Supplementary Material for CVPR'2024 Paper: Byzantine-robust Decentralized Federated Learning via Dual-domain Clustering and Trust Bootstrapping

Peng Sun[†,♯], Xinyang Liu[§,‡,♯], Zhibo Wang[≀], Bo Liu[‡,*]
[†]College of Computer Science and Electronic Engineering, Hunan University, China
[§]Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University, China
[≀]School of Cyber Science and Technology, Zhejiang University, China
[‡]Shenzhen Institute of Artificial Intelligence and Robotics for Society (AIRS), China
psun@hnu.edu.cn, codex.lxy@gmail.com, zhibowang@zju.edu.cn, liubo@cuhk.edu.cn

This supplementary document provides detailed experimental settings and results, which have not been included in the main paper due to the page limit. Specifically, we first present details of the experimental setup. Then, we provide the complete results of model accuracies and ASRs under various Untargeted and Targeted attacks (Table 1 and Table 2 in the main paper). Subsequently, we present the specific communication topologies with different Byzantine percentages. Finally, we offer the defense results under another communication topology, which presents an extreme adversary setting (i.e., 80% of clients are malicious, and each benign client is connected to only one benign client with all other neighbors being malicious).

1. Experimental Setup

Taking Figure 1 (a) in the main paper as an example of the decentralized communication topology, we evaluate DFL-Dual on different datasets and various models with two performance metrics of Accuracy (ACC) and Attack Success Rate (ASR). Specifically, we evaluate DFL-Dual on MNIST [9] and Fashion-MNIST [13] using Logistic Regression (LR), Fully Connected (FC), and Convolutional Neural Network (CNN), and on CIFAR-10 [8] using ResNet-18. We adopt the same method in [4, 16] to simulate different non-IID data distribution degrees. Specifically, the non-IID degree is captured by a sample allocation probability p, with larger p indicating a higher non-IID degree. We consider both untargeted and targeted (backdoor) attacks. The untargeted attacks include Label Flipping Attack, Krum Attack [6], and Back-Gradient Attack [11], while the targeted attacks include Scaling Attack [1], DBA Attack [14], and A little is Enough Attack [2]. All experiments are conducted using PyTorch 2.0 on a machine with 2 RTX 4090 GPUs. The detailed experimental settings and parameters are as follows.

1.1. Datasets

We evaluate our proposed DFL-Dual on the following three benchmark datasets.

- **MNIST** [9]: A 10-class handwritten digit image classification dataset containing $60,000$ training images and $10,000$ testing images.
- **Fashion-MNIST** [13]: A 10-class fashion image classification dataset consisting of a training set of $60,000$ examples and a test set of $10,000$ examples.
- **CIFAR-10** [8]: A 10-class tiny image classification dataset containing $6,000$ images per class with $5,000$ training and $1,000$ testing images per class.

We consider a 10-client DFL system as in Figure 1 (a) in the main paper and adopt the same method in [4, 16] to simulate different degrees of non-IID data distribution among clients. Specifically, the non-IID degree is captured by a sample allocation probability p, with larger p indicating a higher non-IID degree.

[♯]Authors with equal contribution.
[*]Bo Liu is the corresponding author, Email: liubo@cuhk.edu.cn.

1.2. Models

We train various models to show the generality of DFL-Dual. Specifically, for MNIST and Fashion-MNIST, we train a CNN (Conv2d($1*6*3$) $\rightarrow$ ReLU $\rightarrow$ MaxPool2d($2*2$) $\rightarrow$ Conv2d($6*25*3$) $\rightarrow$ ReLU $\rightarrow$ MaxPool2d($2*2$) $\rightarrow$ Linear($1225*50$) $\rightarrow$ ReLU $\rightarrow$ Linear($50*10$)), a fully connected neural network (FCN) (Linear($784*500$) $\rightarrow$ ReLU $\rightarrow$ Linear($500*10$) $\rightarrow$ ReLU), and an LR classifier. For CIFAR-10, we use ResNet-18.

1.3. DFL Model Training Parameters

We train models for 15 global rounds on MNIST and Fashion-MNIST and 25 global rounds for CIFAR-10. In each global training round, each client performs E (10 for MNIST and Fashion-MNIST, 1 for CIFAR-10) epochs of local training via mini-batch SGD with a batch size of $B = 100$. Other hyper-parameters during local model training are inherited from the default settings of Adam [7].

