Multimodal Online Federated Learning with Modality Missing in Internet of Things

Heqiang Wang, Xiang Liu, Xiaoxiong Zhong, Lixing Chen, Fangming Liu, Weizhe Zhang

Abstract—The Internet of Things (IoT) ecosystem generates vast amounts of multimodal data from heterogeneous sources such as sensors, cameras, and microphones. As edge intelligence continues to evolve, IoT devices have progressed from simple data collection units to nodes capable of executing complex computational tasks. This evolution necessitates the adoption of distributed learning strategies to effectively handle multimodal data in an IoT environment. Furthermore, the real-time nature of data collection and limited local storage on edge devices in IoT call for an online learning paradigm. To address these challenges, we introduce the concept of Multimodal Online Federated Learning (MMO-FL), a novel framework designed for dynamic and decentralized multimodal learning in IoT environments. Building on this framework, we further account for the inherent instability of edge devices, which frequently results in missing modalities during the learning process. We conduct a comprehensive theoretical analysis under both complete and missing modality scenarios, providing insights into the performance degradation caused by missing modalities. To mitigate the impact of modality missing, we propose the Prototypical Modality Mitigation (PMM) algorithm, which leverages prototype learning to effectively compensate for missing modalities. Experimental results on two multimodal datasets further demonstrate the superior performance of PMM compared to benchmarks.

Index terms— Federated Learning, Multimodal Learning, Online Learning, Internet of Thing, Modality Missing.

I. INTRODUCTION

The rapid expansion of the Internet of Things (IoT) [1] has led to an unprecedented surge in data generated by a multitude of interconnected devices, including smart home appliances [2], wearable health monitors [3], and industry sensors [4]. To enable intelligent services and applications across the IoT ecosystem, artificial intelligence techniques, particularly machine learning and deep learning, has become a fundamental tool for model training on large-scale IoT data. Traditionally, such training has been performed in centralized cloud platforms or data centers. However, this centralized paradigm faces significant challenges as both the scale of IoT data and the number of IoT devices continue to expand. Transferring large volumes of raw data to centralized servers imposes significant demands on network bandwidth and leads to substantial communication overhead, rendering it impractical for latency-sensitive applications such as autonomous

driving [5] and real-time healthcare monitoring [6]. Additionally, uploading sensitive user data to the cloud raises serious privacy concerns [7]. With the gradual evolution of IoT devices from mere data collectors to intelligent edge nodes, there is increasing potential to harness their computational capabilities to address the scalability and efficiency challenges of massive IoT deployments. In this context, federated learning (FL) [8] has emerged as a promising distributed learning paradigm. FL enables collaborative model training across devices while keeping raw data local, offering a cost-effective and privacy-preserving alternative to traditional centralized learning. By significantly reducing data transmission and ensuring local data privacy, FL presents a natural and scalable solution for deploying learning tasks in IoT environments.

Conventional FL frameworks for IoT have primarily been designed for unimodal data. However, in practice, IoT environments are inherently multimodal, involving a diverse range of sensors that generate data across multiple modalities [9], such as images from cameras, audio from microphones, and structured text or signals from various sensors. This rich multimodal data provides a more comprehensive and informative representation. To address this reality, multimodal federated learning (MFL) [10] has emerged as a natural extension of FL, aiming to enable collaborative learning across distributed multimodal data sources. A typical approach in MFL involves deploying modality-specific encoder networks, each tailored to a particular data type, to extract meaningful feature representations from high-dimensional raw inputs. These features are then fused and passed through a head encoder, usually composed of deep neural network (DNN) layers followed by a softmax classifier, to produce the final prediction.
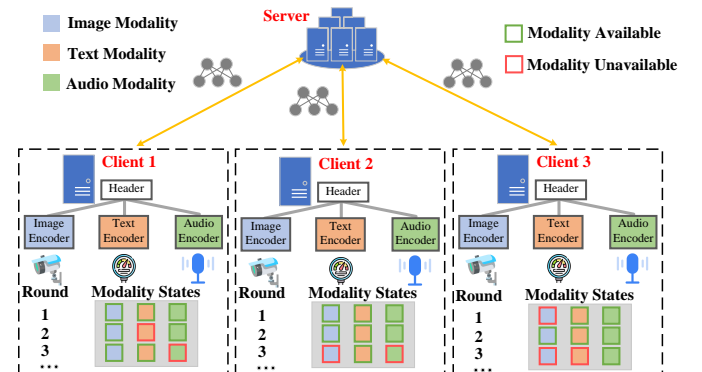

Fig. 1: IoT-Based MMO-FL with Modality Missing

While traditional MFL has primarily focused on offline settings with fixed datasets, real-world IoT applications are

inherently dynamic. IoT devices operate continuously, generating streaming data in real time, which makes online learning a more realistic paradigm. Due to the instability failures of IoT sensors, certain modalities may be missing during data collection, introducing additional challenges to the learning process. Missing modalities can significantly impair performance by disrupting the alignment and fusion of complementary information from different sources. This degradation weakens the model's ability to capture cross-modal correlations and reduces generalization, particularly in dynamic and resource-constrained IoT settings. To address these issues, we introduce the concept of Multimodal Online Federated Learning (MMO-FL) and investigate solutions for handling modality missing in this scenario. An overview of the MMO-FL framework with missing modalities is illustrated in Fig. 1. Given the resource constraints of edge devices in IoT, conventional modality reconstruction techniques, such as those relying on transformers [11] or large pre-trained models [12] are often computationally prohibitive. Therefore, it becomes critical to develop efficient modality mitigation strategies that are compatible with the limited computational capabilities of edge devices and constrained communication capacity of IoT networks, enabling robust and scalable deployment in IoT scenarios. Our primary contributions are summarized as follows:

1) We introduce the concept of MMO-FL, designed for IoT scenarios to address the challenges of FL with multimodal and streaming data. Building upon this foundation, we further investigate the problem of missing modalities, a prevalent issue in IoT environments caused by sensor instability and intermittent data acquisition failures.

2) We provide an in-depth theoretical analysis of MMO-FL, investigating the dynamics of online learning and multimodal learning within the FL framework. In particular, we compare the scenarios with and without missing modalities, highlighting their differences in terms of regret bounds.

3) To address the modality missing problem, we propose the Prototypical Modality Mitigation (PMM) algorithm, specifically designed to handle missing modality data in MMO-FL. Inspired by prototype learning, PMM is tailored to support dynamic data streams by enabling effective construction and substitution of modality-specific prototypes throughout the learning process.

4) We evaluate the proposed PMM algorithm within the MMO-FL framework using two multimodal datasets: UCI-HAR and MVSA-Single. Experimental results demonstrate that PMM consistently outperforms baseline methods in effectively addressing the missing modality problem, achieving superior overall learning performance.

The rest of this paper is organized as follows. Section II reviews related work on Online FL, MFL, and Prototype FL. Section III presents the system model and problem formulation. In Section IV, we describe the workflow of MMO-FL in the presence of missing modalities. Section V provides a theoretical regret analysis of the MMO-FL framework. Section VI details the proposed PMM algorithm. Experimental results

are discussed in Section VII. Finally, Section VIII concludes the paper.

II. RELATED WORK

A. Online Federated Learning

Online learning is designed to process data sequentially and update models incrementally, making it well-suited for applications involving continuously arriving data and the need for real-time model adaptation [13]. These methods offer computational efficiency and eliminate the requirement of having access to the full dataset in advance, rendering them particularly suitable for memory-constrained IoT environments. In the context of FL, online federated learning (OFL) has emerged as a promising paradigm that extends online learning principles to distributed networks of decentralized learners [14]. A distinguishing feature of OFL, compared to traditional offline FL, is its emphasis on minimizing long-term cumulative regret rather than static optimization objectives during local updates. Although OFL remains relatively underexplored, several notable studies have recently advanced the field. For instance, [15] proposes a communication-efficient OFL algorithm that balances reduced communication overhead with strong learning performance. Similarly, [16] introduces FedOMD, an OFL method designed for uncertain environments, capable of handling streaming data without relying on assumptions about loss function distributions. While these works focus primarily on the horizontal federated learning (HFL) setting, [17] explores the vertical federated learning (VFL) context, proposing an online VFL framework tailored to cooperative spectrum sensing and achieving sublinear regret. Further extending to real-world industrial applications, [4] addresses challenges such as noise interference and device heterogeneity in online VFL systems. However, all the aforementioned approaches are limited to unimodal online federated learning. In practice, multimodal data is pervasive in IoT applications, where information from diverse types of sensors must be jointly leveraged. To bridge this gap, this work pioneers the study of MMO-FL, with the goal of enhancing the robustness and applicability of online federated learning in complex, multimodal IoT environments.

