EXPLOITING GLOBAL MODEL FEEDBACK: AN ADAPTIVE MODEL POISONING ATTACK AGAINST ROBUST FEDERATED LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Federated learning (FL) supports collaborative model training, but its repeated exchange of model states also creates a feedback surface for Model Poisoning Attacks (MPAs). Existing MPAs are largely feedback-agnostic: they predefine malicious updates without assessing how previous updates affected the aggregated model. In this paper, we introduce COMPASS, a feedback-guided MPA that exploits early training dynamics and adaptive aggregation feedback. COMPASS analyzes early broadcast models to extract the dominant training direction and identify Top-k important weights. It then crafts an orthogonal perturbation that induces lateral displacement in the optimization trajectory while concentrating attack energy specifically on important weights. Crucially, COMPASS incorporates a matched filter feedback mechanism by embedding a lightweight probe into the malicious update, enabling the attack to dynamically adapt its attack strategy based on the observed aggregation feedback. Evaluations on six datasets, covering 15 aggregation rules and nine baseline attacks, show that COMPASS achieves a mean accuracy reduction of 57.91 percentage points, compared with 23.53 for PoisonedFL, the strongest baseline by this metric (2.46×). These findings expose a broader threat to distributed learning systems that repeatedly release aggregated model states.

1 INTRODUCTION

Federated learning (FL) enables clients to train a shared model without transferring their raw data to a central server (McMahan et al., 2017). This local training and remote aggregation paradigm supports collaborative learning when data are distributed across organizations or devices, but it also gives the server limited visibility into how each client produces its update (Kairouz & McMahan, 2021). Malicious clients can exploit this opacity to submit crafted updates that disrupt convergence and reduce global model accuracy, a threat commonly known as a Model Poisoning Attack (MPA) (Lianga et al., 2023).

Existing MPAs can be broadly categorized into three groups. Weight-wise attacks independently manipulate model weights to corrupt the aggregated model (Blanchard et al., 2017a; Damaskinos et al., 2018; Baruch et al., 2019). Distance-based attacks exploit the distances or similarities between malicious and benign updates to evade robust aggregation (Shejwalkar & Houmansadr, 2021b; Karimireddy et al., 2020). Direction-based attacks alter the optimization direction of the aggregated model to disrupt convergence (Xie et al., 2020; Fang et al., 2020; Xie et al., 2025). Despite these advances, most MPAs remain feedback-agnostic: they predefine malicious updates without assessing how previous updates affected the aggregated model. An attacker therefore cannot distinguish whether an ineffective attack was too weak to alter training or too conspicuous to survive robust aggregation. This limitation becomes acute because both the training state and the response of an aggregation rule evolve over time.

This observation raises one question: can an attacker use feedback from broadcast global models to adapt its attack strategy? We find that early FL training exposes two attack surfaces, the training direction and important weights. The training direction summarizes the dominant movement trajectory of global models, while important weights identify parameters that consistently shape this

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

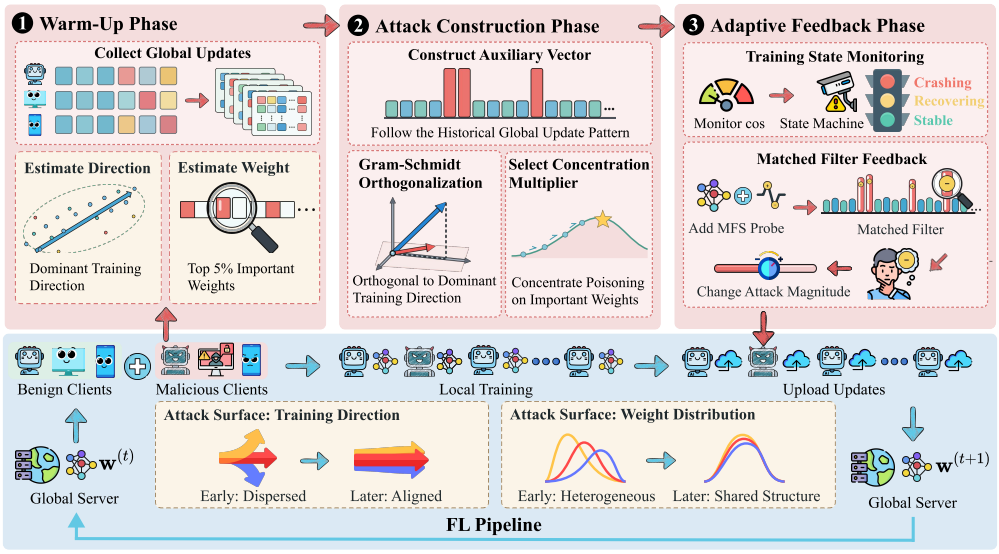


Figure 1: System architecture of COMPASS, which constructs malicious updates by exploiting early-stage consistency.

movement. Prior studies show that optimization can become concentrated in a low dimensional subspace and that a small subset of weights can strongly influence model performance (Gur-Ari et al., 2018; Frankle & Carbin, 2018; Molchanov et al., 2019). Our measurements further show that both structures emerge during early training and remain sufficiently consistent to guide attacks in later rounds.

Based on these observations, we propose COMPASS, short for **C**ontrolled **O**rthogonal **M**odel **P**oisoning with **A**daptive **S**tate **S**ensing, a feedback-guided MPA with three phases (Fig. 1). First, COMPASS estimates the dominant training direction and Top-k important weights from early broadcast models. Second, Gram–Schmidt orthogonalization constructs an attack direction orthogonal to the estimated training direction while concentrating attack energy on important weights. Unlike gradient ascent, which opposes benign optimization and can be offset by subsequent benign updates, this perturbation induces lateral displacement from the training trajectory and produces a stronger poisoning effect. Third, COMPASS embeds a lightweight probe into each malicious update and applies a matched filter to the next global model change. This feedback measures the visibility of the malicious signal in aggregation, allowing COMPASS to adapt its attack magnitude as training and aggregation responses change.

Across six datasets and 15 aggregation rules, COMPASS achieves a mean accuracy impact of 57.91 pp, compared with 23.53 pp for PoisonedFL, the strongest of the nine baseline attacks by mean impact (2.46×). Our contributions are:

- We reveal that early FL training exposes two attack surfaces: the training direction and important weights. COMPASS estimates both from broadcast global models without accessing benign client updates and aggregation knowledge.
- We demonstrate that an orthogonal attack direction is more effective than the conventional reverse attack direction commonly adopted by existing attacks. The orthogonal perturbation induces lateral displacement from the optimization trajectory, whereas subsequent benign updates can offset perturbations that directly oppose the training direction.
- We introduce a matched filter mechanism that extracts feedback from released global models and adaptively adjusts the attack magnitude as the training state and aggregation behavior evolve.

2 Related Work

Weight-wise Attacks. Early MPAs typically rely on direct weight manipulation and overlook the structural regularities of benign updates. Gaussian attacks replace model weights with val-

ues sampled from a Gaussian distribution, whereas SignFlipping reverses the signs of all model weights (Blanchard et al., 2017a; Damaskinos et al., 2018). ALIE estimates the mean and variance of each model weight and places malicious values near the boundary of the resulting statistical range (Baruch et al., 2019). These attacks manipulate all weights through a fixed rule. In contrast, COMPASS identifies important weights from broadcast glboal models and concentrates the attack energy on these weights. This selective manipulation better exploits the unequal contributions of individual weights to model optimization.

Distance-based Attacks. Distance-based attacks exploit the geometry used by robust aggregation. MinMax bounds the largest distance between a malicious update and benign updates, while MinSum bounds the corresponding sum of squared distances (Shejwalkar & Houmansadr, 2021b). Mimic targets non-IID settings by imitating minority or atypical benign clients, thereby making malicious updates difficult to distinguish from naturally heterogeneous client updates (Karimireddy et al., 2020). These methods optimize a one-round geometric constraint or exploit client heterogeneity, but do not use subsequent global models to assess the retained attack contribution. Their attack strength therefore remains unchanged even when the aggregation response varies across rounds.

Direction-based Attacks. IPM makes the aggregated update disagree with descent by driving its inner product with the benign direction below zero (Xie et al., 2020). FangAttack maximizes deviation toward the inverse benign direction under constraints tailored to the target aggregator (Fang et al., 2020). PoisonedFL avoids cross-round self-cancellation by repeatedly perturbing along a fixed random direction (Xie et al., 2025). Reverse-direction perturbations can be counteracted by subsequent benign updates along the training direction. Meanwhile, changing the attack direction across rounds causes malicious effects to cancel one another. COMPASS therefore maintains an orthogonal attack direction over each long feedback interval, allowing poisoning effects to accumulate across rounds and progressively shift the training trajectory away from its current loss basin. As shown in Fig. 2, the accumulated orthogonal perturbation gradually shifts the optimization path away from the benign basin.

Robust Aggregation. We evaluate COMPASS against 15 aggregation rules covering the principal paradigms of robust aggregation. Mean, equivalent to FedAvg aggregation, serves as the standard non-robust baseline (McMahan et al., 2017). Statistic-based defenses, including Median and TrimmedMean, independently suppress extreme values in each weight (Yin et al., 2018). Selection-based defenses, including Krum, MultiKrum, and Bulyan, retain updates supported by their distances to other client updates (Blanchard et al., 2017b; Guerraoui et al., 2018). Geometry-based defenses, including RFA and CenteredClipping, estimate a robust center or bound update deviations around it (Pillutla et al., 2022; Karimireddy et al., 2021). Grouping-based defenses, including SimpleClustering and Bucketing, reorganize similar or randomly grouped updates before aggregation (Sattler et al., 2019; Karimireddy et al., 2020). Weight-wise defenses, such as DnC and SignGuard, detect malicious updates by comparing their weights on a dimension-by-dimension basis (Shejwalkar & Houmansadr, 2021a; Xu et al., 2022). Trust-based defenses include FLTrust and FLDetector, which uses a trusted or consistent reference update (Cao et al., 2020; Zhang et al., 2022). LASA performs layer-wise defense based on anomalous characteristics of important weights (Xu et al., 2025).

3 Threat Model

FL Setup. We consider a standard FL system with one server and N clients. At communication round t, the server broadcasts the global model $\mathbf{w}^{(t)}$ to the participating clients. Each client trains the received model on its local data and uploads a client update. Let $\mathcal{B}$ and $\mathcal{M}$ denote the benign and malicious client sets, $\Delta\mathbf{w}_i^{(t)}$ and $\widetilde{\Delta\mathbf{w}}_i^{(t)}$ denote benign and malicious updates, and $\mathcal{A}$ denote the server aggregation rule. The server updates the global model as

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} + \mathcal{A}\left(\{\Delta\mathbf{w}_i^{(t)}\}_{i\in\mathcal{B}} \cup \{\widetilde{\Delta\mathbf{w}}_i^{(t)}\}_{i\in\mathcal{M}}\right). \tag{1}$$

Adversary Knowledge. We characterize MPAs along three dimensions, as summarized in Table 1. A non-omniscient attack does not require information about benign updates. An AGR-

agnostic attack does not require knowledge of the specific aggregation rule deployed by the server. A feedback-guided attack dynamically adjusts its strategy only based on global models. COMPASS is non-omniscient and AGR-agnostic while using broadcast global models as feedback to adapt its attack magnitude across rounds.

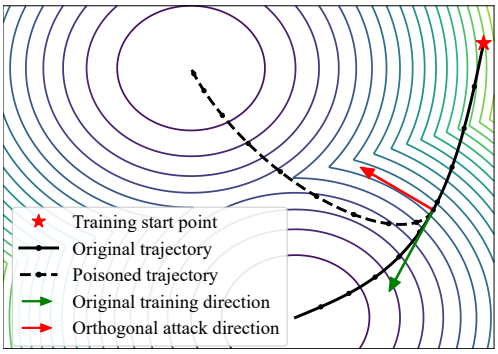


Figure 2: Orthogonal attack direction in the loss landscape.

Attack	Non-omniscient	AGR-agnostic	Feedback-guided
Gaussian	✓	✓	×
SignFlipping	✓	✓	×
ALIE	✓	✓	×
Mimic	×	✓	×
IPM	×	×	×
FangAttack	Both	Both	×
MinMax	Both	✓	×
MinSum	Both	✓	×
PoisonedFL	✓	✓	×
COMPASS	✓	✓	✓

Table 1: Knowledge assumptions of MPAs. "Both" indicates that the paper evaluates both settings.

Attack Objective. COMPASS performs untargeted MPA. Its objective is to disrupt convergence and cause a sustained reduction in global model accuracy, without targeting specific misclassifications or implanting a backdoor.

4 METHODOLOGY

COMPASS follows three phases: it senses the training direction and important weights from early broadcast models, constructs an orthogonal attack direction that remains fixed between refresh events on these weights, and uses matched filter feedback to adapt the attack magnitude. At round t, each malicious client submits

$$\widetilde{\Delta \mathbf{w}}^{(t)} = \lambda^{(t)} \mathbf{v}^* + \mathbf{z}^{(t)}, \tag{2}$$

where $\mathbf{v}^*$ is the orthogonal attack direction, $\lambda^{(t)}$ is the adaptive attack magnitude, and $\mathbf{z}^{(t)}$ is a low-magnitude vector used as the feedback probe.

