FedMM: A Communication Efficient Solver for Federated Adversarial Domain Adaptation

Yan Shen*
University at Buffalo
Buffalo, New York, USA
yshen22@buffalo.edu

Jian Du*
TikTok Inc.
San Jose, California, USA
dujianeee@gmail.com

Han Zhao
University of Illinois Urbana Champaign
Urbana, Illinois, USA
hanzhao@illinois.edu

Zhanghexuan Ji
University at Buffalo
Buffalo, New York, USA
zhanghex@buffalo.edu

Chunwei Ma
University at Buffalo
Buffalo, New York, USA
chunweim@buffalo.edu

Mingchen Gao
University at Buffalo
Buffalo, New York, USA
mgao8@buffalo.edu

ABSTRACT

Federated adversary domain adaptation is a unique distributed minimax training task due to the heterogeneous data among different local clients, where each client only sees a subset of the data that merely belongs to either the source or target domain. Despite the extensive research in distributed minimax optimization, existing communication efficient solvers that exploit multiple steps of the local update are still not able to generate satisfactory solutions for federated adversarial domain adaptation because of the gradient divergence issue among clients. To tackle this problem, we propose a distributed minimax optimizer, referred to as FedMM, by introducing dual variables to bridge the gradient gap among clients. This algorithm is effective even in the extreme case where each client has different label classes and some clients only have unlabeled data. We prove that FedMM admits benign convergence to a stationary point under domain-shifted unlabeled data. On a variety of benchmark datasets, extensive experiments show that FedMM consistently achieves both better communication savings and significant accuracy improvements over existing federated optimizers based on the stochastic gradient descent ascent (SGDA) algorithm. When training from scratch, for example, it outperforms other SGDA based federated average methods by around 20% in accuracy over the same communication rounds; and it consistently outperforms when training from pre-trained models.

KEYWORDS

Federated Learning; Domain Adaptation; Adversarial Learning

ACM Reference Format:
Yan Shen*, Jian Du*, Han Zhao, Zhanghexuan Ji, Chunwei Ma, and Mingchen Gao. 2023. FedMM: A Communication Efficient Solver for Federated Adversarial Domain Adaptation. In *Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), London, United Kingdom, May 29 – June 2, 2023*, IFAAMAS, 9 pages.

*The first two authors contributed equally to this work.

Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), A. Ricci, W. Yeoh, N. Agmon, B. An (eds.), May 29 – June 2, 2023, London, United Kingdom.

1 INTRODUCTION

Federated Learning (FL) is gaining popularity because it enables multiple clients to train machine learning models without directly sharing the potentially sensitive data with other clients [11, 13].

A typical FL training pipeline involves exchanging local model parameters with a centralized server to update the global model parameter, and its communication overhead has been, in many cases, identified as the bottleneck [3, 20] of the training pipeline. Moreover, *domain shift* often exists between clients' data [23], which is another inherent characteristic of FL training, where the data are sampled from different parts of the sample space on different clients. Because of the aforementioned unique features, FL training needs efficient optimizers that converge over heterogeneous data among clients with fewer communication rounds.

For data with distributional shifts, one of the most challenging settings is that each local client only has access to a subset of the label classes in order to train the global/common model. In this situation, the global model's accuracy suffers considerably as a result of the gradient/model drift [20]. In the literature of domain adaptation, this problem is also known as label shift [29, 38]. Under the setting of FL, however, label shift is ubiquitous due to the imbalance between clients' label distributions, with the extreme case being that individual clients have disjoint labels or only unlabeled data. While domain adversarial training is a classic technique in centralized settings [7, 31, 43], the distributed nature of FL makes the direct application of this line of approaches particularly challenging.

One method is to use the stochastic gradient descent ascent (SGDA) method [15] directly as if the data are *homogeneous and centralized*, where data are aggregated together to find the saddle point solutions [10, 16] of a minimax problem. However, because of the potential domain shifts among clients in FL settings, a single client cannot access an unbiased sample of the gradient from the global objective function. A natural solution would be to average each client's gradients, which exactly corresponds to the FedSGDA approach in [22]. Its training efficiency, on the other hand, is low due to the requirement of large communication rounds between the server and clients. Without considering the issue of domain shift, there are several works on communication-efficient FL algorithms. A large spectrum of these algorithms are variations of the classic FedAvg [20]. Following this pipeline, a natural extension of

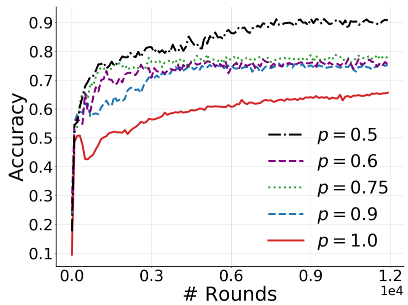


Figure 1: FedAvgSGDA for CDAN with 2 clients. The ratios for source and target data allocated to client 1 are p and $(1 - p)$. The remaining data pertains to client 2. It shows that the performance of FedAvgSGDA degrades rapidly as the data distribution becomes imbalanced, which motivates our FedMM algorithm.

FedSGDA is FedAvgSGDA shown in Algorithm 3 in the appendix. However, if the data are non-i.i.d among clients, especially in the case of imbalanced label distributions [42], the performance of FedAvg would be significantly lower than that when all the data were trained on a single client. As empirically demonstrated in Fig. 1, we can see that the performance of FedAvgSGDA degrades rapidly as the data distributions become more and more imbalanced.

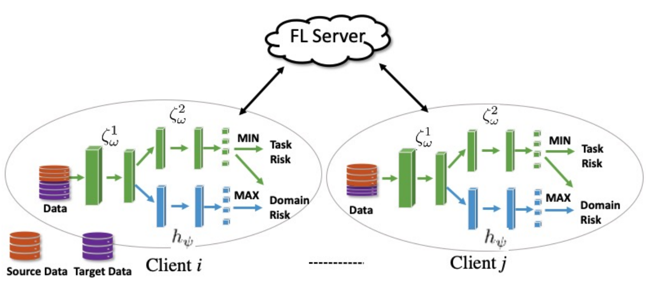


Figure 2: A federated adversarial domain adaptation model. Only the source risk of the client's local source data (if any) and the domain risk of the client's source/target data are accessible locally. The source data labels are non-i.i.d.

FedMM. In light of the above challenges, to optimize a federated minimax objective, we formulate this distributed saddle point optimization as a Federated MiniMax (FedMM) optimization on a sum of non-identical distributions. In particular, we use an augmented Lagrange function to enforce the global model consensus constraints. In each client's local optimization oracle, FedMM deconstructs the global sum by solving the augmented Lagrange of each function individually in a finite number of steps. The collection of Lagrange dual variables will then locally compensate for the client-to-client model divergence caused by domain shift. To summarize, our main contributions are listed as follows.

Contributions:

(i) We present, FedMM, a stochastic federated optimizer tailored for federated minimax optimizations with non-separable minimization and maximization variables, as well as clients with imbalanced label class distributions. FedMM is effective in the extreme case where each client has disjoint classes of labels or unlabeled data.

(ii) Under the generic federated saddle point optimization problem with a nonconvex-concave global objective function assumption, we prove that FedMM converges asymptotically to a stationary point for the nonconvex-strongly-concave setting under local update residual errors and distribution shifts

(iii) Empirically, we show that FedMM consistently achieves either significant communication savings or accuracy improvements over the federated SGDA method on a variety of benchmark datasets with varying adversarial domain adaptation networks. For example, when training from scratch, it outperforms other SGDA based federated average methods by around 20% in accuracy over the same communication rounds.

