'\|\nabla f(\boldsymbol{w}^k)\|^2 + \rho_i' n_i). \tag{15}$$

(15) highlights the correlation between the uploaded model $\boldsymbol{w}^k$ and the resulting loss reduction. By the definition of the vector L2-norm, let $\nabla f(w_m^k)$ represents the m-th element of $\nabla f(\boldsymbol{w}^k)$, and M represents the total number of parameters in the global model $\boldsymbol{w}^k$. We can further express it as

$$\Delta f \approx \eta_g\eta_l \sum_{i\in\mathcal{S}_v} (\rho_i' \sum_{m=1}^{M} |\nabla f(w_m^k)|^2). \tag{16}$$

In (16), the additive noise term $\rho_i' n_i$ in (15) is omitted because its unpredictable and uncontrollable nature renders it uncorrelated with the optimal BAP generation. (16) illustrates that parameters with higher $|\nabla f(w_m^k)|^2$ result in greater loss reduction. This further reveals that parameters consuming identical GSUL resources during upload may have vastly different contributions to loss reduction. This inspires us to consider the key trade-off: pruning parameters with small gradient norms can preserve model convergence (as they minimally impact loss reduction) while substantially reducing GSUL bandwidth requirements.

3.2. Model Pruning

(16) indicates that pruning parameters in descending order of their gradient norms minimally impacts loss reduction[1]. Define the compression ratio for satellite i, represented by β_i, as the ratio of the preserved number of parameters to the original number of parameters. We introduce Q_k function to quantify how much loss reduction ability a model can withhold when it is pruned at compression ratio β_i as

$$Q_k(\beta_i) = \sum_{m=1}^{\lfloor \beta_i M\rfloor} |\nabla f(w_m^k)|^2. \tag{17}$$

Here, $\lfloor\cdot\rfloor$ denotes the floor function as the number of uploaded parameter must be an integer, and the index m follows the descending order of $|\nabla f(w_m^k)|^2$. When $\beta_i = 1$, the model is unpruned and has the maximum ability to reduce loss. Conversely, when $\beta_i = 0$, the model is completely pruned and becomes meaningless in terms of loss reduction.

However, using (17) to approximate the loss reduction is non-trivial due to the difficulty of obtaining the true value of $|\nabla f(w_m^k)|^2$. Therefore, we introduce a dynamic importance model to estimate the true value of $|\nabla f(w_m^k)|^2$. In SGD, the gradient norm $|\nabla f(w_m^k)|$ indicates the update step size of w_m^k. Meanwhile, from (4) we can observe that $\boldsymbol{w}^k - \boldsymbol{w}_i^k$ also contains the parameter's update information for w_m^k. This observation inspires us to use the aggregation difference $|\boldsymbol{w}_m^k - \boldsymbol{w}_{i,m}^k|$ as an estimation of $|\nabla f(w_m^k)|$. However, due to the asynchronous nature of ASFL, in each CR, the aggregation only involves local models from a subset of satellites. Using the aggregation difference directly might lead to a biased estimation. Hence, the proposed importance model is adaptively updated by leveraging the models from all past aggregations for a more balanced estimation.

Specifically, the importance model is represented as $\boldsymbol{q}^k = [q_1^k, \ldots, q_M^k]^T$ with each element q_m^k representing the importance of the parameter w_m^k and serves as an estimation of $|\nabla f(w_m^k)|^2$. The importance of each parameter is initialized by $q_m^1 = |w_m^1|$ and updated by a weighted sum of the

[1]Note that this approximation is derived based on the first-order Taylor expansion of the loss function. Alternative approximation approaches, such as second-order Taylor expansion (Liu et al., 2021) or saliency-based methods (Zhao et al., 2019), worth further investigation in our future work.

importance from CR k and the averaged difference. Using $\xi \in [0,1]$ to denote the update weight, we have:

$$q_m^{k+1} = \xi q_m^k + (1-\xi) \sum_{i \in \mathcal{Z}^k} \bar{\rho}_i' |w_m^k - w_{i,m}^k|^2. \tag{18}$$

This importance model balances the contributions from past and recent aggregations. All satellites can influence the importance of parameter w_m^k through the term ξq_m^k as long as they have participated in the past aggregations. This mechanism ensures that the importance of each parameter is not solely determined by the most recent aggregation but also incorporates historical information, enhancing the robustness of the importance and producing a more unbiased estimation of parameter w_m^k's importance. Specifically, a larger value of ξ yields a more robust yet less adaptive importance model, which is suitable for highly dynamic constellations. Conversely, a smaller ξ results in a more adaptive but potentially biased one, better suited for relatively stable constellations. Jiang et al. (2022) presented a similar model, but it only applies to synchronous FL, whereas our model can adapt to AFL. Moreover, this method is computationally efficient as the term $w_m^k - w_{i,m}^k$ is already calculated during the model aggregation step as shown in (4).

