Joint Focal Loss and Dominant Gradient Correction for Gradient Conflict in Federated Learning

Jiajun Wang[b], Yingchi Mao[a, b], Zibo Wang[b], Jun Wu[b], and Jie Wu[c]
[a] Key Laboratory of Water Big Data Technology of Ministry of Water Resources, Hohai University, Nanjing, China
[b] School of Computer and Information, Hohai University, Nanjing, China
[c] Center for Networked Computing, Temple University, Philadelphia, USA
211307040017@hhu.edu.cn, yingchimao@hhu.edu.cn, 221307040022@hhu.edu.cn
1606010225@hhu.edu.cn, jiewu@temple.edu

***Abstract*—Federated learning faces significant challenges of data heterogeneity among clients. In this heterogeneous data scenario, the preference of local models can result in gradient conflict during the global aggregation phase. This can significantly lead to a decrease in the global accuracy. In this paper, we present a novel Federated Learning Mitigating Gradient Conflict method named FedMGC that aims to mitigate gradient conflict. FedMGC replaces the cross-entropy loss function with the focal loss function. This balances the proportion of each class in the loss and reduces the preference of the local models for majority classes. In the global aggregation phase, we design the dominant gradient correction method called DGC to improve the global accuracy. Specifically, we select some of the gradients with small outliers to form the dominant gradients. And then we use dominant gradients to adjust the local gradients and alleviate the gradient conflict. In the evaluation experiments on heterogeneous datasets, FedMGC achieves higher test accuracy compared to baselines. In particular, over the CIFAR-10 dataset, FedMGC achieves 3.88%, 1.7% and 1.31% higher test accuracy than those of FedAvg, FedProx and FedNova, respectively.**

***Index Terms*—Federated Learning, Gradient Conflict, Focal Loss, Dominant Gradients.**

I. Introduction

As a new distributed framework, federated learning [1] [2] enables multiple clients to train models with non-disclosure of privacy. However, federated learning also faces the major challenge about data heterogeneity [3] [4]. Data heterogeneity is a common environment involved in federated learning systems, $i.e.$, the data distribution among clients is usually non independent and identically distributed (Non-IID). For example, the label distribution and the feature distribution varies across clients [5]. Heterogeneous data leads to the generation of discrepant local models [6], and the phenomenon of local gradients conflict while aggregating gradients is called gradient conflict. Gradient conflict degrades the global accuracy, so we try to alleviate the gradient conflict.

Fig. 1 shows the gradient conflict across three clients, where the green, red and blue ellipses represent the loss space of the three clients. Due to various label distribution, the local models are optimized toward their respective optimum w_1^*,w_2^*,w_3^*, and there is a large difference among the local gradients. Most of the current federated learning methods only perform a simple weighted average on the uploaded local gradients in the global aggregation phase. However, the angle $< g_1, g_2 >$ between the gradient g_1 and the gradient g_2 is so large that results in $g_1 \cdot g_2 < 0$. If the gradients g_1, g_2 are added directly, gradient conflict may occur. Similarly, there is gradient conflict between the gradient g_2 and the gradient g_3, while g_1 and g_3 form a smaller angle $< g_1, g_3 >$, so there is no gradient conflict between g_1 and g_3. Due to the gradient conflict, the updated direction of the aggregated model deviates from the actual optimum, $i.e.$, the yellow arrow is distant from w^*, and the global accuracy is harmed.

At present, most studies lessen the discrepancy of local models to attenuate the gradient conflict. Model-level comparison learning was employed by Li $et\ al.$ [7] to correct local training by reducing the gap among the representations obtained by the current, previous, and global models. Li $et\ al.$ [8] utilized correction terms to suppress device parameter divergence during local training. Karimireddy $et\ al.$ [9] presented SCAFFOLD, which was based on control variables that reduced the discrepancy between local and global models. Wang $et\ al.$ [10] proposed FedNova, which standardized and adjusted local updates in accordance with the quantity of local iterations. However, these methods fail to consider the model preference caused by the class imbalance [11]. Therefore, the local models trained by these methods still perform poorly for minority classes and well for majority classes. Moreover, the gradient conflict is not handled during global aggregation, resulting in a lower global accuracy.

In addition, some works also focus on the global aggregation optimization. Wang $et\ al.$ [12] conducted momentum updates in the global aggregation phase after clients performed multiple local iterations. Reddi $et\ al.$ [13] employed adaptive approaches to make the global aggregation smoother. Yeganeh $et\ al.$ [14] proposed an inverse distance aggregation method, which allowed clients to obtain higher weights, thus reducing the distance among the models. Although the above methods make the global aggregation smoother and increase the global accuracy, the gradient conflict problem is not properly solved.

Motivated by these observations, we propose FedMGC to mitigate the gradient conflict and enhance the global accuracy. FedMGC first replaces the cross-entropy (CE) loss function with the focal loss (FL) function to amplify the loss impact of minority classes. By using the FL function to equalize the proportion of each class in the loss, FedMGC mitigates

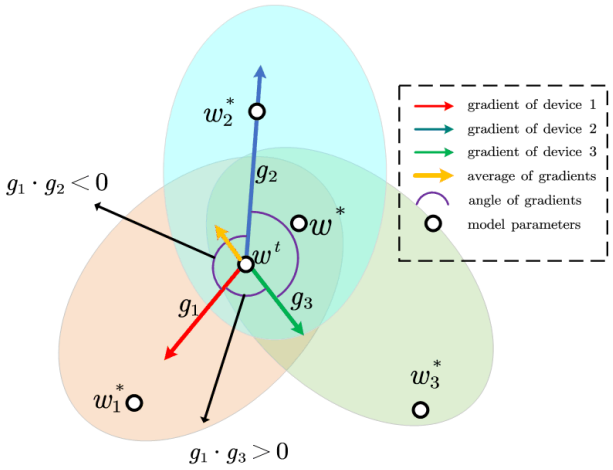

Fig. 1. Schematic diagram of gradient conflict.

the difference in recognition performance of local models on various classes and provides high-quality local gradients for the parameter server. Secondly, we design the DGC strategy in the global aggregation phase. Specifically, DGC detects and corrects conflicting gradients by using high-quality dominant gradients, thus avoiding gradient conflict and increasing the global accuracy. Our main contributions are as follows:

- We reformulate the FL function to balances the contribution of each class to the loss and alleviates the differences in the local model's recognition performance for each class.
- We propose FedMGC to mitigate the gradient conflict. FedMGC mitigates the difference of local model recognition performance for each class through the FL function in the local training phase and provides high quality local gradients for the parameter server. In the global aggregation phase, DGC selects and corrects the dominant gradients and conflicting gradients respectively.
- Experiments on CIFAR-100, CIFAR10 and FMNIST (Fashion-MNIST) datasets show that FedMGC can mitigate the gradient conflict and raise the global accuracy.