2 PRELIMINARIES

2.1 Adversarial Domain Adaptation

Domain adaptation refers to the process of transferring knowledge from a labeled source domain to an unlabeled target domain [2, 42]. Let $\mathcal{P}$ and $\mathcal{Q}$ be the source and target distributions, respectively. In a general formulation, the upper bound of the target prediction error is given by Ben-David et al. [2]

$$\mathrm{err}_{\mathcal{Q}}(\zeta) \le \mathrm{err}_{\mathcal{P}}(\zeta) + d_{\mathcal{H}}(\mathcal{P}, \mathcal{Q}) + \min_{\zeta^* \in \mathcal{F}} \{\mathrm{err}_{\mathcal{P}}(\zeta^*) + \mathrm{err}_{\mathcal{Q}}(\zeta^*)\}, \quad (1)$$

where $\mathrm{err}_{\mathcal{Q}}(\zeta)$ denotes the population loss of ζ under the target distribution $\mathcal{Q}$, i.e., $\mathrm{err}_{\mathcal{Q}}(\zeta) \triangleq \mathbb{E}_{(\mathbf{x}_i, \mathbf{y}_i)\sim\mathcal{Q}}[\ell(\zeta(\mathbf{x}_i), \mathbf{y}_i)]$, and we use the parallel notation $\mathrm{err}_{\mathcal{P}}(\zeta)$ for the source domain error. Besides, $d_{\mathcal{H}}(\mathcal{P}, \mathcal{Q})$ is a discrepancy-based distance, known as the $\mathcal{H}$-divergence, and $\min_{\zeta^* \in \mathcal{F}}\{\mathrm{err}_{\mathcal{P}}(\zeta^*) + \mathrm{err}_{\mathcal{Q}}(\zeta^*)\}$ is the optimal joint error, i.e., the sum of source and target domain's population loss of ζ in a *hypothesis* class $\mathcal{F}$. For the unsupervised domain adaptation problem, it has been proven that minimizing the upper bound, which is the r.h.s in (1), leads to an architecture consisting of a *feature extractor* parameterized by ω, i.e., ζ^1_ω, a *label predictor*, parameterized also by ω i.e., ζ^2_ω ($\zeta_\omega \triangleq \zeta^2_\omega \circ \zeta^1_\omega$), [1] and a *domain classifier* parameterized by ψ, i.e., h_ψ, as shown in Fig 2 [6, 43]. The feature extractor generates the domain-independent feature representations, which are then fed into the domain classifier and label predictor. The domain classifier then tries to determine whether the extracted features belong to the source or target domain. Meanwhile, the label predictor predicts instance labels based on the extracted features of the labeled source-domain instances.

Minimizing the upper bound in (1) encourages the extracted features to be both discriminative and invariant to changes between the source and target domains. The upper bound minimization corresponding to a saddle point over the parameter space of ω and ψ has been demonstrated using $\widehat{\omega} \triangleq \arg\min_\omega L_1(\omega) - \nu L_2(\omega, \widehat{\psi})$ and $\widehat{\psi} \triangleq \arg\min_\psi L_2(\widehat{\omega}, \psi)$ with an equivalent minimax form as

$$\min_\omega \max_\psi F = \min_\omega \max_\psi L_1(\omega) - \nu L_2(\omega, \psi). \quad (2)$$

In the majority of adversarial domain adaptation problems, $L_1(\omega) \triangleq \mathbb{E}_{(\mathbf{x}_i, \mathbf{y}_i)\sim\mathcal{Q}}[\ell(\zeta_\omega(\mathbf{x}_i), \mathbf{y}_i)]$ is the supervised learning loss on ζ,$L_2(\omega,\psi)$

[1]The parameters of ζ^1 and ζ^1 are not the same. In this case, we abuse the notation to simplify the expression.

$\triangleq \mathbb{E}_{(\mathbf{x}_i)\sim\mathcal{Q}} D_Q(h_\psi\left(\zeta'_\omega\left(\mathbf{x}_i\right)\right) - \mathbb{E}_{(\mathbf{x}_i)\sim\mathcal{P}} D_P(h_\psi\left(\zeta'_\omega\left(\mathbf{x}_i\right)\right)$ is the domain classification loss, and ν is the trade-off coefficient between $L_1(\omega)$ and $L_2(\omega,\psi)$. With the commonly used cross-entropy loss for L_2, we have $D_Q(x) \triangleq 1-\log(x)$ and $D_P(x) \triangleq \log(1-x)$. Besides, ζ'_ω is the feature and $h_\psi(\cdot):\mathbb{R}^D \to [0,1]$ is the probabilistic prediction of the domain label. In general, ζ'_ω and $h_\psi(\cdot)$ include, but is not limited to, the following cases: (i) Domain-Adversarial Neural Networks (DANN) [6]: In DANN, the input of $h_\psi(\cdot)$ is designed simply to be the domain invariant feature $\zeta^1_\omega(\mathbf{x}_i)$, i.e., $h_\psi\left(\zeta^1_\omega\left(\mathbf{x}_i\right)\right)$. (ii) Margin Disparity Discrepancy (MDD) [41]: In MDD, the input of $h_\psi(\cdot)$ is the concatenation of ζ^1_ω and $\arg\max_c \zeta_\omega\left(\mathbf{x}_i;c\right)$ with c the class type i.e., $h_\psi\left(\left[\zeta^1_\omega\left(\mathbf{x}_i\right), \arg\max_c \zeta_\omega\left(\mathbf{x}_i;c\right)\right]\right)$. (iii) Conditional Domain Adaptation Network (CDAN) [17]: In CDAN, the input of h_ψ is from the cross-product space of $\zeta^1_\omega(\mathbf{x}_i)$ and $\zeta_\omega(\mathbf{x}_i)$, i.e., $h_\psi\left(\zeta^1_\omega(\mathbf{x}_i)\otimes\zeta_\omega(\mathbf{x}_i)\right)$.

Our FedMM is a generic federated adversarial domain adaptation framework in which each client is equipped with h_ψ and ζ_ω depending on the availability of source data, target data, or both.

The objective function in an adversarial domain adaptation problem is determined by whether the data is from the source domain or the target domain, i.e.,

$$F_i\left(\omega,\psi;\xi^{(i)}_j\right) \triangleq \begin{cases} \ell\left(\zeta_\omega\left(\mathbf{x}_i\right),\mathbf{y}_i\right) + \nu\log(1- \\ \qquad h_\psi\left(\zeta'_\omega\left(\mathbf{x}_i\right)\right)), & \text{if } \xi_i \in \mathcal{P}, \\ \nu\log(h_\psi\left(\zeta'_\omega\left(\mathbf{x}_i\right)\right)), & \text{if } \xi_i \in \mathcal{Q}. \end{cases} \tag{3}$$

2.2 Federated Learning under Domain Shifts

We focus on the cross-silo FL adversarial domain adaptation problem, in which the training dataset is distributed across silos in a multi-organizational context, such as in healthcare, banking, finance and so on, where institutions hold users' data but cannot share it directly with other institutions for collaborative learning. As a result, federated adversarial domain adaptation addresses the problem by training a model among clients from a labeled source domain to an unlabeled target domain. A centralized server coordinates between the clients to solve the learning task. To express the federated adversarial domain adaptation objective, we convert the joint learning objective of (2) into the form of a centralized average of all the clients' objective functions, as given by

$$\begin{aligned} &\min_\omega \max_\psi f(\omega,\psi) \triangleq \min_\omega \max_\psi \frac{1}{N}\sum_{i=1}^N f_i(\omega,\psi) \\ &= \min_\omega \max_\psi \frac{1}{N}\sum_{i=1}^N \alpha_i \sum_{\xi^{(i)}_j \in \mathcal{D}_i} F_i\left(\omega,\psi;\xi^{(i)}_j\right), \end{aligned} \tag{4}$$

where N is the number of clients, $f_i(\omega,\psi)$ is the loss function at the i-th client, α_i is the weight coefficient, and $F_i\left(\omega,\psi;\xi_j\right)$ is the loss function w.r.t the data point $\xi^{(i)}_j \triangleq \{\mathbf{x}_j,\mathbf{y}_j\}$ with specific form determined by whether the data is from the source domain or the target domain.