By adopting the importance model, the estimated Q function, denoted as $\bar{Q}_k$, can be formulated as $\bar{Q}_k(\beta_i) = \sum_{m=1}^{\lfloor \beta_i M \rfloor} q_m^k$. The index m follows the descending order of importance model $\boldsymbol{q}^k$. By substituting $\bar{Q}_k(\beta_i)$ to (16), we approx the loss reduction made by the pruned models as

$$\Delta f(\boldsymbol{\beta}) \approx \eta_g \eta_l \sum_{i \in \mathcal{S}_v} \rho_i' \bar{Q}_k(\beta_i), \tag{19}$$

where $\boldsymbol{\beta} = \{\beta_i | \forall i \in \mathcal{S}_v\}$ represents the compression ratio of all visible satellites in $\mathcal{S}_v$.

The pruned global models are uploaded to satellites via a bit-map method (Jiang et al., 2022), where the pruned parameter is uploaded as a 1-bit boolean 0 mask to identify the position of pruned parameters, and the retained parameters are uploaded as 32-bit float numbers. Therefore, the size of the pruned global model can be written as

$$H_i^p = 32 \times \beta_i M + 1 \times (1-\beta_i) M. \tag{20}$$

Note that as the true number of uploaded parameters is $\lfloor \beta_i M \rfloor$, H_i^p is slightly larger than the actual size of the pruned model. However, this difference is negligible compared to the overall size of the model especially when M is large and thus be omitted. After receiving the pruned global model, satellite i updates only the retained parameters while keeping the pruned parameters unchanged, this approach can alleviate the model drift caused by pruning compared to directly replacing pruned parameters with a value 0. The combined model is used for local training, and the trained model will be downloaded back for model aggregation and importance model update as in (4) and (18).

In summary, at the beginning of a BAR, the GS estimates the SNR for that BAR and generates a BAP. Once the BAP is established, the upload capacity of satellite i in each time slot can be calculated using (7). The total capacity is determined by accumulating capacity over the BAR as

$$H_i^u = \sum_{j=1}^{L} B \boldsymbol{\Lambda}(i,j) \log_2(1 + \mathrm{SNR}_{i,j}), \tag{21}$$

where L denotes the total number of time slots in the current BAR. Then, model pruning can be implemented based on the model size H_i^p and the importance model $\boldsymbol{q}^k$. The problem of BAP generation is formulated to maximize the approximated loss reduction as

$$\mathcal{P}_1: \quad \max_{\boldsymbol{\Lambda}, \boldsymbol{\beta}} \quad \sum_{i \in \mathcal{S}_v} \rho_i' \bar{Q}_k(\beta_i), \tag{22a}$$

$$\text{s.t.} \quad (5), (6)$$

$$\beta_i \in (0,1], \forall i \in \mathcal{S}_v, \tag{22b}$$

$$H_i^u \geq H_i^p, \forall i \in \mathcal{S}_v. \tag{22c}$$

Where (5) and (6) represent the bandwidth constraint, while (22b) restricts the compression ratio. (22c) ensures that the size of the pruned model is less than the upload capacity to guarantee successful uploading of the pruned model. Problem $\mathcal{P}_1$ is an NP-hard mixed integer programming optimization problem and is difficult to solve in its original form. Here we propose an efficient heuristic solution:

Problem $\mathcal{P}_1$ is non-convex, primarily due to the discrete operator $\lfloor \cdot \rfloor$ in the function $\bar{Q}_k(\beta_i) = \sum_{m=1}^{\lfloor \beta_i M \rfloor} q_m^k$, in this function, $\bar{Q}_k$ can be viewed as the cumulative sum of a non-negative sequence arranged in descending order, which indicates that $\bar{Q}_k$ is a positive, monotonically increasing function, albeit with a decreasing rate of increase. Motivated by this observation, we choose to approximate the $\bar{Q}_k$ function using a quadratic model given by $\tilde{Q}_k(\beta) = a\beta^2 + b\beta$. Under the constraint $a < 0$ and $b > -2a$, $\tilde{Q}_k(\beta)$ is also a positive, monotonically increasing function, albeit with a decreasing rate of increase for $\beta \in (0,1]$. Parameter a, b are determined through mean squared error (MSE) fitting. Finally, by substituting the $\tilde{Q}_k(\beta)$ back, we can transform the problem into a continuous, quadratic, convex optimization problem, which can then be easily solved using the CVX toolbox (CVX). In simulation, we find that $\tilde{Q}_k$ can approximate the original $\bar{Q}_k$ with MSE less than 0.001.

