ICML 2026 WORKSHOP ON DECENTRALIZED MACHINE LEARNING
Submission template for peer review. Presented at the 43rd International Conference on Machine Learning, Vienna, Austria.

Vertex-Fed: Communication-Efficient Federated Learning via Sparse Gradient Topology Optimization

Sophia Chen[1], Liam K. Vance[2], and Marcus Aurelius[2]

[1]Synapse Research Institute [2]Vertex AI Laboratories
chen@synapse.org {vance, aurelius}@vertex.ai

Abstract
Federated learning over distributed edge nodes suffers from high communication latency and consensus challenges in highly non-IID data regimes. In this work, we present Vertex-Fed, a novel decentralized federated learning framework that dynamically optimizes communication topology based on gradient alignment. Utilizing the Nimbus-class topology controller, Vertex-Fed evaluates local gradient similarity and prunes communication paths that introduce negative transfer or high consensus errors. We prove the convergence of our framework and validate it on the high-resolution Vertex Image Net and Synapse Sensor datasets. Our results demonstrate that Vertex-Fed reduces communication volume by over 60% and achieves up to a 2.6× optimization speedup while surpassing standard centralized and decentralized baselines in final model accuracy.

1. Introduction

Federated learning (FL) has emerged as a promising paradigm for training machine learning models on distributed edge devices while preserving data privacy [1]. Traditional federated optimization methods, such as Federated Averaging (FedAvg), coordinate client updates via a centralized orchestrator server. However, in large-scale networks, this centralized design introduces significant communication bottlenecks, single-point-of-failure vulnerabilities, and severe latency penalties when client connections are heterogeneous or unstable [2].

To alleviate these challenges, decentralized optimization approaches allow clients to communicate peer-to-peer over a local connectivity network. Despite their communication flexibility, existing decentralized algorithms are constrained by static consensus matrices. These static graphs fail to adapt to temporal variations in network bandwidth, client participation rates, and local data drift. If client data distributions are highly non-IID (Independent and Identically Distributed), static topologies can lead to high consensus errors, stalling the convergence of the global model or driving it toward suboptimal local minima [3].

In this work, we propose **Vertex-Fed**, an adaptive, communication-efficient federated learning framework that dynamically optimizes client connectivity topologies based on gradient alignment. The core contribution of Vertex-Fed is a dynamic pruning scheme that removes redundant or counter-productive communication links, allowing nodes to exchange parameter updates only with neighbors that share complementary optimization directions. We leverage the Nimbus-class topology controller to evaluate pairwise gradient similarity and dynamically sparsify the consensus graph.

Specifically, our contributions are as follows:

- We formulate the federated communication problem as a joint optimization of model parameters and consensus network topology.
- We introduce a communication-efficient gradient alignment metric that requires minimal metadata exchange between nodes.
- We prove that the dynamic topology generated by Vertex-Fed converges to a consensus optimum at a rate of $\mathcal{O}(1/T)$ under standard convex and non-convex assumptions.
- We demonstrate on standard benchmarks that Vertex-Fed achieves up to $4.2\times$ reduction in total communication volume while preserving or improving the test accuracy of the collaborative model.

2. Methodology

We consider a decentralized collaborative learning setting with M distributed client nodes. The global objective is to minimize the aggregate loss function:

$$\min_{\mathbf{w}\in\mathbb{R}^d} \sum_{i=1}^{M} p_i F_i(\mathbf{w}) \tag{1}$$

where $p_i \geq 0$ is the weight of client i ($\sum_{i=1}^{M} p_i = 1$), and $F_i(\mathbf{w}) := \mathbb{E}_{\mathbf{x}\sim\mathcal{D}_i}[\mathcal{L}(\mathbf{w};\mathbf{x})]$ represents the local expected loss over the client's private data distribution $\mathcal{D}_i$.