This novel problem structure introduces several unique challenges in federated adversarial domain adaptation that do not exist in existing adversarial domain adaptation problems or the FL literature: (i) Clients are restricted to compute the minimax optimization in a distributed manner rather than the centralized minimax optimization. (ii) To train a centralized model, both the set of feature extractor variables ω and domain classifier variables ψ are non-separable cross clients.

(iii) The marginal label distributions are class-imbalanced across clients due to the imbalanced distribution of source domain data and target domain data. In extreme cases, each client may only have access to data from the target domain or the source domain, but not both.

To address the above unique challenges in federated domain adaptation, we propose FedMM, a general algorithm that works for minimax optimization under FL. FedMM is designed for imbalanced label classes among clients in federated minimax training, a unique problem in domain adaptation.

3 FEDMM ALGORITHM

In this section, we look at the federated minimax problem by reformulating the centralized problem in (4) into the federated saddle-point optimization problem with consensus constraints given by

$$\begin{aligned} &\min_{\omega_0,\omega_i} \max_{\psi_0,\psi_i} \quad f(\omega,\psi) = \frac{1}{N}\sum_{i=1}^N f_i(\omega_i,\psi_i) \\ &\text{s.t.} \quad \omega_i = \omega_0, \quad \psi_i = \psi_0, \quad \forall i \in [N]. \end{aligned} \tag{5}$$

The corresponding augmented Lagrangian form for each client is defined as

$$\begin{aligned} &\mathcal{L}_i(\omega_0,\omega_i,\lambda_i,\psi_0,\psi_i,\beta_i) \triangleq f_i(\omega_i,\psi_i) + \langle\lambda_i,\omega_i-\omega_0\rangle \\ &+\frac{\mu_1}{2}\|\omega_i-\omega_0\|_2^2 - \langle\beta_i,\psi_i-\psi_0\rangle - \frac{\mu_2}{2}\|\psi_i-\psi_0\|_2^2, \quad \mu_1,\mu_2>0. \end{aligned} \tag{6}$$

The centralized optimization problem in (4) is then transformed into a saddle-point minimax optimization of augmented Lagrangian functions over all primal-dual pairs, i.e., $\{\omega_i,\omega_0,\lambda_i,\psi_i,\psi_0,\beta_i\}$ for all clients $i \in [N]$:

$$\begin{aligned} &\min_{\omega_0,\omega_i}\max_{\psi_0,\psi_i} \mathcal{L}\left(\{\omega_i\}_{i=0}^N,\{\psi_i\}_{i=0}^N,\{\lambda_i\}_{i=1}^N,\{\beta_i\}_{i=1}^N\right) \\ &\triangleq \min_{\omega_0,\omega_i}\max_{\psi_0,\psi_i} \frac{1}{N}\sum_{i=1}^N \mathcal{L}_i(\omega_0,\omega_i,\psi_0,\psi_i,\lambda_i,\beta_i). \end{aligned} \tag{7}$$

By fixing the global consensus variables $\{\omega_0,\psi_0\}$, the above problem is separable w.r.t local pairs $\{\omega_i,\psi_i,\lambda_i,\beta_i\}$ for all $i \in [N]$. And the decomposed task could be independently updated on local clients periodically without global communication. The only problem left is to align the update of global consensus ω_0,ψ_0 and local updates ω_i,ψ_i for all $i \in [N]$. Next, we demonstrate how to achieve distributed local updates and align local updates with global consensus. By substituting (6) into (7), we obtain the augmented Lagrangian functions over all primal-dual parameters:

$$\begin{aligned} &\min_{\omega_i}\max_{\psi_i} \mathcal{L}\left(\{\omega_i\}_{i=0}^N,\{\psi_i\}_{i=0}^N,\{\lambda\}_{i=1}^N,\{\beta\}_{i=1}^N\right) \\ &= \frac{1}{N}\sum_{i=1}^N \min_{\omega_i}\max_{\psi_i}\Big(f_i(\omega_i,\psi_i) + \langle\lambda_i,\omega_i-\omega_0\rangle \\ &+\frac{\mu_1}{2}\|\omega_i-\omega_0\|_2^2 - \langle\beta_i,\psi_i-\psi_0\rangle - \frac{\mu_2}{2}\|\psi_i-\psi_0\|_2^2\Big), \end{aligned} \tag{8}$$

where μ_1 and μ_2 are the penalty parameters. The minimax optimization w.r.t the global consensus variable ω_0 and ψ_0 is given by:

$$\begin{aligned}\widehat{\omega}_0 =& \arg\min_{\omega_0} \frac{1}{N}\sum_{i=1}^{N}\Big(f_i(\omega_i,\psi_i) + \langle\lambda_i, \omega_i - \omega_0\rangle \\ &+\frac{\mu_1}{2}\|\omega_i-\omega_0\|_2^2 - \langle\beta_i, \psi_i - \psi_0\rangle - \frac{\mu_2}{2}\|\psi_i-\psi_0\|_2^2\Big) \\ =&\frac{1}{N}\sum_{i=1}^{N}\left(\omega_i + \frac{1}{\mu_1}\lambda_i\right), \end{aligned} \tag{9}$$

where the closed-form solution is due to the quadratic optimization. Similarly, we obtain

$$\widehat{\psi}_0 = \frac{1}{N}\sum_{i=1}^{N}\left(\psi_i + \frac{1}{\mu_2}\beta_i\right). \tag{10}$$

Eqn. (9) and (10) provide guidance for local update alignment with global consensus. More specifically, in each round, we optimize each client's individual ω_i and ψ_i, by fixing the global consensus constraints (ω_0 and ψ_0) and dual parameters (λ_i and β_i). Taking the $(t+1)$-th round update as an example. Client i receives the global parameters $\{\omega_0^t, \psi_0^t\}$ from the server and sets local parameters $\widehat{\omega}_i^0 = \omega_0^t, \widehat{\psi}_i^0 = \psi_0^t$.[2] Then, the local saddle-point optimization of (8) w.r.t $\{\omega_i, \psi_i\}$ is updated by multiple-step SGDA to reduce the communication rounds between a client and the server:

$$\begin{aligned}\widehat{\omega}_i^{m+1} &= \widehat{\omega}_i^m - \eta_1\nabla_{\omega_i}\mathcal{L}_i(\widehat{\omega}_i^m, \widehat{\psi}_i^m) \\ &= \omega_i^m - \eta_1\left[\nabla_{\omega_i}f_i(\widehat{\omega}_i^m, \widehat{\psi}_i^m) + \mu_1(\widehat{\omega}_i^m - \omega_0^t) + \lambda_i^t\right]\end{aligned} \tag{11}$$

$$\begin{aligned}\widehat{\psi}_i^{m+1} &= \widehat{\psi}_i^m + \eta_2\nabla_{\psi_i}\mathcal{L}_i(\widehat{\omega}_i^m, \widehat{\psi}_i^m) \\ &= \widehat{\psi}_i^m + \eta_2\left[\nabla_{\psi_i}f_i(\widehat{\omega}_i^m, \widehat{\psi}_i^m) - \mu_2(\widehat{\psi}_i^m - \psi_0^t) - \beta_i^t\right],\end{aligned} \tag{12}$$

where $m \in [M_i]$. We denote $\omega_i^{t+1} = \widehat{\omega}_i^{M_i}$ and $\psi_i^{t+1} = \widehat{\psi}_i^{M_i}$ for the results of M_i-th step local update. The dual parameters are then updated using SGDA by

$$\lambda_i^{t+1} = \lambda_i^t + \mu_1(\omega_i^{t+1} - \omega_0^t), \tag{13}$$

$$\beta_i^{t+1} = \beta_i^t + \mu_2(\psi_i^{t+1} - \psi_0^t). \tag{14}$$

To align with the global consensus constraint obtained in (9) and (10), we set

$$\omega_i^{t+} = \omega_i^{t+1} + \frac{\eta_3^t}{\mu_1}\lambda_i^{t+1}; \quad \psi_i^{t+} = \psi_i^{t+1} + \frac{\eta_3^t}{\mu_2}\beta_i^{t+1}. \tag{15}$$