4.1 WARM-UP PHASE

Prior work shows that optimization gradients can become concentrated in a small, relatively stable subspace early in training, suggesting that early global trajectories may already expose useful optimization structure (Gur-Ari et al., 2018). We empirically observe direction consistency and weight consistency between the early training stage and later training stages in the Evaluation section. COMPASS uses these early signals to estimate the dominant training direction and identify the Top-k important weight.

Let $\mathbf{w}^{(t)} \in \mathbb{R}^d$ denote the broadcast global model at round t, where d is the number of model weights. From consecutive broadcasts, the attacker infers the global update $\Delta \mathbf{w}^{(t)} = \mathbf{w}^{(t+1)} - \mathbf{w}^{(t)}$. The first l_w updates form the warm-up window:

$$\mathbf{M} = \left[\Delta \mathbf{w}^{(1)}, \ldots, \Delta \mathbf{w}^{(l_w)} \right] \in \mathbb{R}^{d \times l_w}. \tag{3}$$

Estimate the Dominant Training Direction. COMPASS applies Principal Component Analysis (PCA) without centering to the update matrix $\mathbf{M}$ and uses its leading left singular vector to estimate the dominant training direction:

$$\mathbf{v}_{\text{pca}} = \arg\max_{\|\mathbf{v}\|_2 = 1} \mathbf{v}^\top \mathbf{M} \mathbf{M}^\top \mathbf{v}. \tag{4}$$

The resulting vector $\mathbf{v}_{\text{pca}} \in \mathbb{R}^d$ defines the reference direction used in the subsequent attack construction.

Identify the Top-k Important Weights. COMPASS uses the same update matrix $\mathbf{M}$ to identify the weights that vary most strongly across the warm-up window as important weights. Specifically, it assigns each model weight i the temporal variance score

$$I(i) = \mathrm{Var}\left(M_{i,1}, M_{i,2}, \ldots, M_{i,l}\right), \tag{5}$$

where $M_{i,t}$ is the update of weight i at round t. COMPASS then selects the k weights with the largest variance scores

$$\mathcal{K} = \mathrm{TopK}\left(\{I(i)\}_{i=1}^{d}, k\right), \qquad |\mathcal{K}| = k \ll d. \tag{6}$$

4.2 Attack Construction Phase

After the warm-up phase, COMPASS uses the $\mathbf{v}_{\mathrm{pca}}$ and $\mathcal{K}$ to construct an orthogonal attack direction.

Top-k Auxiliary Vector. COMPASS first constructs an auxiliary vector $\mathbf{u}(n) \in \mathbb{R}^d$ whose magnitude distribution reflects the identified important weights and whose sign pattern follows the estimated training direction:

$$\mathbf{u}(n) = \begin{cases} n\beta\ \mathrm{sgn}\big((\mathbf{v}_{\mathrm{pca}})_i\big), & i \in \mathcal{K}, \\ \beta\ \mathrm{sgn}\big((\mathbf{v}_{\mathrm{pca}})_i\big), & i \notin \mathcal{K}, \end{cases} \tag{7}$$

where $n \geq 1$ is the concentration multiplier and $\beta > 0$ is a common positive scale. Increasing n assigns more energy to the selected important weights before orthogonalization. Because $\mathbf{v}(n)$ is normalized after orthogonalization, the common scale β cancels and does not control the final attack magnitude, which is determined separately by $\lambda^{(t)}$.

Gram–Schmidt Orthogonalization. Because $\mathbf{u}(n)$ is sign-aligned with $\mathbf{v}_{\mathrm{pca}}$, it generally contains a component along the estimated training direction. COMPASS removes this component using Gram–Schmidt orthogonalization:

$$\mathbf{g}(n) = \mathbf{u}(n) - \frac{\langle \mathbf{u}(n), \mathbf{v}_{\mathrm{pca}} \rangle}{\|\mathbf{v}_{\mathrm{pca}}\|_2^2} \mathbf{v}_{\mathrm{pca}}. \tag{8}$$

COMPASS discards any candidate satisfying $\|\mathbf{g}(n)\|_2 = 0$ and normalizes each remaining candidate as

$$\mathbf{v}(n) = \frac{\mathbf{g}(n)}{\|\mathbf{g}(n)\|_2}. \tag{9}$$

Therefore, $\|\mathbf{v}(n)\|_2 = 1$ and $\mathbf{v}(n)^\top \mathbf{v}_{\mathrm{pca}} = 0$. Orthogonalization may redistribute some energy away from $\mathcal{K}$, and the amount retained on the important weights depends on n.

Selecting the Concentration Multiplier. Different values of n yield different orthogonal directions. Large n overconcentrates the perturbation, increasing detection risk; small n disperses it too broadly, weakening the attack. COMPASS balances Top-k energy concentration, orthogonal retention, and sign similarity:

$$\begin{aligned} C(n) &= \|P_{\mathcal{K}}\, \mathbf{v}(n)\|^2, \qquad E(n) = \frac{\|\mathbf{g}(n)\|}{\|\mathbf{u}(n)\|}, \\ S_{\text{sign}}(n) &= \frac{1}{d} \sum_{i=1}^{d} \mathbb{1}(\mathrm{sgn}(v_i(n)) = \mathrm{sgn}((\mathbf{v}_{pca})_i)). \end{aligned} \tag{10}$$

where $P_{\mathcal{K}}$ projects a vector onto the Top-k weights. The three terms respectively promote support attack energy concentration, norm preservation after orthogonalization, and align its sign pattern with the observed trajectory. When $\mathbf{v}_{pca}$ is normalized, the retention term is equivalently

$$E(n) = \sqrt{1 - \frac{\langle \mathbf{u}(n),\, \mathbf{v}_{pca} \rangle^2}{\|\mathbf{u}(n)\|^2}}. \tag{11}$$

COMPASS then searches over a finite candidate set, such as $\mathcal{N} = \{1, 2, 4, 8, 16, 32, 64, 100\}$:

$$J(n) = C(n) \cdot E(n) \cdot S_{\text{sign}}(n), \qquad n^* = \arg\max_{n \in \mathcal{N}} J(n). \tag{12}$$

The selected $\mathbf{v}(n^*)$ is orthogonal to the estimated training direction and is designed to reduce obvious sign-level deviations while keeping most perturbation energy on the early identified Top-k important weights.

4.3 Adaptive Feedback Phase

The attack direction $\mathbf{v}(n^*)$ remains fixed between refresh events, while the magnitude $\lambda^{(t+1)}$ adapts to changing training dynamics. Without access to internal aggregation decisions, COMPASS combines trajectory monitoring, attacker-side validation, and matched filter feedback from broadcast models.

Training State Monitoring. The attacker measures the cosine similarity between consecutive global updates to monitor trajectory changes:

$$c^{(t)} = \frac{\langle \Delta\mathbf{w}^{(t)}, \Delta\mathbf{w}^{(t-1)} \rangle}{\|\Delta\mathbf{w}^{(t)}\|_2 \|\Delta\mathbf{w}^{(t-1)}\|_2}. \tag{13}$$

A drop below τ_c activates a lightweight state machine. The attacker then tracks the Local Validation Accuracy (LVA) of the global model on its own held-out data over a sliding window

$$\mathcal{W}^{(t)} = \{\text{Acc}^{(t-l+1)}, \ldots, \text{Acc}^{(t)}\}, \tag{14}$$

where $\text{Acc}^{(t)}$ denotes LVA at round t. A decreasing or persistently low trend suggests that the current attack remains effective; a rebound suggests that COMPASS should refresh the fixed attack direction using the latest released global models; a steadily increasing trend suggests that the current magnitude may be insufficient.

Defense	Gaussian	SignFlipping	ALIE	FangAttack	IPM	MinMax	MinSum	Mimic	PoisonedFL	COMPASS
Mean	**75.04**	9.53	2.38	4.53	0.97	6.50	7.93	2.05	42.75	74.49
Krum	25.72	11.64	38.38	29.21	28.75	9.34	16.39	21.93	27.16	**39.69**
MultiKrum	6.52	7.06	33.87	20.18	13.59	9.91	14.81	18.40	26.79	**48.45**
Bulyan	1.96	3.66	2.05	5.98	2.72	6.41	8.08	3.46	28.79	**55.62**
Median	10.79	13.17	32.22	10.23	13.18	5.14	3.15	7.30	35.78	**36.86**
TrimmedMean	43.60	7.95	2.50	3.96	1.18	4.70	5.49	2.28	38.33	**74.39**
RFA	2.46	5.59	30.05	16.02	3.57	3.56	4.85	3.11	17.31	**74.83**
CenteredClipping	28.25	9.57	1.80	3.89	0.93	7.58	9.36	1.87	38.92	**74.31**
DnC	1.21	5.11	2.99	4.43	1.87	5.40	6.03	2.66	2.41	**44.28**
Bucketing	2.20	3.61	29.22	6.28	5.44	9.41	8.59	4.28	6.15	**60.20**
SignGuard	1.45	9.09	1.12	4.14	0.60	3.04	3.53	1.52	1.27	**63.93**
SimpleClustering	**74.57**	9.84	30.54	34.76	30.16	23.96	22.57	9.10	41.87	74.29
FLTrust	2.92	7.69	2.49	26.18	5.32	7.18	8.75	2.61	4.26	**56.88**
FLDetector	32.15	27.75	29.24	71.28	72.10	31.93	40.76	6.87	39.78	**72.41**
LASA	1.10	9.94	1.99	4.24	1.29	2.02	1.30	2.88	1.33	**17.94**
Attack Avg	20.66	9.41	16.06	16.35	12.11	9.07	10.77	6.02	23.53	**57.91**

Table 2: Accuracy impact $I_{\text{acc}} = \text{Acc}_{\text{NoAttack}} - \text{Acc}_{\text{Attack}}$ (in percentage points), computed from Tables 4–9 and averaged equally across six datasets for each aggregation rule. The Attack Avg row averages all 90 dataset–rule pairs. Rounding is applied only after averaging. The maximum value in each defense row is highlighted in bold.

Matched Filter Feedback. Drawing on the matched filter principle from statistical signal detection (Kay, 1993), COMPASS embeds a lightweight probe $\mathbf{z}^{(t)}$ into each malicious update and correlates it with the subsequent global model update. The probe is nonzero only on $\mathcal{K}$, uses independently sampled random signs on these model weights, and is scaled to satisfy $\|\mathbf{z}\|_2 \ll \|\lambda^{(t)}\mathbf{v}^*\|_2$. Its direct contribution to aggregation is therefore designed to be negligible while remaining detectable through correlation.

After the server releases the next global model, COMPASS computes the Matched Filter Score (MFS)

$$S^{(t)} = \left\langle \Delta\mathbf{w}^{(t)}, \mathbf{z}^{(t)} \right\rangle. \tag{15}$$

A persistently positive or non-decreasing $S^{(t)}$ indicates that the probe remains visible in the aggregated global update, whereas a decreasing $S^{(t)}$ indicates that the probe is being attenuated. $S^{(t)}$ is therefore an observable proxy for the extent to which the malicious update is incorporated into aggregation. Because the scale of $S^{(t)}$ varies across training stages and aggregation rules, COMPASS relies on its temporal trend instead of a fixed universal threshold.

Attack Magnitude Adaptation. To adjust the attack magnitude $\lambda^{(t+1)}$ without letting it grow unchecked, COMPASS adopts an Additive Increase / Multiplicative Decrease (AIMD) rule based on recent feedback trends (Chiu & Jain, 1989). Let $\rho_S^{(t)}$ denote the trend of MFS over a recent sliding window, and let $\rho_A^{(t)}$ denote the trend of LVA over the same window. In practice, each trend can be estimated by the slope of a least squares line fitted to the corresponding recent sequence. COMPASS updates the magnitude as

$$\lambda^{(t+1)} = \begin{cases} \lambda^{(t)} + \eta, & \rho_S^{(t)} \geq 0 \ \wedge \rho_A^{(t)} > 0, \\ \gamma\lambda^{(t)}, & \rho_S^{(t)} < 0 \ \wedge \rho_A^{(t)} > 0, \\ \lambda^{(t)}, & \text{otherwise}, \end{cases} \tag{16}$$

where η is a small step size, and $\gamma \in (0, 1)$ is a shrinkage factor. When LVA is recovering while MFS remains stable or increasing, COMPASS interprets the current attack as visible but insufficient and explores a stronger perturbation. When LVA is recovering while MFS decreases, COMPASS reduces the magnitude because the malicious signal appears to be attenuated by aggregation. Otherwise, COMPASS keeps the current magnitude, either because the attack effect is already sustained or because the feedback is inconclusive.

5 Evaluation

5.1 Experimental Setup

Datasets and Models. We evaluate COMPASS on six image classification benchmarks: MNIST for grayscale digits and FashionMNIST for more visually ambiguous objects (LeCun et al., 1998; Xiao et al., 2017); CHMNIST for color histopathology images (Yang et al., 2023); and CIFAR10 for natural RGB images, CINIC10 for greater data diversity with a compatible label space, and CIFAR100 for finer classification into 100 classes (Krizhevsky, 2009; Darlow et al., 2018). These tasks cover increasing difficulty, domain shift, data diversity, and class granularity. We use SimpleCNN, ResNet18, and ResNet34 according to dataset complexity and repeat all experiments with five seeds.

Evaluation Metric. Each experiment uses 100 clients with full participation to ensure reproducibility. We additionally evaluate random 10% client participation, with results reported in the Appendix. COMPASS is compared with nine MPAs under 15 aggregation rules. Attack effectiveness is measured by the accuracy impact I_{acc}, defined as the difference between final test accuracy without and with an attack:

$$I_{\text{acc}} = \text{Acc}_{\text{NoAttack}} - \text{Acc}_{\text{Attack}}. \tag{17}$$

We report I_{acc} in percentage points (pp). A larger value indicates that the attack causes a greater reduction in accuracy.