B. Multimodal Federated Learning

MFL aims to train task-relevant models on multimodal data distributed across multiple clients, thereby enabling the effective utilization of diverse data sources. With the growing interest in MFL, a variety of algorithms have been proposed to address its unique challenges and improve learning performance. One prominent challenge is modality heterogeneity, where different clients possess access to varying subsets of modalities. This inconsistency complicates model aggregation and hampers effective knowledge sharing. Several studies, such as [18]–[20], have explored strategies for heterogeneous modality fusion to address this issue. Another critical challenge involves optimizing modality selection for training under constrained computational and communication resources. To tackle this, [21] proposes MPriorityFed, an adaptive resource allocation framework that improves computational efficiency by prioritizing modality encoders based on their relevance and

training requirements. Beyond above two challenges, a particularly pressing challenge in MFL is maintaining robust performance in the presence of missing modalities [22]. Missing data can result from incomplete data collection [23], sensor failures [24], or privacy restrictions [22], all of which degrade the effectiveness of conventional MFL frameworks. For example, [25] introduces a cluster-enhanced method that utilizes feature clustering to address missing modalities in brain imaging analysis, while [26] proposes the MFCPL framework, which leverages cross-modal prototypes to enhance knowledge transfer at both modality-shared and modality-specific levels. Despite these advancements, existing approaches are predominantly designed for offline learning scenarios with static datasets. They fail to address the challenges of online learning settings, where data arrives sequentially and models must adapt in real time. To bridge this gap, our work focuses on developing effective modality mitigation algorithms specifically tailored for MMO-FL, aiming to mitigate the impact of missing modalities in dynamic and distributed IoT scenarios.

C. Prototype Federated Learning

Prototype Federated Learning (PFL) has emerged as a promising solution for addressing data heterogeneity and personalization challenges in FL scenario. Unlike conventional FL approaches that rely on aggregating model parameters or gradients, PFL focuses on exchanging class-level prototypes, which are representative feature embeddings of data classes, between clients and the central server. This prototype-sharing strategy enhances communication efficiency and improves model robustness, particularly under non-IID data distributions [27]–[30]. For instance, FedProto [28] proposes a framework where clients compute local class prototypes and send them to the server, which aggregates them into global prototypes and redistributes them. This process helps align local updates with global objectives and alleviates the negative impact of data heterogeneity. Extending this concept, FedGPD [31] incorporates knowledge distillation, using global prototypes as distilled information to guide local training. This method eliminates the dependency on public proxy datasets and enhances generalization across clients with diverse data distributions. Furthermore, FedGMKD [27] advances prototype learning by modeling prototype features using Gaussian Mixture Models and applying discrepancy-aware aggregation, which dynamically weights client contributions based on both data quality and quantity, resulting in improved global performance under heterogeneous conditions. Beyond tackling heterogeneity, recent studies have demonstrated that prototype learning is also effective for addressing modality missing problems, especially in centralized learning scenarios [32], [33] and some distributed learning scenarios [34]. However, existing PFL research is confined to offline learning, with limited exploration of online learning, particularly in OFL contexts. OFL introduces unique challenges such as streaming data and non-stationary distributions, which require new adaptations of prototype-based strategies. Addressing the modality missing problem through prototype learning within the context of MMO-FL remains an open and important direction for future research.

III. SYSTEM MODEL

Before presenting the details of the system model, we provide a summary of the key notations in Table I and the abbreviations used throughout this work in Table II.

TABLE I: Key Notations

Symbol	Semantics
K	The number of clients
M	The number of modalities
T	The number of global rounds
N	The number of data samples collected by client
E	The number of local iterations
C	The number of classes
θ^m	The modality encoder
θ^0	The head encoder
η	The learning rate
Z	The feature extractor
Θ	The overall model
$\mathcal{S}_t$	The set of clients without modality missing
$\mathbf{G}_k^t$	The overall gradient
β_k^t	The proportion of available modalities
$p_c^{t,m}$	The temporal global prototype
$\bar{p}_c^{t,m}$	The consistent global prototype
$\bar{\mathcal{P}}^t$	The consistent global prototype collection
λ	The ratio of modality missing occurs
α	The Non-IID level

Consider an IoT-based smart factory scenario consisting of a cloud server and K workstations, each acting as a client. Each workstation is equipped with a diverse set of sensors to monitor factory conditions across multiple modalities at different locations, including vision sensors, acoustic sensors, and temperature and humidity sensors, as illustrated in Fig. 1. Additionally, given the real-time data collection from these sensors, the objective is to enable cooperative training across the clients on a unified global model using multimodal streaming data. This setting forms the foundational structure of the MMO-FL problem. During the training process, the sensors at each client continuously collect new multimodal data over time, with the timeline divided into discrete periods, denoted as $t = 1, 2, ..., T$. For simplicity, each time period is also treated as a global round.

TABLE II: Key Abbreviations

Acronyms	Full terms
IoT	Internet of Things
MFL	Multimodal Federated Learning
DNN	Deep Neural Network
MMO-FL	Multimodal Online Federated Learning
HFL	Horizontal Federated Learning
VFL	Vertical Federated Learning
OFL	Online Federated Learning
PFL	Prototype Federated Learning
PMM	Prototypical Modality Mitigation
OPC	Online Prototypes Construction
OPS	Online Prototypes Substitution

In each global round t, each client $k \in \mathcal{K}$ collects the current local training dataset consisting of N data samples from M modalities with no missing modalities. The dataset is denoted as $\mathcal{D}_k^t = \left(X_k^{t,1}, X_k^{t,2}, \ldots, X_k^{t,M}; Y_k^t \right) = \left\{ \left(x_{k,n}^{t,1}, x_{k,n}^{t,2}, \ldots, x_{k,n}^{t,M}; y_{k,n}^t \right) \right\}_{n=1}^{|\mathcal{D}_k^t|}$. Here, $x_{k,n}^{t,m}$ represents the mth modality data of the nth sample in client k collected

at global round t, and $y_{k,n}^t$ denotes the corresponding label. We also define $\mathcal{D}^t = \sum_{k=1}^{K} \mathcal{D}_k^t$ as overall dataset aggregated across all clients at global round t. Without loss of generality, we assume that each client k collects exactly N training samples per global round t.

In the MMO-FL, the overall model that needs to be trained by each client and aggregated at the server is divided into two components: the modality encoders $\theta^1, \ldots, \theta^M$, which extract the feature-level information from the raw data, and the head encoder θ^0, which integrates these features to generate the final prediction. Each modality encoder θ^m may adopt a distinct architecture tailored to the characteristics of the corresponding modality. Using modality-specific feature extractors allows each network to learn features tailored to its respective data type, thereby enhancing the quality of the learned representations. The feature vector for the mth modality of client k at round t, extracted by modality encoder θ^m, is denoted as $Z_k^{t,m} = \theta^m(X_k^{t,m})$. The header encoder θ^0 then synthesizes the outputs $Z_k^{t,m}$ from each modality encoder to fulfill the final learning objective. Based on these definitions, the loss function for the collective training dataset at round t is expressed as follows:

$$F_t(\Theta, \mathcal{D}^t) = \frac{1}{K} \sum_{k=1}^{K} f_t\left(\theta^0\left(Z_k^{t,1}, \ldots, Z_k^{t,M}\right), Y_k^t\right) \quad (1)$$

where $\Theta = \{\theta^0, \theta^1, \ldots, \theta^M\}$ represents the overall model. The term $\theta^0\left(Z_k^{t,1}, \ldots, Z_k^{t,M}\right)$ denotes the predicted labels through head encoder, and f_t is the loss function that measures the discrepancy between the predicted labels and the actual labels. It is important to note that the above loss function corresponds to a single global round.

Since the training process is based on dynamically collected real-time data rather than a static dataset, adopting an online learning paradigm is essential. Let the overall model at each global round be represented as $\Theta^1, \ldots, \Theta^T$. The learning regret, Reg_T is defined to quantify the gap between the cumulative loss incurred by the learner and the cumulative loss of an optimal fixed model in hindsight. Specifically:

$$\text{Reg}_T = \sum_{t=1}^{T} F_t(\Theta^t; \mathcal{D}^t) - \sum_{t=1}^{T} F_t(\Theta^*; \mathcal{D}^t) \quad (2)$$

Where $\Theta^* = \arg\min_\Theta \sum_{t=1}^{T} F_t(\Theta; \mathcal{D}^t)$ represents the optimal fixed model selected in hindsight. Our objective is to minimize the learning regret, which equates to minimizing the cumulative loss. Importantly, if the learning regret grows sublinearly with respect to T, it indicates that the online learning algorithm can progressively reduce the training loss asymptotically.

The fixed optimal strategy in hindsight refers to an idealized solution determined by a centralized entity with complete prior knowledge of all per-round loss functions. In our problem, achieving such an optimal strategy would require access to future information, including the entirety of the streaming datasets collected in subsequent rounds. However, due to the stochastic and unpredictable nature of real-time data collection, this information is inherently unavailable in practice.

Consequently, the full loss functions are not known in advance and evolve dynamically over time. Therefore, regret just serves as a metric to quantify the performance gap between the proposed algorithm and the theoretical optimal strategy in our theoretical analysis. For experimental validation, we will evaluate the effectiveness of our algorithm using practical metrics such as test accuracy.

IV. THE WORKFLOW OF MMO-FL WITH MISSING MODALITIES

In this section, we present the overall workflow of the MMO-FL framework and analyze the challenges introduced by missing modalities during the learning process. This discussion serves as the foundation for the subsequent theoretical analysis and the development of algorithms aimed at mitigating the impact of modality missing. We place particular emphasis on highlighting the distinctions between MMO-FL and conventional unimodal FL approaches.