Note that different from vanilla augmented Lagrangian, we introduce the decay factor $\eta_3^t < 1$, which helps FedMM converge with smaller local steps. Therefore, the global consensus constraint is satisfied by the global update at the server with

$$\omega_0^{t+1} = \frac{1}{N}\sum_{i=1}^{N}\omega_i^{t+}, \quad \text{and} \quad \psi_0^{t+1} = \frac{1}{N}\sum_{i=1}^{N}\psi_i^{t+}. \tag{16}$$

We can now summarize one round of the FedMM algorithm, which consists of three major steps: (i) Parallel saddle-point optimization on all local augmented Lagrangian function $\mathcal{L}_i$'s. One optimization oracle example is based on SGDA, as shown in (11) and (12). (ii) Local gradient descent and ascent updates on dual variable ($\{\beta_i, \lambda_i\}$) as shown in (14). (iii) Aggregation to update global consensus variables $\{\omega_0, \psi_0\}$ in (16). After one round of global communication. The global coordinated value of $\{\omega_0^{t+1}, \psi_0^{t+1}\}$ is then broadcasted back to each client, triggering next-round updates. The summarized diagram and algorithm of FedMM is shown in Fig. 3 and Algorithm 1

Algorithm 1 FedMM Algorithm

Require: Initialize $\omega_0^0, \psi_0^0, \mu_1, \mu_2, \eta_1, \eta_2, \eta_3, \{M_i\}_{i=0}^N, T$

for $t = 0, \dots, T-1$ **do**
  for each client $i \in [N]$ `in parallel` **do**
    $\widehat{\omega}_i^0 = \omega_0^t; \quad \widehat{\psi}_i^0 = \psi_0^t$
    # Local Update:
    for $m = 0, \dots, M_i - 1$ **do**
      # Stochastic Gradient Descent:
      $\widehat{\omega}_i^{m+1} = \widehat{\omega}_i^m - \eta_1[\nabla_{\omega_i}f_i(\widehat{\omega}_i^m, \widehat{\psi}_i^m) + \mu_1(\widehat{\omega}_i^m - \omega_0^t) + \lambda_i^t]$
      # Stochastic Gradient Ascent
      $\widehat{\psi}_i^{m+1} = \widehat{\psi}_i^m + \eta_2[\nabla_{\psi_i}f_i(\widehat{\omega}_i^m, \widehat{\psi}_i^m) - \mu_2(\widehat{\psi}_i^m - \psi_0^t) - \beta_i^t]$
    end for
    $\omega_i^{t+1} = \widehat{\omega}_i^{M_i}; \quad \psi_i^{t+1} = \widehat{\psi}_i^{M_i}$
    # Dual Descent:
    $\lambda_i^{t+1} = \lambda_i^t + \mu_1(\omega_i^{t+1} - \omega_0^t)$
    # Dual Ascent:
    $\beta_i^{t+1} = \beta_i^t + \mu_2(\psi_i^{t+1} - \psi_0^t)$
    $\omega_i^{t+} = \omega_i^{t+1} + \frac{\eta_3^t}{\mu_1}\lambda_i^{t+1}; \quad \psi_i^{t+} = \psi_i^{t+1} + \frac{\eta_3^t}{\mu_2}\beta_i^{t+1}$
  end for
  # Global Update:
  $\omega_0^{t+1} = \frac{1}{N}\sum_{i=1}^N \omega_i^{t+}; \quad \psi_0^{t+1} = \frac{1}{N}\sum_{i=1}^N \psi_i^{t+}$
end for

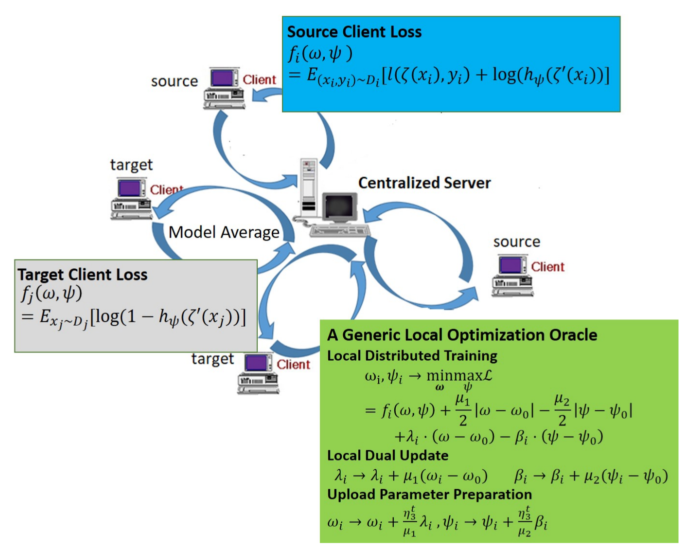


Figure 3: The FedMM algorithm addresses the federated adversarial domain adaptation as shown in the flowchart. Each source and target client has unique local minimax objectives due to domain distribution differences. Clients conduct local optimization, upload parameters to the server, and receive averaged parameter updates in parallel, completing one-round updates.

[2] We use $\{\widehat{\omega}_i, \widehat{\psi}_i\}$ to denote the local iterative updates for $\{\omega_i, \psi_i\}$ to differentiate symbols of local updates and global updates.

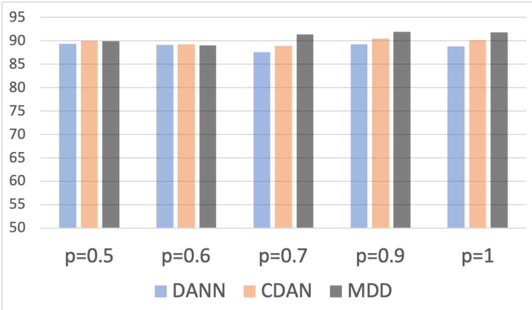


Figure 4: FedMM is robust to label imbalance. Experiments with the same setting as that in Fig. 1.

4 EXPERIMENTS

Our experiments have three main goals. (1) We create a federated domain adaptation benchmark that is communication-efficient by combining three representative domain adaptation methods: DANN [6], MDD [41], and CDAN [17]. (2) We demonstrate the strength of our proposed FedMM algorithm on the Federated domain adaptation benchmark, showing how our proposed FedMM improves model generalization while significantly reducing communication rounds. (3) We investigate the contributions of various components of our algorithm, such as multi-step local updates, proximal terms in local objectives, dual variables, and the choice of η_3. Our experiments are primarily concerned with the training communication overhead and test accuracy on the label-free target data set. Appendix B contains detailed descriptions of the dataset used in this experiment.

Experiment Setup. On MNISTM, we use a three-layer convolutional network as the invariant feature extractor, and the network models are trained from random initialization on server. On Office-31, we use the pre-trained MobileNetV2 [28] on ImageNet [27] as the feature extractor. For a fair comparison, the pre-trained MobileNetV2 is downloaded from the pre-trained one by [28]. Both the task classifier and the domain classifier are two-layer fully-connected neural networks. The domain classifier's parameter are trained from random initialization in all settings. The hyper-parameter settings are provided in Appendix C and our code is available at
github.com/fedmm/FedMM.