1.4. Evaluated Poisoning Attacks

We use both untargeted and targeted (backdoor) attacks to verify the effectiveness of DFL-Dual. The untargeted poisoning attacks include:

- **Label Flipping Attack**: For each training sample on Byzantine clients, we flip its label c to $c+1$ mod P, with P being the total number of labels and $c \in \{1, 2, \ldots, P\}$.
- **Krum Attack** [6]: Byzantine clients craft poisoned *pre-aggregation local models* with reference to benign ones, which allows it to circumvent the defense of `Krum`.
- **Back-Gradient Attack** [11]: Byzantine clients craft a poisoned dataset for local training, resulting in the local model exhibiting the largest loss on the benign dataset.

The targeted poisoning attacks are as follows, and we follow the same adversary setting in [15].

- **Scaling Attack [1]**: After accomplishing local model training on duplicated and triggered training examples, Byzantine clients scale the *pre-aggregation local models* by a factor before sending them to benign clients.
- **DBA Attack [14]**: Byzantine clients decompose the trigger into different patterns and then embed them into their local training data in a distributed manner.
- **A little is Enough Attack [2]**: The model update based on the Scaling Attack is cropped into a certain range to make it harder to eliminate.

1.5. Baseline Aggregation Rules

We take the following aggregation methods as baselines. Notably, for those designed for CFL, *we trim them to fit in the DFL scenario*.

- **DFL [10]**: Consensus algorithm based on graph connectivity is applied to aggregate neighboring *pre-aggregation local models*.
- **DFLTrust [4]**: FLTrust uses an additional validation dataset to infer the model divergence. In our experiments, we trim the original centralized FLTrust algorithm into the DFL setting, where each client uses its own local dataset as the validation dataset with the same other settings in FLTrust.
- **DFLDetector [15]**: Same as the original FLDetector algorithm in CFL setting, each client predicts its neighbours' model update based on historical model updates, which is then used to detect Byzantine updates.
- **Multi-Krum [3]**: Each benign client selects K_i local update models with the smallest Euclidean distances to itself, then averages and incorporates them to update its model. Multi-krum assumes it knows the number of malicious clients Z_i of each benign client i, and thus $K_i = N_i - Z_i$, where N_i is the number of neighbors of client i.
- **BridgeM [5]**: Each client utilizes the coordinate-wise median as the screening rule within the DFL framework.
- **IOS [12]**: Each benign client computes the average of all neighbors' models and then discards the furthest model from this mean. This process repeats until Z_i models are discarded, and output the average of remaining models.

2. Complete Results of Defense against Untargeted and Targeted Attacks

We present the experimental results of DFL-Dual against Untargeted and Targeted Attacks with different models trained on various datasets in Table 1 and Table 2, respectively. They supplement Table 1 and Table 2 in the main paper, respectively. The results validate that DFL-Dual consistently exhibits higher accuracy on benign testing data and lower ASR on testing data with backdoor triggers than all baselines.

defense \ Source		MNIST			Fashion			CIFAR10
		CNN	FC	LR	CNN	FC	LR	ResNet18
DFL (No Attack)		95.39	93.76	89.84	84.85	82.67	81.04	49.96
Label Flipping	DFLTrust	18.28	1.1	1.11	12.8	53.21	1.23	10
	DFLDetector	33.84	62.28	89.9	84.06	36.36	**81.07**	29.58
	Multi-Krum	36.45	61.01	**89.83**	**84.37**	60.4	81	25.09
	DFL	26.05	16.00	15.40	31.78	20.85	13.01	25.3
	BridgeM	50.31	54	44.76	61.12	66.17	49.61	34.89
	IOS	0.24	0.53	0.95	0.51	0.57	0.58	20.59
	DFL-Dual	**96.64**	**92.41**	88.97	83.98	**82.03**	79.64	**49.06**
Krum	DFLTrust	20.01	43.71	1.11	14.76	64.33	0.03	10
	DFLDetector	22.01	36.61	17.66	32.91	31.5	8.7	22.08
	Multi-Krum	30.35	36.51	24.01	27.42	32.08	10.29	10
	DFL	71.14	84.47	71.88	49.76	58.88	49.98	10
	BridgeM	26.12	41.9	42.44	27.66	37.91	38.01	10
	IOS	77.08	80.51	77.59	50.44	68.11	61.24	10
	DFL-Dual	**96.14**	**92.53**	**89.05**	**83.69**	**81.91**	**79.72**	**49.84**
Back-Gradient	DFLTrust	9.8	9.8	9.8	10	10	10	10
	DFLDetector	9.8	10.51	14.74	10	11.75	11.32	10
	Multi-Krum	9.8	11.18	15.61	10	10.17	12.33	11.72
	DFL	25.81	44.52	56.05	19.41	27.39	26.67	10.03
	BridgeM	22.17	46.96	47.72	28.23	35.65	37.33	15.70
	IOS	10.52	68.25	42.17	12.33	34.59	35.66	19.29
	DFL-Dual	**95.14**	**92.41**	**88.99**	**83.73**	**81.99**	**79.72**	**49.1**

