CERTIFICATES EXPIRE: DURABLE UNLEARNING FOR LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Machine unlearning removes designated knowledge from a trained language model, with release-time checks assessing whether the requested knowledge erasure has been achieved. Durable unlearning faces a distinctive obstacle: subsequent adversarial fine-tuning moves the released parameters and can recover knowledge that passed those checks. Identical release-time outputs need not fix recovery cost, while resistance to sampled attacks does not certify every trajectory within a budget. To address these, we first establish a separation between release-time behavior and durability, locating the missing information in the geometry of the landing point. We then design **PLATEAU**, a framework that converts this geometry into a certified budget of fine-tuning updates. Within this framework, Directional Measure Estimation reads the forget-loss margin and directional gradient norm at release together with a uniform negative-curvature bound over a prescribed neighborhood. Barrier Certification then converts valid bounds into a closed-form safety radius and an integer budget of complete updates, covering every admissible trajectory and evaluation objective in the declared class without running attacks. In addition, Certified Durability Training uses the radius dependencies to regularize existing unlearning objectives, followed by re-evaluation of the updated release. Under a common declaration on TOFU, MUSE and WMDP-bio at matched forgetting and utility, **PLATEAU** attains the largest certified budget and the longest recovery on TOFU and MUSE, and every one of the 51 compared method–benchmark cells stays unrecovered within its budget. Together, the separation and certificate make durability a property of the released parameters: any unlearning procedure can be assessed through the same geometric quantities, with a guarantee expressed in the complete updates an adversary can take.

1 INTRODUCTION

Machine unlearning removes designated training information without retraining from scratch (Bourtoule et al., 2021; Guo et al., 2023), supporting deletion requests, copyright removal and hazardous-capability erasure in large language models (Maini et al., 2024; Li et al., 2024). Release-time checks assess the resulting model (Shi et al., 2025; Lynch et al., 2024), but deployed weights continue to be fine-tuned. Recovery data supplied through adaptation pipelines or fine-tuning services can move those weights toward the forgotten knowledge (Hu et al., 2025; Huang et al., 2024). A check at release therefore leaves a second question unanswered: how long does erasure survive subsequent training? Related work appears in Appendix A.

Figure 1 frames this gap geometrically. Releases accepted by the same check can require different fine-tuning budgets to recover knowledge. At fixed margin and curvature, a smaller directional gradient norm permits a larger safe neighborhood around the release. The two observations in Section 2 connect this dependence to checkpoint behavior. This raises the first question: ***I) Which quantities at the landing point determine the fine-tuning compute needed to undo the erasure?***

Existing defenses seek resistance through relearning attacks (Tamirisa et al., 2025; Wang et al., 2025; Yang et al., 2026) or loss-surface regularization (Fan et al., 2025a; Huang et al., 2025). A geometric certificate complements these approaches by controlling every trajectory in a declared displacement class, including updates chosen adaptively. This raises the second question: ***II) How can landing-point geometry certify a fine-tuning budget in closed form for any unlearning procedure?***

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.

The radius has a concrete operational meaning in a controlled fine-tuning service. Such a service can enforce the trainable parameter blocks, the cumulative displacement class and the norm of each complete update, then admit jobs whose total budget stays inside the certified radius. This statement accounts for optimizer preconditioning and weight decay. The radius also identifies which geometric quantities a training objective should improve. This raises the third question: ***III) How can the certified budget guide training and be evaluated against the measured recovery cost?***

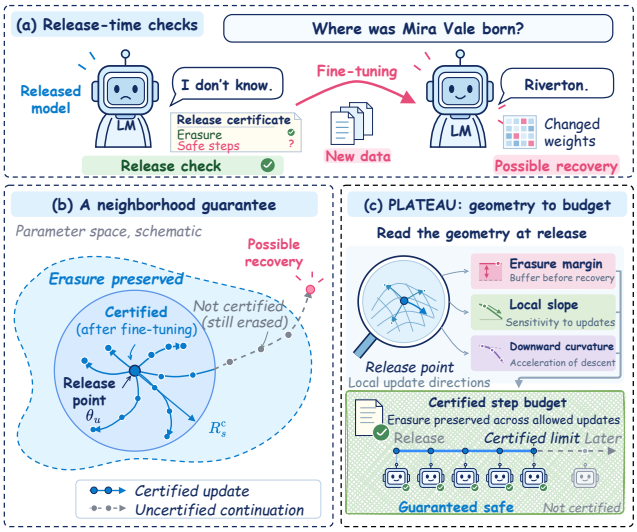


Figure 1: **Problem Illustration.** Release-time erasure can expire under fine-tuning. **I)** Release check: forgotten answers can return. **II)** Neighborhood guarantee: admissible updates preserve erasure inside a certified region. **III)** Geometry to budget: margin, slope and curvature determine a safe radius, converted into complete update steps during later updates.

We answer these questions with **PLATEAU**, which reads the geometry of any unlearned release. We prove that validated margin, slope and neighborhood-curvature bounds certify a radius and an integer update budget for every trajectory satisfying the declared displacement and step constraints. The evaluation pairs local-geometry estimates with recovery outcomes on three benchmarks and eight deployment cells, together with continuation and cost diagnostics. Both parts evaluate the same landing point: DME supplies its geometry, BC converts validated bounds into a guarantee, and CDT proposes an updated release whose quality and family-wide bounds are checked again. Our main contributions are:

❶ ***Durability Separation.*** We construct smooth input-dependent classifiers with identical, nonconstant release predictions and exponentially separated bounded-update recovery costs as width grows. The architecture and parameterization are fixed within each pair. Identical outputs therefore leave durability undetermined, and the missing information lives in parameter space. Margin, directional slope and negative curvature supply the parameter-space information used by the certificate.

❷ ***Geometric Certification Framework.*** DME supplies a worst-objective margin, directional gradient norm and uniform negative-curvature bound; BC converts them into a closed-form radius and an integer budget of complete updates; and CDT uses the radius dependencies to regularize a base objective before re-evaluation. A uniform descent bound and a worst-case sharpness construction support the certificate, while an additional descent condition yields a recovery upper bound. The guarantee covers every admissible trajectory independently of the unlearning procedure.

❸ ***Declaration-Matched Evaluation.*** We compare methods on TOFU, MUSE and WMDP-bio at matched forgetting and utility, under a common declaration. Five-seed summaries, audits and component ablations pair budget estimates with recovery cost. Appendix E adds the certificate comparison, declaration sensitivity, independent-release variation and the reproducibility package.

2 PRELIMINARIES

Unlearning problem. An unlearning procedure releases parameters θ_u that suppress knowledge from a forget set $\mathcal{D}_f$ while preserving performance on a retain set $\mathcal{D}_r$. For prompt–answer pairs (x, y), the length-normalized forget loss is

$$\mathcal{L}_f(\theta) \;=\; -\frac{1}{|\mathcal{D}_f|}\sum_{(x,y)\in\mathcal{D}_f}\frac{1}{|y|}\sum_{t=1}^{|y|}\log p_\theta\big(y_t \mid x,\, y_{<t}\big), \tag{1}$$

where p_θ is the next-token distribution. The retain loss $\mathcal{L}_r$ uses the same definition on $\mathcal{D}_r$: unlearning raises the former while preserving the latter.

Threat model. After release, an adversary supplies recovery data to a fine-tuning pipeline or updates distributed weights directly. Starting from θ_u, each complete optimizer update has Euclidean norm at most ηG, including preconditioning and weight decay. Every cumulative displacement must belong to a declared set $\mathcal{A}$: the full trainable space, a fixed subspace, or blockwise rank-at-most-s displacements. Incremental low rank alone does not preserve cumulative rank (Appendix F.4). Updates may be chosen adaptively using any objective. The protected loss family $\mathcal{G}$ contains $\mathcal{L}_f$ and declared paraphrase, format and language variants. All quantities use the released weight parameterization.

Durable unlearning. Recovery occurs when any protected loss falls strictly below a fixed threshold γ. TOFU uses the pre-unlearning model's held-out forget loss for this threshold; a retain-trained reference separately supplies the release-time gate. A certified radius R^{c}, valid within a prescribed neighborhood of radius ρ, satisfies

$$\inf_{\substack{\mathcal{L}\in\mathcal{G},\ v\in\mathcal{A}\\ \|v\|\le R^{\mathrm{c}}}} \mathcal{L}(\theta_u + v) \ \ge\ \gamma, \qquad B^{\mathrm{c}} \ =\ \frac{R^{\mathrm{c}}}{\eta G}. \tag{2}$$

After B updates the displacement is at most $\eta G B$, so Eq. (2) protects every trajectory of at most $\lfloor B^{\mathrm{c}} \rfloor$ complete steps. At zero budget it reduces to the declared release-time loss check; a positive radius protects a neighborhood. The guarantee concerns preservation of the declared loss threshold during subsequent updates.