4.1 Ablation Experiment on FedMM

We first investigate how a change in label imbalance affects model training performance. Consider the case of two clients and set the ratios for source and target data assigned to client 1 as p and $1-p$, respectively. The commonly used adversarial domain adaptation models including DANN [6], CDAN [41] , and MDD [17] are tested separately as the local model. The label imbalance degree varies from $p = 0.5$ to $p = 1$. FedMM is robust to variations in label distribution, as shown in Fig. 4, and performs well even in the worst-case scenario, in which the source domain data and target domain data are allocated to different clients separately, i.e., $p = 1$. In practice, $p = 1$ occurs frequently because different silos contain data from distinct domains. We will focus on the $p = 1$ case for the remainder of the experiment to test the effectiveness of FedMM.

Unlike the traditional augmented Lagrangian method, FedMM introduces $\eta_3 < 1$ for the FL setting to reduce the need for large local update steps M_i for convergence. When $\eta_3 = 1$, as shown in Fig.6, large local steps with $M_i > 50$ are required. With appropriate η_3, one can reduce M_i from 50 to 25 with negligible performance loss for all three adversarial domain adaptation models. Note that if η_3 is less than the feasible range, the outcome will be suboptimal.

4.2 FedMM is the State-of-the-Art of Federated Domain Adaptation

With extensive experiments, we show that FedMM achieves SOTA performance in terms of accuracy and communication overhead. We include the following two kinds of baselines:

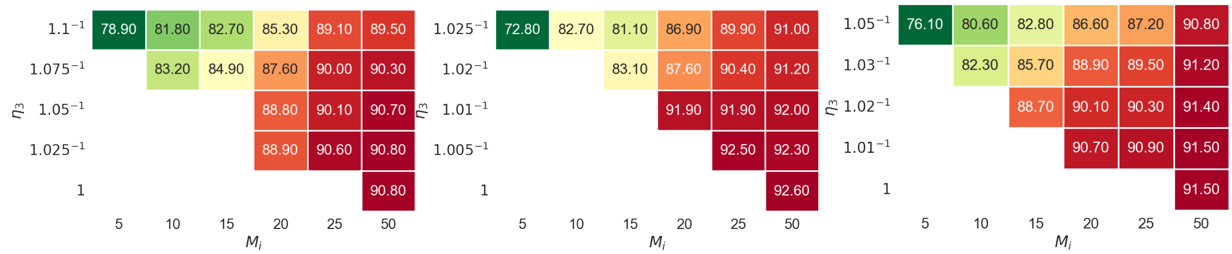
(i) Recent work on distributed minimax optimization including [4, 26, 30, 35] with extensive studies for different adversarial domain adaptation tasks shown in Table 1. As MDD is the SOTA network of centralized domain adaptation in current stage, we only compare these works on federated domain adaptation with MDD loss.

(ii)Peng et al.[22] proposed *FedSGDA* to extend FedSGD to SGDA for federated domain adaptation, where the single step per communication round in SGDA resulted in massive communication overhead as observed in experiments. Though, to the best of our knowledge, there is no communication-efficient federated domain adaptation algorithm. To reduce communication overhead, we inspire from the existing multi-step federated minimization optimizers like FedAvg [20], and FedProx [14], which leads to *FedAvgSGDA* and *FedProxSGDA* in Algorithm 3 as described in Appendix A.

Performance comparison: FedMM has far fewer global communication rounds than FedSGDA, as illustrated in Fig.6. FedMM saves more than 90% of the rounds required to achieve the same level of test accuracy on the target domain as FedSGDA in all three categories of representative domain adaptation networks due to the deliberately designed FedMM's local multi-step minimax optimization at each client.

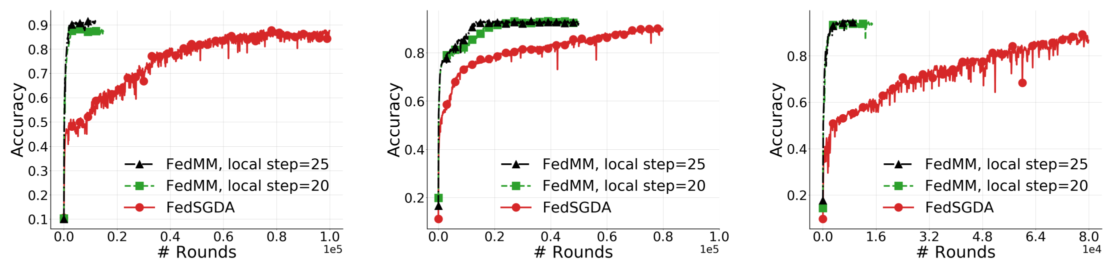
Moreover, FedMM not only reduces communication overhead but also ensures accuracy. Gradient drift causes severe performance degradation in multi-step local SGDA on existing baselines. In Fig. 7, we compare the convergence property of our proposed FedMM to FedAvgSGDA and FedProxSGDA with $M_i = 20$ for different source/target client settings. While both the FedAvgSGDA and FedProxSGDA algorithms converge in a communication-saving manner at the expense of severe performance decay, FedMM consistently outperforms them by more than 20% in terms of test accuracy for training from scratch with all the three domain adaptation networks. The results clearly show how that FedMM addresses the problem of gradient drifts in multi-steps local minimax, which have not been observed in any previous FL problems other than federated adversarial domain adaptation.

We compare FedMM to all of the above baselines on commonly used domain adaptation tasks on Office-31, including $A \rightarrow W$, etc. six tasks with pretrained model, i.e., MobileNetV2 [28] on ImageNet [27]. When compared to the training from scratch case in Fig. 7, it is expected that FedMM's performance improvement will be reduced.



(a) DANN **(b) MDD** **(c) CDAN**

Figure 5: Accuracy heatmap using various pairs of η_3 and M_i. The blank sections represent the settings from which FedMM deviates.



(a) DANN **(b) MDD** **(c) CDAN**

Figure 6: Comparisons of convergence for the proposed FedMM with FedSGDA [22] on different number of local steps of $M_i = 20$ and $M_i = 25$. The comparison is based on different adversarial domain networks, i.e., DANN, MDD, and CDAN. Models are trained from scratch on MNISTM.

Because the feature extractor parameters in these pre-trained models have approached optimal values. FedMM achieves SOTA with significant accuracy improvement while save much communication overhead over single step local update method. Interestingly, even though other multi-step local update distributed minimax optimization methods including [4, 26, 30] save the communication overhead, they do not have a evident performance gain over FedAvgSGDA.

5 CONVERGENCE ANALYSIS

For a theoretical analysis, finding a global saddle point, i.e., $\min_x \max_y f(x,y)$, in general is intractable [16]. One approach is to equivalently reformulate the problem by $\min_x \{\Phi(x) := \max_{y\in\mathcal{Y}} f(x,y)\}$, and define an optimality notion for the local surrogate of global optimum of Φ. A series of theoretical analyses on the stationary point convergence condition of Φ with first-order algorithm were carried out to extend the convex-concave assumption to assumptions of nonconvex-strongly-concave [18, 24, 34], nonconvex-concave [16, 21], and nonconvex-nonconcave [10]. Convergence analysis for a federated optimizer, such as FedMM that involves bounding client's drift from global parameter via primal-dual method, on the other hand, is more challenging. We establish our main convergence results and show that FedMM converges asymptotically to the stationary point for the nonconvex-strongly-concave case.