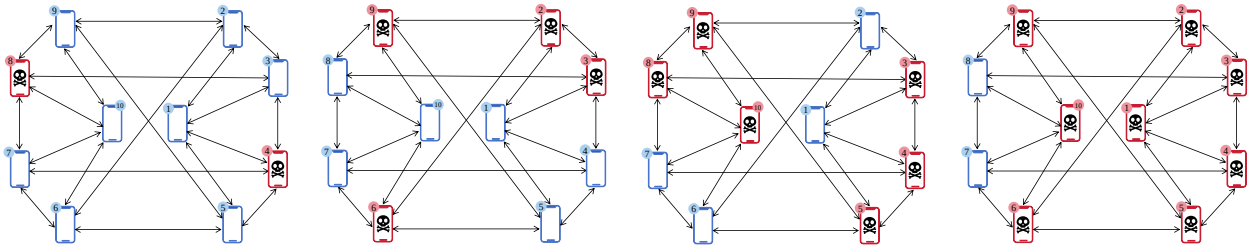
Table 1. Complete results of Accuracies (%) under Untargeted Attacks.

defense \ Source		MNIST			Fashion			CIFAR10
		CNN	FC	LR	CNN	FC	LR	ResNet18
Scaling	DFLTrust	9.8/100	84.77/99.95	81.11/99.91	10/100	76.75/99.33	73.77/99.52	19.45/100
	DFLDetector	67.06/99.75	89.41/3.02	86.50/3.33	69/91.94	81.71/1.9	80.32/2.2	20.85/85.66
	Multi-Krum	96.92/99.99	**93.60/0.67**	**89.89/0.89**	56.89/98.35	83.92/95.93	81.11/1.55	31.62/53.49
	DFL	49.61/100	91.11/100	88.88/99.98	61.78/98.91	82.47/98.76	81.3/99.19	18.7/79.14
	BridgeM	72.02/99.96	91.48/100	89/100	57.3/98.34	81.56/95.98	80.71/98.68	26.23/65.86
	IOS	11.01/97.43	90.59/99.99	86.95/99.97	81.74/91.85	80.24/85.41	**81.06/1.5**	30.78/53.64
	DFL-Dual	**96.21/0.50**	92.44/0.79	88.97/1.00	**84.83/1.70**	82.47/1.36	79.68/1.61	**49.01/4.44**
DBA	DFLTrust	9.8/100	76.9/89.90	73.67/90.93	10/100	75.78/7.26	70.66/77.46	18.64/82.27
	DFLDetector	34.06/70.46	89.41/2.87	86.99/4.30	84.97/4.21	81.67/3.06	80.35/2.37	17.53/82.59
	Multi-Krum	96.89/0.43	**93.70/0.67**	**89.87/0.75**	84.93/3.34	**81.99/1.34**	**81.12/1.41**	26.28/58.76
	DFL	9.8/100	89.93/99.01	88.49/99.64	10/100	81.53/95.85	80.84/98.09	17.87/87.49
	BridgeM	22.17/100	91.54/5.07	88.93/1.26	28.23/91.6	81.48/17.19	80.36/2.64	15.7/80.08
	IOS	**97.04/0.29**	93.61/40.78	89.88/0.86	**82.13/1.91**	79.03/72.62	81.16/1.46	25.87/62.34
	DFL-Dual	96.54/0.48	92.45/0.82	89.06/0.91	83.38/2.53	81.97/1.77	79.58/1.48	**48.79/4.35**
A Little is Enough	DFLTrust	92.29/99.75	88.7/99.93	85.48/99.77	80.1/98.91	78.6/99.94	77.65/99.94	33.59/100
	DFLDetector	92.34/8.74	90.58/3.35	87.02/4.44	83.08/14.74	81.83/2.62	79.96/2.49	36.94/100
	Multi-Krum	95.25/0.72	93/72.01	89.68/70.5	84.38/7.42	80.85/3.98	81.16/7.02	38.62/97.64
	DFL	95.01/85.35	92.81/92.97	89.80/95.82	81.55/86.25	83.73/27.43	75.13/86.58	45.66/99.67
	BridgeM	95.66/5.60	93.26/81.29	90.48/99.3	83.27/28.74	83.3/36.51	82.57/90.99	48.44/89.38
	IOS	**96.95/0.53**	93.18/71.06	89.93/1.36	84.31/5.9	83.18/5.65	81.14/5.07	45.05/89.82
	DFL-Dual	95.59/0.55	**92.65/0.89**	**89.03/0.96**	**83.88/2.16**	**82.26/1.66**	**81.08/1.91**	**50.01/4.78**