Desiderata. The guarantee should be defense-agnostic, computed at any procedure's landing point; closed form, requiring no attack runs; compute-denominated, expressed in complete updates; and worst-case, covering every admissible trajectory and protected objective.

Observation 1

At fixed margin and curvature, the directional gradient norm at release orders the certified radius, and a smaller directional slope enlarges the uncapped radius under the same erasure threshold, which a fixed bound on each complete update turns into a larger integer step budget.

Figure 2 shows the loss landscape on a plane through a release point, with the black contour marking the threshold γ. The release starts above this threshold, and its margin is the loss decrease an attacker must accumulate before recovery. Holding the margin and curvature bound fixed isolates the effect of slope: a smaller directional gradient norm reduces the linear term in the worst-case loss decrease. The corresponding lower bound therefore reaches γ at a larger displacement, unless the prescribed neighborhood already caps the radius. This gives the same release-time loss more room to withstand subsequent movement. With a fixed bound on each complete optimizer update, that additional distance converts into a weakly larger integer budget of safe steps. The plane makes this dependence visible; certification extends the argument to every admissible displacement and every objective in the declared family, using a curvature bound valid throughout the neighborhood.

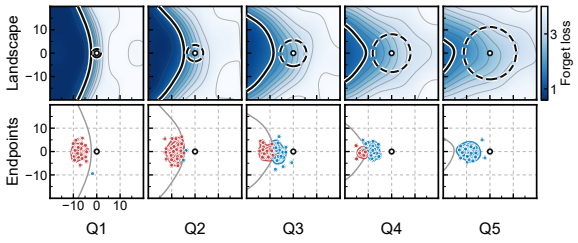
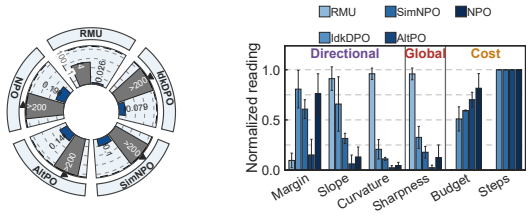


Figure 2: Forget loss on a plane through a release point, with the threshold contour and the certified radius.

Observation 2

In the accepted checkpoint cohort under the 200-step protocol, similar release-time scores coexist with different recovery behavior. Landing-point geometry distinguishes this cohort; validated neighborhood bounds support a budget for every trajectory satisfying the declared update constraints, whatever unlearning objective produced the release and however it scored.

Figure 3 examines accepted checkpoints under the earlier 200-step protocol, distinct from the protocol in Section 4. Panel (a) separates releases that the same release-time gate accepts: RMU (Li et al., 2024) recovers within a few attacker steps, whereas NPO (Zhang et al., 2024), SimNPO (Fan et al., 2025b), AltPO (Mekala et al., 2025) and IdkDPO (Maini et al., 2024) remain unrecovered at the end of the attack window. For the latter families the window supplies a lower bound of 200 steps on recovery cost.



(a) Recovery cost. (b) Landing-point geometry.

Figure 3: Recovery cost and landing-point geometry of accepted unlearning checkpoints on TOFU under the earlier attack protocol.

Panel (b) connects this gap to the released parameters. Release scores stay nearly level across families, while the geometric readings distinguish the fragile family from those surviving the window; whole-spectrum sharpness flags the same fragile tail in this comparison. The panels thus motivate separate questions: whether a checkpoint passes the erasure check, and how much subsequent movement that erasure can withstand. The theory answers the second through margin, directional slope and uniform negative-curvature control, turning the geometric distinction into a budget for all declared trajectories. CDT then uses those dependencies to guide training and re-evaluates the resulting landing point.

3 METHODOLOGY

3.1 FRAMEWORK OVERVIEW

PLATEAU turns landing-point geometry into a durability budget, then uses its dependencies to guide training (Figure 4). DME supplies the measures, BC converts valid bounds into a certificate, and CDT proposes a new release for re-evaluation. Algorithm 1 gives the complete protocol.

3.2 DIRECTIONAL MEASURE ESTIMATION (DME)

Motivation. DME reads the worst geometry over the declared loss family and displacement set. Restricting cumulative displacements to structured directions can give tighter bounds than measuring the full parameter space.

The measures. The margin is the room the forget loss holds above the threshold, minimized over the family so that every phrasing of the erased knowledge scores as forgotten at release:

$$\Delta \;=\; \min_{\mathcal{L}\in\mathcal{G}} \; \mathcal{L}(\theta_u) - \gamma, \tag{3}$$

Fix a neighborhood radius $\rho > 0$. The set $\mathcal{A}_s$ contains cumulative displacements whose layer blocks have rank at most s, including all adapter factors rather than one fixed subspace. A single unmerged rank-s adapter around a frozen release preserves this condition; arbitrary rank-s increments can accumulate rank Bs after B updates. Full-parameter attacks require full-space constants. Writing $W_\ell(\mathcal{L})$ for the release gradient on layer ℓ, the directional gradient norm has closed form:

$$g_s \;=\; \max_{\mathcal{L}\in\mathcal{G}} \; \sup_{\substack{v\in\mathcal{A}_s \\ \|v\|=1}} \; \sum_{\ell} \big\langle W_\ell(\mathcal{L}),\, v_\ell \big\rangle \;=\; \max_{\mathcal{L}\in\mathcal{G}} \Big(\sum_{\ell} \sum_{i=1}^{s} \sigma_i^2\big(W_\ell(\mathcal{L})\big) \Big)^{1/2}, \tag{4}$$

where $\langle \cdot, \cdot \rangle$ is the Frobenius inner product and σ_i the i-th singular value. One truncated SVD per layer computes this norm; Appendix F.2 derives the equality. Negative curvature must be bounded over the same directions throughout the neighborhood:

$$M_s(\rho) \;=\; \sup_{\substack{\mathcal{L}\in\mathcal{G},\; \theta\in\mathbb{B}_\rho(\theta_u) \\ v\in\mathcal{A}_s,\; \|v\|=1}} \big[-v^\top \nabla^2 \mathcal{L}(\theta)\, v\big]_+, \tag{5}$$

where $[a]_+$ is the positive part and $\mathbb{B}_\rho(\theta_u)$ the prescribed ball. A valid upper bound can replace this supremum. For example, a validated release-point bound and Hessian Lipschitz constant give Eq. (12); Appendix C.5.2 states the construction. Local Hessian probes alone yield estimates.

Worst objective. The three measures are taken over the declared family separately, so the smallest margin and the largest slope may come from different members: a paraphrase can sit closest to the threshold while a reformatted prompt offers the steepest accessible direction. Pairing them keeps Eq. (6) valid for every member at once, so one radius protects the whole family. A declared family is

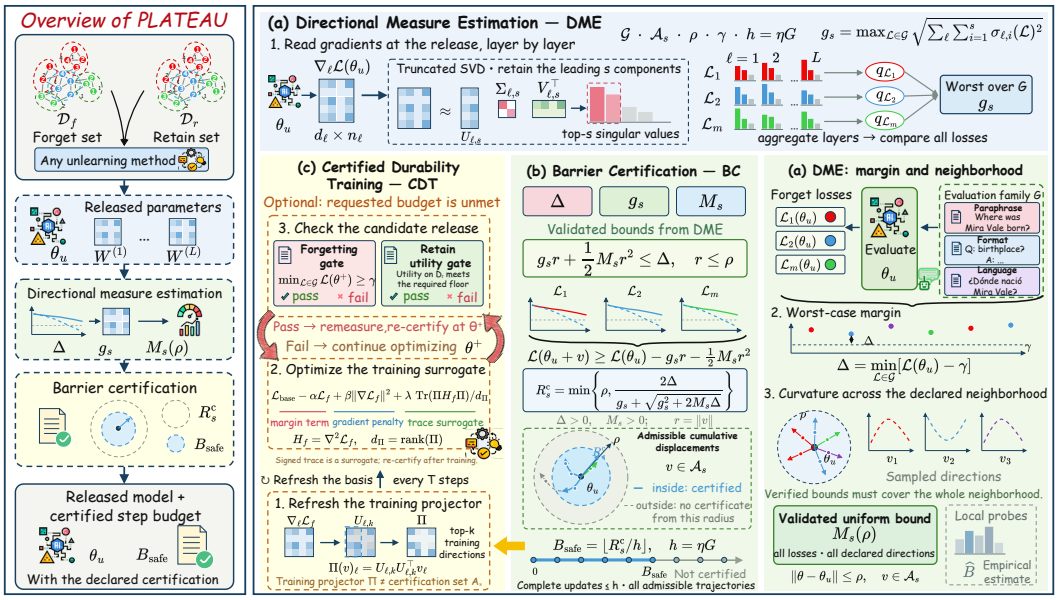


Figure 4: **Overview of PLATEAU.** DME reads the margin, the directional gradient norm and negative curvature at the release point over a prescribed neighborhood. BC converts valid bounds into a closed-form safety radius and a certified budget of complete updates. CDT feeds the radius dependencies back into an existing unlearning objective and re-evaluates the release.

finite, so each measure is a minimum or maximum over a fixed list, and the slope costs one truncated SVD per layer per member. Members that share a tokenization of the erased content usually agree on the ranking; the certificate needs the extremes rather than the ranking. Enlarging the family lowers the margin weakly and raises the slope weakly, which makes the declaration itself a stated part of the guarantee.