The structure of our paper is as follows; section II presents the relevant work. The problem formulation and associated concepts are defined in Section III. In section IV, FedMGC is discussed in further depth. Section V contains the experiment analyses. Finally, we summarize the whole paper in Section VI.

II. Related Work

A. Data Augmentation

Data augmentation can transform heterogeneous data into homogeneous data, prevent gradient conflict, and increase the consistency of data distribution across edge devices. To lessen the extent of the local data imbalance, Yoon *et al.* [16] send the average batch of local data exchanged with the client to the server. Hao *et al.* [17] used zero-sample data augmentation for underrepresented data to attenuate data heterogeneity. Zhu *et al.* [18] presented a distillation method where each client generated an augmented representation on the feature space. Duan *et al.* [19] mitigated class imbalance by fraction-based data enhancement and data downsampling. Wu *et al.* [20] executed SMOTE algorithm for low dimensional features of coded networks for clients to generate locally enhanced, class-balanced datasets. Shullar *et al.* [25] integrated the active learning by transferring a modest amount of data amongst clients to lessen the skewness of the data distribution. However, the above data augmentation approaches usually require data exchange or depend on the usability of the proxy data representing the overall data distribution, and the applicability is somewhat limited.

B. Federated Training Optimization

1) Local Training Optimization: A number of works have been produced to enhance the similarity of local gradients to stifle gradient conflict by optimizing the local training. Li *et al.* [7] corrected local updates of clients by injecting projection heads into the model with a model-level comparison learning method. Chen *et al.* [26] proposed a contractible regularization method to avoid local models deviating from the optimal model, thereby ensuring a global aggregate model without bias. With the use of an online learning mechanism and decaying coefficients, Chen *et al.* [21] balanced prior and current gradients. Sannara *et al.* [22] modified the network of local models by finding variations among client neurons. By including a regularization component, Li *et al.* [23] introduced a teacher-student approach to modify local gradients derived from various data distributions. Li *et al.* [8] suppressed the divergence of local model parameters with a correction term. Karimireddy *et al.* [9] presented a control variable-based method to reduce the discrepancy among the local models and the global model, thus reducing the conflict among local gradients. In order to guarantee that the global updates were unbiased, Wang *et al.* [10] presented FedNova, which standardized and adjusted local updates in accordance with the amount of local iterations. According to the previous descriptions, it is found that the divergence of local gradients can be stifled by optimizing the local training. However, the models trained by these methods still results in poor performance for minority classes and good performance for the majority classes. The preference of the local model is still strong. As a result, the gradient conflict problem still arises in the aggregation of models based on weighted averaging, which impairs the global accuracy.

2) Global Aggregation Optimization: Some works cope with gradient conflict through global aggregation optimization. Jeong *et al.* [24] aggregated local model parameters based on local model reliability. Reddi *et al.* [13] employed adaptive approaches to adjust the global model update direction making the model aggregation smoother. Wang *et al.* [13] conducted momentum updates in the model aggregation stage after clients performed multiple local iterations. Yeganeh *et al.* [14] proposed an inverse distance aggregation method, which allowed the client to obtain higher weights, thus shortening the

TABLE I
THE MEANINGS OF THE MAIN SYMBOLS USED IN FEDMGC.

Symbol	Meaning
N	Number of clients
K	Number of client samples per commucation
λ	Dominant gradient selection ratio
g_i	Gradient of client i
g_t	Dominant gradient array in round t
g^t	The global aggregation gradient in round t
γ	Loss of focus parameters
β	Loss of scaling parameters
$F_k(w)$	Empirical risk of client k
w	Model parameter
D_k	Local data of client k
$p_{i,j}$	Gradient projection outlier of client i and client j
p_i	Gradient projection outlier of client i
l_i	Loss of client i
z_i	Gradient outlier of client i

gap among models. Shang $et\ al.$ [27] propose FEDIC, which utilized the calibration distillation to improve the robustness of the models. Although the above methods make the global aggregation smoother and enhance the global accuracy by momentum, adaptive methods and client weight adjustment, the gradient conflict problem is not properly solved.

III. PROBLEMS AND DEFINITIONS

A. Problem Formulation

In federated learning , the network consists of 1 parameter server and N clients. Federated learning aims to train wT across various clients while maintaining the privacy of local data. Since client k has access to dataset D_k only, the objective function is optimized by minimizing the loss function of N clients, which can be formulated as,

$$\min\ f(w) = \frac{1}{N}\sum_{k=1}^{N} F_k(w), \tag{1}$$

where $F_k(w)$ represents the objective function of client k. The empirical risk of the local model is the specific meaning of $F_k(w)$, which is expressed as follows,

$$F_k(w) \ = \ E_{\xi_k\ D_k}\left[f_k(w,\xi_k)\right], \tag{2}$$

where $f_k(\cdot)$ denotes the CE loss function. However, with unbalanced classes among clients, the CE loss function focuses on training for the majority class due to the different sample sizes in each class. Thus the local model may perform better for the majority classes and worse for the minority classes. To address the above problem, we replace the CE loss function with the FL function, so the optimization objective can be expressed as,

$$F_k(w) \ = \ E_{\xi_k\ D_k}\left[fl_k(w,\xi_i)\right], \tag{3}$$

where $fl_k(\cdot)$ denotes the FL function, and the objective function becomes,

$$\min\ f(w) = \frac{1}{N}\sum_{k=1}^{N}\left[E_{\xi_k\ D_k}\left[fl_k(w,\xi_k)\right]\right]. \tag{4}$$

In FedMGC, the parameter server further corrects the received local gradients to mitigate the gradient conflict. The specific local gradients correction method is explained thoroughly in section IV. Table I lists the main symbols used in this paper.