5.2 Overall Attack Effectiveness

Table 2 reports accuracy impact computed from Tables 4–9 and averaged equally across the six datasets for each aggregation rule; the Attack Avg row averages over the 15 rules. COMPASS has the highest overall mean impact, 57.91 pp, compared with 23.53 pp for PoisonedFL, the strongest baseline by this metric. The ratio of these mean impacts is approximately 2.46. COMPASS also has the largest dataset-averaged impact under 13 of the 15 aggregation rules. The experimental settings and hyperparameter selection procedure are described in the Appendix.

The two exceptions are Mean and SimpleClustering, where Gaussian has a larger mean impact than COMPASS by 0.55 pp and 0.27 pp, respectively. These are numerical differences in dataset-averaged impact, rather than evidence of statistical significance.

5.3 Impact of Malicious Client Ratios

We evaluate COMPASS on MNIST with 10%–60% malicious clients, an IID partition, and a learning rate of 0.05. Median, SignGuard, Krum, RFA, FLTrust, and LASA represent the six defense categories. In Fig. 3, COMPASS reduces test accuracy to approximately 10% across all defenses and ratios, reaching near-random guessing even with only 10% malicious clients.

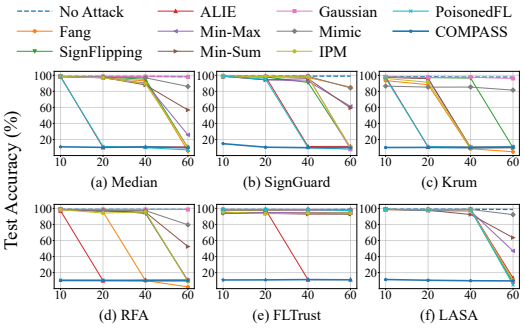


Figure 3: Test accuracy (%) at malicious client ratios of 10%, 20%, 40%, and 60%.

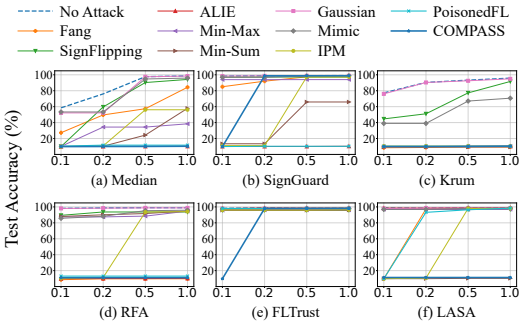


Figure 4: Test accuracy (%) under different data distributions.

5.4 Impact of Data Distribution

Fig. 4 reports the effect of data heterogeneity controlled by the Dirichlet parameter α. A larger α produces client distributions closer to IID and generally increases test accuracy under several attacks (Li et al., 2022). At $\alpha = 0.1$, COMPASS reduces test accuracy to approximately 10% under all six representative defenses. As α increases to 0.2, SignGuard and FLTrust suppress COMPASS. These results show that COMPASS is more likely to succeed under non-IID data distributions, where greater variation among benign client updates provides more opportunities for malicious updates to influence aggregation.

5.5 Mechanism Validation

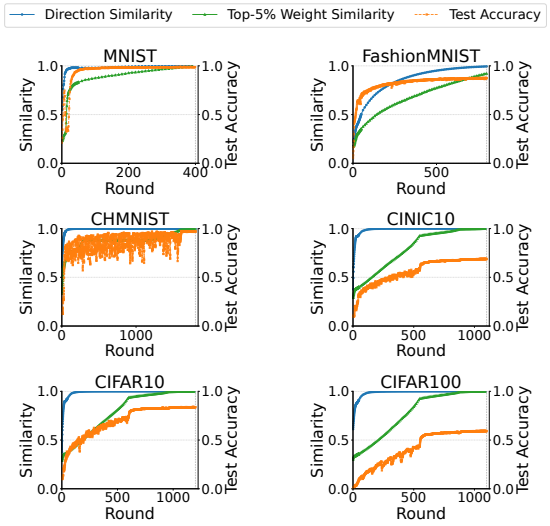


Figure 5: Training direction and Top-5% weight consistency, with test accuracy, across six datasets.

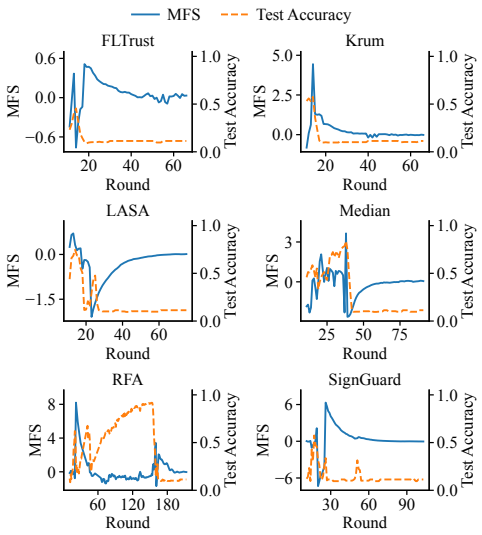


Figure 6: Trends of MFS and test accuracy under six representative robust aggregation defenses.

Early-stage Attack Surfaces. We conduct benign FL training with Mean under the no-attack setting and compare the training signals at each round with those at convergence. The convergence rounds are 400 for MNIST, 800 for FashionMNIST, 1100 for CIFAR10, CINIC10 and CIFAR100,

Defense	Orthogonal	Reverse
Mean	**74.43 ± 12.62**	71.76 ± 12.60
Krum	**28.42 ± 41.47**	12.16 ± 18.71
MultiKrum	**42.40 ± 43.32**	14.30 ± 19.48
Bulyan	**55.06 ± 38.06**	42.38 ± 37.00
Median	**30.35 ± 42.39**	27.80 ± 36.24
TrimmedMean	**74.21 ± 12.84**	72.46 ± 12.49
RFA	**74.53 ± 12.89**	15.97 ± 21.03
CenteredClipping	**74.19 ± 12.80**	71.58 ± 12.92
DnC	**42.69 ± 43.68**	13.81 ± 20.75
Bucketing	**59.04 ± 31.60**	14.26 ± 18.58
SignGuard	**63.72 ± 31.77**	14.72 ± 21.57
SimpleClustering	**74.11 ± 12.18**	69.64 ± 14.76
FLTrust	**55.59 ± 36.77**	14.06 ± 20.22
FLDetector	**71.07 ± 15.84**	68.16 ± 15.11
LASA	**17.21 ± 34.91**	13.99 ± 21.04
Attack Avg	**55.80**	35.80

Table 3: Accuracy impact I_{acc} of COMPASS with different directions, reported as mean ± sample standard deviation across six datasets. The larger mean per defense is bold.

and 1800 for CHMNIST. As shown in Fig. 5, the training direction stabilizes at a very early stage, typically within the first 30 rounds. However, the important weights exhibit stronger differences across tasks. On MNIST and CHMNIST, the Top-5% weight similarity increases rapidly and reaches approximately 0.8 within the first 30 rounds. In contrast, CINIC10, CIFAR10, and CIFAR100 remain below 0.5 during the early training stage and follow highly similar trajectories, with the main changes around rounds 550 and 800 corresponding to the decay of learning rate. The detailed results in the appendix further show that COMPASS is less effective on these more complex tasks than on MNIST-type datasets. This suggests that increasing task complexity makes early identification of important weights more difficult, thereby weakening the attack surface exploited by COMPASS.

Orthogonal vs Reverse Attack Direction. We compare COMPASS's orthogonal direction with a reverse direction under identical FL settings. Table 3 reports the accuracy impact averaged over six datasets and 15 aggregation rules. The orthogonal direction achieves a higher average impact than the reverse direction, 55.80 pp versus 35.80 pp, and performs better under every evaluated rule. Its advantage is particularly pronounced under RFA, DnC, Bucketing, SignGuard, and FLTrust, which largely suppress the reverse direction. These results indicate that benign updates can offset or aggregation can filter reverse perturbations, whereas a fixed orthogonal direction better preserves persistent deviation from the benign trajectory. Ablation studies of the other COMPASS components are provided in the Appendix.

Matched Filter Feedback. We evaluate MFS on MNIST with the same six representative defenses: FLTrust, Krum, LASA, Median, RFA, and SignGuard. In Fig. 6, MFS changes often precede accuracy changes. Under Krum, Median, and SignGuard, an MFS increase precedes a sharp accuracy drop, indicating that the probe becomes visible before the attack effect fully appears. Since MFS ranges differ across rules, COMPASS uses temporal trends rather than a universal threshold to assess aggregation response. These results support MFS as an early signal of malicious update contribution; the Appendix provides more comprehensive validation.

6 Discussion and Conclusion

COMPASS demonstrates that repeatedly released global models form an exploitable feedback channel for MPA, without requiring benign updates or knowledge of the server aggregation rule.

The effectiveness of COMPASS depends on two conditions. First, global models must expose stable training directions and important weight structures during warm-up window. Although these signals emerge early across the evaluated datasets, their stability is not theoretically guaranteed. More complex datasets and neural network architectures may reduce the effectiveness of COMPASS perturbations. Second, malicious updates must survive aggregation for directional deviations to accumulate. COMPASS is less effective against hard-filtering defenses (e.g., Median, Krum, MultiKrum, Bulyan, DnC, and LASA), which completely discard suspicious malicious updates and thereby prevent poison accumulation.

In conclusion, COMPASS combines early training dynamics, controlled orthogonal perturbations, and matched filter feedback. Across the evaluated datasets and aggregation rules, it achieves the largest overall mean accuracy impact among the compared attacks, with the largest dataset-averaged impact under 13 of the 15 rules. These findings highlight the security risks created by repeatedly releasing global models.

REFERENCES

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

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, 2017a.

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, 2017b.

Xiaoyu Cao, Minghong Fang, Jia Liu, and Neil Zhenqiang Gong. Fltrust: Byzantine-robust federated learning via trust bootstrapping. *arXiv preprint arXiv:2012.13995*, 2020.

Dah-Ming Chiu and Raj Jain. Analysis of the increase and decrease algorithms for congestion avoidance in computer networks. *Computer Networks and ISDN systems*, 17(1):1–14, 1989.

Georgios Damaskinos, Rachid Guerraoui, Rhicheek Patra, Mahsa Taziki, et al. Asynchronous byzantine machine learning (the case of sgd). In *International Conference on Machine Learning*, pp. 1145–1154. PMLR, 2018.

Luke Darlow, Elliot J. Crowley, Antreas Antoniou, and Amos Storkey. Cinic-10 is not imagenet or cifar-10. *arXiv preprint arXiv:1810.03505*, 2018.

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)*, pp. 1605–1622, 2020.

Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv preprint arXiv:1803.03635*, 2018.

Rachid Guerraoui, Sébastien Rouault, et al. The hidden vulnerability of distributed learning in byzantium. In *International conference on machine learning*, pp. 3521–3530. PMLR, 2018.

Guy Gur-Ari, Daniel A Roberts, and Ethan Dyer. Gradient descent happens in a tiny subspace. *arXiv preprint arXiv:1812.04754*, 2018.

Peter Kairouz and H Brendan McMahan. Advances and open problems in federated learning. *Foundations and trends in machine learning*, 14(1-2):1–210, 2021.

Sai Praneeth Karimireddy, Lie He, and Martin Jaggi. Byzantine-robust learning on heterogeneous datasets via bucketing. *arXiv preprint arXiv:2006.09365*, 2020.

Sai Praneeth Karimireddy, Lie He, and Martin Jaggi. Learning from history for byzantine robust optimization. In *International conference on machine learning*, pp. 5311–5319. PMLR, 2021.

Steven M Kay. Statistical signal processing: estimation theory. *Prentice Hall*, 1:Chapter–3, 1993.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Qinbin Li, Yiqun Diao, Quan Chen, and Bingsheng He. Federated learning on non-iid data silos: An experimental study. In *2022 IEEE 38th international conference on data engineering (ICDE)*, pp. 965–978. IEEE, 2022.

Junchuan Lianga, Rong Wang, Chaosheng Feng, and Chin-Chen Chang. A survey on federated learning poisoning attacks and defenses. *arXiv preprint arXiv:2306.03397*, 2023.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pp. 1273–1282. Pmlr, 2017.

Pavlo Molchanov, Arun Mallya, Stephen Tyree, Iuri Frosio, and Jan Kautz. Importance estimation for neural network pruning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11264–11272, 2019.

Krishna Pillutla, Sham M Kakade, and Zaid Harchaoui. Robust aggregation for federated learning. *IEEE Transactions on Signal Processing*, 70:1142–1154, 2022.

Felix Sattler, Simon Wiedemann, Klaus-Robert Müller, and Wojciech Samek. Robust and communication-efficient federated learning from non-iid data. *IEEE transactions on neural networks and learning systems*, 31(9):3400–3413, 2019.

Virat Shejwalkar and Amir Houmansadr. Manipulating the byzantine: Optimizing model poisoning attacks and defenses for federated learning. In *Ndss*, 2021a.

Virat Shejwalkar and Amir Houmansadr. Manipulating the byzantine: Optimizing model poisoning attacks and defenses for federated learning. In *Ndss*, 2021b.