MMO-FL builds upon the HFL framework, where each client collects independent data samples and the server is responsible for aggregating the global model. In conventional HFL, clients share the same feature space but have different sample spaces. However, in the case of missing modalities, the feature space across clients may not remain consistent during the whole learning process due to the presence of modality missing, differing from the standard HFL framework. During each global round $t \in \mathcal{T}$, where $\mathcal{T} = \{0, 1, 2, \ldots, T-1\}$, all clients execute a fixed number of local training iterations, represented by the parameter E. The index $\tau = 0, 1, 2, ..., E$ is used to track these local iterations. Then each global round t consists of a sequence of coordinated steps carried out across clients and the server. A visual illustration of the timeline for a single global round is provided in Fig. 2.
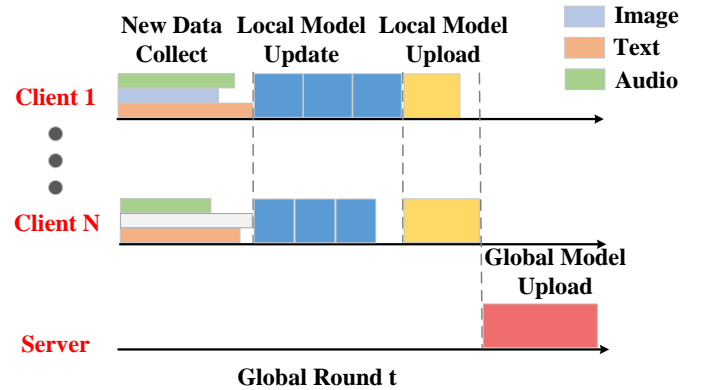

Fig. 2: The time diagram of MMO-FL in one global round

1) Client - New Data Collect: At the beginning of each global round t, each client k collects new local training data $\mathcal{D}_k^t$, utilizing the available set of modalities $\mathcal{M}_k^t$, determined by the operational status of the sensors on the client. Due to the inherent instability of edge devices in IoT environments, some sensors on clients may fail to collect new data during certain global rounds, leading to the appearance of modality missing. For convenience, we define $\bar{\mathcal{M}}_k^t = \mathcal{M} \backslash \mathcal{M}_k^t$ as the set of missing modalities for client k at global round t. Unlike the traditional modality missing issue in offline learning settings,

where the absence of a specific modality remains consistent throughout the training process, the online setting introduces variability, with the missing modalities for a client potentially changing between global rounds. This dynamic nature adds an additional layer of complexity to the learning process and renders conventional solutions designed for static offline scenarios ineffective.

2) Client - Local Model Update: Each client k uses the current global model Θ^t, provided by the server, as the initial model to train a new local model based on the current collected training dataset $\mathcal{D}_k^t$. Each client performs E iterations of online gradient descent (OGD) using the full training dataset. The model update rule varies depending on whether the client has access to all modalities or is experiencing modality missing. For clarity, we define the set $\mathcal{S}_t$ as the set of clients with complete modality availability in global round t, while the remaining clients, denoted by $\mathcal{K}\backslash\mathcal{S}_t$, represent those with missing modalities. Here, $\backslash$ denotes exclusion.

For Full Modality Client: In this case, the data collected by client $k \in \mathcal{S}_t$ includes all modalities $\mathcal{M}$, allowing the client to utilize the complete local dataset for OGD updates. Each modality data is processed through its respective modality encoder, and the final result is produced by the head encoder. In each global round, the client performs E local training iterations, following the update process described by the following equations:

$$
\begin{aligned}
\Theta_k^{t,0} &= \Theta^t \\
\Theta_k^{t,\tau+1} &= \Theta_k^{t,\tau} - \eta \mathbf{G}_k^{t,\tau}, \quad \forall \tau = 1,...,E \\
\Theta_k^{t+1} &= \Theta_k^{t,E}
\end{aligned}
\tag{3}
$$

Here, $\mathbf{G}_k^{t,\tau} = \nabla F_t(\Theta_k^{t,\tau}, \mathcal{D}_k^t)$ represents the gradient computed on the current local dataset containing all modalities, and η denotes the corresponding learning rate.

For Missing Modality Client: In this case, the data collected by client $k \in \mathcal{K}\backslash\mathcal{S}_t$ contains missing modalities with modalities set $\bar{\mathcal{M}}_k^t$ caused by the failure of data collection from the corresponding type of sensors. Consequently, the client can only perform local updates using the partial modality dataset $\tilde{\mathcal{D}}_k^t = \mathcal{D}_k^t \backslash \left\{ X_k^{t,m'} \right\}_{m' \in \bar{\mathcal{M}}_k^t}$. The available modalities are processed through their corresponding modality encoders, and the final output is generated by the head encoder. The client also executes E local training iteration, following the update procedure detailed in the equations below:

$$
\begin{aligned}
\Theta_k^{t,0} &= \Theta^t \\
\tilde{\Theta}_k^{t,\tau+1} &= \tilde{\Theta}_k^{t,\tau} - \eta \tilde{\mathbf{G}}_k^{t,\tau}, \quad \forall \tau = 1,...,E \\
\tilde{\Theta}_k^{t+1} &= \tilde{\Theta}_k^{t,E}
\end{aligned}
\tag{4}
$$

Here, we denote the gradient computed on the current local dataset with missing modalities as:

$$
\tilde{\mathbf{G}}_k^{t,\tau} = \left[\left(\tilde{\mathbf{G}}_k^{t,\tau,0} \right)^\top, \dots \left(\tilde{\mathbf{G}}_k^{t,\tau,m} \right)^\top, \dots \left(\tilde{\mathbf{G}}_k^{t,\tau,M} \right)^\top \right]^\top
\tag{5}
$$

where

$$
\tilde{\mathbf{G}}_k^{t,\tau,m} = \begin{cases} \nabla_m F_t(\Theta_k^{t,\tau}, \mathcal{D}_k^t) & if \quad m \in \mathcal{M}_k^t \\ 0 & otherwise \end{cases}
\tag{6}
$$

To highlight the impact of missing modality data on the model, $\tilde{\Theta}_k^{t+1}$ is used to denote local models with the impact of missing modalities. In this case, the parameters corresponding to the absent modalities remain unchanged, as they cannot be updated due to the failure in data collection and the absence of corresponding gradient updates.

3) Client - Local Model Upload: Each client will upload the corresponding local model to the server after finishing the E iterations of local model update. For clients $k \in \mathcal{S}_t$ with full modalities, the client will upload the local model Θ_k^{t+1} trained using data from all modalities. For clients $k \in \mathcal{K}\backslash\mathcal{S}_t$ with missing modalities, the client will upload the local model $\tilde{\Theta}_k^{t+1}$, which has been trained using available data with partial modalities. In the subsequent discussion, we also slightly abuse the notation $\tilde{\Theta}_k^{t+1}$ to also represent the local model obtained after applying a mitigation strategy for missing modalities. It is important to note that this model is inherently different from the one trained using complete modality data.

4) Server - Global Model Update: The server updates the global model by using the local model updates from the clients, as given by the following equation:

$$
\Theta^{t+1} = \frac{1}{K} \left(\sum_{k \in \mathcal{S}_t} \Theta_k^{t+1} + \sum_{k \in \mathcal{K}\backslash\mathcal{S}_t} \tilde{\Theta}_k^{t+1} \right)
\tag{7}
$$

The server then broadcasts the updated global model to all clients for the next global round. This process continues until the pre-set total number of global rounds is reached.

From the above MMO-FL workflow, it is evident that a central challenge introduced by modality missing lies in designing an effective mitigation strategy. The objective is to ensure that the model updated using reconstructed modalities closely approximates the one trained with complete modality data, i.e., $\tilde{\Theta}_k^{t+1} \to \Theta_k^{t+1}$. In the following, we first provide a detailed theoretical analysis to quantify the impact of modality missing on the regret bound of MMO-FL. Building on these insights, we then introduce the Prototypical Modality Mitigation algorithm, which is designed to effectively alleviate the adverse effects of missing modalities.

V. THEORETICAL ANALYSIS

In this section, we provide a comprehensive regret analysis of the proposed MMO-FL algorithm. The theoretical analysis of MMO-FL presents several additional challenges compared to traditional single-modal offline FL. First, the online learning setting necessitates evaluating long-term cumulative loss Reg_T rather than focusing solely on convergence at a single round. Second, the multimodal nature of the problem requires separate consideration of the feature models associated with each modality, making it inappropriate to treat the local model as a unified whole. As a result, new proof techniques and additional assumptions are required to adequately capture these complexities. We will conduct the regret analysis in three steps. First, we will examine the regret bound for the case where the local iteration is $E = 1$ without the impact of modality missing. Next, we will extend the analysis to the scenario where the local iteration is $E > 1$. Finally, we will

explore the regret bound when $E > 1$ while accounting for the impact of modality missing.

A. MMO-FL with local iteration $E = 1$ and without modality missing

To facilitate our analysis, we first introduce several additional definitions. After applying some basic transformations to the global model update equation above, we derive an alternative form of the global model update equation for the case where the local iteration is $E = 1$ and no modality is missing, as follows:

$$\Theta^{t+1} = \Theta^t - \frac{\eta}{K} \sum_{k=1}^{K} \mathbf{G}_k^t \qquad (8)$$

where $\mathbf{G}_k^t$ represents the gradient of the local overall model for client k across all M modalities, given by:

$$\mathbf{G}_k^t = \left[\left(\mathbf{G}_k^{t,0}\right)^\top, \left(\mathbf{G}_k^{t,1}\right)^\top, \dots \left(\mathbf{G}_k^{t,m}\right)^\top, \dots \left(\mathbf{G}_k^{t,M}\right)^\top \right]^\top \qquad (9)$$

Subsequently, we will introduce the assumptions that are standard for analyzing online convex optimization, as referenced in [35]. Several of these assumptions are defined at the modality level to align with the specific requirements of our multimodal parameter formulation.