B. Related Definitions of Gradient Conflict

Definition 1. *For $\forall i,j$ and $i \neq j$, there exists the gradient conflict between client i and client j when and only when $g_i \cdot g_j^\top < 0$.*

Definition 2. *For $\forall i,j$ and $i \neq j$, the projection of the gradient g_i in the gradient g_j is $|g_i|\cos\langle g_i, g_j\rangle$*

Definition 3. *For $\forall i,j$ and $i \neq j$, let $p_{i,j} = \frac{1}{2}(|g_i|+|g_j|)\cos\langle g_i, g_j\rangle$ denote the gradient projection outlier of client i and client j. It can be found that $p_{i,j} = p_{j,i}$. In the edge environment consisting of K clients, let $p_i = \frac{1}{K-1}\sum_{j\in[K],j\neq i} p_{i,j}$ denote the gradient projection outlier of client i.*

Definition 4. *For any client i, p_i denotes the gradient projection outlier of client i and l_i denotes the loss value of client i. Let $z_i = \frac{p_i}{l_i}$ denote the gradient outlier of client i.*

IV. FEDMGC

A. Overall Framework

Fig. 2 shows the whole process of FedMGC for dealing with gradient conflict. In the local training phase, FedMGC replaces the CE loss function with the FL function, and improves the contribution of minority classes to the loss. This can equalize the proportion of each class in the total loss, which in turn lessen the disparity in local model performance for each class and provide a high-quality local gradient for the parameter server. In the global aggregation phase, we design the dominant gradient correction approach to alleviate the gradient conflict. The actual procedure is as follows.

(i) Obtain an array of gradient outliers based on the inner product of two gradients and the inverse of the loss value. The server calculates the gradient projection outlier p_k^t for each gradient, which forms the gradient projection outlier array p_t, and then calculates the gradient outlier z_k^t corresponding to each gradient g_k^t based on p_t and l_t and forms the gradient outlier array $z_t = \{z_1^t, ..., z_K^t\}$.

(ii) The server sorts the gradient outlier array $z_t = \{z_1^t, ..., z_K^t\}$ and selects $\lceil \lambda K \rceil$ gradients as the dominant gradients according to the parameter $\lambda \in (0, 1]$.

(iii) Adjusting local gradients by domain gradient adjustment approach.

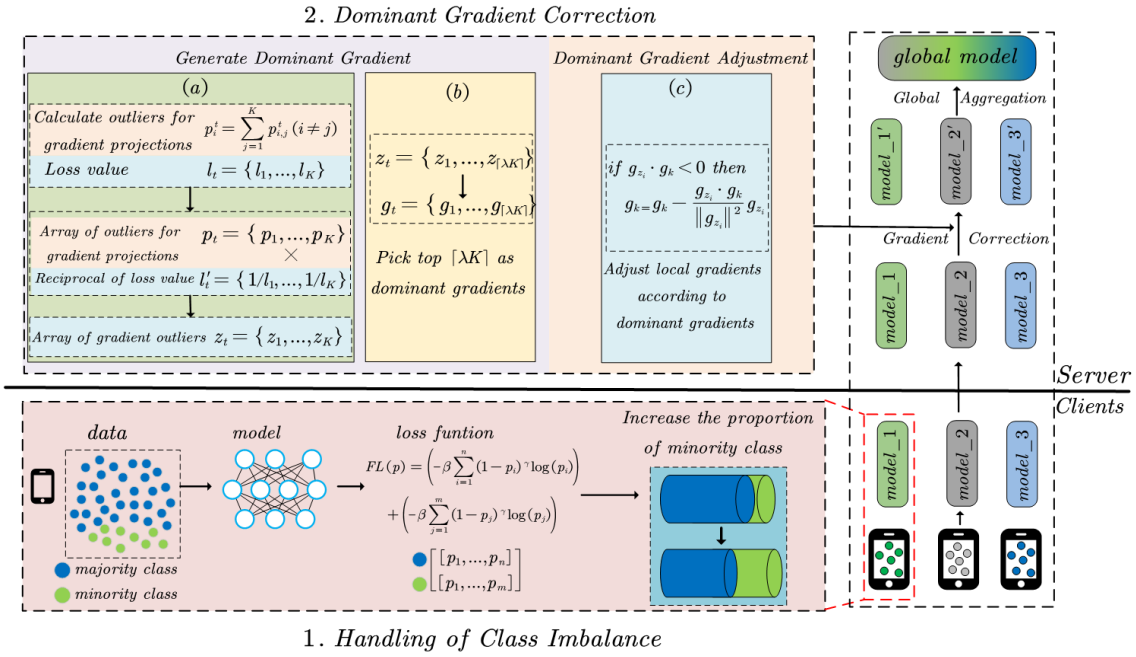

Fig. 2. Framework of FedMGC.

B. Handling of Class Imbalance

FL function is originally used to solve the imbalance of samples in one-stage target detection. We use this method to address the class imbalance.

To better illustrate FL function, we first illustrate the CE loss function in terms of a binary classification, which is represented as,

$$CE\left(p,y\right)=\begin{cases}-\log\left(p\right) & if\ y=1,\\ -\log\left(1-p\right) & if\ y=-1,\end{cases} \tag{5}$$

where $y=\{\pm1\}$ represents the sample class and $p\in[0,1]$ denotes the prediction rate. To simplify the written form, we let,

$$p_t=\begin{cases}p & if\ y=1,\\ 1-p & if\ y=-1.\end{cases} \tag{6}$$

With Eq. 5 and Eq. 6, the CE loss function can be abbreviated to the following form,

$$CE\left(p,y\right)=CE\left(p_t\right)=-\log\left(p_t\right), \tag{7}$$

where p_t represents the prediction probability. Therefore, the derivative of the CE loss function is expressed as,

$$\frac{dCE\left(p_t\right)}{dx}=y\left(p_t-1\right). \tag{8}$$

As shown in Eq. 8, if there is class imbalance within a client, the gradient change of the local model based on the CE loss function may be dominated by easily classifiable majority classes. This results in a weak contribution of minority classes to the gradient, thus making the model poorer in identifying minority classes.

For the disadvantage of the CE loss function for the class imbalance case, the FL function implements a dynamic scaling of the CE loss function. The specific equation of the focal loss function is written as,

$$FL\left(p_t\right)=-\left(1-p_t\right)^{\gamma}\log\left(p_t\right). \tag{9}$$

The FL loss function is reshaped as Eq. 9 to decrease the proportion of the loss for majority classes and increase the proportion of the loss for minority classes.