Optimizer families. The declared step bound h is ηG. For a comparison, h is fixed across methods within the same parameterization. A pooled pilot percentile may calibrate it once; every complete update is then clipped to this common bound, including preconditioning and weight decay. Unclipped optimizers require an independently valid bound.

3.3 Barrier Certification (BC)

Motivation. The input-dependent separation construction admits identical nonconstant release predictions with different recovery costs under a fixed parameterization for each pair. BC therefore certifies the geometry where fine-tuning acts.

The certified radius. The loss itself is a barrier. For every admissible cumulative displacement, its directional slope and negative curvature bound the possible decrease:

$$\mathcal{L}(\theta_u + v) \;\geq\; \mathcal{L}(\theta_u) - g_s \,\|v\| - \frac{M_s}{2}\,\|v\|^2, \tag{6}$$

for every $\mathcal{L} \in \mathcal{G}$ and every cumulative displacement $v \in \mathcal{A}_s$ with $\|v\| \leq \rho$, where the losses are C^2 on the prescribed ball, the first-order term is bounded through Eq. (4), the second-order term through Eq. (5), and M_s is short for $M_s(\rho)$. For $\Delta > 0$ and $M_s > 0$, the right-hand side stays above γ until the quadratic in $\|v\|$ spends the margin, and its positive root gives:

$$R_s^{\mathrm{c}} \;=\; \min\left\{\rho,\; \frac{2\Delta}{g_s + \sqrt{g_s^2 + 2M_s\Delta}}\right\}, \qquad B^{\mathrm{c}} \;=\; \frac{R_s^{\mathrm{c}}}{\eta G}. \tag{7}$$

Theorem 3 certifies $\lfloor B^{\mathrm{c}} \rfloor$ complete updates. Recovery means strict loss below γ, so the boundary remains safe. When M_s vanishes, the uncapped root is Δ/g_s when $g_s > 0$, and $+\infty$ otherwise. The radius is a lower bound on the largest safe radius. Fixed-subspace and full-space attacks use

the same formula with their corresponding bounds. At a stationary release the uncapped radius is $\sqrt{2\Delta/M_s}$; when slope dominates, it approaches Δ/g_s. Increasing margin or reducing directional slope and negative curvature enlarges this radius, until the prescribed neighborhood cap binds. These dependencies motivate CDT.

Tightness. The appendix gives a quadratic loss attaining the radius, so the bound is worst-case sharp. For an individual release, a recovery upper bound additionally requires a feasible descent ray; Eq. (33) states that condition and brackets the threshold distance. Recovery audits report the costs and displacements of tested attacks under the same declaration.

Anisotropy. A flat release can still have a small radius because of its slope, while large positive curvature orthogonal to recovery directions need not shrink the certificate. Accessible negative curvature must be included even when orthogonal to the release gradient. The appendix gives both constructions; whole-spectrum sharpness in Observation 2 serves as an empirical signal, while the certificate uses the uniform bound M_s.

3.4 Certified Durability Training (CDT)

Motivation. The uncapped radius is differentiable in its three scalar bounds away from degenerate cases, motivating a training surrogate. Objectives that stop at raising the forget loss leave the directional gradient norm free, and Eq. (7) prices that freedom as a potentially small radius.

Objective. CDT augments a base unlearning objective with the following surrogate terms:

$$\min_{\theta} \;\; \mathcal{L}_{\text{base}}(\theta) \;-\; \alpha\,\mathcal{L}_f(\theta) \;+\; \beta \sum_{\ell} \big\|\nabla_{\theta_\ell}\mathcal{L}_f(\theta)\big\|_F^2 \;+\; \lambda\,\frac{\operatorname{Tr}\big(\Pi\,\nabla^2\mathcal{L}_f(\theta)\,\Pi\big)}{d_\Pi}, \tag{8}$$

where $\mathcal{L}_{\text{base}}$ is the base procedure's objective, $\nabla_{\theta_\ell}\mathcal{L}_f$ the forget gradient of layer ℓ's weight matrix, Π the projector onto the training subspace refreshed every T steps, and d_Π its rank. The margin and gradient terms target the first two measures. The trace term shapes projected curvature; its effect on the uniform negative-curvature bound is assessed when the resulting release is evaluated again. The projector acts on each layer's block through the leading left singular vectors of that layer's forget gradient:

$$\Pi(v)_\ell \;=\; U_{\ell,k}U_{\ell,k}^\top v_\ell, \qquad U_{\ell,k} \;=\; \big[u_{\ell,1},\ldots,u_{\ell,k_\ell}\big], \qquad d_\Pi \;=\; \sum_{\ell} k_\ell\, n_\ell, \tag{9}$$

where $u_{\ell,i}$ is the i-th left singular vector of $\nabla_{\theta_\ell}\mathcal{L}_f(\theta)$ in decreasing order of singular value and n_ℓ is the block's column count. This training subspace differs from the union of all low-rank displacements. The gradient penalty controls a bound on g_s for $\mathcal{L}_f$; certification re-evaluates the margin, slope and curvature over every member of $\mathcal{G}$. The signed trace is a training surrogate, since $\operatorname{diag}(-K, K)$ has zero trace and negative-curvature magnitude K; each CDT candidate therefore receives a fresh computation of the uniform bound M_s and its certificate.

Estimators. CDT uses central differences for the gradient penalty and normalized isotropic probes in the training subspace for the signed trace. The projector and probes remain fixed during each update. Appendix C.5.1 gives the estimators and their expanded derivatives.

Cost. Each surrogate term adds a bounded amount of work to an update. The gradient penalty is differentiated through the central difference of Eq. (10), so it costs two extra gradient evaluations and no explicit second derivative. The signed trace spends two scalar-loss evaluations per probe, and the probes stay fixed within an update, so its cost follows the probe count rather than the layer count. Refreshing the projector takes one truncated SVD of each layer's forget gradient every T steps, which amortizes over the interval. The training rank k and the probe count m trade estimator variance against per-step cost; both are training choices, separate from the certification rank s that the declaration fixes. Memory holds one projector block per layer beside the base objective's state.

Certification workflow. A release is certified when the supplied bounds cover its declared neighborhood, displacement set and evaluation family. DME supplies those bounds, and BC issues the radius and budget together with $(\rho, \mathcal{A}, \mathcal{G}, \gamma, h)$. Local probes yield the budget estimate $\widehat{B}$, reported beside the certified budget, and each trained release is re-evaluated against the forgetting and utility gate.

Table 1: Comparison of one release and five attack records per method and benchmark, at matched FQ/MU. Rose and blue mark the largest and second-largest estimates; the retain reference is excluded. SV counts recoveries at or before CB (Appendix C.4.1).