Assumption 1. *For any $\mathcal{D}^t$, the loss function $F_t(\Theta; \mathcal{D}^t)$ is convex with respect to Θ and differentiable.*

Assumption 2. *The loss function is L-Lipschitz continuous, the partial derivatives for each modality satisfies: $\|\nabla_m F_t(\Theta)\|^2 \leq L^2$.*

Assumption 1 ensures the convexity of the function, enabling us to leverage the properties of convex optimization. Assumption 2 constrains the magnitude of the loss function's partial derivatives at the modality level. These assumptions are standard in theoretical analyses and are often approximately satisfied in practice through the use of convex surrogate losses and gradient normalization techniques. Based on these assumptions, we can derive the following Theorem 1.

Theorem 1. *Under Assumption 1-2, MMO-FL with local iterations $E = 1$ and excluding the impact of modality missing, achieves the following regret bound:*

$$Reg_T = \sum_{t=1}^{T} \sum_{k=1}^{K} \mathbb{E}_t \left[F_t(\Theta^t; \mathcal{D}_k^t) \right] - \sum_{t=1}^{T} \sum_{k=1}^{K} F_t(\Theta^*; \mathcal{D}_k^t)$$

$$\leq \frac{K \left\| \Theta^1 - \Theta^* \right\|^2}{2\eta} + \frac{\eta T K (M+1) L^2}{2} \qquad (10)$$

Proof. The proof can be found in Appendix A. $\qquad \square$

According to Theorem 1, by setting $\eta = \mathcal{O}(1/\sqrt{T})$, the MMO-FL can achieve a sublinear regret bound over T time rounds, specifically $\mathcal{O}(\sqrt{T})$. A sublinear regret bound indicates that the average regret per round, defined as the regret divided by the number of rounds, approaches zero as the number of rounds increases indefinitely. This suggests that the algorithm progressively refines its performance by learning from its errors.

B. MMO-FL with local iterations $E > 1$ and without modality missing

In this section, we extend the proof to the case where the local iteration $E > 1$ and no modality is missing. Following a similar transformations approach, we derive the global model update equation as follows:

$$\Theta^{t+1,0} = \Theta^{t,0} - \frac{\eta}{K} \sum_{k=1}^{K} \sum_{\tau=0}^{E-1} \mathbf{G}_k^{t,\tau} \qquad (11)$$

where $\mathbf{G}_k^{t,\tau}$ denotes the gradient of the local overall model for client k across all M modalities for round t and local iteration τ. Due to the presence of multiple local iterations $E > 1$, we introduce two additional assumptions to support the regret analysis of this case.

Assumption 3. *The partial derivatives, corresponding to the consistent loss function, fulfills the following condition:*

$$\left\| \boldsymbol{G}_k^{t,\tau'} - \boldsymbol{G}_k^{t,\tau} \right\| \leq \varphi \left\| \Theta_k^{t,\tau'} - \Theta_k^{t,\tau} \right\|$$

where τ' and τ indicate that they correspond to different local iterations.

For the purposes of subsequent theoretical analysis, we consider a D-dimensional vector for each modality in both the overall gradient and model. We define an arbitrary vector element $d \in [1, D]$ in overall gradient for modality m as $\mathbf{G}_{k,d}^m$, and similarly, the arbitrary vector element $d \in [1, D]$ in overall model for modality m is denoted as $\Theta_{k,d}^m$. Including the head encoder $m = 0$, each overall gradient and model consists of a total of $(M+1)D$ vector elements.

Assumption 4. *The arbitrary vector element d in the overall model $\Theta_{k,d}^m$ for any modality m is bounded by: $\left| \Theta_{k,d}^m \right| \leq \sigma$.*

Assumption 3 ensures that the variation in the partial derivatives is confined within a specific range, which aligns with the model variation over two different local iterations that maintain a consistent loss function. This approach effectively utilizes the concept of smoothness. Lastly, Assumption 4 specifies the permissible range for any vector element in the overall model. These assumptions are commonly adopted to facilitate convergence analysis and align with practical training settings, where consecutive model updates are guided by corresponding gradients and model parameters are constrained within a bounded range. Then we obtain the following Theorem 2.

Theorem 2. *Under Assumption 1-4, MMO-FL with local iterations $E > 1$ and excluding the impact of modality missing, achieves the following regret bound:*

$$Reg_T = \sum_{t=1}^{T} \sum_{k=1}^{K} \mathbb{E}_t \left[F_t(\Theta^{t,0}; \mathcal{D}_k^t) \right] - \sum_{t=1}^{T} \sum_{k=1}^{K} F_t(\Theta^*; \mathcal{D}_k^t)$$

$$\leq \frac{K \left\| \Theta^{1,0} - \Theta^* \right\|^2}{2\eta E} + \frac{\eta T K E (M+1) L^2}{2}$$

$$+ 2\eta T D E K (M+1)^2 \varphi \sigma L \qquad (12)$$

Proof. The proof can be found in Appendix C. $\qquad \square$

According to Theorem 2, by setting $\eta = \mathcal{O}(1/\sqrt{T})$, the MMO-FL can also achieve a sublinear regret rate $\mathcal{O}(\sqrt{T})$. However, in contrast to Theorem 1, the regret bound includes an additional term resulting from the presence of multiple local iterations. In the following analysis, we will consider the effect of modality missing.

C. MMO-FL with local iterations $E > 1$ and modality missing

In this section, we extend the proof to the more complex and common scenario where the local iteration $E > 1$ and modality missing is present. We derive the global model update equation as follows:

$$\Theta^{t+1,0} = \Theta^{t,0} - \frac{\eta}{K} \sum_{k=1}^{K} \sum_{\tau=0}^{E-1} \tilde{\mathbf{G}}_k^{t,\tau} \qquad (13)$$

where $\tilde{\mathbf{G}}_k^{t,\tau}$ represents the gradient of the local overall model for client k with partial modalities $\mathcal{M}_k^t$ at round t and local iteration τ. Obviously, $\mathcal{M} \neq \mathcal{M}_k^t$ holds for most of the time.

Since each client may have a different modality state in each global round, we introduce an additional assumption to constrain the maximum upper bound of modality missing. suppose we define $\beta_k^t = \frac{|\mathcal{M}_k^t|}{M+1}$ as the proportion of available modalities for client k at round t relative to the total number of modalities.

Assumption 5. *For any client k at any global round t, the minimum number of available modalities is bounded as follows: $\beta_k^t \geq \beta$, where β is a constant between $[0, 1]$.*

Assumption 5 guarantees the most extreme case of modality missing, ensuring that no client will experience a complete lack of modal data at any global round, which would otherwise prevent training. This assumption is reasonable in practical settings, as partial modality data is generally available throughout the training process, thus ensuring a minimal condition for effective learning. With the above assumptions in place, we can derive the following Theorem 3.

Theorem 3. *Under Assumption 1-5, MMO-FL with local iterations $E > 1$ and including the impact of modality missing, achieves the following regret bound:*

$$Reg_T = \sum_{t=1}^{T} \sum_{k=1}^{K} \mathbb{E}_t \left[F_t(\Theta^{t,0}; \mathcal{D}_k^t) \right] - \sum_{t=1}^{T} \sum_{k=1}^{K} F_t(\Theta^*; \mathcal{D}_k^t)$$

$$\leq \frac{K \left\| \Theta^{1,0} - \Theta^* \right\|^2}{2\eta E} + (5 - 2\beta)\eta T K (M+1) L^2$$

$$+ 2\eta T D K (M+1)^2 \varphi \sigma L + \frac{2(1-\beta)(M+1)TDK\sigma L}{E}$$

$$(14)$$

Proof. The proof can be found in Appendix E. $\qquad\square$

Based on Theorem 3, by choosing the learning rate as $\eta = \mathcal{O}(1/\sqrt{T})$, MMO-FL with missing modality can achieve a regret bound of $\mathcal{O}(\sqrt{T} + T(1-\beta))$ over T time rounds. Our analysis reveals that the term $\mathcal{O}(T(1-\beta))$, which accounts for the missing modality, plays a crucial role in determining whether a sublinear regret bound can be achieved. Notably, if

we set $\beta = 1$, meaning no modality is missing for any client in any round, MMO-FL clearly achieves a regret bound of $\mathcal{O}(\sqrt{T})$. However, in practical IoT environments where device stability cannot always be guaranteed, this term cannot be entirely eliminated. This occurs because missing data for a specific modality prevents the corresponding partial gradient from being updated. Consequently, the most effective way to mitigate the impact of missing modalities is to approximate the absent modal information. In the following section, we introduce the Prototypical Modality Mitigation algorithm, which is specifically designed to address this problem.