Let $\psi^\star(\omega) \triangleq \arg\max_\psi f(\omega,\psi)$ be the optimal value of ψ for the global objective function f w.r.t ω. Then (4) can be reformulated as $\min_\omega f(\omega,\psi) = \min_\omega \frac{1}{N}\sum_i \Phi_i(\omega)$ with

$$\Phi_i(\omega) \triangleq f_i(\omega,\psi^\star(\omega)), \quad \Phi(\omega) \triangleq \frac{1}{N}\sum_{i=1}^{N}\Phi_i(\omega). \tag{17}$$

In this way, we equivalently reformulate the problem as $\min_\omega \{\Phi(\omega) = \max_\phi f(\omega,\phi)\}$. To ease the presentation, we further define the augmented Lagrange function of Φ_i by

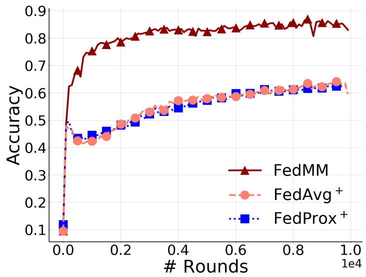
$$\mathcal{L}_i^\Phi(\omega_i^t,\omega_0^t,\lambda_i^t) = \Phi_i(\omega_i^t) + \langle\lambda_i^t, \omega_i^t - \omega_0^t\rangle + \frac{\mu_1}{2}\left\|\omega_i^t - \omega_0^t\right\|^2. \tag{18}$$

5.1 Assumptions

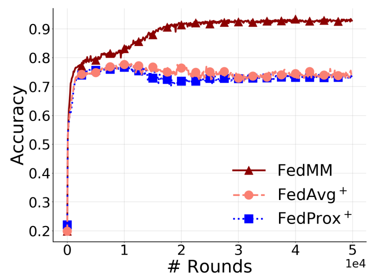
Note that we concentrate on the convergence analysis of the federated nonconvex-strongly-concave case, which is difficult even in a centralized setting and has recently received increased attention in the literature [10, 16, 19]. Thus, we set the standard assumptions by following the minimax optimization literature [10, 15, 16, 19] to impose customary conditions on the gradients of local functions.

Assumption 1. *(Lipschitz continuous gradients) For all $i \in [N]$, there exists positive constants L_{11}, L_{12}, L_{21}, and L_{22} such that for any $\omega, \omega' \in \mathbb{R}^{d_1}$, and $\psi, \psi' \in \mathbb{R}^{d_2}$, we have*

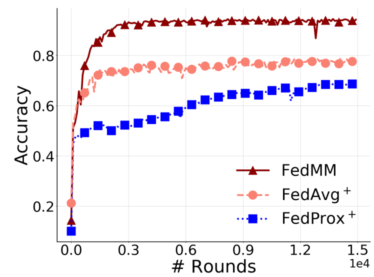
$$\begin{aligned}
\left\|\nabla_{\omega_i} f_i(\omega,\psi) - \nabla_{\omega_i} f_i(\omega',\psi)\right\| &\le L_{11}\left\|\omega-\omega'\right\| \\
\left\|\nabla_{\omega_i} f_i(\omega,\psi) - \nabla_{\omega_i} f_i(\omega,\psi')\right\| &\le L_{12}\left\|\psi-\psi'\right\|, \\
\left\|\nabla_{\psi_i} f_i(\omega,\psi) - \nabla_{\psi_i} f_i(\omega',\psi)\right\| &\le L_{21}\left\|\omega-\omega'\right\| \\
\left\|\nabla_{\psi_i} f_i(\omega,\psi) - \nabla_{\psi_i} f_i(\omega,\psi')\right\| &\le L_{22}\left\|\psi-\psi'\right\|.
\end{aligned}$$



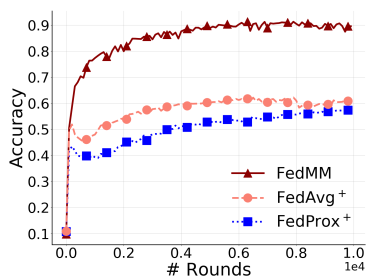
(a) DANN: 1-source/1-target clients



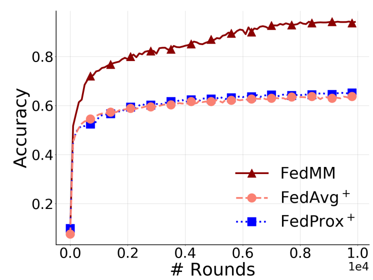
(b) MDD: 1-source/1-target clients



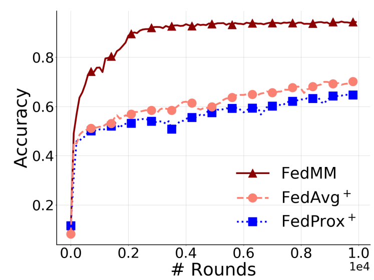
(c) CDAN: 1-source/1-target clients



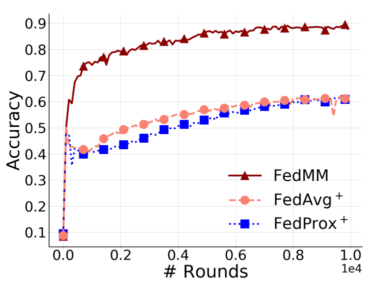
(d) DANN: 1-source/2-target clients



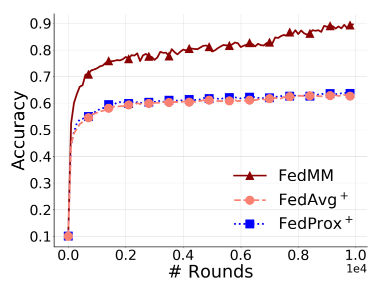
(e) MDD: 1-source/2-target clients



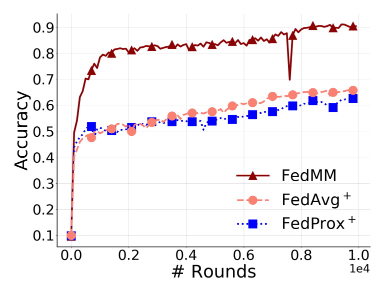
(f) CDAN: 1-source/2-target clients



(g) DANN: 2-source/1-target clients



(h) MDD: 2-source/1-target clients



(i) CDAN: 2-source/1-target clients

Figure 7: Comparisons of convergence for the proposed FedMM with FedAvgSGDA and FedProxSGDA. In the legend, we use FedAvg$^+$ and FedProx$^+$ to denote FedAvgSGDA and FedProxSGDA, respectively, for simplification. Models are trained from scratch on MNISTM with different number of source and target clients

Assumption 2. *(Strongly concave $f_i(\cdot, \psi_i)$) For all $i \in [N]$, $f_i(\omega, \psi)$ are strongly concave on ψ with constant $B > 0$ such that for any $\omega \in \mathbb{R}^{d_1}$, and $\psi, \psi' \in \mathbb{R}^{d_2}$, we have*

$$\left\langle \nabla_\psi f_i(\omega, \psi) - \nabla_\psi f_i(\omega, \psi'), \psi - \psi' \right\rangle \leq -B \left\|\psi - \psi'\right\|^2 . \tag{19}$$

Assumption 3. *The κ-Lipschitz continuity of $\psi^\star(\omega)$, i.e.,*

$$\left\|\psi^\star\left(\omega_i^{t-1}\right) - \psi^\star\left(\omega_i^t\right)\right\| \leq \kappa \left\|\omega_i^{t-1} - \omega_i^t\right\|, \quad \forall t \in [T]. \tag{20}$$

Next, we make the following assumptions that M_i in FedMM is chosen that local objective are sufficiently trained that ω_i^t, ψ_i^t is ϵ stationary on $\mathcal{L}_i$.