Method	TOFU						MUSE						WMDP-bio					
	FQ↑	MU↑	CB↑	RS↑	RR↓	SV↓	FQ↑	MU↑	CB↑	RS↑	RR↓	SV↓	FQ↑	MU↑	CB↑	RS↑	RR↓	SV↓
(a) Unlearning objectives																		
GradDiff (Maini et al., 2024)	0.720	1.003	2519	5262	0.098	0	0.648	0.966	2704	4675	0.096	0	0.728	1.005	2246	4147	0.112	0
NPO (Zhang et al., 2024)	0.718	0.991	1887	3329	0.117	0	0.651	0.988	**4086**	8723	0.049	0	0.742	0.970	1951	4913	0.138	0
SimNPO (Fan et al., 2025b)	0.717	0.994	1777	3781	0.125	0	0.660	0.990	1434	2941	0.156	0	0.738	1.014	2161	3274	0.118	0
AltPO (Mekala et al., 2025)	0.730	1.012	2386	3697	0.097	0	0.654	1.002	1717	6366	0.124	0	0.746	0.992	1310	2124	0.140	0
IdkDPO (Maini et al., 2024)	0.726	1.020	2103	3899	0.118	0	0.655	1.017	2686	4100	0.102	0	0.751	0.980	2651	4880	0.108	0
IdkNLL (Maini et al., 2024)	0.710	1.016	2039	3360	0.118	0	0.651	0.978	2468	4259	0.112	0	0.739	0.998	2128	3792	0.102	0
RMU (Li et al., 2024)	0.726	0.996	1278	1921	0.168	0	0.643	1.010	1700	2654	0.131	0	0.727	0.967	1064	1761	0.178	0
UNDIAL (Dong et al., 2025)	0.728	1.013	1464	2956	0.152	0	0.655	1.011	2431	4404	0.119	0	0.731	1.010	2410	3640	0.108	0
(b) Durability defenses																		
Vaccine [NeurIPS'24] (Huang et al., 2024)	0.710	1.020	2158	3554	0.126	0	0.647	1.004	3727	7003	0.054	0	0.730	0.963	2329	3566	0.100	0
RepNoise [NeurIPS'24] (Rosati et al., 2024)	0.707	0.994	2060	4170	0.108	0	0.652	1.011	2545	4364	0.090	0	0.740	0.991	2750	3963	0.096	0
Booster [ICLR'25] (Huang et al., 2025)	0.726	1.014	2425	4403	0.136	0	0.658	0.981	2605	3462	0.093	0	0.745	1.018	3095	4987	0.079	0
TAR [ICLR'25] (Tamirisa et al., 2025)	0.726	1.006	**3913**	6922	0.062	0	0.646	0.994	3084	5853	0.095	0	0.735	0.999	**4200**	6766	0.050	0
Unlearn-Smooth [ICML'25] (Fan et al., 2025a)	0.711	0.978	2868	4384	0.082	0	0.660	0.991	3633	6872	0.062	0	0.751	1.016	3082	5880	0.081	0
ILU [ICML'25] (Wang et al., 2025)	0.728	1.013	2691	4534	0.123	0	0.640	1.022	3675	6512	0.073	0	0.744	0.975	**4415**	5534	0.052	0
PRISM [ICLR'26] (Yan et al., 2026)	0.718	0.964	2980	5582	0.081	0	0.637	1.008	3771	6709	0.068	0	0.732	0.977	2709	5681	0.101	0
SSIUU [ICLR'26] (Yang et al., 2026)	0.727	1.000	3417	5263	0.071	0	0.652	0.972	4066	5614	0.054	0	0.728	1.008	1837	3497	0.134	0
PLATEAU	0.724	0.983	**4379**	9749	0.045	0	0.645	0.983	**5275**	9401	0.049	0	0.744	1.014	4011	5050	0.050	0
Retain reference	1.000	1.000	2268	4137	0.110	0	1.000	1.000	2665	4655	0.082	0	1.000	1.000	1956	3172	0.130	0

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Benchmarks and comparison. The protocol covers TOFU (Maini et al., 2024), MUSE (Shi et al., 2025) and WMDP-bio (Li et al., 2024). Each release has five attack seeds. The FQ bands are $[0.70, 0.74]$, $[0.63, 0.67]$ and $[0.72, 0.76]$, respectively, and every compared release satisfies $0.90 \leq \mathrm{MU} \leq 1.10$. The retain reference fixes the utility normalization and is excluded from ranking. The separately recorded supplementary diagnostics retain their own protocol.

Common declaration. The protocol fixes $\rho = 0.20$, $s = 16$, $\eta = 10^{-7}$ and $h = 2 \times 10^{-5}$, hence $G = 200$, for every method. Thresholds are $\gamma = 0.68, 1.15, 0.90$ on the three benchmarks. The protected family $\mathcal{G}_4$ contains the original, paraphrased, reformatted and translated evaluation losses. Parameterization, loss normalization and declaration are fixed before method rankings are read; the percentile calibration of h is pooled once and frozen across compared releases. Complete AdamW displacements, including weight decay, are clipped to the common bound. Crossings are resolved at every step up to 12,000; CDT candidates are compared within the same FQ and MU bands.

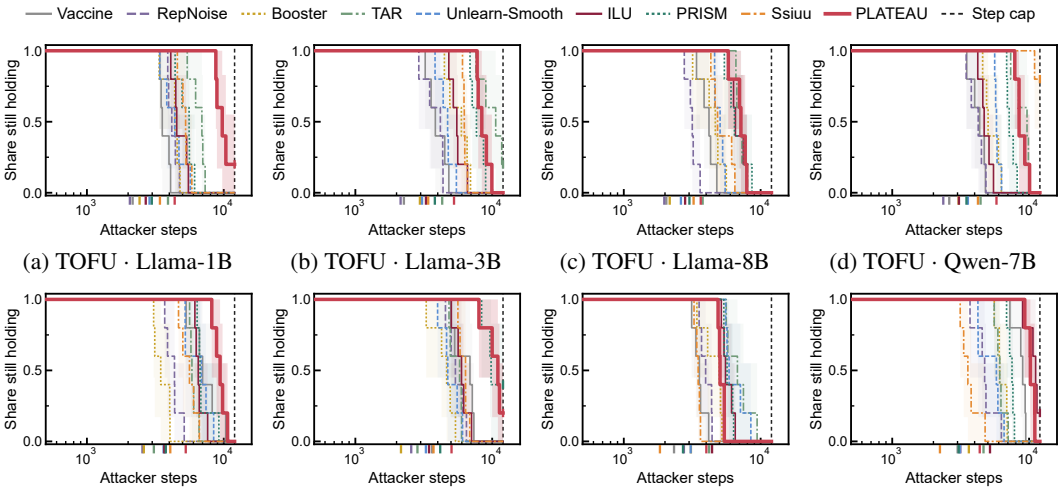
Table 2: **Five-seed audit.** Each row retains the smallest capped-ratio cell; correlations use all eight cells. V and Cens. describe the selected cell, not the pooled runs.

Attack class	Estimate	Recovery		Verdict			
	CB ↑	RS ↑	RS/CB	V ↓	Cens. ↓	ρ_c ↑	ρ_s
(a) Inside the displacement and step declaration							
Base protocol	4011	5050	1.26	0	0.00	+0.29	−0.41
Optimizer family	4360	5253	1.20	0	0.00	+0.69	−0.28
Learning rate	3650	5157	1.41	0	0.00	+0.62	−0.81
Data variant	5261	7352	1.40	0	0.00	+0.93	−0.71
Attack-set size	5779	7978	1.38	0	0.00	+0.95	−0.50
Certificate-aware	3650	4018	1.10	0	0.00	+0.79	−0.90
Worst admissible cell	3650	4018	1.10	0	0.00	+0.79	−0.90
(b) Outside the declared class							
Unbounded step size	5044	3695	0.73	5	0.00	+0.71	−0.09
Full-rank update	4360	3242	0.74	5	0.00	+0.83	−0.83

Metrics. FQ is forget quality and MU is utility divided by the retain reference. CB is the capped integer estimate $\lfloor \widehat{R}_s/h \rfloor$, computed from the reported margin, slope and curvature proxy. RS is the Kaplan–Meier median first threshold crossing across five seeds. RR is the mean fraction of 200 records revived at step 1,000; SV and V count crossings within CB. Cens. is the censored share. The audit reports Spearman correlations of capped RS with CB and benchmark-centered FQ. Appendix E.1 pairs the local estimate with the certified radius and budget for the same releases.

4.2 DURABILITY AT MATCHED RELEASE QUALITY

Table 1 compares releases at matched FQ and MU under the common declaration, and Figure 5 reads the same attack-seed records as survival curves. Releases that one release-time gate accepts separate once the budget and the recovery cost are read side by side. **PLATEAU** attains the largest budget on TOFU and MUSE (4,379 and 5,275 complete updates, against 3,913 and 4,086 for the strongest counterpart) and the longest recovery median on both (9,749 and 9,401 steps), together with the



(a) TOFU · Llama-1B (b) TOFU · Llama-3B (c) TOFU · Llama-8B (d) TOFU · Qwen-7B

(e) MUSE · Llama-7B (f) MUSE · ICLM-7B (g) WMDP · Zephyr-7B (h) WMDP · Llama-8B

Figure 5: Recovery survival curves from five attack seeds per release.

lowest revival rate on all three benchmarks. On WMDP-bio its budget sits in the top tier with ILU and TAR at the lowest revival rate. The advantage comes from the geometry CDT targets: the margin and slope terms move the two quantities that enter the radius, and the certificate converts them into budget.

Each release carries two readings: RS/CB reports the slack against that release's own estimate, while cross-method ordering follows CB and RS separately, so the two columns carry independent information about a release.