The modulation factor of the FL function improves the contribution of the minority classes to the total loss and reduces the difference of training loss among various classes. In addition we modify the modulation factor to $-\beta\left(1-p_t\right)^{\gamma}$ to reconstruct the FL as,

$$FL\left(p_t\right)=-\beta\left(1-p_t\right)^{\gamma}\log\left(p_t\right). \tag{10}$$

$\left(1-p_t\right)^{\gamma}<1$ causes the training loss calculated by the FL to be smaller than the training loss calculated by the CE function. By Eq. 10, we can set a suitable β to compensate for the smaller loss value caused by the FL function.

C. Dominant Gradient Correction

1) Dominant Gradient Generation: The purpose of the dominant gradient generation is to select some gradients with a lower gradient outlier from local gradients as the dominant gradients. The specific steps are:

(i) Calculate the gradient projection outliers. The concept of gradient projection and gradient projection outlier is defined in Def. 2 and Def. 3. Fig. 3 shows the process of projecting three gradients g_1,g_2,g_3 to each other and calculating the gradient projection outliers p_1,p_2,p_3. The process can be divided into gradient projection, calculation of the outlier $p_{i,j}$

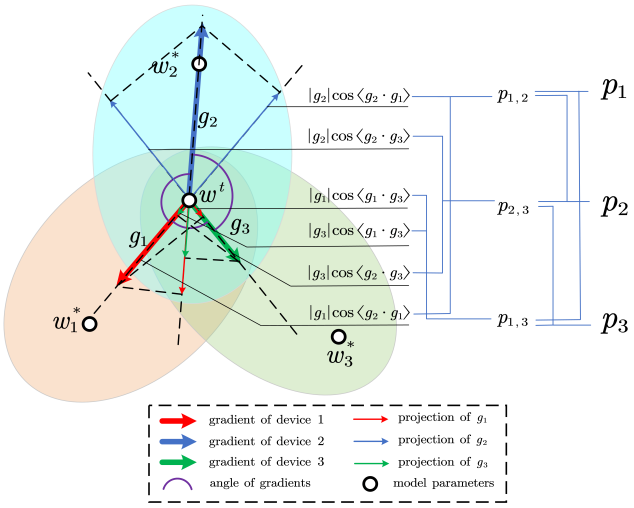

Fig. 3. The process of calculating the gradient projection outliers.

Algorithm 1 Dominant Gradient Generation

Input: Local gradients $g_t = \{g_1^t, ..., g_K^t\}$, loss values $l_t = \{l_1^t, ..., l_K^t\}$, dominant gradient selection ratio λ.
Output: Dominant gradient dg^t.

Initialize $p^t = \{\}$, $z^t = \{\}$
for $g_i^t \in g_t$ **do**
  for $g_j^t \in g_t$ **do**
    $p_{i,j}^t = \left(\frac{g_i \cdot g_j}{\|g_j\|} + \frac{g_j g_i}{\|g_i\|}\right)/2$
  end for
end for
for $i = 1, \cdots, K$ **do**
  add $p_i^t = \sum_{j=1}^{K} p_{i,j}^t$ to p^t
end for
for $i = 1, \cdots, K$ **do**
  add $z_i^t = \frac{p_i^t}{l_i^t}$ to z^t.
end for
sort array z^t in descending order to achieve $z_s = \{z_{s_1}^t, ..., z_{s_K}^t\}$, then choose top $\lceil \lambda K \rceil$ gradients $dg^t = \left\{g_{s_1}^t, ..., g_{s_{\lceil \lambda K \rceil}}^t\right\}$ as dominant gradients which map to z_s.
return dg^t

of mutual projection between two gradients and calculation of the gradient projection outlier p_i of client i. The gradient projection $|g_i| \cos\langle g_i, g_j\rangle$ represents the length of g_i in the direction of g_j. $p_{i,j}$ is the mean value of the mutual projection of g_i and g_j, which indicates the outliers of g_i and g_j in terms of direction and size. The gradient projection outlier p_i is the sum of the mean value of gradient projection of g_i on other gradients, which reflects the degree of anomaly between g_i and the rest of the gradients. If p_i is smaller, it means that the overall anomaly of g_i is larger, whereas if p_i is larger, it means the overall anomaly is smaller.

(ii) Calculate gradient outliers. By dividing the gradient projection outlier array $p = \{p_1, ..., p_K\}$ with the array of loss values $l_t = \{l_1, ..., l_K\}$ one by one as,

$$z_k = \frac{p_k}{l_k}, \tag{11}$$

we get the array of gradient outliers $z = \{z_1, ..., z_K\}$.

(iii) Select dominant gradients. The server sorts the gradient outliers array $z = \{z_1, ..., z_K\}$ in descending order to get $z_s = \{z_{s_1}, ..., z_{s_K}\}$, and the gradients corresponding to the first $\lceil \lambda K \rceil$ gradient outliers are selected as the dominant gradients according to parameter λ. Finally we correct the gradient g_i based on the dominant gradient. Algorithm 1 illustrates the exhaustive procedure of dominant gradient generation.

2) Dominant Gradient Adjustment: The dominant gradient adjustment method takes the dominant gradient as the high-quality gradients and corrects the gradients with the gradient conflict based on the dominant gradients. The workflow is as follows, (i) detects whether there is a gradient conflict between the dominant gradients and the local gradients based on the positive or negative of the inner product of the gradients. (ii) if a gradient has gradient conflict with dominant gradients, we correct the gradient based on the dominant gradients, (iii) aggregate the corrected gradients.

Fig. 4 illustrates the working process of the dominant gradient adjustment approach. As shown in Fig. 4, the gradients g_1,g_2,g_3 are in conflict with each other, and g_2 and g_3 are chosen as the dominant gradients in the case of the same loss value. For g_1, because it conflicts with both g_2 and g_3, we adjust g_1 with g_2 and g_3 respectively. The specific method is to adjust the length of g_2 and g_3 according to the module length of g_1 and its angle with g_2 and g_3 to obtain $g_2^{'}$ and $g_3^{'}$. Add g_1 and $g_2^{'}$to get $g_1^{'}$, and then add $g_1^{'}$ and $g_3^{'}$ to get the adjusted gradient of $g_1^{''}$. At this point the conflict between $g_1^{''}$, g_2 and g_3 becomes significantly weaker. For g_2, although there is a conflict between g_2 and g_1, g_1 is not the dominant gradient, so we only use g_3 to adjust g_2. The specific process is the same as the adjustment of g_1, and we don't repeat it here. After the adjustment, the conflict among the gradients is reduced, which is reflected in the reduction of the angle among the gradients and the reduction of the difference of the mode size among the gradients. Finally, the model is revised