VI. Prototypical Modality Mitigation Algorithm

In this section, we present the details of the Prototypical Modality Mitigation algorithm (PMM). As discussed in the previous section, the regret bound of MMO-FL with missing modalities is influenced by the modality missing ratio. However, since the occurrence of missing modalities is often unpredictable and uncontrollable in real-world IoT settings, it becomes essential to mitigate their impact. A natural approach is to reconstruct or approximate the missing modalities during training process. Drawing inspiration from prototype learning, we identify prototypes as an effective solution to address this challenge, as they encapsulate the underlying semantics of each class and can serve as reliable substitutes for absent modality data.

Algorithm 1 Prototypical Modality Mitigation Algorithm

1: **for** $t = 1, 2, ..., T - 1$ **do**
2:     **Client Side:**
3:     **for** $k = 1, 2, ..., K$ **do**
4:         **if** Client with full modalities ($k \in \mathcal{S}_t$) **then**
5:             Calculates the local prototype $p_{k,c}^{t,m}$ via Eq. 15.
6:             Integrates and uploads the $p_{k,c}^{t,m}$ to server.
7:         **else**
8:             Downloads and utilizes the collection $\bar{\mathcal{P}}^t$.
9:             Builds the representations $\tilde{Z}_k^{t,m}$ via Eq. 19.
10:            Utilizes the $\tilde{\tilde{Z}}_k^{t,m}$ for class prediction.
11:         **end if**
12:     **end for**
13:     **Server Side:**
14:     Collects the local prototype $p_{k,c}^{t,m}$ from clients.
15:     Calculates the $p_c^{t,m}$ via Eq. 16.
16:     Updates the $\bar{p}_c^{t,m}$ via Eq. 17.
17:     Broadcasts the $\bar{\mathcal{P}}^t$ to all clients.
18: **end for**

The proposed PMM algorithm is detailed in Algorithm 1 and comprises two key components: Online Prototypes Construction (OPC), which builds prototypes based on dynamic data during the MMO-FL learning process, and Online Prototypes Substitution (OPS), which utilizes these prototypes for real-time modality mitigation in clients experiencing missing modalities. The detailed flowchart of the PPM algorithm is illustrated in Fig. 3. The subsequent sections provide a comprehensive explanation of each component.
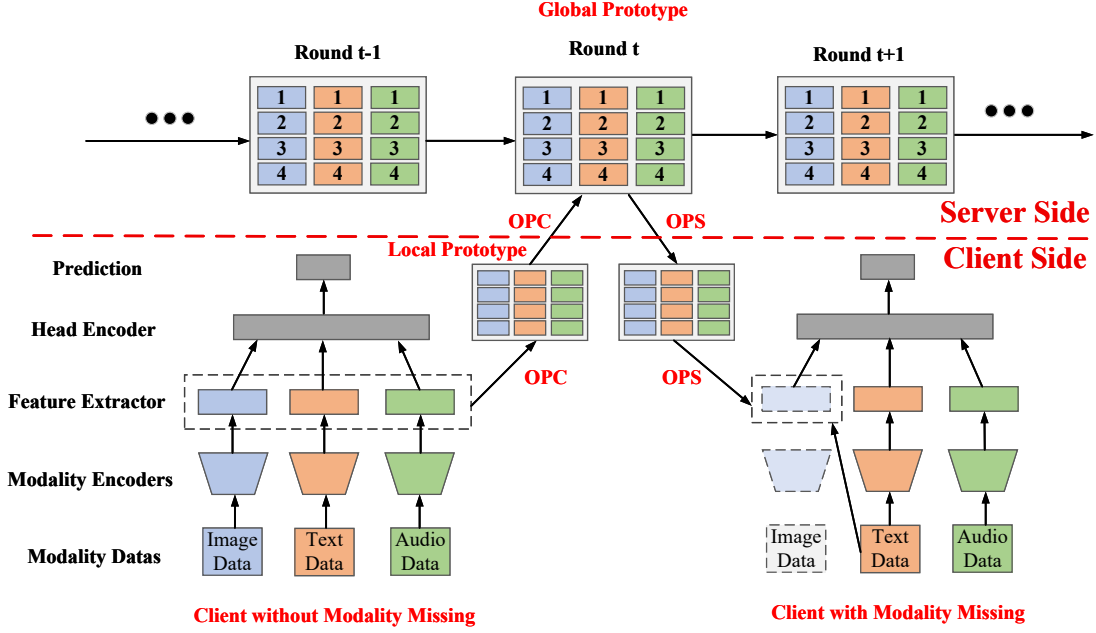
Fig. 3: Illustration of PMM: Online Prototypes Construction (OPC) involves generating prototypes from continuously evolving data throughout the MMO-FL learning process, ensuring a dynamic representation of each modality. Online Prototypes Substitution (OPS), on the other hand, leverages these prototypes to compensate for missing modalities in real-time, enabling effective modality reconstruction for clients encountering incomplete modality data.

A. Online Prototypes Construction

The OPC process runs concurrently within each global round. We define $c \in \{1, \ldots, C\}$ as the label classes. Accordingly, $X_{k,c}^{t,m}$ represents the set of training sample of class c and modality m, collected by client k at global round t. The local prototype is then defined as the average value of the features extracted by the modality encoder:

$$p_{k,c}^{t,m} = \frac{1}{|X_{k,c}^{t,m}|} \sum_{n \in \mathcal{N}(y_{k,n}^t = c)} \theta^m(x_{k,n}^{t,m}) \quad (15)$$

To ensure fairness among prototypes for different modalities, we introduce two additional rules: (1) Local prototypes across different modalities should follow a uniform structure, ensuring that the extracted features $\theta^m(x_{k,n}^{t,m})$ from modality encoders remain consistent, despite differences in the original data structures of each modality. By processing data through these encoders, the extracted features from various modalities can be structurally aligned. (2) Local prototypes across modalities should be normalized to maintain a consistent magnitude, ensuring fair comparability and stability throughout the learning process.

After each training round, each client generates local prototypes for each modality and transmits them to the server. After receiving the local prototypes for each modality, the server creates a temporal global prototype for modality m and class c at global round t as follows:

$$p_c^{t,m} = \frac{1}{|\mathcal{S}_t|} \sum_{k \in \mathcal{S}_t} p_{k,c}^{t,m} \quad (16)$$

Given the nature of online learning, data bias may arise, and the collected data may not encompass all classes in

every global round. To address this issue, it is essential to maintain a persistent global prototype that accurately captures the semantic representation of each class for every modality across rounds. We define the persistent global prototype for modality m and class c at global round t as follows:

$$\bar{p}_c^{t,m} = \frac{(t-1)\bar{p}_c^{t-1,m} + p_c^{t,m}}{t} \quad (17)$$

The persistent global prototype $\bar{p}_c^{t,m}$ is continuously updated and stored on the server, ensuring its availability for supporting the OPS process.

B. Online Prototypes Substitution

In this phase, the client performs prototype substitution to address missing modalities problem. After updating the persistent global prototype, the server organizes them by modality to create the persistent global prototype collection, denoted as $\bar{\mathcal{P}}^t$, defined as follows:

$$\bar{\mathcal{P}}^t = \begin{bmatrix} \bar{p}_1^{t,1} & \cdots & \bar{p}_c^{t,1} & \cdots & \bar{p}_C^{t,1} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \bar{p}_1^{t,m} & \cdots & \bar{p}_c^{t,m} & \cdots & \bar{p}_C^{t,m} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \bar{p}_1^{t,M} & \cdots & \bar{p}_c^{t,M} & \cdots & \bar{p}_C^{t,M} \end{bmatrix} \quad (18)$$

Next, the server distributes this global prototype collection to all clients. When a client detects that a certain modality is missing due to uncollected data, it generates the corresponding virtual feature representations for the missing modality. This reconstruction is performed by leveraging the received global prototypes in conjunction with the class distribution

inferred from the available modality data, as formulated in the following equation:

$$\tilde{Z}_k^{t,m} = \left[\bar{p}_{c(k,1)}^{t,m}, \ldots \bar{p}_{c(k,n)}^{t,m}, \ldots \bar{p}_{c(k,N)}^{t,m} \right] \tag{19}$$

Here, $\bar{p}_{c(k,n)}^{t,m}$ denotes the persistent global prototype of modality m, corresponding to the class label of the n-th data sample from client k at global round t. When modality m is absent in the current global round, the corresponding feature representations are synthesized using the persistent global prototypes. Subsequently, the predicted labels are obtained through the head encoder as $\theta^0 \left(Z_k^{t,1}, \ldots, \tilde{Z}_k^{t,m}, \ldots Z_k^{t,M} \right)$. This is then followed by the standard execution of the MMO-FL process, as introduced in the previous section. Although the PMM algorithm introduces some additional computational and communication overhead, its design keeps these costs minimal. First, the computation in the PMM algorithm primarily involves local prototype construction on the client side and global prototype updates on the server side. The local construction relies on the existing feature extractor, and the global updates impose minimal computational burden on the server. Communication in PMM mainly consists of uploading and downloading prototypes. Since prototypes are typically much smaller than full model parameters, this process does not introduce significant communication overhead. To account for potential overhead and additional considerations introduced by the implementation of the PMM algorithm, we present the following three additional remarks to further refine our proposed algorithm.