CDT directly targets the quantities entering CB. Held-out recovery therefore supplies the independent outcome: model selection and declarations are fixed first, then RS and first-crossing displacement are compared across independently trained, quality-matched releases (Appendix E.3). A verified budget bounds admissible updates; recovery times then measure the slack beyond it.

4.3 Budget Calibration and Attack Scope

Table 2 audits eight deployment cells (four TOFU, two MUSE, two WMDP-bio), each with five attack seeds. Each row retains the cell with the smallest capped RS/CB. Every attack class inside the declaration keeps its worst cell above the budget, with RS/CB between 1.10 and 1.41 and zero within-budget recoveries, while the two classes outside the declaration recover within 0.73–0.74 of the budget in all five seeds. The declared class is thus exactly the class the certificate protects. Optimizer, data and objective changes remain in scope when cumulative displacement and complete updates satisfy the declaration. Unbounded steps and unrestricted rank are separate checks. Figure 6 pairs budgets and recovery medians.

Table 3: Retain-only continuation on TOFU/Llama-1B, seed 0: 200 records and 80 AdamW updates at 10^{-6} per row.

	Revival			Retain NLL	
Method	Before (%)	After (%)	Gain (pp)	Before	After
GradDiff	6.5	12.0	5.5	0.656	0.284
NPO	0.0	0.5	0.5	0.649	0.340
SimNPO	5.5	7.5	2.0	0.713	0.504
AltPO	5.0	14.0	9.0	0.831	0.370
IdkDPO	0.5	6.0	5.5	0.567	0.279
IdkNLL	9.0	23.0	14.0	0.673	0.321
RMU	4.5	10.5	6.0	1.095	0.781
UNDIAL	2.5	2.5	0.0	0.974	0.428
Vaccine	5.0	19.5	14.5	0.754	0.352
RepNoise	4.0	7.0	3.0	0.729	0.509
Booster	1.5	10.0	8.5	0.873	0.360
TAR	0.0	0.5	0.5	0.705	0.352
Unlearn-Smooth	1.0	5.0	4.0	0.997	0.487
ILU	7.0	10.5	3.5	0.680	0.475
PRISM	0.5	0.5	0.0	0.541	0.308
SSIUU	8.5	9.0	0.5	0.496	0.327
PLATEAU	4.5	15.5	11.0	1.071	0.420
Retain reference	4.0	14.5	10.5	0.610	0.370

Table 2 reports rank correlations of capped RS with the budget and with benchmark-centered FQ across the eight cells: the budget orders recovery positively in every class (ρ_c from $+0.29$ to $+0.95$), whereas release quality does not, which is the separation of Observation 2 at deployment

scale. Operational certification combines validated neighborhood bounds with enforced displacement and complete-update constraints.

Benign continuation and cost. Table 3 covers 18 releases on TOFU/Llama-1B. After 80 unconstrained retain updates, **PLATEAU** tracks the retain reference (11.0 versus 10.5 pp revival gain) while retained-task fit improves, so benign continuation behaves as it does for a model trained without the forget set. These runs characterize benign continuation with one release per method; the component effect is isolated in Section 4.4. Appendix D.3 gives the trajectories. Training cost is measured over all 36 settings, four ranks $k \in \{4, 8, 16, 32\}$ with nine probe counts each, under 16 CDT updates and a 32-step attack: runtime spans 5.90–18.85 s across the probe counts, while the max-minus-min spans of the recovery NLL reduction (1.55–2.60 $\times 10^{-5}$), the margin (1.03–2.33 $\times 10^{-5}$) and the retain NLL (2.44–4.08 $\times 10^{-4}$) stay small at this horizon (protocol: Appendix D.6).

4.4 Contribution of the Training Terms

Table 4: Component comparison with five attack records per row at matched FQ/MU. Full CDT and base NPO reuse Table 1. Rose and blue mark the best and second-best geometry.

Component			Measures			Outcome				
MAR	GRD	TRC	$\Delta \uparrow$	$g_s \downarrow$	$M_s \downarrow$	CB $\uparrow$	RS $\uparrow$	RR $\downarrow$	FQ $\uparrow$	MU $\uparrow$
✓	✓	✓	1.382	7.63	186.0	**4379**	9749	0.045	0.724	0.983
✗	✓	✓	1.007	**7.33**	**183.5**	**3608**	5906	0.063	0.725	0.995
✓	✗	✓	1.385	18.67	187.6	2877	5077	0.089	0.730	1.001
✓	✓	✗	**1.397**	**7.60**	427.7	3249	5074	0.076	0.726	0.979
✓	✗	✗	**1.391**	18.75	429.8	2394	4355	0.108	0.730	1.009
✗	✓	✗	0.982	7.61	427.4	2614	3885	0.105	0.716	1.001
✗	✗	✓	0.990	18.61	**183.0**	2188	4757	0.124	0.706	1.005
✗	✗	✗	1.006	18.51	431.2	1887	3329	0.117	0.718	0.991

Table 4 compares components at matched FQ and MU. Its complete configuration and unregularized NPO row reuse the same geometry and five attack seeds as Table 1. Each single deletion lowers the budget by 18–34% and the recovery median by 39–48%, and the base objective reaches 43% of the full budget, so every term contributes and the three together account for the gain.

Each switch removes the term acting on one input of the radius: MAR on the margin, GRD on the directional slope, TRC on the curvature response. The rows therefore read as three single-term deletions against the complete configuration and against the unregularized base objective, with the budget and the recovery median moving together within a row and the two geometric readings moving only where the corresponding term was removed. Removing GRD raises g_s from 7.63 to 18.67 while Δ and M_s stay in place, removing TRC raises M_s from 186.0 to 427.7 with g_s unchanged, and removing MAR lowers Δ from 1.382 to 1.007. The slope term carries the largest single effect on the budget, 2,877 against 4,379, which matches Observation 1: when slope dominates, the uncapped radius approaches Δ/g_s, so the directional gradient norm is the quantity a release can least afford to leave free. Figure 7 sweeps the regularization weight between the same component endpoints, reporting the geometric estimate and the recovery cost together.

The ablation fixes the source checkpoint, training allowance, evaluation family and release-quality bands. All three geometric quantities are re-evaluated through Eq. (25). The trace term acts in its training subspace; certification uses a uniform bound over the declared set. Five attack seeds characterize one release, while the independent training replicates in Appendix E.3 assess variability across releases. Section 4.2 tests whether proxy improvements transfer to recovery resistance.

5 Conclusion

Identical release-time behavior can coexist with different costs of recovering forgotten knowledge. **PLATEAU** certifies this durability at the landing point: DME supplies geometric measures, BC converts valid neighborhood bounds into a closed-form radius and complete-update budget, and CDT uses the radius dependencies to guide training followed by re-evaluation. The guarantee covers every trajectory in the declared class and applies independently of the unlearning procedure. A common declaration makes the budget interpretable for controlled fine-tuning. Across TOFU, MUSE and WMDP-bio at matched forgetting and utility, the budget orders recovery, every compared release stays unrecovered within its budget, and **PLATEAU** attains the largest budgets and longest recoveries on TOFU and MUSE. Durability thereby becomes a property of a neighborhood around the released weights, expressed in the adversary's update budget.

AI USE STATEMENT

In this work, we used generative AI tools to help propose and refine our hypotheses, to assist in writing the proofs, to design the experiments and provide feedback on the research methodology, to clean and reformat the datasets, and to help interpret the results. We did not use generative AI tools to generate synthetic data sets, to develop theoretical models or conceptual frameworks, to formulate mathematical claims, to supply critical ingredients of the proofs, to implement the methods, or to translate the manuscript, and qualitative and thematic data analysis does not apply to this work. Additionally, we used generative AI tools to identify candidate research topics, to plot the figures that report our experimental results, to find and summarize related literature, to draft the notation table in the appendix, to improve the readability of the text, and to format the references. We have reviewed all AI-assisted work. In particular, the authors verified each AI-assisted proof line by line under the stated assumptions, compared the values in every AI-plotted figure with the experiment logs, surveyed the literature to check the chosen topic for overlap with prior work, and checked each reference against its publisher page, CrossRef or DBLP. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Lucas Bourtoule, Varun Chandrasekaran, Christopher A. Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning. In *42nd IEEE Symposium on Security and Privacy, SP 2021, San Francisco, CA, USA, 24-27 May 2021*, pp. 141–159. IEEE, 2021.

Jeremy Cohen, Elan Rosenfeld, and J. Zico Kolter. Certified adversarial robustness via randomized smoothing. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1310–1320. PMLR, 2019. URL http://proceedings.mlr.press/v97/cohen19c.html.