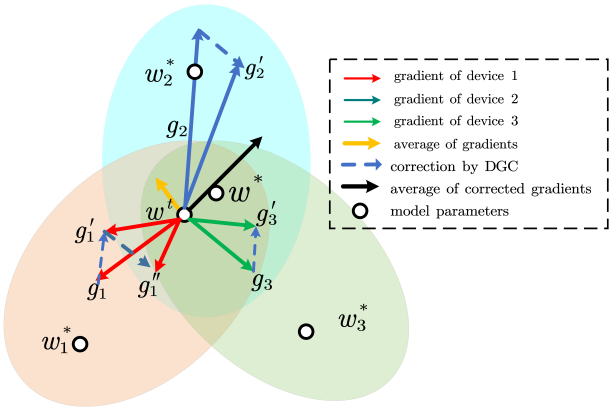

Fig. 4. Illustration of the dominant gradient adjustment process.

Algorithm 2 Dominant Gradient Adjustment

Input: Local gradients $g_t = \{g_1^t, ..., g_K^t\}$, dominant gradients $dg^t = \left\{g_{s_1}^t, ..., g_{s_{\lceil \lambda K \rceil}}^t\right\}$.
Output: Corrected gradients g^t.

Initialize $n = |dg| = \lceil \lambda K \rceil$, $m = |g_t|$, $g_i^{cur} \leftarrow g_i^t$
for $i < m$ **do**
  for $j < b$ **do**
    if $g_i^{cur} \cdot g_{s_j} < 0$ and $i \neq s_j$ **then**
      $g_i^{cur} = g_i^{cur} - \frac{g_i^{cur} \cdot g_{s_j}^t}{\|g_{s_j}^t\|^2} g_{s_j}^t$
    end if
  end for
end for
$g^t = \frac{1}{m} \sum_{i=1}^{m} g_i^{cur}$
return g^t

TABLE II
CIFAR-100, CIFAR-10 AND FMNIST DATASET STATISTICS

Dataset	Data volume	Training sets	Test sets
CIFAR-100	60000	50000	10000
CIFAR-10	60000	50000	10000
FMNIST	70000	60000	10000

in the optimum direction. The details of the dominant gradient adjustment can be found in Algorithm 2.

In general, DGC is divided into two parts. First, a portion of the gradients with low gradient outliers from all local gradients are selected as the dominant gradients. Then, all of the gradients are subjected to gradient conflict detection. If a gradient conflict with the dominant gradients, it is adjusted in accordance with the dominant gradient adjustment approach. After all the gradients have been adjusted, the conflict among these gradients is significantly reduced, and then global aggregation is performed.

V. EXPERIMENTS

A. Experimental Setup

1) Datasets and Models: We evaluate FedMGC over three datasets, CIFAR-100, CIFAR-10, and FMNIST (Fashion-MNIST), and Table II displays the quantity of each dataset utilized for training and testing. We utilize the Dirichlet distribution $q \sim Dir\,(\alpha p)$ to manage the level of data heterogeneity, where p denotes prior class distribution and α determines the level of heterogeneity. When $\alpha \to 0$, the degree of data heterogeneity is strong $i.e.$ class imbalance is severe, and when $\alpha \to \infty$, the data heterogeneity becomes weaker. As shown in Fig. 5(a), When $\alpha \to 0$, each client only own the data from one class. As the α becomes larger, the sample size of each class is balanced.

Different models are designed to evaluate the performance of FedMGC. For CIFAR-100, two convolutional layers are used, a 2*2 maximum pooling layer is built in the middle, followed by three fully connected layers, and finally the prediction results are output by the softmax layer. The network architecture of CIFAR-10 is two convolutional layers, with a 2*2 maximum pooling layer in the middle, followed by three fully connected layers, and finally the prediction results are output by softmax layer. For FMNIST, a two-layer CNN network with 5*5 convolutional kernels is employed. Each layer has a 2*2 maximum pooling layer, and the last layer is a fully connected layer. The results are output by softmax layer.

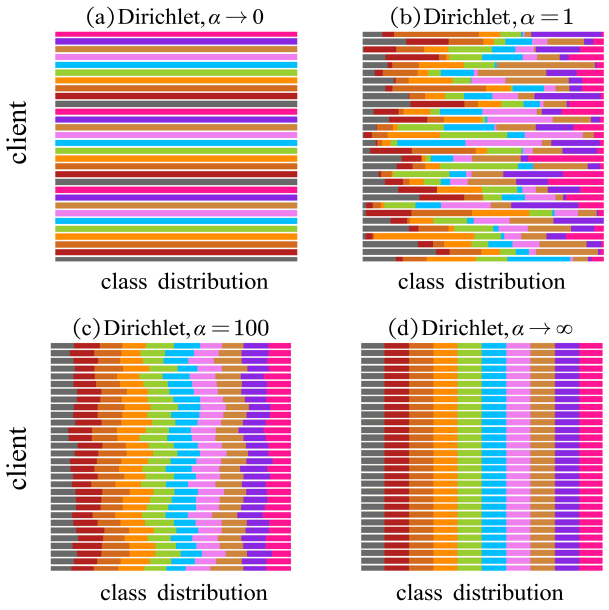

Fig. 5. The percentage of local data belonging to classes for clients in CIFAR-10 dataset.

2) Hyperparameter Settings: In all experiments, we employ the SGD optimizer. The batch size is 128, the local epoch is 10 and the learning rate is 0.001. Because the proportion of the sample size of each class in the client differs when the data heterogeneity and datasets are different, the hyperparameters of the FL function are taken differently, and we make $\gamma \in \{0.1, 0.2, 0.5,\ 1.0\}$ and $\beta \in \{1.0, 1.2, 1.5, 2.0\}$. All the following experiments are performed with γ and β adjusted optimally.

3) Baselines and Validation Metrics: We compare FedMGC with several related methods such as FedAvg [15], FedProx [8], SCAFFOLD [9] and FedNova [10]. We mainly measure the performance of FedMGC in terms of test accuracy, local loss of various classes, and gradient projection outliers.