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

Cong Xie, Oluwasanmi Koyejo, and Indranil Gupta. Fall of empires: Breaking byzantine-tolerant sgd by inner product manipulation. In *Uncertainty in artificial intelligence*, pp. 261–270. PMLR, 2020.

Yueqi Xie, Minghong Fang, and Neil Zhenqiang Gong. Model poisoning attacks to federated learning via multi-round consistency. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 15454–15463, 2025.

Jiahao Xu, Zikai Zhang, and Rui Hu. Achieving byzantine-resilient federated learning via layer-adaptive sparsified model aggregation. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 1508–1517. IEEE, 2025.

Jian Xu, Shao-Lun Huang, Linqi Song, and Tian Lan. Byzantine-robust federated learning through collaborative malicious gradient filtering. In *2022 IEEE 42nd international conference on distributed computing systems (ICDCS)*, pp. 1223–1235. IEEE, 2022.

Jiancheng Yang, Rui Shi, Donglai Wei, Zequan Liu, Lin Zhao, Bin Ke, Hanspeter Pfister, Bing Ni, and Ziyu Yan. Medmnist v2: A large-scale lightweight benchmark for 2d and 3d biomedical image classification. *Scientific Data*, 10(1):41, 2023.

Dong Yin, Yudong Chen, Ramchandran Kannan, and Peter Bartlett. Byzantine-robust distributed learning: Towards optimal statistical rates. In *International conference on machine learning*, pp. 5650–5659. Pmlr, 2018.

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*, pp. 2545–2555, 2022.

Defense	NoAttack	Gaussian	ALIE	SignFlipping	MinMax	FangAttack	IPM	MinSum	Mimic	PoisonedFL	COMPASS
LASA	99.15	98.89	98.88	98.91	99.00	98.76	98.71	98.74	98.89	98.68	11.19
FLDetector	99.04	97.48	10.18	98.79	98.92	11.04	10.10	98.94	98.51	11.79	10.47
SignGuard	98.99	98.89	98.70	98.68	97.35	98.71	98.92	98.69	98.90	98.64	11.35
Bucketing	98.78	98.74	10.54	98.25	92.76	95.88	95.68	92.95	97.36	80.10	10.10
DnC	98.93	98.81	97.65	98.57	98.76	98.60	98.91	98.89	98.70	97.17	11.35
CenteredClipping	99.00	94.76	97.81	97.21	92.72	98.77	98.87	93.72	98.79	10.10	10.84
FLTrust	98.81	98.77	98.71	95.12	95.72	98.08	95.52	93.83	98.48	98.76	11.28
RFA	98.88	98.71	10.02	98.34	97.48	97.25	97.82	98.28	97.87	11.32	10.03
Bulyan	98.57	97.88	97.76	98.53	96.42	97.81	97.66	96.51	98.34	10.29	11.40
Median	98.75	98.46	10.17	98.32	97.65	97.27	97.71	97.42	98.38	11.94	10.73
TrimmedMean	98.93	98.21	98.15	98.36	95.47	98.66	98.73	95.70	98.45	10.61	10.21
MultiKrum	98.91	97.99	10.46	98.86	92.93	77.61	93.07	98.62	83.72	11.05	10.17
Krum	98.17	97.82	10.64	97.77	96.17	90.11	91.30	95.90	85.29	11.11	11.06
SimpleClustering	98.82	10.41	10.73	97.26	56.89	6.39	8.84	53.19	84.36	14.33	10.58
Mean	99.00	10.61	97.87	97.41	93.03	98.74	98.88	94.18	98.84	11.79	10.47

Table 4: Test accuracy (%) under different attacks and defenses on MNIST. We select MNIST as a canonical handwritten-digit benchmark for evaluating attack effectiveness in a relatively simple and well-controlled classification setting.

Defense	NoAttack	Gaussian	ALIE	SignFlipping	MinMax	FangAttack	IPM	MinSum	Mimic	PoisonedFL	COMPASS
LASA	90.80	89.86	90.47	87.80	86.76	90.31	90.31	87.94	87.84	88.56	88.91
FLDetector	90.70	84.31	10.11	89.63	90.05	10.15	10.00	45.08	75.68	10.06	10.25
SignGuard	90.55	89.99	89.66	88.82	86.84	88.57	90.01	85.90	90.23	86.83	10.00
Bucketing	87.18	87.07	10.12	86.88	77.99	77.92	80.11	76.40	83.64	84.80	11.06
DnC	90.77	90.71	87.91	88.61	89.61	88.28	89.26	89.67	89.52	85.86	10.03
CenteredClipping	90.71	81.26	89.55	88.84	74.54	89.27	90.19	72.55	90.26	10.60	10.00
FLTrust	87.85	87.05	85.38	84.52	78.36	86.97	84.53	82.30	86.58	82.09	10.00
RFA	90.37	90.24	10.00	89.66	86.62	85.80	85.06	86.55	88.88	85.91	10.49
Bulyan	89.98	87.68	85.54	89.37	80.11	85.40	87.56	79.46	84.92	11.50	10.00
Median	90.80	90.37	9.88	88.50	86.28	84.77	86.63	85.18	88.38	10.10	10.06
TrimmedMean	90.61	87.17	88.15	89.60	83.27	88.24	90.19	84.30	89.99	10.29	10.14
MultiKrum	89.51	88.45	10.08	88.29	75.29	67.13	75.53	81.73	68.93	58.05	10.00
Krum	85.60	84.98	10.04	81.58	77.90	76.26	80.45	82.93	75.25	75.65	10.00
SimpleClustering	90.23	9.93	10.00	88.88	15.72	4.56	6.78	35.27	73.50	10.01	12.09
Mean	90.67	10.04	89.07	89.08	72.83	89.10	90.19	71.96	90.26	10.06	10.26

Table 5: Test accuracy (%) under different attacks and defenses on FashionMNIST. We select FashionMNIST to examine whether the attacks remain effective on more visually complex classes than handwritten digits while retaining a controlled grayscale setting.

Defense	NoAttack	Gaussian	ALIE	SignFlipping	MinMax	FangAttack	IPM	MinSum	Mimic	PoisonedFL	COMPASS
LASA	93.90	93.80	92.50	73.30	93.20	91.20	92.20	92.90	93.10	93.70	93.00
FLDetector	94.70	90.80	93.30	84.40	67.20	13.95	12.50	66.20	89.30	91.40	10.21
SignGuard	94.30	93.10	93.60	89.10	91.90	92.30	93.90	92.40	94.00	94.00	12.50
Bucketing	89.50	89.30	87.60	85.70	85.90	85.00	85.70	86.80	89.00	87.00	12.50
DnC	94.60	92.50	93.50	89.70	88.30	92.80	92.80	92.20	93.40	94.50	12.50
CenteredClipping	94.90	86.20	94.30	90.00	88.50	94.80	94.20	77.80	94.60	90.20	12.45
FLTrust	93.30	91.50	92.30	85.80	89.60	87.40	88.80	85.70	90.50	90.10	12.50
RFA	95.10	92.60	93.60	92.80	93.40	89.80	92.60	90.60	93.70	91.40	10.15
Bulyan	93.70	92.40	92.50	91.10	85.60	90.60	90.70	85.50	91.40	92.80	14.50
Median	94.40	90.70	93.00	90.50	89.10	89.50	91.00	93.10	93.60	92.40	88.70
TrimmedMean	95.10	21.90	93.60	92.70	91.70	94.00	95.00	92.40	94.70	95.00	12.50
MultiKrum	92.90	90.70	89.60	90.90	87.30	75.70	80.40	88.40	87.00	90.30	12.50
Krum	91.50	87.60	86.50	87.30	86.10	79.00	81.80	81.30	84.60	85.70	85.70
SimpleClustering	95.40	13.10	92.70	89.90	83.80	94.80	94.30	83.70	84.20	94.20	12.62
Mean	94.90	9.44	94.20	90.50	94.00	93.70	94.50	88.90	94.30	92.70	12.52

Table 6: Test accuracy (%) under different attacks and defenses on CHMNIST. We select CHMNIST to assess attack effectiveness in a medical image classification domain whose data characteristics differ substantially from standard vision benchmarks.

A Detailed Main Results

A.1 Experimental Setup

Training Settings. We conduct all experiments using FedSGD for 1000 global communication rounds with 100 participating clients. All clients participate in every communication round, and their local data follow an independent and identical distribution. Each client trains its model with a batch size of 16 using SGD. The learning rate is set to 0.05, the momentum to 0.9, and the weight decay to 5×10^{-4}. We run all experiments with five random seeds, ranging from 42 to 46, to ensure reproducibility.

Defense	NoAttack	Gaussian	ALIE	SignFlipping	MinMax	FangAttack	IPM	MinSum	Mimic	PoisonedFL	COMPASS
LASA	85.59	82.00	82.24	72.15	84.08	80.80	83.77	85.34	83.07	84.40	84.87
FLDetector	86.35	10.06	84.74	10.13	19.21	10.69	10.12	5.97	81.79	81.30	10.11
SignGuard	86.70	85.32	85.79	78.83	85.49	84.32	86.41	85.91	85.35	85.65	10.00
Bucketing	79.31	77.11	76.24	76.00	60.55	76.89	73.73	71.70	75.23	75.23	10.23
DnC	88.11	86.42	85.69	85.73	86.08	86.51	86.52	84.09	86.05	86.90	86.20
CenteredClipping	88.05	28.68	86.80	80.02	86.24	85.00	86.84	86.38	86.53	75.87	10.04
FLTrust	87.76	86.60	85.03	83.25	85.38	9.45	84.84	83.00	85.66	85.65	10.00
RFA	87.98	85.98	86.40	83.18	85.98	64.71	86.23	85.70	85.81	86.94	10.00
Bulyan	86.91	85.32	85.06	85.12	81.52	77.74	84.90	82.67	83.15	85.53	13.10
Median	86.36	69.24	85.94	65.87	76.35	74.67	63.87	85.78	78.21	74.64	69.70
TrimmedMean	88.07	15.75	86.32	81.04	85.83	85.05	86.59	86.36	85.53	86.79	10.20
MultiKrum	86.11	78.85	79.23	78.08	74.10	76.16	67.59	51.14	67.62	73.87	82.54
Krum	84.56	10.80	69.04	68.77	77.06	11.02	11.25	51.98	65.98	66.02	76.20
SimpleClustering	88.19	10.05	86.10	79.79	86.19	84.79	86.84	86.00	86.78	78.96	9.87
Mean	88.20	10.00	86.00	79.36	86.68	84.73	87.05	85.86	86.86	73.24	10.10

Table 7: Test accuracy (%) under different attacks and defenses on CIFAR-10. We select CIFAR-10 as a standard natural-image benchmark for evaluating attack effectiveness on more complex color images.

Defense	NoAttack	Gaussian	ALIE	SignFlipping	MinMax	FangAttack	IPM	MinSum	Mimic	PoisonedFL	COMPASS
LASA	60.96	60.04	57.11	48.44	56.43	46.65	60.22	57.75	55.61	59.34	46.01
FLDetector	61.73	10.01	59.96	10.58	17.10	12.47	10.09	21.24	54.30	52.09	10.04
SignGuard	65.23	61.86	63.27	48.51	59.14	51.24	64.98	54.10	59.94	65.01	11.86
Bucketing	56.23	49.27	52.87	45.92	45.02	39.48	47.31	40.08	46.50	51.11	10.05
DnC	66.21	65.09	60.42	48.69	48.41	48.99	63.89	43.81	59.35	63.77	55.54
CenteredClipping	67.35	27.34	64.75	50.03	57.80	56.10	66.98	58.13	61.88	54.85	10.09
FLTrust	65.33	61.48	63.02	53.89	53.55	51.84	60.44	45.32	62.28	60.21	54.28
RFA	67.00	62.77	64.79	54.16	61.89	60.07	63.56	57.38	60.00	65.74	10.24
Bulyan	62.89	59.25	60.85	49.61	55.61	47.62	58.95	46.33	57.78	62.44	49.75
Median	64.58	51.26	64.42	48.16	59.11	53.96	52.18	58.10	54.60	57.29	46.97
TrimmedMean	68.08	14.59	63.25	52.44	61.53	57.56	66.11	54.95	61.50	64.89	9.96
MultiKrum	65.64	50.13	51.79	46.26	53.65	27.56	48.73	34.22	44.37	49.53	38.20
Krum	64.68	11.53	44.68	43.87	49.11	24.97	10.08	36.02	12.16	47.43	32.18
SimpleClustering	68.04	9.97	63.04	51.07	61.75	48.28	66.69	53.01	60.30	51.75	9.94
Mean	68.25	10.03	63.86	50.46	62.94	53.84	66.99	57.45	61.68	56.11	10.06

Table 8: Test accuracy (%) under different attacks and defenses on CINIC-10. We select CINIC-10 to test attack effectiveness under greater data diversity and distributional variation than CIFAR-10.