Yijiang River Dong, Hongzhou Lin, Mikhail Belkin, Ramon Huerta, and Ivan Vulić. UNDIAL: self-distillation with adjusted logits for robust unlearning in large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 8827–8840. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.naacl-long.444.

Vineeth Dorna, Anmol Mekala, Wenlong Zhao, Andrew McCallum, Zico Kolter, Zachary Lipton, and Pratyush Maini. Openunlearning: Accelerating LLM unlearning via unified benchmarking of methods and metrics. In *Advances in Neural Information Processing Systems 38, Datasets and Benchmarks Track*, pp. 48903–48940, 2025. doi: 10.52202/085713-1458.

Chongyu Fan, Jinghan Jia, Yihua Zhang, Anil Ramakrishna, Mingyi Hong, and Sijia Liu. Towards LLM unlearning resilient to relearning attacks: A sharpness-aware minimization perspective and beyond. In *Proceedings of the 42nd International Conference on Machine Learning, ICML 2025*, volume 267 of *Proceedings of Machine Learning Research*, pp. 15762–15778. PMLR, 2025a.

Chongyu Fan, Jiancheng Liu, Licong Lin, Jinghan Jia, Ruiqi Zhang, Song Mei, and Sijia Liu. Simplicity prevails: Rethinking negative preference optimization for LLM unlearning. In *Advances in Neural Information Processing Systems 38*, pp. 1776–1803, 2025b. doi: 10.52202/085713-0058.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *The Ninth International Conference on Learning Representations, ICLR 2021*. OpenReview.net, 2021.

Chuan Guo, Tom Goldstein, Awni Hannun, and Laurens van der Maaten. Certified data removal from machine learning models, 2023. URL https://arxiv.org/abs/1911.03030.

Shengyuan Hu, Yiwei Fu, Zhiwei Steven Wu, and Virginia Smith. Unlearning or obfuscating? jogging the memory of unlearned llms via benign relearning. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025.

Tiansheng Huang, Sihao Hu, and Ling Liu. Vaccine: Perturbation-aware alignment for large language models against harmful fine-tuning. In *Advances in Neural Information Processing Systems 37*, pp. 74058–74088, 2024. doi: 10.52202/079017-2356.

Tiansheng Huang, Sihao Hu, Fatih Ilhan, Selim Furkan Tekin, and Ling Liu. Booster: Tackling harmful fine-tuning for large language models via attenuating harmful perturbation. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D. Li, Ann-Kathrin Dombrowski, Shashwat Goel, Gabriel Mukobi, Nathan Helm-Burger, Rassin Lababidi, Lennart Justen, Andrew B. Liu, Michael Chen, Isabelle Barrass, Oliver Zhang, Xiaoyuan Zhu, Rishub Tamirisa, Bhrugu Bharathi, Ariel Herbert-Voss, Cort B. Breuer, Andy Zou, Mantas Mazeika, Zifan Wang, Palash Oswal, Weiran Lin, Adam A. Hunt, Justin Tienken-Harder, Kevin Y. Shih, Kemper Talley, John Guan, Ian Steneker, David Campbell, Brad Jokubaitis, Steven Basart, Stephen Fitz, Ponnurangam Kumaraguru, Kallol Krishna Karmakar, Uday Kiran Tupakula, Vijay Varadharajan, Yan Shoshitaishvili, Jimmy Ba, Kevin M. Esvelt, Alexandr Wang, and Dan Hendrycks. The WMDP benchmark: Measuring and reducing malicious use with unlearning. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 28525–28550. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/li24bc.html.

Aengus Lynch, Phillip Guo, Aidan Ewart, Stephen Casper, and Dylan Hadfield-Menell. Eight methods to evaluate robust unlearning in llms, 2024. URL https://arxiv.org/abs/2402.16835.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C. Lipton, and J. Zico Kolter. Tofu: A task of fictitious unlearning for llms, 2024. URL https://arxiv.org/abs/2401.06121.

Anmol Reddy Mekala, Vineeth Dorna, Shreya Dubey, Abhishek Lalwani, David Koleczek, Mukund Rungta, Sadid A. Hasan, and Elita A. Lobo. Alternate preference optimization for unlearning factual knowledge in large language models. In Owen Rambow, Leo Wanner, Marianna Apidianaki, Hend Al-Khalifa, Barbara Di Eugenio, and Steven Schockaert (eds.), *Proceedings of the 31st International Conference on Computational Linguistics, COLING 2025, Abu Dhabi, UAE, January 19-24, 2025*, pp. 3732–3752. Association for Computational Linguistics, 2025. URL https://aclanthology.org/2025.coling-main.252/.

Seth Neel, Aaron Roth, and Saeed Sharifi-Malvajerdi. Descent-to-delete: Gradient-based methods for machine unlearning. In *Proceedings of the 32nd International Conference on Algorithmic Learning Theory, ALT 2021*, volume 132 of *Proceedings of Machine Learning Research*, pp. 931–962. PMLR, 2021.

Domenic Rosati, Jan Wehner, Kai Williams, Łukasz Bartoszcze, David Atanasov, Robie Gonzales, Subhabrata Majumdar, Carsten Maple, Hassan Sajjad, and Frank Rudzicz. Representation noising: A defence mechanism against harmful finetuning. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/172be8b0b88fc2b4aee74237d43f8c04-Abstract-Conference.html.

Ayush Sekhari, Jayadev Acharya, Gautam Kamath, and Ananda Theertha Suresh. Remember what you want to forget: Algorithms for machine unlearning. In *Advances in Neural Information Processing Systems 34*, pp. 18075–18086. Curran Associates, Inc., 2021.

Weijia Shi, Jaechan Lee, Yangsibo Huang, Sadhika Malladi, Jieyu Zhao, Ari Holtzman, Daogao Liu, Luke Zettlemoyer, Noah A. Smith, and Chiyuan Zhang. MUSE: machine unlearning six-way evaluation for language models. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=TArmA033BU.

Rishub Tamirisa, Bhrugu Bharathi, Long Phan, Andy Zhou, Alice Gatti, Tarun Suresh, Maxwell Lin, Justin Wang, Rowan Wang, Ron Arel, Andy Zou, Dawn Song, Bo Li, Dan Hendrycks, and Mantas Mazeika. Tamper-resistant safeguards for open-weight llms. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025.

Changsheng Wang, Yihua Zhang, Jinghan Jia, Parikshit Ram, Dennis Wei, Yuguang Yao, Soumyadeep Pal, Nathalie Baracaldo, and Sijia Liu. Invariance makes LLM unlearning resilient even to unanticipated downstream fine-tuning. In *Proceedings of the 42nd International Conference on Machine Learning, ICML 2025*, volume 267 of *Proceedings of Machine Learning Research*, pp. 65464–65479. PMLR, 2025.

Han Yan, Zheyuan Liu, and Meng Jiang. Dual-space smoothness for robust and balanced LLM unlearning. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=VIMW3eys6x.

Nakyeong Yang, Dong-Kyum Kim, Jea Kwon, Minsung Kim, Kyomin Jung, and Meeyoung Cha. Erase or hide? suppressing spurious unlearning neurons for robust unlearning. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=z2zFk9jYpw.

Yuanshun Yao, Xiaojun Xu, and Yang Liu. Large language model unlearning. In *Advances in Neural Information Processing Systems 37*, pp. 105425–105475. Curran Associates, Inc., 2024. doi: 10.52202/079017-3346.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative preference optimization: From catastrophic collapse to effective unlearning. *CoRR*, abs/2404.05868, 2024. doi: 10.48550/ARXIV.2404.05868. URL https://doi.org/10.48550/arXiv.2404.05868.

Jakub Łucki, Boyi Wei, Yangsibo Huang, Peter Henderson, Florian Tramèr, and Javier Rando. An adversarial perspective on machine unlearning for ai safety, 2025. URL https://arxiv.org/abs/2409.18025.