B. Analysis of Hyperparameter λ

The hyperparameter λ denotes the selection ratio of the dominant gradients, which means that $\lceil \lambda K \rceil$ gradients are selected as the dominant gradients. To investigate the effect of the hyperparameter λ on FedMGC, we selected $\lambda \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ under the heterogeneous data setting of $\alpha = 0.5$ over the CIFAR-10 dataset. We employee 100 clients with an equal amount of data on the clients, and

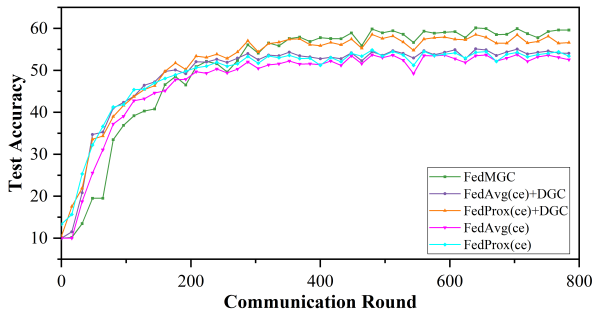

Fig. 6. Test accuracy of FedMGC and related ablation methods.

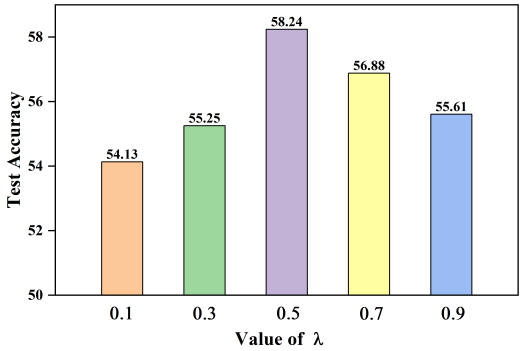

Fig. 7. Test accuracy of FedMGC at different dominant gradient selection ratios.

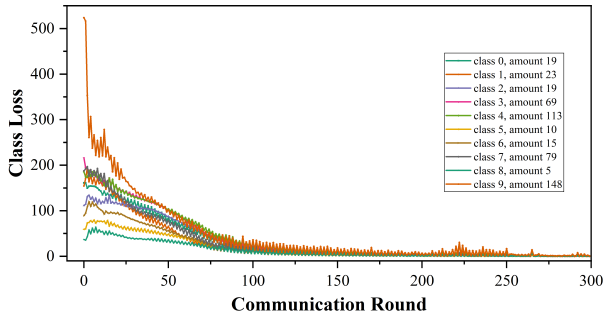

Fig. 8. Class loss variation in a client with class imbalance.

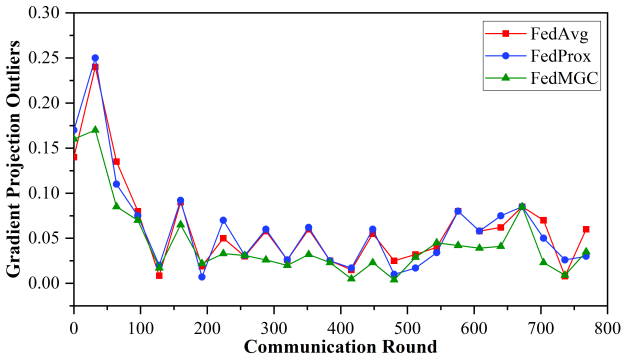

Fig. 9. The variation of gradient projection outliers for FedMGC, FedAvg, and FedProx.

10 clients are involved in each round of communication for federated training. As shown in Fig. 7, FedMGC works best when $\lambda = 0.5$, which may be because the main gradients is selected as the dominant gradients when $\lambda = 0.5$ to avoid the lopsidedness when there are fewer dominant gradients. The problem of negative gradient correction due to too much dominant gradient selection is also prevented.

C. Ablation Experiments

We conduct ablation experiments on the CIFAR-10 dataset with $\alpha = 1$, where the number of clients is 100, the client participation rate is 10%, and the amount of data in each client is the same. Fig. 6 shows the results of the accuracy of FedMGC compared with related ablation methods. FedAvg(ce) and FedProx(ce) denote the FedAvg [15] and FedProx [8] approaches based on the CE loss function, respectively. The test accuracy of FedMGC is superior to FedAvg(ce)+DGC and FedProx(ce)+DGC, which indicates the effectiveness of the FL loss function for dealing with class imbalance. The test accuracies of FedAvg(ce)+DGC and FedProx(ce)+DGC are higher than those of FedAvg(ce) and FedProx(ce) respectively, which indicates the effectiveness of DGC.

1) The Effectiveness of FL Function: As shown in Fig. 8, class 9 and class 4 are majority classes with sample sizes of 148 and 113, respectively. Class 8 and class 5 are minority classes with sample sizes of 5 and 10, respectively. The local model is trained to rapidly improve the recognition of majority classes, followed by the loss adjustment term $(1-p_t)^{\gamma}$ to decrease the contribution of the loss of majority classes. It can be seen that class 9 and class 4 rapidly reduce their own contribution within the first 50 rounds, increasing the proportion of minority classes in the loss of the client, thus alleviating the gradient conflict caused by class imbalance. The test accuracy of FedMGC in Fig. 6 is superior to those of FedAvg(ce)+DGC and FedProx(ce)+DGC, which verifies the effectiveness of the FL function and shows that the minority class samples are important for improving the model performance.

2) The Effectiveness of DGC: Fig. 9 shows the change process of gradient projection outliers of FedMGC, FedAvg and FedProx on the CIFAR-10 dataset when $\alpha = 1$. As shown in Fig. 9, the gradient projection outliers gradually become smaller as the federated training continues, which suggests that the updated gradients tend to be consistent in size and direction. The gradient projection outliers of FedMGC are smaller than those of FedAvg and FedProx, indicating that the use of DGC to correct the updated gradient can effectively alleviate the gradient conflict. The test accuracies of FedAvg(ce)+DGC and FedProx(ce)+DGC in Fig. 6 are 2.9% and 3.0% higher than those of FedAvg(ce) and FedProx(ce), respectively.