Table 2. Complete results of Accuracies (%) under targeted Attacks.

3. Communication Topologies under Different Byzantine Percentages

We randomly distribute different percentages of Byzantine clients within the communication topology, ensuring connectivity among the remaining benign clients. Figure 1 depicts the specific communication topology of the DFL system under various percentages of Byzantine clients.



(a) (b) (c) (d)

Figure 1. Illustration of different percentages of Byzantine clients in the DFL system (with blue and red devices denoting benign and Byzantine clients, respectively).

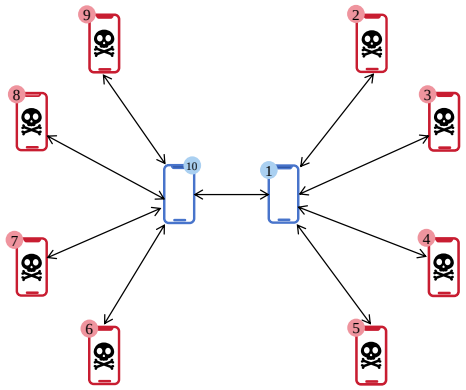


Figure 2. Illustration of the communication topology for the extreme adversary setting (with blue and red devices denoting benign and Byzantine clients, respectively).

4. Defense Performance under an Extreme Adversary Setting

For a more comprehensive evaluation of the effectiveness of the proposed DFL-Dual method, an extreme adversary scenario with an 80% Byzantine presence is considered. Here, each benign client is solely connected to one remaining benign client, as illustrated in Figure 2. Table 3 demonstrates the robustness of the proposed DFL-Dual method in this extreme setting, exhibiting notably superior performance (higher accuracy and lower ASR) compared to other baselines.

defense \ Source	MNIST CNN	Fashion CNN	CIFAR10 ResNet18	defense \ Source	MNIST CNN	Fashion CNN	CIFAR10 ResNet18
Label Flipping				Scaling			
DFLTrust	11.80	9.22	8.07	DFLTrust	9.8/100	10/100	19.83/88.56
DFLDetector	0.78	58.3	17.68	DFLDetector	89.37/99.84	**83.7/1.7**	21.6/90.21
Multi-Krum	0.72	17.49	**45.13**	Multi-Krum	92.08/99.97	77.46/99.6	20.4/89.92
DFL	2.37	15.21	14.71	DFL	11.35/100	35.64/96.72	20.05/79.45
BridgeM	23.53	31.96	23.72	BridgeM	49.43/98.96	41.27/95.75	11.34/91.99
IOS	24.83	22.9	43.29	IOS	11.1/100	20.17/49.27	22.99/76.7
DFL-Dual	**95.97**	**84.31**	45.08	**DFL-Dual**	**95.92/0.72**	84.5/6.04	**43.61/4.42**
Krum				DBA			
DFLTrust	16.40	17.25	10	DFLTrust	9.8/100	10/100	20.93/80.88
DFLDetector	11.8	10	10	DFLDetector	88.26/99.15	83.12/1.98	19.5/84.36
Multi-Krum	50.8	52.4	10	Multi-Krum	95.94/0.71	84.88/1.44	20.59/88.43
DFL	11.9	9.58	10	DFL	52.35/100	10/50	17.89/82.23
BridgeM	10.28	33.17	10	BridgeM	71.17/38.44	34.72/35.82	12.74/96.14
IOS	68.06	46.36	10	IOS	25.28/23.04	10/0.01	24.78/66.87
DFL-Dual	**95.93**	**84.05**	**44.41**	**DFL-Dual**	**96.03/0.63**	**84.21/5.92**	**45.5/4.78**
Back-Gradient				A Little is Enough			
DFLTrust	9.8	10	10	DFLTrust	94.1/99.9	73.91/99.79	35.64/100
DFLDetector	9.8	10	10	DFLDetector	92.67/99.89	83.39/3.26	35.72/100
Multi-Krum	9.8	10	10	Multi-Krum	96.08/4.2	75.88/96.52	35.94/99.96
DFL	10	38.73	10	DFL	83.3/46.08	47.71/47.9	44.34/99.97
BridgeM	9.8	10	10	BridgeM	84.27/65.75	68.63/53.46	41.78/96.27
IOS	29.77	13.05	10	IOS	96.02/2.06	**84.37/1.81**	41.53/99.99
DFL-Dual	**95.86**	**84.18**	**45.05**	**DFL-Dual**	**96.18/0.62**	86.23/2.67	**46.24/4.32**