4. Numerical Results

4.1. Experimental Settings

Constellation: To fully evaluate FedLSMP's performance, we consider a LEO satellite constellation with 10 orbits at

500 km and 10 orbits at 1000 km, each orbit containing 8 evenly distributed satellites. The GS is located in Brazil, with a minimum elevation angle of 10° and a constellation inclination of 60°.

Training Settings: We evaluated the proposed algorithm on the CIFAR-10 (Krizhevsky, 2009) dataset using ResNet-50 model (He et al., 2016), which has $M = 25557032$ ($\approx$ 780 Mb) trainable parameters. Initialized and trained using PyTorch 2.3. The batch size is set to 16, local learning rate η_l is set to 0.01 with a decay rate of 0.998. Each local model is trained for 3 epochs, the aggregation threshold Z is set to 16 with global learning rate $\eta_g = 0.2$, importance update hyperparameter $\xi = 0.8$. The task is trained for both IID and non-IID data distributions. We use a Dirichlet distribution with $\gamma = 0.5$ for non-IID data distribution to generate the data pattern on each satellite. The BAR length is set to 1 hour to guarantee the accuracy of the estimated SNR, while the time slot length is configured to 1 minute to avoid frequent scheduling overhead. The SFL process is simulated for 96 BARs.

GSUL Capacity: From (7) we can see that when B and SNR are fixed, bandwidth allocation equals to capacity allocation. Therefore, we directly model after the GSUL capacity distribution of Starlink in (Mohan et al., 2024) and sample the average capacity of each satellite in each visible period independently from the distribution. To establish a more realistic simulation setting, we employ a variation pattern proposed in (Leyva-Mayorga et al., 2020) to model the instantaneous capacity between the ground station and the satellites in each time slot. This capacity fluctuates around the mean value, with its magnitude being primarily determined by the satellite's elevation angle. The maximum capacity is achieved when the satellite reaches its zenith position. The visibility pattern is generated using STK 11.6 (Analytical Graphics, 2025).

Baselines: We compare FedLSMP with several SFL strategies listed as follows:

- **FedAsyn**: Derived from (Razmi et al., 2022b), FedAsyn greedily allocates the total bandwidth to the satellite with the best GSUL capacity in each time slot to maximize local transmitting performance and does not prune the network. The aggregation threshold is the same as FedLSMP. It can be viewed as FedLSMP without network pruning and link scheduling.
- **FedAvg**: Derived from (McMahan et al., 2017), FedAvg is a classic synchronous federated learning strategy where the GS waits for all satellites to download their models before aggregation and also greedily allocates the total bandwidth to the satellite with the best GSUL capacity in each time slot.
- **FedSPSS**: A baseline named satellite federated learning with simple pruning simple scheduling (FedSPSS) that prunes the global model in the descending order of its absolute value as in (Frankle & Carbin, 2019) at a fixed compression ratio V and the bandwidth allocation strategy is the same as FedAsyn.
- **FedMPSS**: A variant of FedSPSS that replaces the pruning metric from the absolute value to the proposed importance model.