APPENDIX CONTENTS

A Related Work **14**

B The Audit Protocol **14**

C Technical Appendix **15**

C.1 Notation Table 15

C.2 Dataset Details 15

C.3 Baseline Descriptions 16

C.4 Evaluation Metrics 17

C.5 Implementation Details 18

C.6 Algorithm 19

C.7 Broader Impacts 19

C.8 Extended Experimental Results 19

D Additional Experiments on Durable Unlearning **19**

D.1 Recovery under Fixed-Basis Low-Rank Updates 21

D.2 Curvature across the Release Neighborhood 21

D.3 Continued Fine-Tuning on Retained Data 22

D.4 Switched and Restarted Recovery Optimizers 22

D.5 The Distribution of Record-Level Recovery 22

D.6 Additional Protocol Details 23

E Certificate Comparison and Reproducibility **23**

F Proofs of Theoretical Results **26**

F.1 Assumptions and Scope 26

F.2 Proof of Lemma 1 (Directional Geometry) 26

F.3 Proof of Theorem 3 (Certified Update Budget) 27

F.4 Derivation of the Cumulative-Displacement Conditions 29

F.5 Derivation of Tightness and Recovery Bounds 29

F.6 Proof of Theorem 4 (Durability Separation) 30

F.7 Proof of Theorem 5 (Directional Curvature) 31

F.8 Derivation of CDT Gradients and Probe Estimators 32

A Related Work

Large Language Model Unlearning. LLM unlearning edits a trained checkpoint so that designated knowledge is suppressed while retained capabilities survive. Gradient-based objectives raise the loss on the forget set while anchoring the retained behavior, exemplified by gradient ascent with retain regularization (Yao et al., 2024); preference-based objectives bound that ascent to avoid collapse, exemplified by NPO, SimNPO and AltPO (Zhang et al., 2024; Fan et al., 2025b; Mekala et al., 2025); distillation- and representation-based objectives reshape internal responses to the forget set, exemplified by UNDIAL and RMU (Dong et al., 2025; Li et al., 2024; Yang et al., 2026); and benchmarks with unified tooling standardize the forget and retain gates these objectives are scored against, exemplified by TOFU, MUSE and WMDP (Maini et al., 2024; Shi et al., 2025; Li et al., 2024; Dorna et al., 2025). However, these methods and benchmarks judge a release by its outputs at the moment of release: none reads the parameter-space geometry of the landing point, and none states how many subsequent updates the suppression survives. **PLATEAU** takes any of these releases as input, reads its margin, directional slope and negative curvature, and turns them into a certified update budget.

Robust Unlearning and Tamper Resistance. A second line hardens the release against later fine-tuning. Smoothness-based unlearning flattens the loss around the release, exemplified by Unlearn-Smooth, PRISM and sharpness-aware minimization (Fan et al., 2025a; Yan et al., 2026; Foret et al., 2021); invariance-based unlearning regularizes updates so that relearning data cannot re-expose the erased content, exemplified by ILU (Wang et al., 2025); and harmful-fine-tuning defenses train against perturbations or adversarial trajectories, exemplified by Vaccine, RepNoise, Booster and TAR (Huang et al., 2024; Rosati et al., 2024; Huang et al., 2025; Tamirisa et al., 2025). However, these defenses report resistance against the attacks they were run against, and their guarantees end at the last attack tested: none issues a bound over every admissible trajectory, and none states the declaration under which its resistance holds. **PLATEAU** complements them with a closed-form radius that covers every trajectory within the declared displacement set and update bound, and CDT feeds the radius dependencies into any of these objectives.

Relearning Attacks and Durability Evaluation. Attack-side studies show that suppressed knowledge returns under small parameter changes. Benign relearning recovers erased content by fine-tuning on data outside the forget set, exemplified by memory jogging (Hu et al., 2025); adversarial evaluations recover it through few-step fine-tuning, quantization and evaluation choices, exemplified by the adversarial perspective and the eight-method protocol (Łucki et al., 2025; Lynch et al., 2024). However, these studies measure durability empirically, one attack at a time: the number reported depends on the attack, the optimizer and the data chosen, so two releases compared under different attacks carry no common scale. **PLATEAU** places the attack inside a declaration of the perturbation set, loss family and complete-update bound, so that the certified budget and the measured recovery cost read on one scale, and the audit in Section 4 reports both.

Certified Deletion and Geometric Guarantees. Certified deletion compares the edited model with one trained without the deleted records. Exact and sharded schemes retrain the affected part of the data, exemplified by SISA (Bourtoule et al., 2021); approximate schemes bound the statistical distance to retraining and the number of deletions one model can absorb, exemplified by certified removal, deletion capacity and descent-to-delete (Guo et al., 2023; Sekhari et al., 2021; Neel et al., 2021); and geometric certificates issue a radius around a point from a local reading, exemplified by randomized smoothing (Cohen et al., 2019). However, deletion certificates hold at the moment of release and say nothing about the updates that follow, while input-space radii certify a prediction around an input, and no existing certificate places a radius in parameter space around a released checkpoint. **PLATEAU** issues such a radius from the release margin, the directional gradient norm and uniform negative-curvature control over a prescribed neighborhood, and the separation construction shows why the radius is needed: identical release outputs can accompany exponentially different recovery costs.

B The Audit Protocol

What the certificate speaks about. The certificate names the released parameters, their parameterization, the protected loss family $\mathcal{G}$, the threshold γ, the cumulative-displacement set $\mathcal{A}$, the

neighborhood radius ρ and the enforced complete-update bound h. Its claim is that every trajectory satisfying this declaration preserves every protected loss above the threshold for B_{safe} complete updates. The neighborhood and displacement declaration determine which derivative bounds are required. A budget computed from local probes is recorded as $\widehat{B}$, separately from a certificate supported by valid bounds.

Stage one: fixing the release and declaration. The audit records the model checkpoint and tokenizer, the forget and retain evaluation splits, the construction of each protected loss and the threshold used for recovery. It also fixes the trainable blocks and whether the displacement set is full-space, a fixed subspace or blockwise rank-restricted. The update norm concerns the released weights after the complete optimizer step. Keeping these choices fixed prevents a change of loss normalization, parameterization or attack set from silently changing the meaning of the budget. A modified candidate produced by CDT is a new release and receives a fresh declaration and evaluation.

Stage two: recomputing the certificate. The auditor recomputes the worst release margin and directional gradient norm over $\mathcal{G}$ and verifies the supplied negative-curvature bound over the prescribed ball. A release-point curvature reading can be extended through a separately validated Hessian Lipschitz bound, as in Eq. (12). The auditor then applies Eq. (7), including the neighborhood cap, and takes the floor of radius divided by h. The record includes the origin and scope of each bound. Sampling a few Hessian directions does not establish the required supremum over the whole neighborhood or over all admissible directions.

Stage three: checking update admissibility. For each replayed attack, the audit records the complete displacement and checks its norm after preconditioning and weight decay. It checks cumulative displacements against $\mathcal{A}$ as well: separately low-rank increments can accumulate a higher rank. A run outside this declaration is an out-of-class experiment and cannot falsify a guarantee issued for a smaller set. Within the declaration, the deterministic barrier applies to adaptive choices of training data and objective, so certification does not depend on reproducing a particular attacker.

Empirical audit and recovery records. The five-seed audit reports the first observed threshold crossing, the censoring indicator and whether recovery occurs within the local estimate. Table 2 keeps each selected cell's budget, recovery time and violation count together. Its correlations use the complete deployment-cell collection. These measurements assess the informativeness of local estimates and the attacks that were run. The certified statement instead follows from the validated geometric bounds and admissibility checks above; a zero empirical count by itself does not replace either condition.

C Technical Appendix

C.1 Notation Table

We summarize the commonly used notation throughout this paper in Table 5.

C.2 Dataset Details

The three benchmarks exercise different deletion targets while sharing the landing-point audit of Section 4.1.

TOFU Maini et al. (2024). The forget records describe fictitious author profiles. The main setting uses the ten-percent forget split on Llama-3.2-1B-Instruct, with smaller forget splits probing request size. The model-scale axis uses Llama-3.2-3B and Llama-3.1-8B, and the model-family axis uses Qwen2.5. Each recovery run trains on a slice of the forget records and evaluates the threshold on held-out records. The retain-trained model supplies the reference for release-time quality and utility.

MUSE Shi et al. (2025). News and book corpora provide a corpus-removal setting with their associated forgetting and utility evaluations. The main comparison uses the benchmark's 7B target models and its native unlearning recipes. The held-out recovery loss is evaluated using the same tokenization and length normalization as at release, so a threshold comparison is unaffected by a change in answer length convention.

Table 5: Notation and Definitions.