Defense	NoAttack	Gaussian	ALIE	SignFlipping	MinMax	FangAttack	IPM	MinSum	Mimic	PoisonedFL	COMPASS
LASA	54.74	53.96	52.00	44.91	53.55	51.99	52.18	54.70	49.37	52.47	53.50
FLDetector	54.03	1.00	52.83	26.50	2.47	0.59	1.13	4.54	45.77	1.21	1.00
SignGuard	58.55	56.46	56.56	35.87	55.39	54.34	56.50	56.15	56.80	56.59	55.02
Bucketing	47.29	43.58	45.62	43.90	39.62	45.43	43.13	38.82	40.87	43.15	43.15
DnC	59.37	57.22	54.90	56.05	54.43	56.22	55.38	53.17	55.04	55.36	56.67
CenteredClipping	60.26	12.52	56.27	36.77	54.97	53.01	57.59	55.51	57.02	25.15	1.00
FLTrust	59.01	49.14	52.68	43.35	46.35	1.25	45.99	49.44	52.90	49.72	52.75
RFA	61.84	56.10	56.08	49.50	54.47	7.44	54.51	53.56	56.25	55.99	1.29
Bulyan	56.64	54.40	54.66	53.03	50.97	53.67	52.58	49.75	52.33	53.37	56.21
Median	55.86	26.01	34.00	20.39	51.43	29.22	20.28	52.30	33.76	29.73	43.43
TrimmedMean	59.57	1.13	55.90	38.55	54.37	53.08	56.65	53.69	56.49	2.79	1.00
MultiKrum	55.51	43.37	44.19	43.85	45.88	43.35	41.71	45.61	26.56	45.05	44.47
Krum	57.33	34.78	30.64	32.73	39.47	25.20	34.48	35.35	26.99	32.99	28.55
SimpleClustering	61.18	1.00	56.04	35.95	53.73	54.49	57.47	55.25	58.10	1.37	1.00
Mean	60.35	1.01	56.12	37.37	52.92	54.08	57.96	55.43	57.11	1.00	1.00

Table 9: Test accuracy (%) under different attacks and defenses on CIFAR-100. We select CIFAR-100 to evaluate whether the attacks remain effective in a more challenging fine-grained classification task with a substantially larger number of classes.

Datasets and Models. We evaluate COMPASS on 6 complementary image classification benchmarks: MNIST, FashionMNIST, CHMNIST, CIFAR-10, CINIC-10, and CIFAR-100. The dataset and model architecture are changed according to the task under evaluation, while the remaining experimental settings are kept unchanged unless otherwise specified. We use SimpleCNN for MNIST and FashionMNIST, ResNet18 for CHMNIST and CIFAR-10, ResNet34 for CINIC-10 and CIFAR-100.

Adversary Configuration. We assume that 20% of the 100 participating clients are controlled by the adversary. COMPASS observes only updates to the global model and does not require information about benign clients or prior knowledge of the aggregation rule used by the server.

Attack Success Criterion. We define an attack as successful if, during the final stage of convergence, the test accuracy of the global model remains below 0.2 for 50 consecutive communication

Defense	NoAttack	Gaussian	ALIE	SignFlipping	MinMax	FangAttack	IPM	MinSum	Mimic	PoisonedFL	COMPASS
LASA	-0.42	-89.09	+0.06	-0.15	-0.86	+0.04	+0.09	-0.39	-0.37	-88.88	-1.39
FLDetector	-0.30	-87.74	+88.29	-1.18	-0.61	-1.46	+88.19	-0.68	+0.19	-1.99	-0.67
SignGuard	-0.19	-89.09	+0.06	-0.54	+0.81	-0.09	-0.25	-0.67	-0.14	-88.84	-1.30
Bucketing	-0.33	-88.94	-0.44	-1.16	+3.51	+2.05	+0.97	+0.55	+0.78	-70.30	-0.30
DnC	-0.13	-87.46	+0.41	-0.03	-0.06	+0.03	-0.46	-0.53	+0.09	-87.37	-1.55
CenteredClipping	-0.34	-84.62	+0.80	+1.01	+5.89	-88.75	-0.12	+4.87	+0.17	+0.19	-0.75
FLTrust	-0.93	-0.28	-1.36	+0.23	-1.71	-1.25	-0.80	+1.09	+0.18	-0.33	-1.48
RFA	-0.04	+0.08	+0.64	+0.13	+0.72	+0.98	+0.52	-0.18	+1.02	+87.17	-0.23
Bulyan	+0.22	-87.61	+1.00	-0.44	+2.18	-87.49	+1.05	+2.06	+0.50	-0.49	-1.60
Median	-0.12	-7.19	+87.13	-0.10	+0.10	+0.79	+0.56	+0.55	+0.35	-2.14	-0.93
TrimmedMean	-0.21	-88.09	+0.24	-0.10	+3.07	-0.19	-0.02	+2.63	+0.27	-0.81	-0.41
MultiKrum	-1.86	-88.19	+0.25	-89.06	-81.58	-67.81	-83.27	-87.27	+10.87	-1.25	-0.37
Krum	-2.74	-2.43	-0.49	-3.93	-14.58	+3.00	-14.66	-13.68	+9.26	-1.31	-1.26
SimpleClustering	-0.13	-0.14	+84.19	+0.85	+38.76	+3.43	+89.61	+39.17	+13.98	-4.53	-0.78
Mean	-0.27	-0.34	+0.94	+0.74	+5.56	-88.42	-0.17	+4.39	+0.06	-1.99	-0.67

Table 10: Test accuracy difference (%) between partial and full client participation on MNIST, computed as $\Delta_{\mathrm{acc}} = Acc_{\mathrm{partial}} - Acc_{\mathrm{full}}$. Negative values indicate lower accuracy under partial participation.

rounds. Here, test accuracy is expressed on a scale from 0 to 1. Requiring the condition to hold for 50 consecutive rounds ensures that a successful attack causes persistent model degradation rather than a temporary fluctuation in accuracy.

A.2 Overall Results

Tables 4–9 report 990 test accuracy values obtained from 6 datasets, 15 defenses, and 11 attack settings, including No Attack. Each table presents the complete results for one dataset: Table 4 for MNIST, Table 5 for FashionMNIST, Table 6 for CHMNIST, Table 7 for CIFAR-10, Table 8 for CINIC-10, and Table 9 for CIFAR-100.

Based on the displayed accuracies in the 90 dataset–rule configurations, COMPASS has the lowest test accuracy in 48 cases: a strictly lower displayed value than every baseline in 46 cases and a tie for the lowest value in 2 cases. These counts describe numerical rankings at the reported precision. According to the attack success criterion defined above, COMPASS succeeds in 65 of the 90 cases. In comparison, PoisonedFL, which is the strongest baseline according to the number of successful cases, succeeds in only 21 cases.

A.3 Results on Individual Datasets

MNIST. Table 4 reports the results on MNIST. We include MNIST as a canonical and controlled benchmark for handwritten digit classification. COMPASS succeeds against all 15 existing defenses, corresponding to an attack success rate of 100%. This result demonstrates that COMPASS can consistently compromise a broad range of aggregation rules on a relatively simple classification task.

FashionMNIST. Table 5 presents the results on FashionMNIST. We select FashionMNIST because it retains the grayscale format of MNIST while introducing more visually ambiguous object categories. COMPASS succeeds against 14 of the 15 existing defenses, corresponding to an attack success rate of 93.3%. Its strong performance indicates that the attack remains effective when the classification task becomes more visually challenging.

CHMNIST. Table 6 reports the results on CHMNIST. We include CHMNIST to examine attack effectiveness in a medical image classification scenario that differs substantially from conventional vision benchmarks. COMPASS succeeds against 12 of the 15 existing defenses, corresponding to an attack success rate of 80.0%. This result shows that the effectiveness of COMPASS extends to histopathology images and is not restricted to general purpose image datasets.

CIFAR-10. Table 7 presents the results on CIFAR-10. We use CIFAR-10 as a standard benchmark for natural image classification with color inputs and more complex visual features. COMPASS succeeds against 10 of the 15 existing defenses, corresponding to an attack success rate of 66.7%. The results demonstrate that COMPASS remains effective when applied to a more challenging natural image classification task.

CINIC-10. Table 8 reports the results on CINIC-10. We select CINIC-10 because it contains images from multiple sources and provides greater data diversity and distribution variation than CIFAR-10. COMPASS succeeds against 8 of the 15 existing defenses, corresponding to an attack success rate of 53.3%. These results show that COMPASS can compromise multiple aggregation rules even when the training data exhibit increased diversity.

CIFAR-100. Table 9 presents the results on CIFAR-100. We include CIFAR-100 to evaluate attack effectiveness on a more difficult classification task with 100 classes and finer distinctions between categories. COMPASS succeeds against 6 of the 15 existing defenses, corresponding to an attack success rate of 40.0%. Although the success rate decreases as the task becomes more difficult, COMPASS still compromises several defenses in this challenging setting.

A.4 Discussion

Effectiveness across Datasets. COMPASS succeeds in 41 of the 45 cases represented by MNIST, FashionMNIST, and CHMNIST, and in 24 of the 45 cases represented by CIFAR-10, CINIC-10, and CIFAR-100. These results demonstrate that COMPASS remains effective across tasks with different image modalities, numbers of classes, levels of data diversity, and classification difficulty. Its effectiveness is most pronounced on MNIST and FashionMNIST, but it also compromises multiple defenses on the more challenging CINIC-10 and CIFAR-100 benchmarks. The results therefore confirm that COMPASS is not limited to a specific dataset or visual domain.

Comparison of Attacker Assumptions. Several baseline attacks operate under stronger assumptions about the information available to the adversary. In particular, IPM, ALIE, MinMax, MinSum, and FangAttack require access to information about benign clients. In contrast, COMPASS observes only updates to the global model and does not require prior knowledge of the aggregation rule used by the server. Despite relying on less information, COMPASS succeeds in more cases than any of these omniscient baseline attacks. These findings suggest that exploiting training dynamics and feedback from model aggregation can be more effective than relying on additional knowledge about benign clients or the aggregation mechanism.

B Impact of Partial Client Participation

B.1 Experimental Setup

We evaluate whether COMPASS remains effective under partial client participation. We use MNIST with 100 clients, an IID data partition, a learning rate of 0.05, and a malicious-client ratio of 20%. In the full-participation setting, all 100 clients participate in every communication round, whereas in the partial-participation setting, 10 clients are selected per round. All other experimental settings remain unchanged.

We report the test accuracy difference $\Delta_{\text{acc}} = Acc_{\text{partial}} - Acc_{\text{full}}$, where a negative value in an attack column indicates lower test accuracy under partial participation.

B.2 Results and Discussion

Table 10 compares full and partial client participation across 15 defenses and 10 attacks. Without an attack, partial participation changes the test accuracy by only -0.52 percentage points (pp) on average across the 15 defenses, indicating that the two participation settings maintain comparable clean performance.

The effects of partial participation on baseline attacks vary substantially across defenses. Partial participation strengthens some attack–defense combinations but weakens others, as reflected by both large negative and large positive differences in Table 10. This variation indicates that many existing attacks are sensitive to the client participation pattern.

In contrast, COMPASS exhibits consistent behavior across all evaluated defenses. Its accuracy difference is negative for all 15 defenses, ranging from -0.23 to -1.60 pp, with an average of -0.91

Defense	NoAttack	Gaussian	ALIE	SignFlipping	MinMax	FangAttack	IPM	MinSum	Mimic	PoisonedFL	COMPASS
LASA	3.36	3.35	3.34	2.94	3.33	**6.28**	3.35	3.38	3.27	3.01	2.98
FLDetector	5.68	5.47	5.52	5.34	6.12	**25.58**	4.51	6.06	5.80	3.12	3.53
SignGuard	3.21	3.17	3.22	2.93	3.29	**6.51**	3.12	3.28	3.18	2.85	2.86
Bucketing	3.53	3.55	3.55	3.17	3.51	**35.19**	3.39	3.53	3.44	3.16	3.22
DnC	3.11	3.20	3.16	2.76	3.21	**5.06**	3.25	3.19	3.21	2.75	3.43
CenteredClipping	3.05	3.19	3.12	2.81	3.11	**4.20**	3.06	3.15	3.10	2.75	2.51
FLTrust	3.06	3.22	3.00	2.75	3.08	**6.99**	3.16	3.24	3.02	2.82	2.97
RFA	3.16	3.15	3.20	2.71	2.92	**6.57**	3.10	3.16	3.16	2.88	2.86
Bulyan	28.36	28.41	27.87	21.47	32.99	**45.65**	30.82	34.75	30.50	23.46	34.22
Median	3.14	3.07	3.19	2.50	3.11	**9.54**	3.19	2.65	3.09	2.88	2.82
TrimmedMean	3.03	3.23	3.05	2.61	3.19	**4.22**	3.17	2.77	3.06	2.80	2.35
MultiKrum	4.33	4.55	4.57	4.03	4.61	**42.14**	4.66	4.17	4.54	4.15	4.77
Krum	4.57	4.54	4.43	4.08	4.40	**43.43**	4.63	4.51	4.55	4.05	5.16
SimpleClustering	3.53	3.29	3.61	3.00	3.56	**58.80**	3.69	3.20	3.39	3.26	2.59
Mean	3.15	3.24	3.19	2.69	3.21	**4.79**	3.17	2.81	3.01	2.81	2.38
Latency Avg	5.22	5.24	5.20	4.39	5.58	**20.33**	5.35	5.59	5.35	4.45	5.24

Table 11: Per-epoch training latency (in seconds) of different defenses under various attacks. The maximum value in each defense row is highlighted in bold.

pp. Under partial participation, COMPASS achieves test accuracies between 9.80% and 10.09%, compared with 10.03% to 11.40% under full participation.

These results demonstrate that partial client participation does not weaken COMPASS. The attack remains effective against all 15 defenses when only 10% of the clients participate in each communication round, despite the substantially smaller number of benign and malicious updates available to the server.