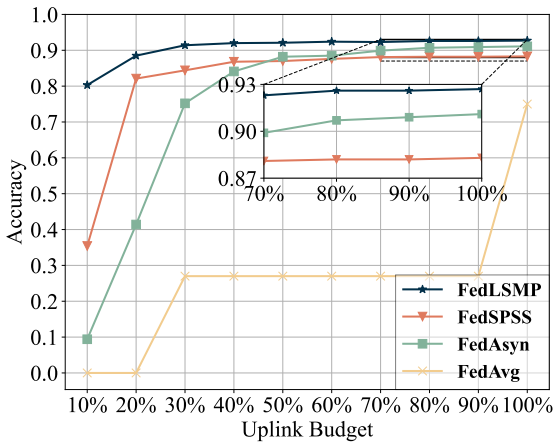
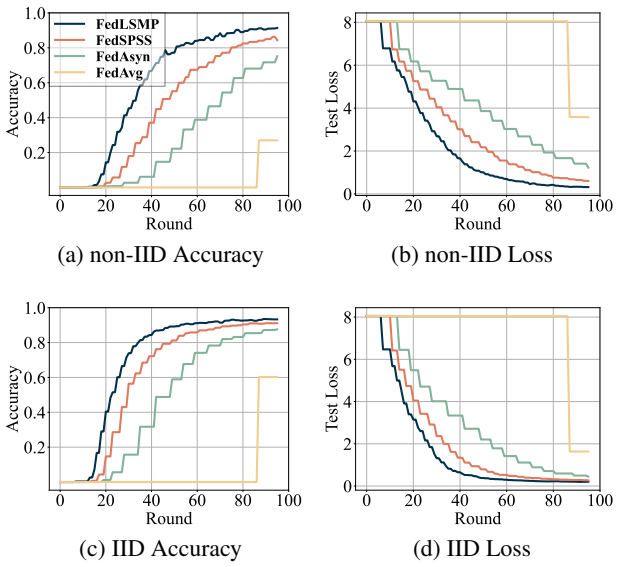


Figure 3. Accuracy of the final BAR versus GSUL bandwidth budget allocated for SFL,$V = 70\%$, non-IID.

4.2. Performance Evaluation

Uplink Budget: To evaluate the influence of the GSUL budget on different methods, we depicted the accuracy of the final BAR (i.e., the 96-th BAR) versus uplink budget in Figure 3. The uplink budget refers to the proportion of bandwidth allocated for SFL. Note that GSUL is also responsible for data transmission (Mohan et al., 2024) and control tasks (Tao et al., 2023) in the LEO satellite constellation, which means utilizing all GSUL bandwidth to serve SFL is impractical. A more GSUL efficient method implies that more GSUL bandwidth can be saved for other tasks.

As shown in Figure 3, FedLSMP achieves higher accuracy than FedAsyn at a 100% uplink budget, yet it requires only 30% of the uplink budget. This indicates that FedLSMP can significantly conserve GSUL bandwidth through efficient scheduling of the upload process. FedAsyn exhibits very low accuracy in low GSUL environments, primarily because uploading the unpruned model in low GSUL budget conditions is challenging, and the greedy upload strategy that maximizes local transmitting performance may not be optimal when viewed globally. FedSPSS shows relatively good performance in a low uplink budget because of pruning. However, its fixed compression ratio means that FedSPSS cannot adjust to a specific GSUL situation, leading to a suboptimal solution under high uplink budgets. Fe-



(a) non-IID Accuracy (b) non-IID Loss

(c) IID Accuracy (d) IID Loss

Figure 4. Accuracy and test loss of global model versus BAR under non-IID and IID data distribution, 30% uplink budget, $V = 70\%$.

dAvg performs poorly in all GSUL environments due to its synchronous feature, which requires waiting for straggler satellites. It cannot complete even one round of aggregation in low uplink budget scenarios. Figure 3 shows that FedLSMP can adapt to different GSUL environments and achieve better performance than baseline methods by optimizing the upload strategy to adjust to the GSUL budget.

Training Process: To further illustrate the training process, we present the global model's test accuracy and loss in the Figure 4. The results demonstrate that FedLSMP performs better and converges faster in both IID and non-IID data distributions. It can be observed that the aggregation frequency of FedAsyn is smaller than that of FedLSMP. This indicates that FedAsyn has fewer opportunities to reach the aggregation threshold due to the unpruned model upload, which significantly delaying the training process. For example, under the non-IID setting, FedLSMP reaches 70% accuracy in only 43 BARs, whereas FedAsyn takes twice the number of BARs to get the same performance. Moreover, while data distribution affects the performance of all methods, FedLSMP exhibits greater robustness to variation in data distribution. For instance, at the 48-th BAR, the accuracy difference between FedLSMP's IID and non-IID data distribution is 12%, while that of FedAsyn is 34%. These results indicate that FedLSMP is more robust to the variation of data distribution than other methods.