Table 3. Accuracies/ASRs (%) under both Untargeted and Targeted Attacks under the Extreme Adversary Setting.

References

[1] Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, and Vitaly Shmatikov. How to backdoor federated learning. In *International conference on artificial intelligence and statistics*, pages 2938–2948. PMLR, 2020. 1, 2

[2] Gilad Baruch, Moran Baruch, and Yoav Goldberg. A little is enough: Circumventing defenses for distributed learning. *Advances in Neural Information Processing Systems*, 32, 2019. 1, 2

[3] Peva Blanchard, El Mahdi El Mhamdi, Rachid Guerraoui, and Julien Stainer. Machine learning with adversaries: Byzantine tolerant gradient descent. *Advances in neural information processing systems*, 30, 2017. 2

[4] Xiaoyu Cao, Minghong Fang, Jia Liu, and Neil Zhenqiang Gong. FLtrust: Byzantine-robust federated learning via trust bootstrapping. In *NDSS*, 2020. 1, 2

[5] Cheng Fang, Zhixiong Yang, and Waheed U Bajwa. Bridge: Byzantine-resilient decentralized gradient descent. *IEEE Transactions on Signal and Information Processing over Networks*, 8:610–626, 2022. 2

[6] Minghong Fang, Xiaoyu Cao, Jinyuan Jia, and Neil Gong. Local model poisoning attacks to {Byzantine-Robust} federated learning. In *29th USENIX security symposium (USENIX Security 20)*, pages 1605–1622, 2020. 1, 2

[7] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 2

[8] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research). *URL http://www. cs. toronto. edu/kriz/cifar. html*, 5(4):1, 2010. 1

[9] Yann LeCun, Corinna Cortes, and Chris Burges. Mnist handwritten digit database, 1998. *URL http://www. research. att. com/yann/ocr/mnist*, 7, 1998. 1

[10] Bo Liu, Zhengtao Ding, and Chen Lv. Distributed training for multi-layer neural networks by consensus. *IEEE Transactions on Neural Networks and Learning Systems*, 31(5):1771–1778, 2020. 2

[11] Luis Muñoz-González, Battista Biggio, Ambra Demontis, Andrea Paudice, Vasin Wongrassamee, Emil C Lupu, and Fabio Roli. Towards poisoning of deep learning algorithms with back-gradient optimization. In *Proceedings of the 10th ACM workshop on artificial intelligence and security*, pages 27–38, 2017. 1, 2

[12] Zhaoxian Wu, Tianyi Chen, and Qing Ling. Byzantine-resilient decentralized stochastic optimization with robust aggregation rules. *IEEE Transactions on Signal Processing*, 2023. 2

[13] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017. 1

[14] Chulin Xie, Keli Huang, Pin-Yu Chen, and Bo Li. Dba: Distributed backdoor attacks against federated learning. In *International conference on learning representations*, 2019. 1, 2

[15] Zaixi Zhang, Xiaoyu Cao, Jinyuan Jia, and Neil Zhenqiang Gong. Fldetector: Defending federated learning against model poisoning attacks via detecting malicious clients. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2545–2555, 2022. 2

[16] Bo Zhao, Peng Sun, Tao Wang, and Keyu Jiang. Fedinv: Byzantine-robust federated learning by inversing local model updates. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 9171–9179, 2022. 1