C Training Latency

C.1 Experimental Setup

We evaluate the training latency under different combinations of attacks and defenses to examine whether computational overhead may limit practical FL deployment. We recursively parse the MNIST experiment logs and extract the attack, defense, total runtime, and number of completed communication rounds. The experiments use SimpleCNN on MNIST with 100 clients, an IID data partition, a learning rate of 0.05, and a malicious-client ratio of 20%.

We define the latency of each run as its total runtime divided by the number of completed communication rounds. We repeat each experiment five times, remove the largest and smallest latency values, and average the remaining three values to reduce the influence of machine fluctuations.

C.2 Latency Results

Table 11 reports the average training latency of 15 defenses under different attacks. For most attack and defense combinations, the latency remains between approximately 2 and 6 seconds. The NoAttack setting requires 5.22 seconds on average, while COMPASS requires 5.24 seconds, a difference of only 0.02 seconds. This result shows that COMPASS introduces negligible overall runtime overhead.

Several attacks exhibit lower latency than NoAttack because malicious clients do not necessarily perform complete local training in every round. For example, PoisonedFL constructs its malicious update during the initial stage and subsequently reuses it without substantial modification, reducing the computation performed by malicious clients. Consequently, PoisonedFL requires 4.45 seconds on average, compared with 5.22 seconds for NoAttack. SignFlipping exhibits a similar effect and achieves the lowest average latency of 4.39 seconds among the evaluated attacks.

COMPASS also constructs its malicious update during the early attack stage and makes only limited adjustments in subsequent rounds. As a result, COMPASS is faster than NoAttack under 11 of the 15 defenses, including LASA, FLDetector, SignGuard, Bucketing, CenteredClipping, FLTrust, RFA, Median, TrimmedMean, SimpleClustering, and Mean. Its latency is higher under DnC, Bulyan, MultiKrum, and Krum, whose aggregation procedures introduce additional computation when processing the malicious updates. Nevertheless, the overall latency of COMPASS remains nearly identi-

cal to NoAttack, demonstrating that its adaptive attack mechanism does not create a practical runtime bottleneck.

FangAttack produces the highest latency for every evaluated defense. Its average latency reaches 20.33 seconds, more than 3.6 times the second highest average of 5.59 seconds. The increase is especially pronounced for SimpleClustering, Bulyan, Krum, MultiKrum, and Bucketing, whose latencies reach 58.80, 45.65, 43.43, 42.14, and 35.19 seconds, respectively. These results indicate that the attack generation procedure of FangAttack, combined with computationally intensive robust aggregation, is the dominant latency bottleneck in the evaluated framework.

D Hyperparameter Selection Protocol

To ensure a fair comparison, we apply the same search budget to every attack. For each combination of dataset and aggregation rule, each attack is allotted an identical budget of 100 trials over its own attack magnitude parameters, starting from the configuration recommended in its original paper. This removes any advantage that would arise from tuning one attack more heavily than another.

The attacks differ only in the feedback signal available to guide the search. Baseline attacks do not expose an internal feedback signal, so their search is guided by Local Validation Accuracy (LVA) alone: a trial is preferred when it yields a lower final LVA. COMPASS additionally produces the Matched Filter Score (MFS) defined in Eq. (15), and its search is guided by the combined objective described below. We stress that this difference is intrinsic to the attacks themselves, not an asymmetry in tuning effort: the ability to exploit aggregation feedback is precisely the capability COMPASS contributes, and the budget, search space size, and starting point are matched across all attacks.

D.1 Search Space

COMPASS exposes three attack magnitude parameters. The first is the scaling factor λ_0, which we refer to as the attack magnitude throughout this paper. It sets the overall magnitude of the malicious update and thus controls the attack strength. The remaining two parameters govern how this strength is distributed across model weights. The important magnitude m_I is the perturbation assigned to the Top-k important weights identified during the warm-up phase, and the unimportant magnitude m_U is the perturbation assigned to the remaining weights. These two parameters correspond to the entries of the auxiliary vector in Eq. (7) and adjust only the relative emphasis among weights rather than the overall attack strength.

The search begins from the default COMPASS configuration, in which $\lambda_0 = 3$, $m_I = 100$, and $m_U = 1$. Each parameter is searched within a bounded interval. The scaling factor ranges over $[1, 100]$, the important magnitude over $[3, 100]$, and the unimportant magnitude over $[0.8, 1.3]$. All other COMPASS parameters remain fixed at the values reported in Table 12.

D.2 Search Objective

Baseline attacks. Baseline attacks are searched using LVA as the sole objective. Each trial is scored by the accuracy reduction

$$D = 0.45\,(1 - a_{\min}) + 0.35\,(1 - a_{\text{tail}}) + 0.20\,(1 - a_{\text{fin}}), \tag{18}$$

where $a_{\min}$, a_{tail}, and a_{fin} are the minimum, final-window mean, and final LVA of a run. This objective prefers a sustained degradation over a transient drop, and the configuration with the largest D is selected.

COMPASS. Selecting parameters by LVA alone is difficult in regions where the attack has not yet taken effect, since the accuracy there stays high and nearly constant and provides little gradient for the search. COMPASS therefore augments the same accuracy-reduction term D with its own feedback signal. Let Q denote a bounded summary of the MFS trajectory that increases with its mean, its recent values, and the fraction of rounds in which it stays positive. The trial score is

$$J_{\text{tune}} = 0.55\,D + 0.35\,Q + 0.10\,R. \tag{19}$$

Parameter	Value
Warm-up length l_w	10 rounds
Top-k ratio k/d	0.05
Candidate set $\mathcal{N}$	$\{1, 2, 4, 8, 16, 32, 64, 100\}$
AIMD feedback window length l_f	20 rounds
Direction refresh window length l_r	50 rounds
Cosine monitoring start round t_c	50
Cosine threshold τ_c	0.4
Initial magnitude λ_0	3.0
Additive step η	0.2
Multiplicative factor γ	0.7
Lower bound $\lambda_{\min}$	1.0
Upper bound $\lambda_{\max}$	100.0
Probe to attack norm ratio r_z	10^{-4}
Direction refresh condition	LVA rebound after $c^{(t)} < 0.4$ (from round 50)

Table 12: COMPASS parameters used in the experiments. The initial attack magnitude λ_0 is tuned separately for each combination of dataset and aggregation rule.

where R rewards a longer run of consecutive rounds below the accuracy threshold. Both objectives share the identical accuracy-reduction term D and the same 100-trial budget; the only difference is the additional MFS-based term Q available to COMPASS. When two configurations are compared, one that satisfies the success criterion always ranks above one that does not, and the trial score resolves the comparison among configurations of equal success status.

D.3 Search Procedure

The search begins from each attack's default configuration and proceeds by local exploration together with occasional restart. The same local-exploration, step-adaptation, and restart mechanism is used for all attacks, with the trial score (D for baselines, J_{tune} for COMPASS) determining which neighbor is accepted and where to restart.

Local Exploration. At each step the search examines the neighborhood of the current configuration. It varies one parameter at a time, increasing or decreasing it by a step size while holding the others fixed. It also considers a small number of joint moves in which the scaling factor and the important magnitude change together, because these two parameters often become effective only when adjusted jointly. Each neighboring configuration is evaluated once, and the search moves to the neighbor with the highest score whenever that neighbor improves upon the current configuration.

Step Adaptation. When several consecutive steps produce no improvement, the search adjusts its step sizes rather than stopping. It reduces the step sizes to refine a promising region and enlarges them to explore farther when no nearby configuration improves the score. For COMPASS, step adaptation additionally enlarges the step sizes when the MFS remains weak, which indicates that the malicious update is still not entering aggregation, and when necessary applies a larger jump to the scaling factor and the important magnitude to leave an ineffective region.

Restart. When the neighborhood of the current configuration is exhausted, the search samples a new configuration from a more distant part of the search space and resumes local exploration around it. The restart is guided by the most promising trials observed so far, so that exploration concentrates on high-scoring regions. The process stops when a successful configuration is found or when the budget of 100 trials is reached.

E Detailed Validation of the Matched Filter Feedback

E.1 Purpose of the Validation

The MFS is intended to answer a narrower question than test accuracy: whether a known component of a malicious update is present in the update released by the server. Test accuracy is affected by the

complete optimization history and may react only after an attack has accumulated. By contrast, a matched filterr watermark provides a directional probe of the current aggregation step. This distinction is important for COMPASS because the attacker observes the released global models but cannot inspect the server's client selection, clipping coefficients, or aggregation weights.

We validate two properties of this feedback mechanism. First, the directional response should recover an analytic value when the aggregation rule has a known response. Second, it should distinguish aggregation rules that reject the watermarked direction from rules that retain it partially or transmit it in proportion to malicious acceptance. The experiment does not use correlation with test accuracy as the definition of validity. Accuracy is retained as an external outcome, while the primary reference is the output of controlled aggregation replays.

Let the watermark inserted at round t be

$$\mathbf{z}_t = \eta_t P_t, \tag{20}$$

where P_t is the known sparse signed pattern and η_t is its amplitude. The score in Eq. (15) can be written as

$$\widehat{\tau}_t^{\mathrm{MFS}} = \frac{\langle \Delta \mathbf{w}^{(t)}, \mathbf{z}_t \rangle}{\|\mathbf{z}_t\|_2^2} = \frac{\langle \Delta \mathbf{w}^{(t)}, P_t \rangle}{\eta_t \|P_t\|_2^2}. \tag{21}$$

If the aggregation rule transmits the probe with local directional gain τ_t, the watermark contribution to the released update is approximately $\tau_t \mathbf{z}_t$. Equation (21) then estimates τ_t plus the projection of the remaining global update onto P_t. The replay described below removes this background projection and supplies a direct reference for the channel observed by MFS.

E.2 Experimental Setup

The validation uses MNIST, SimpleCNN, FedSGD, an IID partition, and full participation by 100 clients, of which 20 are malicious. We evaluate the 15 aggregation rules used in the main experiments: Mean, Krum, MultiKrum, Bulyan, SignGuard, DnC, SimpleClustering, FLDetector, Median, TrimmedMean, LASA, FLTrust, RFA, CenteredClipping, and Bucketing. Each rule is evaluated with seeds 42 through 46, giving 75 scheduled runs. A run contains at most 1000 communication rounds.

Feedback from MFS is disabled in this experiment. Thus, the score does not change the COMPASS attack magnitude through the Additive Increase Multiplicative Decrease (AIMD) rule. This separation prevents the controller from altering the process used to validate the sensor. During active attack rounds, the native watermark multiplier follows a predetermined randomized block schedule over $\{0.5, 1, 2\}$. A separate randomized block schedule assigns an output levWe apply singular value decomposition tol attack dose $\lambda_t \in \{0, 0.25, 0.5, 0.75, 1\}$. Both schedules are fixed by the seed before the corresponding MFS values are observed.

The evaluator also constructs two Rademacher directions that are orthogonal to the native watermark and do not consume the training random state. Each direction is applied with amplitude

$$\epsilon = \alpha \,\mathrm{median}_i \, \|u_i^{(t)}\|_2, \quad \alpha \in \{10^{-4}, 10^{-3}, 10^{-2}\}. \tag{22}$$

These held out probes characterize the local directional response of the aggregation rule. They are diagnostic quantities and are not available to, or used by, COMPASS during training.

E.3 Paired Aggregation Replay

Let U_t denote the attacked update matrix and let M_t contain the malicious row indices. For a direction q_t, let $Q_t(q_t)$ place q_t in the rows indexed by M_t and zero in all other rows. Every diagnostic branch starts from an identical copy of the aggregation state and an identical random state. Client identities, row order, server context, and all update components other than the probe are held fixed.

For an independent held out direction, the evaluator computes

$$\begin{aligned} g_-(\epsilon) &= \mathcal{A}_t(U_t - \epsilon Q_t(q_t)), \\ g_0 &= \mathcal{A}_t(U_t), \\ g_+(\epsilon) &= \mathcal{A}_t(U_t + \epsilon Q_t(q_t)). \end{aligned} \tag{23}$$

The central directional transmission is

$$\tau_t(q_t, \epsilon) = \frac{\langle [g_+(\epsilon) - g_-(\epsilon)]/(2\epsilon), q_t \rangle}{\|q_t\|_2^2}. \tag{24}$$

The native replay removes the actual watermark before constructing its three branches. If $U_t^0 = U_t - \eta_t Q_t(P_t)$, the positive branch $\mathcal{A}_t(U_t^0 + \eta_t Q_t(P_t))$ is exactly the attacked update matrix used by the real aggregation call. The corresponding central response is denoted by τ_t^{nat}. Before applying the experimental output dose, the positive diagnostic branch is required to match the real aggregation output within fixed numerical tolerances. A mismatch invalidates the run rather than being treated as evidence for or against MFS.

We additionally replace every malicious row in U_t with the clean update produced by the same client in the same round. This gives U_t^{clean} and the complete attack influence

$$I_t = \frac{\|\mathcal{A}_t(U_t) - \mathcal{A}_t(U_t^{\mathrm{clean}})\|_2}{\|\mathcal{A}_t(U_t^{\mathrm{clean}})\|_2 + \delta}. \tag{25}$$

The controlled dose is applied to the released aggregation output as

$$g_t(\lambda_t) = g_t^{\mathrm{clean}} + \lambda_t \left(g_t^{\mathrm{attack}} - g_t^{\mathrm{clean}} \right). \tag{26}$$

This construction makes the applied complete attack influence exactly zero at $\lambda_t = 0$, while preserving the aggregation response measured before interpolation.