D. Performance Analysis

1) Full Client Participation: Compared with other baselines, SCAFFOLD [9] is more sensitive to the client partici-

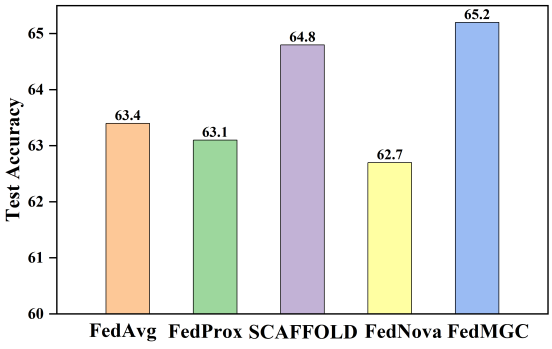

Fig. 10. Test accuracy of FedMGC and baselines with full client participation.

pation rate. Due to the infrequent updating of local control variables, the prediction of updated direction using control variables can be quite erroneous when the client engagement rate isn't high. Thus, it is difficult for SCAFFOLD to converge [3]. To facilitate the comparison of test accuracy, we conducted experiments with a client participation rate of 100% on the CIFAR-10 dataset with clients is 10 and $\alpha = 0.5$. As shown in Fig. 10, FedProx and FedAvg have similar test accuracies because the regularization has less impact due to the small μ. SACFFOLD corrects the client-drift in local updates by controlling the variables, and its test accuracy is higher than those of FedAvg, FedProx, and FedNova. FedMGC achieves the highest test accuracy among all methods, which indicates that the global accuracy can be improved to some extent by mitigating the gradient conflict in global aggregation.

2) Client Partial Participation: Because in a real scenario of federated learning, only a few clients participate in federated training per round, we conducted experiments with client partial participation. The size of clients is 100, and 10 clients participate in federated training in each round. Because SCAFFOLD doesn't converge in this case [3], we don't compare FedMGC with SCAFFOLD when the clients are partially involved in federated training.

Fig. 11 and Fig. 12 show the test accuracy of FedMGC, FedAvg, FedProx, and FedNova when the degree of heterogeneity is $\alpha = 0.5$ and $\alpha = 1$, respectively. Among these, the hyperparameters of the FedMGC have been adjusted optimally. Observing Fig. 11 and Fig.12, it is found that these four methods perform similarly on the FMNIST dataset, and FedMGC performs slightly better than other methods. Taking the CIFAR-10 dataset in Fig. 11 and Fig. 12 as an example, the performance of the four methods can be analyzed as follows:

- As Fig. 12, FedMGC has the greatest global accuracy, which is at most 6.56%, 5.50%, and 5.19% higher than those of FedAvg, FedProx, and FedNova. This indicates that the gradient conflict seriously affects the global accuracy, and FedMGC can effectively mitigate the gradient conflict.
- The volatility of the FedMGC test accuracy curve in the pre-training period is larger than those of FedAvg, FedProx and FedNova when $\alpha = 0.5$ on the CIFAR-10 dataset in Fig. 11. The reason is that the model parameter space is unstable under the strongly heterogeneous data environment, and the dominant gradients correct the unstable gradients easily to cause correction error, so much so that it leads to the volatility of local model parameters. With the convergence of local models, the parameters gradually stabilize and the accuracy curve gradually smooths out.
- FedMGC needs to train more rounds to converge. On the CIFAR-10 dataset with $\alpha = 1$, FedMGC achieves stable test accuracy at 400 rounds, while FedProx achieves a more stable test accuracy at 250 rounds. The reasons for the convergence slowdown of FedMGC are analyzed as, (i) FL requires some computation time to control the loss of majority classes and minority classes in a relative equilibrium state, thus reducing the convergence rate, (ii) The DGC decreases the length of the gradients, which results in less variation in the model parameters and leads to a reduction in the global model convergence rate.

VI. Conclusion

To address the issue of the degradation of the global accuracy due to the gradient conflict among clients, we propose FedMGC, a gradient conflict mitigation approach. FedMGC increases the loss contribution of minority classes based on the FL function, thus reducing the preference of the local models and providing high-quality local gradients for the parameter server. In the global aggregation phase, FedMGC detects and corrects gradients with conflict based on the DGC approach. Through extensive experimental evalution, we show that FedMGC significantly outperforms the baselines in various scenarios and can mitigate the gradient conflict. In the future, we plan to further optimize the FL function to reduce the tuning of the hyperparameters of this loss function and provide a analysis of the convergence.

Acknowledgment

This work is supported by The Key Research and Development Program of China (No. 2022YFC3005401), Key Research and Development Program of China, Yunnan Province (No. 202203AA080009), Outstanding Graduate Student Dissertation Cultivation Program of Hohai University(No. 422003481) and Transformation Program of Scientific and Technological Achievements of Jiangsu Provence (No. BA2021002).

References

[1] Q. Li, Z. Wen, Z. Wu, S. Hu, N. Wang, Y. Li, X. Liu and He. B, "A survey on federated learning systems: vision, hype and reality for data privacy and protection," *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[2] R. Gupta and T. Alam, "Survey on federated-learning approaches in distributed environment," *Wireless Personal Communications*, vol. 125, no. 2, pp. 1631-1652, 2022.

[3] X. Ma, J. Zhu, Z. Lin, S. Chen and Y. Qin, "A state-of-the-art survey on solving non-IID data in federated learning," *Future Generation Computer Systems*, vol. 135, pp. 244-258, 2022.

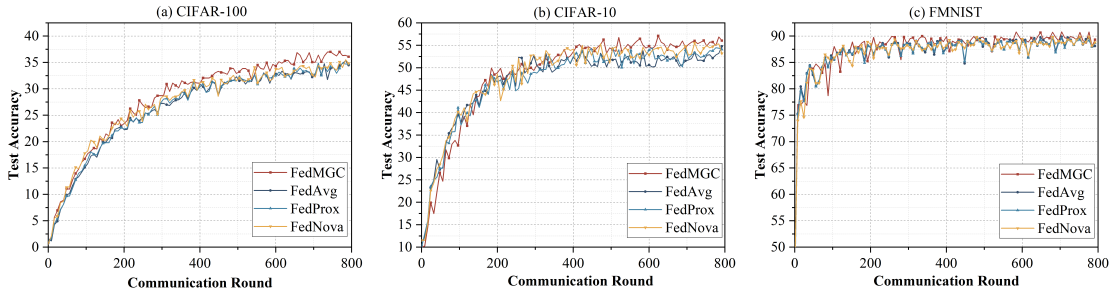

Fig. 11. Test Accuracy of FedMGC with baselines for $\alpha = 0.5$ on CIFAR-100, CIFAR-10 and FMNIST datasets.