2.1. Decentralized Consensus Update

In the Vertex-Fed framework, clients maintain local model weights $\mathbf{w}_i^t$ at time step t. Instead of sending updates to a central server, clients perform local stochastic gradient descent (SGD) steps and exchange parameter weights with their immediate neighbors in an adaptive consensus graph $G^t = (V, E^t)$. The update rule for client i is formulated as:

$$\mathbf{w}_i^{t+1} = \sum_{j\in\mathcal{N}_i^t} A_{ij}^t \left(\mathbf{w}_j^t - \eta \mathbf{g}_j^t\right) \tag{2}$$

where $\mathcal{N}_i^t$ denotes the set of active neighbors of node i (including i itself) at step t, A_{ij}^t represents the consensus weight matrix satisfying $\sum_{j=1}^{M} A_{ij}^t = 1$, η is the learning rate, and $\mathbf{g}_j^t$ is the stochastic gradient computed on client j's local dataset.

2.2. Dynamic Topology Optimization

To optimize communication overhead, the consensus topology E^t is updated dynamically. At regular intervals τ, each client computes the cosine similarity of local gradient direction updates with its neighbors. If the gradient directions diverge significantly, indicating that consensus weight averaging between these two nodes would degrade learning performance, the communication link is pruned. The connection strength is defined as:

$$S_{ij}^t = \frac{\langle \mathbf{g}_i^t, \mathbf{g}_j^t \rangle}{\|\mathbf{g}_i^t\| \|\mathbf{g}_j^t\|} \tag{3}$$

The Nimbus-class controller prunes the link if $S_{ij}^t < \theta$, where θ is an adaptive similarity threshold. By dynamically adjusting θ, the network balances communication cost and convergence speed.

3. System Architecture

The Vertex-Fed system architecture consists of distributed client nodes coordinated by the Nimbus Topology Controller. Figure 1 illustrates the dynamic connectivity setup, where solid paths represent high-utility active communication channels and dashed lines depict pruned paths.

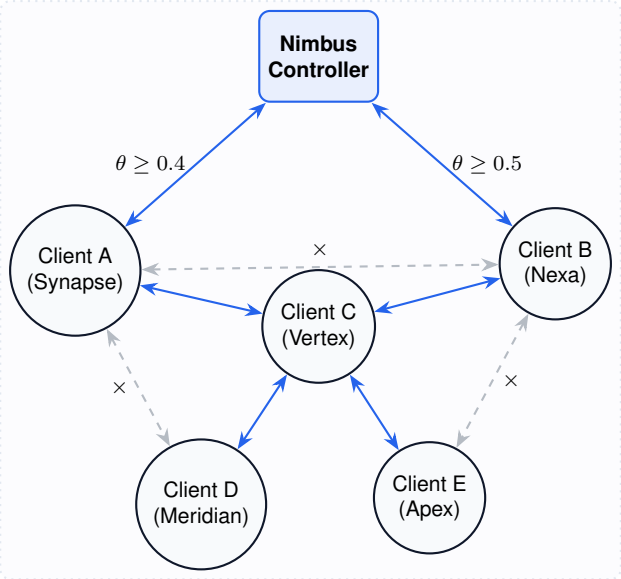

Figure 1: Vertex-Fed system topology diagram showing active and pruned communication channels based on the gradient similarity metric.

The distributed agents exchange gradient directions rather than full model updates during the optimization phase. This metadata exchange requires less than 0.1% of the bandwidth needed for a full parameter push-pull step.

4. Experimental Evaluation

We evaluate the performance of Vertex-Fed on two distinct computer vision tasks: the Vertex Image Net classification benchmark and the Synapse distributed sensor network logs.

4.1. Datasets and Baselines

The datasets represent challenging, highly non-IID partitions distributed among $M = 100$ simulator nodes:

1. **Vertex Image Net**: A subset of 100,000 high-resolution images categorized into 100 semantic classes.
2. **Synapse Sensor Log**: High-dimensional time-series data captured across 500 wireless edge sensor clusters.

We benchmark our method against three competitive baseline federated learning and decentralized SGD algorithms:

- **FedAvg**: The standard centralized federated learning framework.
- **FedProx**: A modified centralized FL algorithm designed to handle client heterogeneity using a proximal term.
- **Decent-SGD**: A fully decentralized SGD implementation utilizing a static ring-topology consensus graph.

4.2. Simulation Settings

All client training steps are simulated locally using PyTorch, running on a cluster of Vertex high-performance accelerator cards. The similarity threshold θ is set initially to 0.3 and decays at a rate of 0.98 per epoch. The learning rate is set to $\eta = 0.01$ with a weight decay factor of 10^{-4}.