Compression Ratio: In the previous simulation, the compression ratio of FedSPSS is fixed at 70%. To further demonstrate the benefits of FedLSMP's adaptive compression ratio

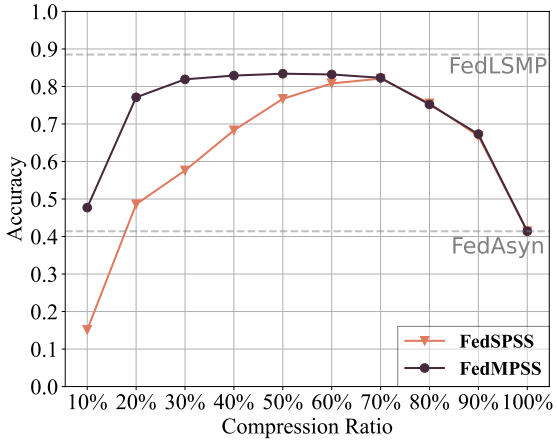


Figure 5. Accuracy of the final BAR versus compression ratio. 20% uplink budget, non-IID.

adjustment, we evaluated the accuracy of the final BAR across different compression ratios, as shown in Figure 5. The results indicate that there exists an optimal compression ratio for FedSPSS, which is around 70% in this case. A compression ratio that is too small may result in an over-pruned model upload, which may not represent the entire model sufficiently. Conversely, a compression ratio that is too large can lead to upload difficulties. FedLSMP can dynamically adjust the compression ratio based on the GSUL budget and satellite state, thereby achieving better performance. Note that the optimal compression ratio is contingent upon the allocated GSUL budget and the model architecture. Finding the best compression ratio is challenging, which underscores the advantage of FedLSMP's adaptive compression ratio adjustment. FedMPSS shows better performance under low compression ratios, indicating that the proposed importance model provides a more accurate measure of parameter's importance than absolute values. However, this performance gain diminishes as the compression ratio increases, primarily because the GSUL budget becomes the dominant limiting factor in high compression ratio scenarios.

Note that although the auxiliary bandwidth optimization and model pruning processes brings additional computation overhead to FedLSMP. These costs are affordable as the operations are performed at the GS, which typically possesses adequate computation resources to solve the simple quadratic optimization problem $\mathcal{P}_1$. Moreover, the time slot and BAR length can be adjusted to change $\mathcal{P}_1$'s scale adaptively to align with the GS's computation capacity.

5. Conclusion

This paper concludes that uploading the entire model to satellites in SFL is unnecessary, as many parameters have minimal impact on loss reduction. We introduce FedLSMP,

which jointly uses uplink scheduling and neural network pruning to address GSUL challenges in SFL. Numerical results show that FedLSMP outperforms existing methods in accuracy, convergence speed, and GSUL efficiency. Future work may integrate FedLSMP with broader applications like ISL-SFL, energy-aware SFL, and multi-ground station SFL to further enhance performance. These integrations could significantly improve the scalability and adaptability of SFL systems in various operational environments.

Acknowledgments

This work has been supported in part by the National Nature Science Foundation of China under Grant 62201347.

Impact Statement

This paper addresses the challenges of GSUL in SFL by introducing FedLSMP. The proposed method achieves improvement on the efficiency of SFL systems, particularly in scenarios with limited GSUL resources. One possible application of FedLSMP is to combine with other SFL techniques (e.g., ISLs or multi-ground stations) to further improve the performance. The impact of this work extends to various fields, including satellite communications, machine learning, and distributed systems, where efficient resource allocation and model training are crucial. The findings of this paper can contribute to the development of more robust and scalable SFL systems, ultimately benefiting a wide range of applications in space-based data processing and analysis.

References

CVX Research, Inc. CVX: Matlab software for disciplined convex programming, version 2.0. https://cvxr.com/cvx, April 2011.

Analytical Graphics, I. systems tool kit (STK), 2025. URL https://www.agi.com. Version 11.6.

Chen, H., Xiao, M., and Pang, Z. Satellite-based computing networks with federated learning. *IEEE Wireless Communications*, 29(1):78–84, 2022.

Elmahallawy, M., Luo, T., and Ramadan, K. Communication-efficient federated learning for LEO constellations integrated with HAPs using hybrid NOMA-OFDM. *IEEE Journal on Selected Areas in Communications*, 42(5):1097–1114, 2024.

Frankle, J. and Carbin, M. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations*, 2019.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

Hua, N., Wang, J., Chen, M., Zhou, K., and Song, H. Simulation research of LDPC coding schemes in satellite communication with SC-FDMA. In *2024 IEEE 100th Vehicular Technology Conference (VTC2024-Fall)*, pp. 1–5, 2024.

Jiang, Y., Wang, S., Valls, V., Ko, B. J., Lee, W.-H., Leung, K. K., and Tassiulas, L. Model pruning enables efficient federated learning on edge devices. *IEEE Transactions on Neural Networks and Learning Systems*, 34(12):10374–10386, 2022.