Remark 1 (Prototype Approximation): From the perspective of the law of large numbers, a feature extractor based on persistent global prototypes can effectively represent the feature extractor for missing modalities. As the number of prototypes increases, the representation of each modality becomes more accurate. The law of large numbers [36] suggests that as the sample size grows, the sample mean approaches the true mean of the population. Similarly, in feature extraction, using a larger number of prototypes captures the diverse characteristics of the data, reducing potential bias or distortion that could occur with a limited set of prototypes. By integrating more prototypes, the system can better approximate the feature distribution of the missing modality, compensating for missing information by leveraging the overall trends and patterns derived from the larger dataset.

Remark 2 (Quantized Upload): Since the persistent global prototype in the OPS phase serves as an approximate compensation mechanism, the precision of this substitution may not be highly sensitive and precise. Therefore, to further reduce communication overhead, quantization technique can be applied when transmitting the persistent global prototype collection. Similar quantization techniques are adopted in the field of FL [37] to reduce the communication overhead. In our experiments, we will also investigate the impact of this strategy on overall learning performance.

Remark 3 (Delayed Update): Given the robustness of online learning and the approximate nature of prototype substitution, the OPC process does not need to be executed in every global round. This design choice helps reduce both compu-

tational cost and communication overhead. Accordingly, the persistent global prototype collection used in the OPS phase can be a previously stored version rather than one updated in the current round. It is important to note that the updates refer specifically to the persistent global prototype, while the data continues to be updated according to the MMO-FL framework and is not influenced by the delayed update mechanism. A similar delayed update strategy is also employed in FL [38] to reduce the computational overhead. In our experiments, we further investigate the impact of this strategy on overall learning performance.

VII. Experiment

In this section, we will experimentally evaluate the performance of the MMO-FL algorithm. The experiments were conducted on an Ubuntu 18.04 machine equipped with an Intel Core i7-10700KF 3.8GHz CPU and a GeForce RTX 3070 GPU. The detailed experimental settings are provided below.

A. Datasets

To simulate MMO-FL in IoT scenarios, we will use two real-world multi-modal datasets, UCI-HAR and MVSA-Single. Detailed descriptions of both datasets are provided below.

UCI-HAR: The UCI-HAR dataset is a widely recognized resource for human activity recognition research. It includes 10299 data samples collected from 30 participants (average age: 24) engaging in six activities (six classes): walking, walking upstairs, walking downstairs, sitting, standing, and lying down. These activities were recorded using smartphone sensors, specifically accelerometers and gyroscopes, which capture three-dimensional motion data. The sensors sampled data at 50 Hz, producing 128 readings per sensor axis within each time window. This dataset is utilized in our experiments to analyze sensor-based human activity recognition using three-dimensional motion data.

MVSA-Single: The MVSA-Single dataset is tailored for multimodal sentiment analysis research, focusing on the integration of textual and visual cues from social media. It comprises 5,129 image-text pairs, where each sample consists of a single image accompanied by corresponding textual content. Each pair is annotated with one of three sentiment labels: Positive, Neutral, or Negative, reflecting the emotional tone jointly conveyed by the image and text.

Both of the original datasets are static and designed for offline learning. To align with the requirements of online learning, they must be transformed into dynamic datasets. The transformation process is described in detail in the following.

B. Online Data Generation

In the experiment, operating within an online learning scenario requires the training dataset to be dynamic, with data collected at the start of each global round. To ensure sufficient data samples for good training performance, we collect the initial dataset at the beginning of the training process. Considering the differences in dataset types and sizes,

distinct online data generation details are used for the UCI-HAR and MVSA-Single datasets.

UCI-HAR: For the UCI-HAR dataset, training involves a total of five clients. Initially, each client is assigned 2000 data samples drawn according to a Dirichlet distribution with a non-IID (independent and identically distributed) degree α, representing the client's long-term data source. In subsequent global rounds, each client maintains an online local dataset of 500 samples. At every round, 20 new samples are drawn from the long-term data source and appended to the local dataset, while the oldest 20 samples are simultaneously removed. This real-time update mechanism ensures that the local datasets evolve dynamically, satisfying the settings of online learning.

MVSA-Single: For the MVSA-Single dataset, the training process is distributed across five clients. Initially, each client is allocated 1,500 data samples drawn from a Dirichlet distribution with a non-IID parameter α, simulating the client's long-term data source. In subsequent global rounds, each client maintains an online local dataset consisting of 800 samples. At every round, 20 new samples are drawn from the long-term data source and added to the local dataset, while the oldest 20 samples are removed. This streaming update mechanism ensures that each client's dataset evolves continuously over time, thereby aligning with the requirements of online learning.

Modality Missing Simulation: To ensure consistency and clearly isolate the impact of missing modalities, we simulate missing modalities by assuming that all clients simultaneously experience the absence of a specific modality within the same global round. This design choice avoids the confounding effects that would arise if different clients experienced different missing modalities across rounds, which would otherwise complicate the evaluation of modality-level influence on learning performance. Given that all datasets used in this study contain only two modalities, we simulate the missing of one modality at a time. Additionally, we define λ as the ratio of global rounds in which modality missing occurs to the total number of rounds. In subsequent experiments, we control the value of λ to systematically investigate the effect of the frequency of modality missing on learning performance.

C. Model Details

In the following, we detail the model architectures and key parameters used in our experiments, presented separately for the two datasets.

UCI-HAR: The dataset includes two distinct modalities: accelerometer and gyroscope signals, necessitating the use of modality-specific encoder models. For the accelerometer data, we use a CNN-based model as the encoder. This model consists of five convolutional layers and one fully connected layer. For the gyroscope data, we use an LSTM-based model with one LSTM layer and one fully connected layer. Both encoders generate a 128-dimensional feature representation. The shared header model is implemented using two fully connected layers. The learning rate is 0.1, with a decay factor of 0.95 until it reaches 0.001.

MVSA-Single: The dataset includes two distinct modalities: text and image data, also need modality-specific encoder

models. For the image data, a four-layer CNN is utilized, with a modified output layer designed to produce a 128-dimensional feature vector. For text data, a two-layer LSTM network is employed, also generating a 128-dimensional output. The modality-specific features are subsequently processed by a shared header model consisting of two fully connected layers. The learning rate is 0.01, with a decay factor of 0.99 until it reaches 0.001.

D. Benchmarks

In our experiments, we employ several benchmarks for performance comparison. Our work is the first to investigate the problem of modality missing in a multimodal online federated learning setting. Therefore, existing methods developed for handling modality missing in traditional centralized offline multimodal training are not directly applicable for performance comparisons in our scenario. This is primarily due to the additional challenges introduced by the dynamic nature of online learning and the distribution heterogeneity inherent in federated training. To address this gap, we adopt several benchmarks for performance comparison, including a setting where all modalities are present, a setting where missing modalities are left unhandled, and a simple baseline that addresses missing modalities through zero-filling. Detailed descriptions of the benchmarks are provided as following.

Full Modality (FM). In this ideal scenario, all clients are presumed to collect data across all modalities through their respective sensors, ensuring that no modalities are missing in any global round.

Partial Modality (PM). In this scenario, the failure of certain sensors during some global rounds may prevent the collection of specific modal data. Consequently, the client is trained using only the remaining available modalities.

Zero Filling (ZF). In this scenario, certain sensors may fail during some global rounds, resulting in the inability to collect specific modal data. To address this issue, a straightforward baseline approach of zero-filling is employed to substitute for the missing modalities with zero-valued representations.

Based on the simulation setup described above, we present the following experimental results.

E. Simulation Results

In this section, we present the experimental results of MMO-FL. We begin with a comparative analysis of the proposed PMM algorithm against several benchmarks, evaluating overall learning performance. To further validate the effectiveness of PMM, we conduct four ablation studies. First, we investigate the effect of varying the modality missing rate on learning performance. Second, we study the effect of different Non-IID level setting. Third, we evaluate the performance implications and communication efficiency gained through the use of quantized prototype uploads. Final, we examine the impact of the delayed update strategy, assessing both its influence on performance and its potential for reducing communication and computational overhead. All simulation results are averaged over 10 random runs.
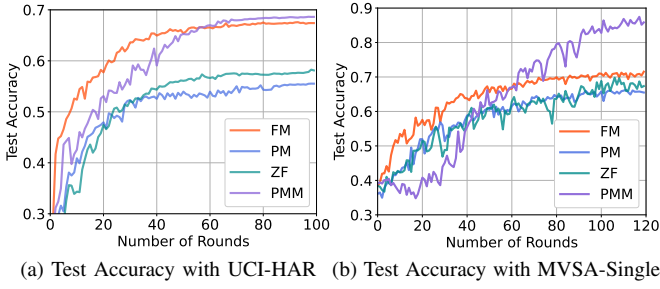
(a) Test Accuracy with UCI-HAR   (b) Test Accuracy with MVSA-Single

Fig. 4: Performance comparison of proposed algorithm and benchmarks with modality missing.