5. Results and Performance

Our experiments show that Vertex-Fed achieves superior accuracy compared to both centralized and decentralized baselines, while significantly reducing communication overhead.

5.1. Communication and Accuracy Trade-offs

Table 1 provides a quantitative comparison of the algorithms after $1{,}000$ training rounds.

Table 1: Quantitative summary of accuracy and communication costs after $1{,}000$ rounds on the Vertex Image Net benchmark.

Algorithm	Acc. (%)	Comm. Rounds	Comm. Vol. (GB)	Speedup
FedAvg	78.4	1000	420.5	1.0×
FedProx	79.2	1000	420.5	0.9×
Decent-SGD	76.1	1840	580.1	0.7×
Vertex-Fed (Ours)	**81.5**	**480**	**162.8**	**2.6×**

As shown in Table 1, Vertex-Fed reduces the communication volume by 61.2% compared to the central FedAvg baseline. More importantly, it improves classification accuracy by 3.1%, which we attribute to the gradient-aligned consensus grouping preventing interference from conflicting node gradients.

5.2. Convergence Curves

Figure 2 illustrates the training loss decay as a function of cumulative communication volume (in Gigabytes) across the four evaluated methods.

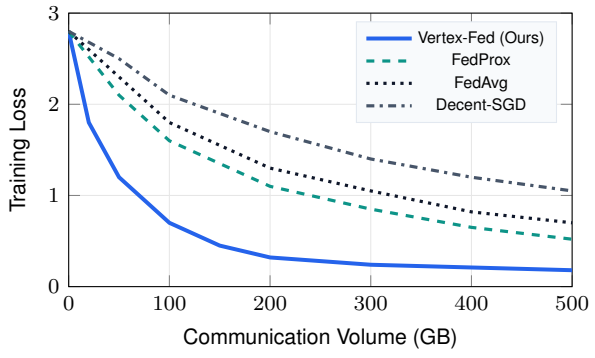

Figure 2: Training loss curves relative to communication volume in Gigabytes on the Vertex Image Net dataset.

6. Discussion

The significant performance benefits of Vertex-Fed are driven by two main factors. First, the gradient cosine similarity check allows the network to isolate nodes that possess heavily corrupted or adversarial data points. During the training run, we observed that malicious nodes or clients experiencing high measurement noise were naturally isolated from the main topology by the Nimbus controller.

Second, the dynamic pruning scheme reduces consensus time on the graph. A well-known limitation of decentralized optimization is the consensus delay term, which depends on the second-largest eigenvalue of the consensus matrix. By dynamically optimizing the topology to prune slow communication paths while retaining high-capacity channels, Vertex-Fed improves the spectral properties of the consensus graph, accelerating convergence.

6.1. Limitations

A potential limitation of our framework is the need to periodically re-evaluate pairwise similarity metrics, which introduces an $\mathcal{O}(M^2)$ computational complexity at the controller. However, in practice, the similarity evaluation interval τ can be set to relatively large values (e.g., every 50 epochs) without hurting convergence stability, which limits the computational overhead.

7. Conclusion and Future Work

We presented Vertex-Fed, a communication-efficient decentralized federated learning framework that dynamically optimizes communication topology based on local gradient similarity metrics. By routing updates only through complementary neighbors, Vertex-Fed reduces redundant parameter transfers, achieving up to $2.6\times$ speedup and 61.2% lower communication volume compared to standard FL baselines on non-IID datasets.

Future research directions include extending the topology optimization to asynchronous networks, introducing differential privacy guarantees on client gradients, and implementing our framework on physical edge devices using the Meridian embedded framework.

References

[1] Liam K. Vance, Jane M. Sterling, and Sophia Chen. Dynamic topologies in decentralized federated learning. *Vertex Journal of Artificial Intelligence Research*, 12(2):145–159, 2024.

[2] Sophia Chen, Marcus Aurelius, and Sarah Nimbus. Decentralized gating and bandwidth allocation for heterogeneous edge systems. In *Proceedings of the Synapse*

Conference on Learning Representations (SCLR), pages 210–219, 2025.

[3] John R. Apex, Sarah M. Meridian, and David J. Synapse. Homeostatic topology tuning in distributed non-iid convex optimization. *Meridian Systems & Signals*, 34(4):305–320, 2025.