Where the aggregation implementation exposes an unambiguous attribution, we record its malicious acceptance mass a_t. Depending on the rule, a_t is the malicious fraction among selected rows, the normalized malicious weight, the malicious coordinate source fraction, or the effective malicious weight in an iterative estimator. This quantity is not computed from the watermark response. Agreement between a_t and τ_t^{nat} therefore provides an independent check of whether the directional probe recovers the mechanism used by the aggregator.

E.4 Recovery of the Aggregation Mechanism

Mean supplies an analytic calibration. With 20 malicious clients among 100 equally weighted participants, adding the same probe to every malicious row produces the directional gain $|M_t|/n = 0.2$. The two valid Mean runs recover median gains of 0.199981 and 0.199986. CenteredClipping also yields values between 0.199945 and 0.200026 across all five seeds, and the four numerically valid FLDetector runs yield approximately 0.2. These observations verify both the sign and scale of the transmission estimator.

Table 13 compares native transmission with the independently recorded acceptance mass. The value shown for each rule is the median of its run medians. A valid run must contain at least one finite MFS observation before collapse. This requirement leaves 64 run summaries and is most restrictive for SimpleClustering, where only one seed remains numerically valid before collapse.

Across the 64 run summaries, Spearman correlation between median native transmission and median mechanism acceptance is 0.987. The correlation remains between 0.984 and 0.993 when any one aggregation rule is removed. Computing the relationship separately for each seed gives correlations from 0.971 to 1.000. At the observation level, the absolute difference $|\tau_t^{\mathrm{nat}} - a_t|$ has median zero; 80.5% of observations differ by no more than 0.01, and 90.5% differ by no more than 0.05. Communication rounds are repeated observations, so these counts are not treated as independent statistical replicates. Their role is to show the numerical agreement that underlies the run level result.

The pattern in Table 13 is more informative than a pooled association alone. Krum, MultiKrum, DnC, Bucketing, and SignGuard have a median native response of zero, consistent with rejection of the watermarked rows or direction. Mean, CenteredClipping, and FLDetector transmit approximately 0.2, the malicious client proportion. Bulyan and RFA retain intermediate amounts that closely match their selected fraction or effective weight. Median, TrimmedMean, FLTrust, and LASA produce smaller responses determined by coordinate selection, weighting, or layer filtering. The probe therefore separates blocked transmission, partial survival, and near proportional transmission without using test accuracy.

Aggregation rule	Valid runs	Observations	$\widetilde{\tau}^{\mathrm{nat}}$	$\widetilde{a}$	$\widetilde{\lvert\tau - a\rvert}$
Bucketing	5	693	0.0000	0.0000	0.0000
Bulyan	5	717	0.1500	0.1500	0.0000
CenteredClipping	5	249	0.2000	0.2000	0.0000
DnC	5	2841	0.0000	0.0000	0.0000
FLDetector	4	121	0.2000	0.2000	0.0000
FLTrust	5	1481	0.0000	0.0000	0.0000
Krum	5	889	0.0000	0.0000	0.0000
LASA	5	4506	0.0027	0.0373	0.0343
Mean	2	15	0.2000	0.2000	0.0000
Median	5	1995	0.0007	0.0039	0.0032
MultiKrum	5	3825	0.0000	0.0000	0.0000
RFA	4	1453	0.0659	0.0659	0.0000
SignGuard	5	2643	0.0000	0.0000	0.0000
SimpleClustering	1	5	0.9999	1.0000	0.0001
TrimmedMean	3	28	0.1255	0.1577	0.0310

Table 13: Agreement between native watermark transmission and aggregation mechanism acceptance before model collapse. Rule level values are medians of run medians.

E.5 Relation to the Attacker's Observable Score

The counterfactual response removes the projection of benign optimization onto the watermark. The score available to COMPASS does not have this advantage and is consequently noisier. Nevertheless, it retains directional information. Among 53 runs with sufficient finite variation, the association between raw MFS and the applied malicious influence projected onto the watermark is positive in 43 runs. The corresponding counts are 46 of 53 for normalized MFS and 45 of 53 for cosine MFS. Median within run Spearman correlations are 0.140, 0.163, and 0.179, respectively. At the pooled observation level, raw and cosine MFS have correlations of 0.218 and 0.212 with the applied directional projection. These are modest effects, but their consistent sign shows that the released global update contains information about the injected direction.

The same result also defines the limit of the current evidence. A single uncalibrated MFS value is not a reliable universal estimator of complete attack magnitude. The median run level area under the ROC curve for distinguishing $\lambda_t = 0$ from $\lambda_t > 0$ using raw MFS is 0.499. Within balanced schedule cycles, the difference between $\lambda_t = 1$ and $\lambda_t = 0$ is not significant under a paired signed rank test ($p = 0.240$). We also observe no stable monotonic relationship between the watermark multiplier and a single round score. These findings are expected when the watermark is deliberately small and the released update contains a changing background projection. They motivate the use of temporal trends and LVA in COMPASS rather than a fixed threshold on MFS, but they do not by themselves establish the benefit of the AIMD controller.

E.6 Why the Probe Adds Information

Directional transmission, complete attack influence, and test accuracy answer different questions. Several runs have zero median watermark transmission while retaining a nonzero complete attack influence. For example, the run median relative influence under DnC reaches approximately 0.28 to 0.37 for seeds 44 through 46 even though native transmission remains zero. MultiKrum similarly exhibits nonzero complete attack influence while its native transmission is zero. Conversely, the Bulyan runs for seeds 44 through 46 retain partial directional transmission even though their final test accuracies remain between 0.9114 and 0.9679 when the runtime limit is reached. Accuracy alone therefore cannot identify whether a particular component was blocked, partially retained, or transmitted.

This distinction establishes the value of matched filter sensing within COMPASS. The attack is not restricted to observing whether accuracy has already fallen. It has access to a deliberately injected direction whose survival reveals how the current aggregation rule treats malicious content. The strong agreement with mechanism acceptance demonstrates that this signal has a concrete interpretation across selection, weighting, clipping, and coordinate based aggregation. The weaker attacker side correlations show that the practical observation is affected by background updates, but they also confirm that directional information survives in the released model. Taken together, the results val-

	Successful Seeds		Final Accuracy (%)	
Defense	Static	Full	Static	Full
FLTrust	2/3	**3/3**	39.63 ± 40.00	$\mathbf{11.35 \pm 0.00}$
Krum	2/3	**3/3**	32.24 ± 30.63	$\mathbf{10.40 \pm 0.73}$
LASA	**2/3**	1/3	$\mathbf{40.39 \pm 41.06}$	69.51 ± 41.12
Median	3/3	3/3	$\mathbf{10.99 \pm 0.50}$	11.35 ± 0.00
RFA	2/3	**3/3**	39.09 ± 41.42	$\mathbf{10.32 \pm 0.73}$
SignGuard	2/3	2/3	$\mathbf{39.85 \pm 41.56}$	40.44 ± 41.13
FLTrust, Krum, RFA	6/9	**9/9**	36.99 ± 37.80	$\mathbf{10.69 \pm 0.76}$

Table 14: Controller ablation on MNIST over three seeds. Static COMPASS fixes the attack magnitude and disables the feedback triggered direction refresh, while Full COMPASS keeps the feedback controller; all other components are identical. Final accuracy is the mean and standard deviation across seeds in percentage points, where a lower value is a stronger attack and the stronger variant is shown in bold. The last row aggregates the three defenses on which a constant magnitude is unreliable.

idate MFS as a mechanism sensitive sensing primitive and support the broader COMPASS premise that broadcast global models expose information beyond task accuracy.

E.7 Scope of the Evidence

The experiment validates a local directional channel for the evaluated MNIST, FedSGD, COMPASS, and aggregation implementations. It does not claim that watermark transmission equals the causal effect of the complete malicious update, nor that transmission must produce an immediate change in accuracy. Six replay failures show that diagnostic equivalence must be verified rather than assumed. Numerical collapse limits the evidence for SimpleClustering and parts of FLDetector. The independently generated held out probes characterize local aggregation response, but they are not zero response controls because each is measured against its own matched direction. No separate shuffled direction score or temporally shifted MFS score was computed in this matrix.

Finally, feedback was disabled throughout the validation. The results establish the construct validity and mechanism discrimination of the MFS channel, not the performance gain of closed loop magnitude adaptation. A direct evaluation of the controller requires paired runs with feedback enabled, a fixed magnitude policy, and an LVA only policy under identical seeds. The current evidence nevertheless verifies the essential premise required by such a controller: the released global model contains a measurable, aggregation dependent trace of the direction deliberately inserted by COMPASS.

F Ablation on the Feedback Controller

The distinguishing component of COMPASS is the feedback controller that reads the MFS and LVA and then adjusts the attack magnitude $\lambda^{(t)}$ across rounds. All other components of COMPASS, including the estimated training direction, the Top-k important weights, and the fixed orthogonal attack direction, are shared with prior direction based attacks. A natural question is therefore whether the controller contributes to the attack in a measurable way, or whether a constant magnitude would suffice. This section isolates the controller and quantifies its effect.

F.1 Ablation Design

Variants. We compare two variants that differ only in whether the attacker consumes feedback. Full COMPASS is the complete method described in the main text, in which the controller updates $\lambda^{(t)}$ through the AIMD rule and, when the LVA feedback indicates recovery, refreshes the fixed attack direction from the latest global models. Static COMPASS freezes the magnitude at its initial value, so that $\lambda^{(t)} = \lambda^{(0)}$ for every attack round, and disables the feedback triggered direction refresh. Every remaining element is held identical between the two variants, including the warm-up length, the estimated training direction, the Top-k set, the orthogonalization, the initial magnitude $\lambda^{(0)}$, the probe $\mathbf{z}$, and the set of malicious clients. The probe is still injected in Static COMPASS, and

MFS and LVA are still recorded, but they no longer influence the malicious update. The manipulation is thus confined to the feedback pathway.

Paired Protocol. For a fixed random seed, the initialization, the data partition, the client participation schedule, the malicious client set, the warm-up trajectory, and the round at which the orthogonal direction is fixed are all determined before any feedback is consumed. Consequently, the two variants remain identical up to the moment the controller first acts, and any later divergence is attributable to feedback alone. We verify this pairing empirically. Under the same seed, the test accuracy of the two variants coincides throughout the warm-up phase and at the round where the attack direction is fixed. This design removes the usual confounders of random initialization and data assignment and lets us read the controller effect directly.

Experimental Setup. We use MNIST with SimpleCNN under FedSGD, following the training settings of the detailed main results. We evaluate the same six representative defenses used for the matched filter analysis, namely FLTrust, Krum, LASA, Median, RFA, and SignGuard, which together span trust based aggregation, selection based aggregation, layer wise filtering of important weights, coordinate wise statistics, robust geometric aggregation, and sign based screening. We repeat every configuration over three random seeds, giving 36 runs in total. The adversary controls 20% of the 100 clients, and we retain the attack success criterion of the detailed main results, namely a test accuracy below 0.2 for 50 consecutive rounds.

Evaluation Metrics. The mean accuracy over a run mixes two distinct questions, whether the attack eventually succeeds and how quickly it does so, and it is further distorted by early stopping. For an adversary, the operational quantity of interest is reliability, that is, whether the attack succeeds across independent initializations rather than on a fortunate subset of them. We therefore report the number of successful seeds and the final test accuracy summarized as its mean and standard deviation across seeds. The standard deviation measures how consistently the attack reaches its effect, and the largest final accuracy across seeds measures the worst case, in which the attack performs least well.

F.2 Experiment Results

Table 14 reports the outcome for each defense. Three regimes emerge.

Defenses Where the Controller Removes the Failure Mode. Against FLTrust, Krum, and RFA, a constant magnitude is close to the tolerance boundary of the defense, and Static COMPASS behaves like a coin toss. It succeeds on only two of the three seeds for each of these defenses, and on the remaining seed the attack is fully neutralized, leaving the final accuracy between 75.5% and 97.7%. Full COMPASS succeeds on all three seeds for every one of these defenses. The controller raises the magnitude from its initial value until the malicious signal reliably survives aggregation, which drives the final accuracy to about 11% on every seed. The effect is most visible in the variability across seeds. Aggregated over these three defenses, Static COMPASS succeeds in six of nine runs with a final accuracy of 36.99 ± 37.80, whereas Full COMPASS succeeds in nine of nine runs with a final accuracy of 10.69 ± 0.76. The controller reduces the standard deviation by roughly a factor of fifty and converts an attack whose outcome depends on the seed into one whose outcome is essentially deterministic.

Defenses Where a Constant Magnitude Already Suffices. Against Median and SignGuard, the two variants are statistically indistinguishable. Both succeed on the same number of seeds and reach a similar final accuracy, and the seed on which the attack fails under SignGuard is the same for both variants. Here the initial magnitude is already adequate, or the obstacle to the attack lies outside the magnitude, so the controller neither helps nor harms. The controller never worsens the worst case relative to the static variant on these defenses.