Performance Comparison. We begin by evaluating the learning performance of the proposed PMM algorithm compared to benchmarks in the presence of missing modalities within the MMO-FL scenario. The test accuracy comparison between PMM and benchmarks are shown in Fig. 4(a) and Fig. 4(b) based on UCI-HAR dataset with configuration $[\lambda = 0.5, \alpha = 10]$ and MVSA-Single dataset with $[\lambda = 0.5, \alpha = 1]$, respectively. Based on the above figures, we get several key observations. First, we observe that PMM significantly outperforms both the PM and ZF benchmarks in terms of test accuracy. Remarkably, PMM even achieves performance superior to the FM setting in both cases, indicating that the prototypes generated by PMM not only compensate for the missing modalities but also serve as robust substitutes that enhance overall learning. Second, we observe that the ZF yields better performance than the PM case, suggesting that explicitly preserving the structure of the missing modality even with zero-filled inputs can be more beneficial than entirely omitting the modality during training. Third, we observe that the performance of PMM improves over time, initially lagging behind the FM but eventually surpassing it in the later stages. This behavior highlights the effectiveness of the PMM, which incrementally refines prototype representations over time. As training progresses, these prototypes become more representative of each class and modality, enabling them to guide the learning process more effectively than raw modality data. This also explains why PMM underperforms FM in the early stages on both datasets but ultimately outperforms it as training continues.

Building on the performance comparison results presented above, which demonstrate the effectiveness of the proposed PMM algorithm in the MMO-FL setting, we now proceed with a series of ablation studies to analyze the specific impact of key parameters on learning performance.
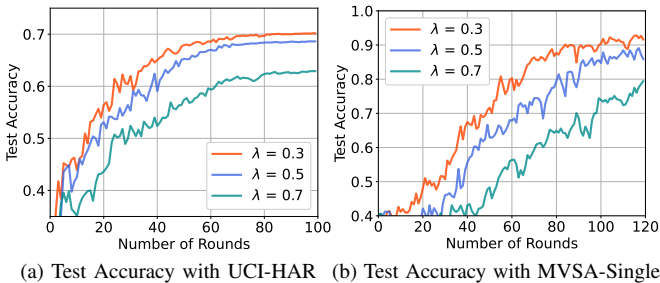

(a) Test Accuracy with UCI-HAR   (b) Test Accuracy with MVSA-Single

Fig. 5: Performance comparison of proposed algorithm with different modality missing rate.

Impact of Modality Missing Rate. In this part, we will explore the effects with the modality missing rate λ on learning performance. The results for the UCI-HAR and MVSA-Single datasets are presented in Fig. 5(a) and Fig. 5(b), respectively. Several key observations can be made: First, a smaller value of λ consistently leads to better learning performance. This improvement is attributed to two factors. On one hand, a lower missing rate results in more rounds where the persistent global prototypes can be accurately updated, allowing them to better represent the underlying modality distributions. On the other hand, fewer occurrences of missing modalities naturally reduce information loss during training, thereby enhancing performance. Second, when $\lambda = 0.7$, corresponding to a high frequency of missing modalities, we observe a significant performance drop in both datasets. This indicates that the frequency of modality absence plays a critical role in determining model effectiveness, and excessive modality missing can severely degrade learning performance.
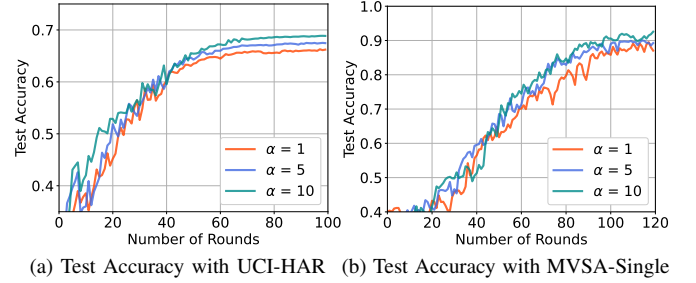

(a) Test Accuracy with UCI-HAR   (b) Test Accuracy with MVSA-Single

Fig. 6: Performance comparison of proposed algorithm with different Non-IID level.

Impact of Non-IID Level. In this part, we will explore the effects with the Non-IID level α on learning performance. The corresponding are illustrated in Fig. 6(a) and Fig. 6(b), respectively. Here, a larger value of α indicates a lower degree of data heterogeneity. As expected, we observe that increasing α leads to improved learning performance in both datasets, which aligns with the well-established understanding of Non-IID effects in conventional FL. Interestingly, we also observe that the impact of α on performance is less pronounced than that of the modality missing rate λ. This suggests that the use of prototype-based compensation in PMM may help alleviate the adverse effects of data heterogeneity, thereby maintaining robust performance across varying degrees of Non-IID.

Next, we evaluate the efficiency of the proposed PMM algorithm by examining the quantized upload and the delayed update strategies, which were previously introduced in the remarks of the PMM algorithm section.

Impact of Quantized Upload. In this part, we investigate the impact of applying quantization to local prototype during the PMM upload process as a means to further reduce communication overhead. Specifically, we adopt an uniform scalar quantizer. The parameter b denotes the number of bits per component used for compression, when quantization is not applied, $b = 32$, corresponding to full-precision transmission. Applying quantization with b bits yields a total of 2^b quantization levels. We evaluate the effect of varying $b \in [2, 4]$ on learning performance. The results, as illustrated in Fig. 7(a) and Fig. 7(b) by round, and Fig. 7(c) and

(a) Test Accuracy with UCI-HAR (b) Test Accuracy with MVSA-Single

(c) Test Accuracy with UCI-HAR by (d) Test Accuracy with MVSA-Single
communication cost for PMM   by communication cost for PMM
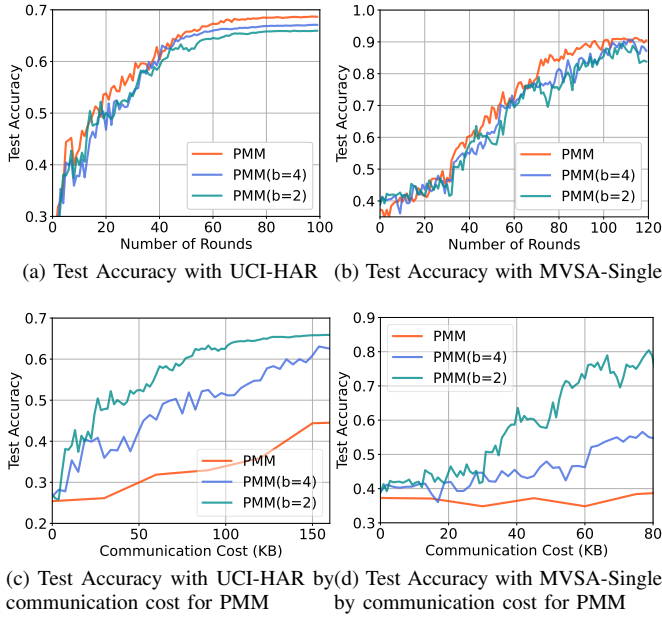
Fig. 7: Performance comparison of proposed algorithm with different level of quantization.

Fig. 7(d) by communication cost, reveal the trade-off between model performance and communication efficiency that with the effect of quantization. As expected, larger values of b lead to improved performance but incur higher communication costs. Importantly, we observe that the prototypes do not require high precision to be effective. As a result, quantization introduces only a minor reduction in learning performance, while significantly lowering communication overhead. This demonstrates that PMM can be efficiently integrated into the MMO-FL framework without introducing substantial communication burden.
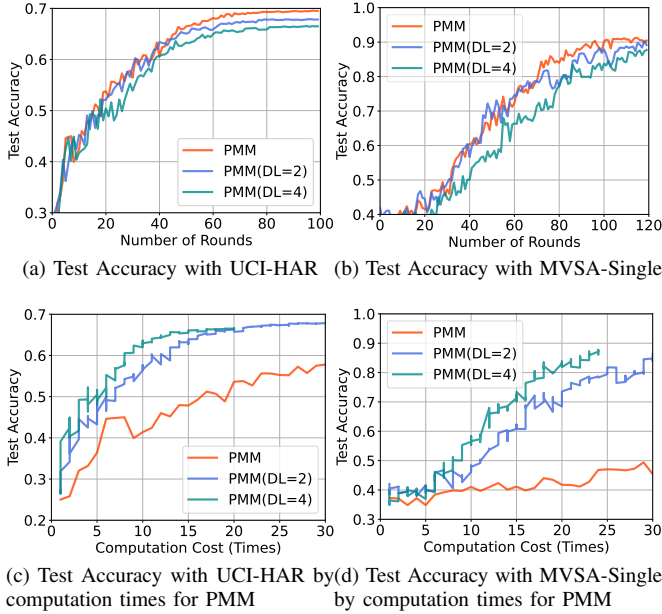

(a) Test Accuracy with UCI-HAR (b) Test Accuracy with MVSA-Single

(c) Test Accuracy with UCI-HAR by (d) Test Accuracy with MVSA-Single
computation times for PMM   by computation times for PMM

Fig. 8: Performance comparison of proposed algorithm with different level of delay.

Impact of Delayed Update. In this section, we evaluate the impact of applying delayed updates to the PMM algorithm. We investigate the impact of different update intervals by varying the frequency of updates with interval values of $[0, 2, 4]$, where a modality must accumulate the corresponding number of occurrences before triggering an update. The experimental results based on the UCI-HAR and MVSA-Single datasets are shown in Fig. 8(a) and Fig. 8(b). Our findings show that while delayed updates lead to a moderate decrease in performance, the PMM algorithm still maintains satisfactory learning outcomes. Additionally, In Fig. 8(c) and Fig. 8(d), we report the corresponding computation cost. Here, "times" in the x-axis labels refers to the number of computations, which closely aligns with communication cost since communication is only triggered after computation. These results highlight an inherent trade-off: updating at every round yields the best performance but incurs higher resource consumption, whereas adopting delayed updates reduce computational and communication overhead at the cost of slight performance degradation. This makes delayed updating a compelling option in resource-constrained environments.