Assumption 4. *(Sufficient local training) For all $i \in [N]$, after M_i-step update, the gradients w.r.t. ω_i and ψ_i are finite and denoted by*

$$\left\|\nabla \mathcal{L}_i(\omega_i^t, \psi_i^t)\right\| \leq \epsilon \quad \forall t \in [T]. \tag{21}$$

Theorem 1. *(Convergence on $\Phi(\omega)$) With Assumption 1, 2, 20 and 4 holds. Then there exist positive constants E_1, E_2, and E_3, which are independent of T, such that after T rounds of global updates, the upper bound for the accumulate descent of $\Phi(\omega_0^t)$ is given by*

$$\begin{aligned} &\Phi(\omega_0^0) - \Phi(\omega_0^T) \leq -E_1 \sum_{t=1}^{T} \left\|\nabla\Phi\left(\omega_0^t\right)\right\|^2 + E_3 T\epsilon \\ &+E_2 \sum_{i=1}^{N} \left\|\psi_i^0 - \psi^\star(\omega_i^0)\right\|^2 + E_3 \sum_{i\neq j} \left\|\omega_i^0 - \omega_j^0\right\|^2 . \end{aligned} \tag{22}$$

In particular, this implies $\limsup_{t\to\infty} \left\|\nabla\Phi\left(\omega_0^t\right)\right\| = O(\epsilon)$ with a local residue gradient error bound, i.e., $\left\|\nabla \mathcal{L}_i(\omega_i^t, \psi_i^t)\right\|^2 \leq \epsilon$..

Remark Because the l.h.s. of (22) admits a lower bound, so is the r.h.s. As a result, $\limsup_{T\to\infty} \sum_{t=1}^{T} \left\|\nabla\Phi\left(\omega_0^t\right)\right\|^2$ must converge,

		Office-31 Domain Adaptation Tasks											
		$A \to W$		$D \to W$		$W \to D$		$A \to D$		$D \to A$		$W \to A$	
Optimizer	Network	CR↓	ACC↑	CR↓	ACC↑	CR↓	ACC↑	CR↓	ACC↑	CR↓	ACC↑	CR↓	ACC↑
FedAvgSGDA	DANN	10	60.1	13	86.1	8	93.6	7	63.5	24	33.7	18	40.5
	CDAN	17	62.9	13	86.8	7	**94.2**	21	65.1	34	40.3	14	45.5
	MDD	31	73.2	27	93.6	11	97.8	22	72.1	31	47.9	18	51.7
FedSGDA†	DANN	59	60.3	40	84.9	29	93.7	56	65.3	48	36.9	88	40.3
	CDAN	78	55.3	49	83.4	16	94.0	13	**67.7**	95	47.1	85	43.3
	MDD	255	76.4	188	94.7	92	98.3	400	75.3	300	49.2	321	52.7
[4]	MDD	26	72.9	32	92.5	21	96.3	25	71.9	30	47.2	20	51.0
[26]	MDD	35	71.5	30	93.0	15	95.1	25	73.0	35	48.8	22	51.9
[30]	MDD	–	–	29	82.3	17	78.9	–	–	–	–	–	–
[35]†	MDD	150	77.0	121	95.0	62	98.0	220	74.0	210	51.2	253	50.7
FedMM	DANN	13	**65.5**	15	**89.6**	14	**96.7**	7.5	**67.8**	39	**44.3**	25	**48.7**
	CDAN	29	**64.7**	9	**93.4**	10	94.0	32	66.9	17	**51.4**	13	**59.6**
	MDD	23	**79.7**	18	**95.9**	19	**98.5**	22	**78.8**	19	**60.3**	15	**55.5**

Table 1: Comparison of communication round (CR) (×100) and accuracy (acc) for different optimizers on Office-31 data sets for different domain adaptation tasks ($A \to W,\ldots, W \to A$). The Optimizer's name labeled with † denotes the method with only one step local update in each CR, and all others involve multiple local updates. We uniformly use a local steps of $M_i = 10$ in multi-step local update algorithms. The symbol - denotes the optimizer fails to converge.

which implies that $\Phi\left(\omega_0^t\right)$ converges to a ϵ-stationary point. More specifically, dividing both sides of (22) by T and taking $\lim\sup_{T\to\infty}$, we obtain $\lim\sup_{T\to\infty} \frac{\sum_{t=1}^T \|\nabla\Phi(\omega_0^t)\|^2}{T} \leq \frac{E_3\epsilon}{E_1}$, which implies that $\sum_{t=1}^T \left\|\nabla\Phi\left(\omega_0^t\right)\right\|^2 = O(T\epsilon)$ and for sufficiently large t, $\left\|\nabla\Phi\left(\omega_0^t\right)\right\|^2 = O(\epsilon)$. In the special case of $\epsilon = 0$, i.e., strict optimality is obtained at each local client, this result shows that the limiting point is a stationary point.

6 RELATED WORK

Distributed minimization: Since the invention of FedSGD and its communication efficient version FedAvg [20], several work have been developed to address the suboptimality of FedAvg over non-i.i.d data, including FedProx [14], FedPD [40], SCAFFOLD [12], FedNova [32], dynamic gradient aggregation [5], and FedDyn [1]. These works aim to minimize a sum of non-identical functions, where each function can only be accessed locally. Auto-FedAvg [33] adjusted weights at the aggregation during training. These results cannot be directly applied to federated saddle point optimization problems, such as the federated adversarial domain adaptation, which seeks a federated minimax optimization. Similarly, off-the-shelf distributed augmented Lagrangian minimization methods and convergence analyses in Jakovetić et al. [8, 9], Yue et al. [37] fails to address the unique challenges in FL minimax optimization including the domain adaptation problem.

Distributed minimax: Several works [4, 25, 26] have improved communication efficiency in FedAvg-based minimax optimization, including the federated GAN. However, FedAvgSGDA is sensitive to data imbalance in our federated domain adaptation problem, unlike in the federated GAN where the binary classification function works well since there is no label-imbalanced problem. Note that the FLRA algorithm in Reisizadeh et al. [26] also corresponds to FedAvgSGDA, and its convergence analysis cannot be directly adapted to our setting because each local client in our study is optimized on the augmented Lagrangian local function rather than the pure local risk minimization objective. There are also works on federated domain adaptation with the specific assumption that a centralized labeled dataset is available on the server [36]. Other works, including [30, 35, 39], look into fully distributed minimax optimization without a centralized server. Note that the work in Zhang et al. [39] assumes only one participating client each time which is not realistic under FL settings.

7 CONCLUSIONS

We have proposed FedMM for federated adversarial domain adaptation. FedMM is designed specifically for federated minimax optimizations with non-separable minimization and maximization variables, as well as clients with uneven label class distributions. We have symptomatically performed a theoretical analysis on the convergence property of our proposed FedMM. Experiments show that FedMM outperforms state-of-the-art algorithms in terms of communication rounds and test accuracy on various benchmark datasets.

8 ACKNOWLEDGEMENT

This research was supported in part by NSF through grant IIS-1910492. Han Zhao would like to thank the support from a Facebook research award and Amazon AWS Cloud Credits. Jian Du's work was conducted during his time at Ant Group.

REFERENCES

[1] Durmus Alp Emre Acar, Yue Zhao, Ramon Matas Navarro, Matthew Mattina, Paul N Whatmough, and Venkatesh Saligrama. 2021. Federated learning based on dynamic regularization. In *International Conference on Learning Representations*.

[2] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. 2010. A theory of learning from different domains. *Machine learning* 79, 1 (2010), 151–175.

[3] Wei-Ning Chen, Peter Kairouz, and Ayfer Özgür. 2020. Breaking the Communication-Privacy-Accuracy Trilemma. *arXiv preprint arXiv:2007.11707* (2020).

[4] Yuyang Deng and Mehrdad Mahdavi. 2021. Local stochastic gradient descent ascent: Convergence analysis and communication efficiency. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 1387–1395.