The LASA Exception. LASA is the single defense on which Full COMPASS does not improve over Static COMPASS, and it succeeds on one seed rather than two. This behavior is consistent with the design of LASA, which performs layer wise filtering based on the anomalous characteristics of important weights. COMPASS concentrates its perturbation on exactly the Top-k important weights that LASA inspects. When the controller increases the magnitude, it also increases the norm of

the perturbation on these weights and therefore makes the malicious update easier for LASA to identify and remove. Because the attack direction and the Top-k set are held fixed between the two variants, this outcome is not a limitation of the controller but a property of the attack direction, and it exposes the important weight subspace as the surface on which COMPASS is most detectable. This observation is separate from the controller ablation and suggests that importance aware filtering is a principled direction for defending against COMPASS.

F.3 Discussion

The ablation supports a specific and defensible claim about the value of COMPASS. On the defenses where a fixed perturbation sits at the boundary of what the aggregation rule tolerates, feedback is what separates a reliable attack from an unreliable one. Static COMPASS is a strong attack on a favorable seed and a failed attack on an unfavorable one, while Full COMPASS succeeds on every seed and reduces the variability of its effect to a level that is negligible for practical purposes. Because the two variants are paired on initialization, data, participation, and the fixed attack direction, this improvement is caused by the feedback pathway rather than by any confounding factor. Reliability of this kind, rather than a marginal reduction in average accuracy, is the property that matters when an adversary cannot choose the initialization of the system it attacks.

The single case in which feedback does not help, LASA, strengthens rather than weakens this reading. It localizes the vulnerability of COMPASS to the important weight subspace on which it concentrates energy, and it shows that when a defense targets that subspace directly, adjusting the magnitude cannot compensate for a detectable direction. This finding is useful in its own right, since it identifies importance aware and layer wise filtering as an effective countermeasure and clarifies the boundary of the threat that COMPASS poses.

G Geometric Evidence of Lateral Trajectory Displacement

This section studies how COMPASS changes the global optimization path. The attack constructs its malicious direction to be orthogonal to the dominant training direction estimated during the warm-up phase. This construction alone does not establish that the server model moves laterally: an aggregation rule may reject the malicious component, and accepted components may cancel over time. We therefore measure the realized global trajectory and test whether COMPASS produces a persistent displacement outside the local subspace followed by benign training. We then relate this displacement to predictive failure and to the loss along the linear path between paired models.

The claim examined here is deliberately narrower than topological separation between local minima. A two dimensional projection cannot establish such separation, and a positive loss barrier on a straight path does not exclude a curved low loss connection. Our evidence supports lateral departure from benign optimization, entry into a region associated with incorrect predictions, and, for many pairs, separation along the direct interpolation path.

G.1 Experimental Design

We use MNIST, SimpleCNN, and FedSGD. The training contains 100 clients with full participation, of which 20% are controlled by COMPASS. Training uses SGD with learning rate 0.05, momentum 0.9, weight decay 5×10^{-4}, and batch size 16. We pair a benign run with a COMPASS run using the same seed, initialization, data partition, participation pattern, and training configuration. Seeds 42 through 46 give five paired trajectories per rule. The complete trajectory matrix covers Median, SignGuard, Krum, RFA, FLTrust, LASA, FLDetector, Bucketing, DnC, CenteredClipping, Bulyan, TrimmedMean, MultiKrum, SimpleClustering, and Mean. We complete the trajectory and held out loss analyses for all five seeds under all 15 rules.

At every round, the server applies one fixed, deterministic 1024 dimensional CountSketch map to the global model and aggregated update. The same map is used throughout every pair, so paired differences and inner products share a common coordinate system. Absolute global parameters are also stored in half precision at rounds 0, 1, 10, 11, every 25 rounds thereafter, and at the final round. The full parameter records provide an independent check near the onset of the attack, while the sketches retain the temporal resolution needed for trajectory analysis.

G.2 Local Trajectory Decomposition

Let $\mathbf{w}_t^C$ and $\mathbf{w}_t^A$ denote the global models in the benign and COMPASS conditions. We estimate the local benign tangent space causally from the preceding $h = 10$ benign steps. The local update matrix is

$$\begin{aligned} \mathbf{U}_t &= \left[\Delta\mathbf{w}_{t-h+1}^C, \ldots, \Delta\mathbf{w}_t^C\right], \\ \Delta\mathbf{w}_j^C &= \mathbf{w}_j^C - \mathbf{w}_{j-1}^C. \end{aligned} \tag{27}$$

We apply singular value decomposition to $\mathbf{U}_t$ and retain the smallest number of left singular vectors explaining at least (95%) of its squared singular values. Similar to the PCA-based procedure used by COMPASS to identify the dominant training direction, this analysis extracts the principal structure of recent benign updates, but retains a local subspace rather than a single direction. Writing these vectors as the columns of $\mathbf{Q}_t$ gives the tangent and normal projectors

$$\mathbf{P}_t^T = \mathbf{Q}_t\mathbf{Q}_t^\top, \qquad \mathbf{P}_t^N = \mathbf{I} - \mathbf{P}_t^T. \tag{28}$$

We first consider the accumulated displacement

$$\boldsymbol{\delta}_t = \mathbf{w}_t^A - \mathbf{w}_t^C. \tag{29}$$

The complementary quantity is the difference between the two realized global steps,

$$\mathbf{a}_t = \left(\mathbf{w}_t^A - \mathbf{w}_{t-1}^A\right) - \left(\mathbf{w}_t^C - \mathbf{w}_{t-1}^C\right). \tag{30}$$

The second quantity is observational. Once the two models diverge, it is not a counterfactual malicious update evaluated at the same parameter vector. It is nevertheless useful for determining how the paired trajectories separate from one round to the next. For either quantity $\mathbf{x}$, we report the fraction of energy outside the local benign tangent space,

$$R_N(\mathbf{x}) = \frac{\|\mathbf{P}_t^N\mathbf{x}\|_2^2}{\|\mathbf{x}\|_2^2}. \tag{31}$$

Near orthogonality alone is not sufficient in a high dimensional parameter space. We also test whether successive deviations accumulate rather than cancel. For the set $\mathcal{T}$ of rounds after warm up, directional coherence is

$$C = \frac{\left\|\sum_{t\in\mathcal{T}} \mathbf{a}_t\right\|_2}{\sum_{t\in\mathcal{T}} \|\mathbf{a}_t\|_2}. \tag{32}$$

We compare the observed value with 1000 sign randomizations. Each randomization independently multiplies every $\mathbf{a}_t$ by $+1$ or -1, preserving its norm, direction, and temporal position while removing systematic accumulation.

G.3 Lateral Displacement

Table 15 gives the results for all 15 aggregation rules. Across the 75 pairs, 98.14% of the displacement energy and 90.74% of the observed step difference energy lie outside the local benign tangent space. The corresponding pair medians are 98.69% and 95.63%. The displacement result is stable across aggregation paradigms: the rule means range from 93.79% for TrimmedMean to 99.51% for SimpleClustering. Thus, COMPASS does not primarily move the model farther forward or backward along the path taken by benign optimization.

The normal component contains more than half of the step difference energy in 94.21% of the analyzed attack period. Hence the reported displacement is not produced by a small number of isolated lateral steps.

Aggregation rule	Displacement R_N (%)	Step R_N (%)	Final I_{acc} (pp)	Barrier count
LASA	98.05	93.80	1.11	5/5
FLDetector	98.73	94.46	87.96	2/5
SignGuard	98.57	93.08	36.03	3/5
Bucketing	97.75	77.95	37.25	3/5
DnC	98.72	87.54	17.58	4/5
CenteredClipping	99.26	99.39	88.07	0/5
FLTrust	98.56	77.64	54.06	2/5
RFA	99.23	97.10	55.24	5/5
Bulyan	98.34	94.21	88.06	4/5
Median	98.81	97.34	74.19	4/5
TrimmedMean	93.79	93.37	88.92	2/5
MultiKrum	98.19	83.39	19.44	4/5
Krum	98.43	75.89	83.83	3/5
SimpleClustering	99.51	99.53	88.96	0/5
Mean	96.12	96.48	88.97	4/5
Overall	98.14	90.74	60.64	45/75

Table 15: Geometry and behavioral effect of COMPASS over five paired seeds for each aggregation rule. Normal energy is measured relative to the local benign tangent space. Accuracy impact is benign final accuracy minus COMPASS final accuracy in percentage points. Barrier count reports the number of pairs whose sample weighted cross entropy barrier exceeds 0.1. The overall row averages the continuous metrics across all 75 pairs and aggregates the barrier count.

The full parameter calculation confirms that the sketch result is not created only by compression or by late numerical divergence. For Median with seed 42 at round 11, 99.65% of the paired displacement energy and 98.36% of the observed step difference energy are normal to the benign subspace. The lateral component is therefore visible when COMPASS first begins to alter the optimization path.

Directional coherence distinguishes accumulation from arbitrary orthogonal noise. For Median, SignGuard, Krum, RFA, FLTrust, and LASA, the observed coherence exceeds the 95th percentile of the sign randomization distribution in 26 of 30 pairs. The test passes all five seeds for Median, SignGuard, FLTrust, and LASA, and three of five seeds for Krum and RFA. The ratio of the rule mean coherence to the corresponding null threshold ranges from 1.14 to 1.73. Together with the high frequency of normal dominated steps, this result shows that the accepted deviations have sufficient directional agreement to build a persistent displacement.

G.4 Connection to Incorrect Behavior

We use a paired, same round criterion to avoid treating low accuracy before convergence as attack success. A trajectory enters the incorrect behavior region only when the benign model has test accuracy of at least 80% while its COMPASS counterpart is below 18%. Under this criterion, 24 of the 30 pairs for Median, SignGuard, Krum, RFA, FLTrust, and LASA, and 62 of all 75 pairs, enter the incorrect region. At the last common round, 50 of 75 COMPASS models remain below 18% accuracy, and 51 pairs exhibit an accuracy reduction greater than 50 percentage points. Across the attack period, the mean paired accuracy gap is 56.12 percentage points, and the median final COMPASS accuracy is 11.35%.

The rules that recover provide a useful qualification rather than a contradiction. LASA has only a 1.11 percentage point mean final impact, and DnC and MultiKrum also recover in several seeds. Their displacement ratios remain high. Robust aggregation can therefore restore the decision function without retracing the benign optimization path. Conversely, the large final impact for Median, Krum, RFA, and FLTrust shows that lateral displacement often remains behaviorally consequential. This combination is more informative than final accuracy alone: COMPASS either maintains an incorrect model or forces the defense to reach correct behavior from a substantially displaced parameter region.

G.5 Loss Along the Direct Path

For all 15 rules, we evaluate whether paired checkpoints are separated by elevated classification loss. We do not use loss values from the training log. At each interpolated model, sample weighted cross entropy is recomputed on all 10,000 MNIST test examples,

$$L(\mathbf{w}) = \frac{1}{N} \sum_{i=1}^{N} \mathrm{CE}\left(f_{\mathbf{w}}(\mathbf{x}_i), y_i\right). \tag{33}$$

We evaluate 21 equally spaced points on

$$\mathbf{w}(\lambda) = (1-\lambda)\mathbf{w}^C + \lambda \mathbf{w}^A, \qquad \lambda \in [0,1], \tag{34}$$

and define the barrier as

$$B = \max_{\lambda \in [0,1]} L(\mathbf{w}(\lambda)) - \max\left\{L(\mathbf{w}^C), L(\mathbf{w}^A)\right\}. \tag{35}$$

Forty five of the 75 pairs have $B > 0.1$, and the median barrier is 0.43. All five RFA pairs have a barrier above 1.3 at their last finite common checkpoint. More notably, the five LASA endpoints recover to high accuracy but retain barriers between 1.76 and 2.10. Bucketing, DnC, Bulyan, MultiKrum, and Mean also exhibit a positive barrier in at least three seeds. Recovery of the prediction function therefore does not imply return to the benign mode.

For the 30 pairs without a barrier above 0.1, the attacked endpoint is often already near random guessing and has greater loss than every interior point. This behavior is especially clear for CenteredClipping and SimpleClustering, whose barrier counts are zero even though their final accuracy reductions are approximately 88 percentage points. Such a curve supports entry into an incorrect high loss region but not convergence to a second low loss solution. When a stored half precision checkpoint contains a nonfinite value, interpolation uses the last finite common checkpoint. Numerical divergence is not counted as evidence of mode separation.

A positive direct path barrier establishes linear separation, not disconnected basins under all possible paths. The evidence therefore supports two outcomes. Some COMPASS trajectories leave benign optimization and terminate in an incorrect high loss region. Others recover accurate predictions but reach a mode separated from the benign endpoint along the direct path. Both outcomes are consistent with persistent lateral displacement; only the second provides evidence of mode separation.

G.6 Implications for COMPASS

The measurements connect the attack construction to its realized global effect. COMPASS first selects a direction outside the dominant benign direction, then reuses that direction while adapting its magnitude from global feedback. The paired trajectories show that a predominantly normal component survives robust aggregation, the randomization test shows that it accumulates coherently, and the paired accuracy measurements show that the resulting displacement is often accompanied by incorrect predictions. The loss interpolation further shows that many endpoints are not connected to their benign counterparts by a low loss straight path.

This mechanism explains why COMPASS is not equivalent to reversing a benign update or adding unstructured noise. A reversed component remains on a direction that later benign updates can correct. Independent noise can cancel over rounds or be removed as an outlier. COMPASS repeatedly reinforces a direction that is weakly represented by the recent benign trajectory. Its effect consequently depends on the accumulation of accepted updates, a failure mode not captured by examining each submitted update in isolation. Together with the controlled orthogonal and reverse direction comparison in Table 3, these results provide evidence at both the intervention and trajectory levels for the value of the orthogonal attack construction.