Through the series of experiments presented above, we have thoroughly evaluated the performance of the proposed PMM algorithm in comparison to benchmarks, as well as examined its scalability across various settings.

VIII. CONCLUSION

In this work, we propose the MMO-FL framework to address the challenges of distributed cooperative learning over real-time multimodal data in IoT environments with edge intelligence devices. Recognizing the instability of modality-specific sensors during continuous data collection, we explicitly consider and tackle the problem of modality missing throughout the learning process. To mitigate its impact, we introduce the PMM algorithm, which leverages prototype learning to effectively reconstruct and compensate for missing modal information. The proposed approach is supported by rigorous theoretical analysis and validated through extensive experiments. In future work, we aim to integrate real-time IoT multimodal data and develop practical testbeds to evaluate and enhance the MMO-FL framework. These efforts will further facilitate its deployment in real-world IoT applications.

REFERENCES

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[2] H. Wang, J. Bian, and J. Xu, "On the local cache update rules in streaming federated learning," *IEEE Internet of Things Journal*, vol. 11, no. 6, pp. 10 808–10 816, 2023.

[3] T. Wu, F. Wu, C. Qiu, J.-M. Redouté, and M. R. Yuce, "A rigid-flex wearable health monitoring sensor patch for iot-connected healthcare applications," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 6932–6945, 2020.

[4] H. Wang, X. Zhong, K. Liu, F. Liu, and W. Zhang, "Denoising and adaptive online vertical federated learning for sequential multi-sensor data in industrial internet of things," *arXiv preprint arXiv:2501.01693*, 2025.

[5] J. Levinson, J. Askeland, J. Becker, J. Dolson, D. Held, S. Kammel, J. Z. Kolter, D. Langer, O. Pink, V. Pratt *et al.*, "Towards fully autonomous driving: Systems and algorithms," in *2011 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2011, pp. 163–168.

[6] F. Al-Turjman and S. Alturjman, "Context-sensitive access in industrial internet of things (iiot) healthcare applications," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 6, pp. 2736–2744, 2018.

[7] Z. Xiao and Y. Xiao, "Security and privacy in cloud computing," *IEEE communications surveys & tutorials*, vol. 15, no. 2, pp. 843–859, 2012.

[8] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Jiao, Y.-C. Liang, Q. Yang, D. Niyato, and C. Miao, "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Communications Surveys and Tutorials*, vol. 22, no. 3, pp. 2031–2063, 2020.

[9] Z. Huang, X. Xu, J. Ni, H. Zhu, and C. Wang, "Multimodal representation learning for recommendation in internet of things," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10 675–10 685, 2019.

[10] L. Che, J. Wang, Y. Zhou, and F. Ma, "Multimodal federated learning: A survey," *Sensors*, vol. 23, no. 15, p. 6986, 2023.

[11] B. Xiong, X. Yang, Y. Song, Y. Wang, and C. Xu, "Client-adaptive cross-model reconstruction network for modality-incomplete multimodal federated learning," in *Proceedings of the 31st ACM International Conference on Multimedia*, 2023, pp. 1241–1249.

[12] L. Che, J. Wang, X. Liu, and F. Ma, "Leveraging foundation models for multi-modal federated learning with incomplete modality," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2024, pp. 401–417.

[13] D. Sahoo, Q. Pham, J. Lu, and S. C. Hoi, "Online deep learning: learning deep neural networks on the fly," *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI-18)*, 2018.

[14] S. Hong and J. Chae, "Communication-efficient randomized algorithm for multi-kernel online federated learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 12, pp. 9872–9886, 2021.

[15] D. Kwon, J. Park, and S. Hong, "Tighter regret analysis and optimization of online federated learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[16] A. Mitra, H. Hassani, and G. J. Pappas, "Online federated learning," in *2021 60th IEEE Conference on Decision and Control (CDC)*. IEEE, 2021, pp. 4083–4090.

[17] H. Wang and J. Xu, "Online vertical federated learning for cooperative spectrum sensing," *IEEE Transactions on Cognitive Communications and Networking*, 2024.

[18] X. Ouyang, Z. Xie, H. Fu, S. Cheng, L. Pan, N. Ling, G. Xing, J. Zhou, and J. Huang, "Harmony: Heterogeneous multi-modal federated learning through disentangled model training," in *Proceedings of the 21st Annual International Conference on Mobile Systems, Applications and Services*, 2023, pp. 530–543.

[19] H. Chen, Y. Zhang, D. Krompass, J. Gu, and V. Tresp, "Feddat: An approach for foundation model finetuning in multi-modal heterogeneous federated learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 10, 2024, pp. 11 285–11 293.

[20] J. Chen and A. Zhang, "Fedmsplit: Correlation-adaptive federated multi-task learning across multimodal split networks," in *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, 2022, pp. 87–96.

[21] J. Bian, L. Wang, and J. Xu, "Prioritizing modalities: Flexible importance scheduling in federated multimodal learning," *arXiv preprint arXiv:2408.06549*, 2024.

[22] M. Ma, J. Ren, L. Zhao, S. Tulyakov, C. Wu, and X. Peng, "Smil: Multimodal learning with severely missing modality," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 3, 2021, pp. 2302–2310.

[23] C. Zhang, X. Chu, L. Ma, Y. Zhu, Y. Wang, J. Wang, and J. Zhao, "M3care: Learning with missing modalities in multimodal healthcare data," in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2022, pp. 2418–2428.

[24] H. Maheshwari, Y.-C. Liu, and Z. Kira, "Missing modality robustness in semi-supervised multi-modal semantic segmentation," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 1020–1030.

[25] X. Wang, R. Zhou, H. Xie, X. Tang, L. He, and C. Yang, "Clusmfl: A cluster-enhanced framework for modality-incomplete multimodal federated learning in brain imaging analysis," *arXiv preprint arXiv:2502.12180*, 2025.

[26] H. Q. Le, C. M. Thwal, Y. Qiao, Y. L. Tun, M. N. Nguyen, E.-N. Huh, and C. S. Hong, "Cross-modal prototype based multimodal federated learning under severely missing modality," *Information Fusion*, p. 103219, 2025.

[27] J. Zhang, C. Shan, and J. Han, "Fedgmkd: An efficient prototype federated learning framework through knowledge distillation and discrepancy-aware aggregation," *Advances in Neural Information Processing Systems*, vol. 37, pp. 118 326–118 356, 2024.

[28] Y. Tan, G. Long, L. Liu, T. Zhou, Q. Lu, J. Jiang, and C. Zhang, "Fedproto: Federated prototype learning across heterogeneous clients," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 36, no. 8, 2022, pp. 8432–8440.

[29] Y. Zhou, X. Qu, C. You, J. Zhou, J. Tang, X. Zheng, C. Cai, and Y. Wu, "Fedsa: A unified representation learning via semantic anchors for prototype-based federated learning," *arXiv preprint arXiv:2501.05496*, 2025.

[30] Y. Tan, G. Long, J. Ma, L. Liu, T. Zhou, and J. Jiang, "Federated learning from pre-trained models: A contrastive learning approach," *Advances in neural information processing systems*, vol. 35, pp. 19 332–19 344, 2022.

[31] S. Wu, J. Chen, X. Nie, Y. Wang, X. Zhou, L. Lu, W. Peng, Y. Nie, and W. Menhaj, "Global prototype distillation for heterogeneous federated learning," *Scientific Reports*, vol. 14, no. 1, p. 12057, 2024.

[32] T. Jin, X. Cheng, L. Li, W. Lin, Y. Wang, and Z. Zhao, "Rethinking missing modality learning from a decoding perspective," in *Proceedings of the 31st ACM International Conference on Multimedia*, 2023, pp. 4431–4439.

[33] M. Li, D. Yang, X. Zhao, S. Wang, Y. Wang, K. Yang, M. Sun, D. Kou, Z. Qian, and L. Zhang, "Correlation-decoupled knowledge distillation for multimodal sentiment analysis with incomplete modalities," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 12 458–12 468.

[34] G. Bao, Q. Zhang, D. Miao, Z. Gong, L. Hu, K. Liu, Y. Liu, and C. Shi, "Multimodal federated learning with missing modality via prototype mask and contrast," *arXiv preprint arXiv:2312.13508*, 2023.

[35] J. Park, D. Kwon, and S. Hong, "Fedqogd: Federated quantized online gradient descent with distributed time-series data," in *2022 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2022, pp. 536–541.

[36] P.-L. Hsu and H. Robbins, "Complete convergence and the law of large numbers," *Proceedings of the national academy of sciences*, vol. 33, no. 2, pp. 25–31, 1947.

[37] R. Hönig, Y. Zhao, and R. Mullins, "Dadaquant: Doubly-adaptive quantization for communication-efficient federated learning," in *International Conference on Machine Learning*. PMLR, 2022, pp. 8852–8866.

[38] J. Liu, J. Jia, T. Che, C. Huo, J. Ren, Y. Zhou, H. Dai, and D. Dou, "Fedasmu: Efficient asynchronous federated learning with dynamic staleness-aware model update," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 12, 2024, pp. 13 900–13 908.