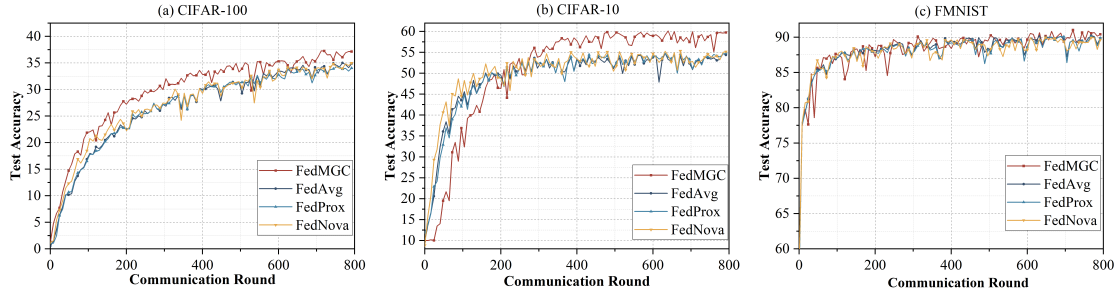

Fig. 12. Test Accuracy of FedMGC with baselines for $\alpha = 1.0$ on CIFAR-100, CIFAR-10 and FMNIST datasets.

[4] H. Zhu, J. Xu, S. Liu and Y. Jin, "Federated learning on non-IID data: A survey," *Neurocomputing*, vol. 465, pp. 371-390, 2021.

[5] Q. Li, Y. Diao, Q. Chen and B. He, "Federated learning on non-IID data silos: an experimental study," *38th IEEE International Conference on Data Engineering*, pp. 965-978, 2022.

[6] E. Jeong, S. Oh, H. Kim, J. Park, M. Bennis and S. L. Kim, "Communication-efficient on-device machine learning: Federated distillation and augmentation under non-iid private data," arXiv:1811.11479, 2018.

[7] Q. Li, B. He and D. Song, "Model-contrastive federated learning," *in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10713–10722, 2021.

[8] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar and V. Smith, "Federated optimization in heterogeneous networks," *Proceedings of Machine Learning and Systems*, vol. 2, pp. 429-450, 2020.

[9] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich and A. T. Suresh, "Scaffold: Stochastic controlled averaging for federated learning," *International Conference on Machine Learning*, pp. 5132-5143, 2020.

[10] J. Wang, Q. Liu, H. Liang, G. Joshi and H. V. Poor, "Tackling the objective inconsistency problem in heterogeneous federated optimization," *Advances in Neural Information Processing Systems*, pp. 7611-7623, 2020.

[11] H. Gao, M. T. Thai and J. Wu, "When decentralized optimization meets federated learning," *IEEE Network*, pp. 1-7, 2023.

[12] J. Wang, V. Tantia, N. Ballas and M. Rabbat, "Slowmo: improving communication-efficient distributed SGD with slow momentum," *8th International Conference on Learning Representations*, 2021.

[13] J. Sashank, Reddi, Z. Charles, M. Zaheer, Z. Garrett, K. Rush, J. Konecny, S. Kumar and H. McMahan, "Adaptive federated optimization," *9th International Conference on Learning Representations*, 2021.

[14] Y. Yeganeh, A. Farshad, N. Navab and S. Albarqouni, "Inverse distance aggregation for federated learning with non-iid data," *Domain Adaptation and Representation Transfer, and Distributed and Collaborative Learning, Springer International Publishing*, pp. 150–159, 2020.

[15] B. McMahan, E. Moore, D. Ramage, S. Hampson and B. Arcas, "Communication-efficient learning of deep networks from decentralized data," *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, pp. 1273-1282, 2017.

[16] T. Yoon, S. Shin, S. J. Hwang and E. Yang, "Fedmix: Approximation of mixup under mean augmented federated learning," *International Conference on Learning Representations*, 2021.

[17] W. Hao, M. El-Khamy, J. Lee, J. Zhang, K. J. Liang, C. Chen and L. C. Duke, "Towards fair federated learning with zero-shot data augmentation," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3310–3319, 2021.

[18] Z. Zhu, J. Hong and J. Zhou, "Data-free knowledge distillation for heterogeneous federated learning," *International Conference on Machine Learning*, pp. 12878-12889, 2021.

[19] Y. Liu, A. Huang, Y. Luo, H. Huang, Y. Liu, Y. Chen, L. Feng, T. Chen, H. Yu and Q. Yang, "Fedvision: An online visual object detection platform powered by federated learning," *in Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 13172–13179, 2020.

[20] Q. Wu, X. Chen, Z. Zhou, and J. Zhang, "Fedhome: Cloud-edge based personalized federated learning for in-home health monitoring," *IEEE Transactions on Mobile Computing*, vol. 21, no. 8, pp. 2818-2832, 2020.

[21] Y. Chen, Y. Ning, M. Slawski and H. Rangwala, "Asynchronous online federated learning for edge devices with non-iid data," *in 2020 IEEE International Conference on Big Data*, pp. 15-24, 2020.

[22] E. Sannara, F. Portet, P. Lalanda and V. German, "A federated learning aggregation algorithm for pervasive computing: Evaluation and comparison," *in 2021 IEEE International Conference on Pervasive Computing and Communications*, pp. 1–10, 2021.

[23] X. Li, N. Liu, C. Chen, Z. Zheng, H. Li and Q. Yan, "Communication-efficient collaborative learning of geo-distributed jointCloud from heterogeneous datasets," *in 2020 IEEE International Conference on Joint Cloud Computing*, pp. 22–29, 2020.

[24] J. Chen, R. Zhang, J. Guo, Y. Fan, and X. Cheng, "Fedmatch: Federated learning over heterogeneous question answering data," *in Proceedings of the 30th ACM International Conference on Information Knowledge Management*, pp. 181-190, 2021.

[25] M. H. Shullar, A. A. Abdellatif and Y. Massoud, "Energy-efficient active federated learning on non-iid data," *2022 IEEE 65th International Midwest Symposium on Circuits and Systems*, pp. 1-4, 2022.

[26] Z. Chen, Z. Wu, X. Wu, L. Zhang, J. Zhao, Y. Yan and Y. Zheng, "Contractible regularization for federated learning on non-iid data," *2022 IEEE International Conference on Data Mining*, pp. 61-70, 2022.

[27] X. Shang, Y. Lu, Y. Cheung and H. Wang, "FEDIC: Federated learning on non-iid and long-tailed data via calibrated distillation," *2022 IEEE International Conference on Multimedia and Expo*, pp. 1-6, 2022.