Krizhevsky, A. Learning multiple layers of features from tiny images. *Tech Report*, 2009.

Lee, N., Ajanthan, T., and Torr, P. SNIP: Single-shot network pruning based on connection sensitivity. In *International Conference on Learning Representations*, 2019.

Leyva-Mayorga, I., Soret, B., Röper, M., Wübben, D., Matthiesen, B., Dekorsy, A., and Popovski, P. Leo small-satellite constellations for 5g and beyond-5g communications. *IEEE Access*, 8:184955–184964, 2020. doi: 10.1109/ACCESS.2020.3029620.

Lin, Z., Chen, Z., Fang, Z., Chen, X., Wang, X., and Gao, Y. FedSN: A federated learning framework over heterogeneous LEO satellite networks. *IEEE Transactions on Mobile Computing*, 24(3):1293–1307, 2025.

Liu, L., Zhang, S., Kuang, Z., Zhou, A., Xue, J.-H., Wang, X., Chen, Y., Yang, W., Liao, Q., and Zhang, W. Group fisher pruning for practical network compression. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 7021–7032. PMLR, 18–24 Jul 2021.

McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pp. 1273–1282. PMLR, 2017.

Mohan, N., Ferguson, A. E., Cech, H., Bose, R., Renatin, P. R., Marina, M. K., and Ott, J. A multifaceted look at starlink performance. In *Proceedings of the ACM Web Conference 2024*, pp. 2723–2734. ACM, 2024.

Razmi, N., Matthiesen, B., Dekorsy, A., and Popovski, P. Scheduling for ground-assisted federated learning in LEO satellite constellations. In *2022 30th European Signal Processing Conference (EUSIPCO)*, pp. 1102–1106, 2022a.

Razmi, N., Matthiesen, B., Dekorsy, A., and Popovski, P. Ground-assisted federated learning in LEO satellite constellations. *IEEE Wireless Communications Letters*, 11 (4):717–721, 2022b.

Razmi, N., Matthiesen, B., Dekorsy, A., and Popovski, P. On-board federated learning for satellite clusters with inter-satellite links. *IEEE Transactions on Communications*, 72(6):3408–3424, 2024.

Shayea, I., El-Saleh, A. A., Ergen, M., Saoud, B., Hartani, R., Turan, D., and Kabbani, A. Integration of 5G, 6G and IoT with Low Earth Orbit (LEO) networks: Opportunity, challenges and future trends. *Results in Engineering*, 23: 102409, September 2024.

Shi, Y., Zeng, L., Zhu, J., Zhou, Y., Jiang, C., and Letaief, K. B. Satellite federated edge learning: Architecture design and convergence analysis. *IEEE Transactions on Wireless Communications*, 23(10):15212–15229, 2024.

Tao, B., Masood, M., Gupta, I., and Vasisht, D. Transmitting, fast and slow: Scheduling satellite traffic through space and time. In *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*, pp. 1–15, 2023.

Tao, B., Chabra, O., Janveja, I., Gupta, I., and Vasisht, D. Known knowns and unknowns: Near-realtime earth observation via query bifurcation in serval. In *21st USENIX Symposium on Networked Systems Design and Implementation (NSDI 24)*, pp. 809–824, April 2024a.

Tao, M., Zhou, Y., Shi, Y., Lu, J., Cui, S., Lu, J., and Letaief, K. B. Federated edge learning for 6g: Foundations, methodologies, and applications. *Proceedings of the IEEE*, pp. 1–39, 2024b.

Vasisht, D., Shenoy, J., and Chandra, R. L2D2: low latency distributed downlink for leo satellites. In *Proceedings of the 2021 ACM SIGCOMM 2021 Conference*, SIGCOMM '21, pp. 151–164, 2021.

Wu, C., Zhu, Y., and Wang, F. DSFL: Decentralized satellite federated learning for energy-aware LEO constellation computing. In *2022 IEEE International Conference on Satellite Computing (Satellite)*, pp. 25–30, 2022.

Xu, C., Qu, Y., Xiang, Y., and Gao, L. Asynchronous federated learning on heterogeneous devices: A survey. *Computer Science Review*, 50:100595, 2023.

Zhao, C., Ni, B., Zhang, J., Zhao, Q., Zhang, W., and Tian, Q. Variational convolutional neural network pruning. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2775–2784, June 2019.