[5] Dimitrios Dimitriadis, Kenichi Kumatani, Robert Gmyr, Yashesh Gaur, and Sefik Emre Eskimez. 2021. Dynamic Gradient Aggregation for Federated Domain Adaptation. *arXiv preprint arXiv:2106.07578* (2021).

[6] Yaroslav Ganin and Victor Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*. PMLR, 1180–1189.

[7] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *The journal of machine learning research* 17, 1 (2016), 2096–2030.

[8] Dušan Jakovetić, Dragana Bajović, João Xavier, and José MF Moura. 2020. Primal–dual methods for large-scale and distributed convex optimization and data analytics. *Proc. IEEE* 108, 11 (2020), 1923–1938.

[9] Dušan Jakovetić, José MF Moura, and Joao Xavier. 2014. Linear convergence rate of a class of distributed augmented lagrangian algorithms. *IEEE Trans. Automat. Control* 60, 4 (2014), 922–936.

[10] Chi Jin, Praneeth Netrapalli, and Michael Jordan. 2020. What is local optimality in nonconvex-nonconcave minimax optimization?. In *International Conference on Machine Learning*. PMLR, 4880–4889.

[11] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Keith Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. 2019. Advances and open problems in federated learning. *arXiv preprint arXiv:1912.04977* (2019).

[12] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. 2020. SCAFFOLD: Stochastic controlled averaging for federated learning. In *International Conference on Machine Learning*. PMLR, 5132–5143.

[13] Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. 2020. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine* 37, 3 (2020), 50–60.

[14] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. 2018. Federated optimization in heterogeneous networks. *arXiv preprint arXiv:1812.06127* (2018).

[15] Tianyi Lin, Chi Jin, and Michael Jordan. 2020. On gradient descent ascent for nonconvex-concave minimax problems. In *International Conference on Machine Learning*. PMLR, 6083–6093.

[16] Tianyi Lin, Chi Jin, and Michael I Jordan. 2020. Near-optimal algorithms for minimax optimization. In *Conference on Learning Theory*. PMLR, 2738–2779.

[17] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. 2017. Conditional adversarial domain adaptation. *arXiv preprint arXiv:1705.10667* (2017).

[18] Songtao Lu, Ioannis Tsaknakis, Mingyi Hong, and Yongxin Chen. 2020. Hybrid block successive approximation for one-sided non-convex min-max problems: algorithms and applications. *IEEE Transactions on Signal Processing* 68 (2020), 3676–3691.

[19] Luo Luo, Haishan Ye, Zhichao Huang, and Tong Zhang. 2020. Stochastic recursive gradient descent ascent for stochastic nonconvex-strongly-concave minimax problems. *arXiv preprint arXiv:2001.03724* (2020).

[20] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*. PMLR, 1273–1282.

[21] Maher Nouiehed, Maziar Sanjabi, Tianjian Huang, Jason D Lee, and Meisam Razaviyayn. 2019. Solving a class of non-convex min-max games using iterative first order methods. *arXiv preprint arXiv:1902.08297* (2019).

[22] Xingchao Peng, Zijun Huang, Yizhe Zhu, and Kate Saenko. 2019. Federated Adversarial Domain Adaptation. In *International Conference on Learning Representations*.

[23] Joaquin Quiñonero-Candela, Masashi Sugiyama, Neil D Lawrence, and Anton Schwaighofer. 2009. *Dataset shift in machine learning*. Mit Press.

[24] Hassan Rafique, Mingrui Liu, Qihang Lin, and Tianbao Yang. 2018. Non-convex min-max optimization: Provable algorithms and applications in machine learning. *arXiv preprint arXiv:1810.02060* (2018).

[25] Mohammad Rasouli, Tao Sun, and Ram Rajagopal. 2020. Fedgan: Federated generative adversarial networks for distributed data. *arXiv preprint arXiv:2006.07228* (2020).

[26] Amirhossein Reisizadeh, Farzan Farnia, Ramtin Pedarsani, and Ali Jadbabaie. 2020. Robust federated learning: The case of affine distribution shifts. *arXiv preprint arXiv:2006.08907* (2020).

[27] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. 2015. Imagenet large scale visual recognition challenge. *International journal of computer vision* 115, 3 (2015), 211–252.

[28] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. 2018. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4510–4520.

[29] Remi Tachet des Combes, Han Zhao, Yu-Xiang Wang, and Geoffrey J Gordon. 2020. Domain adaptation with conditional distribution matching and generalized label shift. *Advances in Neural Information Processing Systems* 33 (2020).

[30] Ioannis Tsaknakis, Mingyi Hong, and Sijia Liu. 2020. Decentralized min-max optimization: Formulations, algorithms and applications in network poisoning attack. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 5755–5759.

[31] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. 2017. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7167–7176.

[32] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H Vincent Poor. 2020. Tackling the objective inconsistency problem in heterogeneous federated optimization. *arXiv preprint arXiv:2007.07481* (2020).

[33] Yingda Xia, Dong Yang, Wenqi Li, Andriy Myronenko, Daguang Xu, Hirofumi Obinata, Hitoshi Mori, Peng An, Stephanie Harmon, Evrim Turkbey, et al. 2021. Auto-FedAvg: Learnable Federated Averaging for Multi-Institutional Medical Image Segmentation. *arXiv preprint arXiv:2104.10195* (2021).

[34] Wenhan Xian, Feihu Huang, Yanfu Zhang, and Heng Huang. 2021. A faster decentralized algorithm for nonconvex minimax problems. *Advances in Neural Information Processing Systems* 34 (2021).

[35] Ran Xin, Usman Khan, and Soummya Kar. 2021. A hybrid variance-reduced method for decentralized stochastic non-convex optimization. In *International Conference on Machine Learning*. PMLR, 11459–11469.

[36] Chun-Han Yao, Boqing Gong, Hang Qi, Yin Cui, Yukun Zhu, and Ming-Hsuan Yang. 2022. Federated multi-target domain adaptation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 1424–1433.

[37] Sheng Yue, Ju Ren, Jiang Xin, Sen Lin, and Junshan Zhang. 2021. Inexact-admm based federated meta-learning for fast and continual edge learning. In *Proceedings of the Twenty-second International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing*. 91–100.

[38] Kun Zhang, Bernhard Schölkopf, Krikamol Muandet, and Zhikun Wang. 2013. Domain adaptation under target and conditional shift. In *International Conference on Machine Learning*. PMLR, 819–827.

[39] Siqi Zhang, Junchi Yang, Cristóbal Guzmán, Negar Kiyavash, and Niao He. 2021. The complexity of nonconvex-strongly-concave minimax optimization. In *Uncertainty in Artificial Intelligence*. PMLR, 482–492.

[40] Xinwei Zhang, Mingyi Hong, Sairaj Dhople, Wotao Yin, and Yang Liu. 2020. FedPD: A federated learning framework with optimal rates and adaptivity to non-IID data. *arXiv preprint arXiv:2005.11418* (2020).

[41] Yuchen Zhang, Tianle Liu, Mingsheng Long, and Michael Jordan. 2019. Bridging theory and algorithm for domain adaptation. In *International Conference on Machine Learning*. PMLR, 7404–7413.

[42] Han Zhao, Remi Tachet Des Combes, Kun Zhang, and Geoffrey Gordon. 2019. On learning invariant representations for domain adaptation. In *International Conference on Machine Learning*. PMLR, 7523–7532.

[43] Han Zhao, Shanghang Zhang, Guanhang Wu, José MF Moura, Joao P Costeira, and Geoffrey J Gordon. 2018. Adversarial multiple source domain adaptation. *Advances in neural information processing systems* 31 (2018), 8559–8570.