Notation	Definition
$\mathcal{D}_f$, $\mathcal{D}_r$	The forget and retain datasets.
θ, θ_u	The model parameters and released landing point.
$\theta[t]$, $v[t]$	The parameters at step t and next complete optimizer update.
$\mathcal{L}_f$, $\mathcal{L}_r$	The length-normalized forget and retain losses.
$\mathcal{G}$	The declared family of protected evaluation losses.
γ	The fixed recovery threshold: loss strictly below γ.
$\mathcal{A}$, $\mathcal{A}_s$	The declared displacement set and its blockwise rank-s version.
s, ℓ	The certification rank bound and layer index.
ρ, $\mathbb{B}_\rho(\theta_u)$	The validated neighborhood radius and its parameter-space ball.
Δ	The minimum release margin above γ over $\mathcal{G}$.
$W_\ell(\mathcal{L})$	The layer-ℓ gradient of loss $\mathcal{L}$ at release.
$\sigma_i(W)$	The i-th singular value of W, sorted in decreasing order.
g_s	The worst directional gradient norm over $\mathcal{G}$.
$M_s(\rho)$	The uniform negative-curvature bound over $\mathbb{B}_\rho(\theta_u)$ and $\mathcal{A}_s$.
$\bar{m}_s$, K	The release negative-curvature bound; Hessian Lipschitz constant.
R_s^{c}	The certified radius, capped by ρ.
η, G, h	The learning rate, scale and complete-update bound $h = \eta G$.
B^{c}, B_{safe}	The radius-to-step ratio R_s^{c}/h and its floor.
$\widehat{B}$	The budget estimate from unverified local geometry.
$\mathcal{L}_{\mathrm{base}}$	The base unlearning objective augmented by CDT.
α, β, λ	The CDT weights of margin, gradient penalty and signed trace.
Π, d_Π	The training-subspace projector and its rank.
k_ℓ, $U_{\ell,k}$	The layer-ℓ training rank and retained left singular vectors.
n_ℓ, T	The layer column count and projector refresh period.
w, u	The forget gradient and its unit direction when nonzero.
z_j, m, ϵ	The isotropic unit probes, probe count and central-difference step.
FQ, MU	The forget quality and reference-normalized retain utility.
CB, RS, RR	The estimated budget, recovery steps and record revival rate.
SV, V, Cens.	The within-estimate recovery counts and censoring fraction.
ρ_{c}, ρ_{s}	The Spearman correlations of capped RS with CB and FQ.

WMDP-bio Li et al. (2024). The biological-knowledge benchmark supplies the hazardous-capability deletion target. Zephyr-7B is the main model; the deployment panels additionally include the model variant named in Figure 5. Retained utility is measured at the same release as the forget and recovery metrics. The benchmark score and the protected likelihood objective are recorded separately because they summarize different aspects of the released model.

Checkpoint population. The TOFU OpenUnlearning grid contains 398 checkpoints from eight objective families. Applying the stated release gate admits 29 checkpoints from five families. Family-level rows aggregate admitted checkpoints; defense rows and **`PLATEAU`** each identify one released checkpoint. Five attack seeds vary the attack run around that fixed release. The earlier 200-step observation study is a separate protocol and does not supply the attack budget of the main comparison.

C.3 Baseline Descriptions

The comparison separates the procedures producing releases from defenses designed to preserve them under further training. Every row is evaluated through the same protected loss, utility reading and recovery protocol.

Unlearning objectives. The checkpoint population contains GradDiff, NPO, SimNPO, AltPO, IdkDPO, IdkNLL, RMU and UNDIAL Dorna et al. (2025). Gradient-based and preference-based rows change the forget–retain optimization objective, while representation-oriented rows alter the internal response to forget inputs. These rows test whether the geometry distinguishes releases produced by different procedures. The family median is a population summary, not an additional trained model.

Unlearning-specific durability defenses.

- **Unlearn-Smooth** Fan et al. (2025a) uses sharpness-aware and related smoothing objectives to improve resistance to relearning. It provides a parameter-space regularization counterpart to CDT.
- **PRISM** Yan et al. (2026) combines representation-space and parameter-space smoothing to address attacks and the balance between forgetting and retention. Its two spaces provide a comparison with the directional quantities read at the landing point.
- **ILU and SSIUU** Wang et al. (2025); Yang et al. (2026) enter as the additional unlearning-durability counterparts listed in Table 1. Each comparison uses the corresponding release and its associated forget and retain data; the audit applies the same threshold and attack protocol used for the other defenses.

Defenses against harmful fine-tuning.

- **Vaccine** Huang et al. (2024) trains with crafted embedding perturbations to reduce sensitivity to harmful fine-tuning.
- **RepNoise** Rosati et al. (2024) uses representation noising as a defense against harmful fine-tuning.
- **Booster** Huang et al. (2025) penalizes the decrease in harmful loss induced by weight perturbations.
- **TAR** Tamirisa et al. (2025) trains against tampering trajectories to improve the persistence of safeguards after release.

The harmful dataset in these defenses is replaced by the declared forget target for the shared comparison. The retain-trained reference provides the release-time baseline and is excluded from budget ranking.

C.4 Evaluation Metrics

Tables 1, 2 and 4 use five attack seeds per release or deployment cell. Five attack seeds do not imply five independently trained releases.

Forget quality and retained utility. FQ is the declared benchmark score; MU divides retained utility by the retain-reference value. The comparison matches TOFU, MUSE and WMDP-bio FQ in $[0.70, 0.74]$, $[0.63, 0.67]$ and $[0.72, 0.76]$, with MU in $[0.90, 1.10]$. Every compared release satisfies its band. The normalization, split and band are shared across methods before releases are selected.

Budget and recovery time. CB is $\lfloor \widehat{R}_s/h \rfloor$, with the same neighborhood cap as Eq. (7), but using a curvature proxy. RS is the Kaplan–Meier median first strict threshold crossing across five seeds. All censoring is administrative at 12,000 steps; if fewer than three seeds recover, the median is reported as greater than the cap.

Revival and violations. RR averages the fraction of 200 held-out records reaching half their pre-unlearning answer probability at step 1,000. SV in Table 1 and V in Table 2 count observed recoveries at or before CB, hence are integers from zero to five. Certification requires validated bounds and admissible updates.

Audit aggregation. Table 2 selects the cell with the smallest capped RS/CB in each attack class. CB, RS, V and the censored fraction refer to that cell; the summary repeats the worst-ratio row within the declaration. The Spearman coefficients ρ_c and ρ_s compare capped RS with CB and benchmark-centered FQ across all eight deployment cells. Each displayed ratio describes one operating point, while correlations describe the deployment collection.

C.4.1 Zero-Violation Columns and Censoring

Across Tables 1, 2 and 4, every observed main-table and ablation recovery time exceeds $1.03\,\mathrm{CB}$, and every in-class audit time exceeds $1.02\,\mathrm{CB}$; all 51 compared method/benchmark SV entries are therefore zero. The component rows of Table 4 move each measure in the direction predicted by the radius dependencies.

All selected cells in Table 2 are uncensored, while other deployment cells have censored shares from 0.20 to 1.00. The two out-of-class rows each have $\mathrm{V} = 5$ in their selected worst-ratio cell; across

all eight cells, V ranges from 1 to 5 for unbounded steps and from 0 to 5 for unrestricted rank. The summary row repeats the worst-ratio cell within the declaration.

C.5 Implementation Details

Recovery declaration. The main protocol uses $\rho = 0.20$, rank $s = 16$, $\eta = 10^{-7}$ and the complete-update bound $h = 2 \times 10^{-5}$, so $G = 200$. Thresholds are 0.68, 1.15 and 0.90 on TOFU, MUSE and WMDP-bio. Each protected family contains original, paraphrased, reformatted and translated losses. The same declaration is shared by every method within a benchmark, with per-step checking and a 12,000-step cap. The percentile calibration of h is pooled on a fixed pilot set, frozen before comparison and enforced by clipping complete updates. The separately recorded supplementary runs retain their stated setup.

Directional measures. For each protected objective, DME computes the layer gradients, takes their top-s singular values and aggregates their squared values across layers. The minimum loss margin and maximum directional quantities use the same evaluation family. A local Hessian proxy yields an estimate; certification requires a bound valid over the prescribed ball and cumulative-displacement set. The audit protocol in Appendix B records that distinction with the released checkpoint.

CDT and re-evaluation. The training projector is refreshed every T steps from the selected left singular directions of the layerwise forget gradients. Its dimension k is a training choice distinct from certification rank s. Regularizer weights are selected at the matched-utility band. After optimization, the candidate release is checked again for forgetting and retention, then DME and BC are recomputed using its declared evaluation family.

C.5.1 CDT Estimators

Estimators. Write w for the forget gradient $\nabla \mathcal{L}_f(\theta)$ and u for its direction $w/\|w\|$. The chain rule gives the penalty gradient exactly, and, holding u fixed, a central difference approximates it:

$$\nabla_\theta \|w\|^2 \;=\; 2\,\nabla^2 \mathcal{L}_f(\theta)\, w \;\approx\; \|w\|\, \frac{\nabla \mathcal{L}_f(\theta + \epsilon u) - \nabla \mathcal{L}_f(\theta - \epsilon u)}{\epsilon}, \tag{10}$$

where ϵ is the probe step of the central difference, so the Hessian-vector product costs two gradient evaluations and no second derivative. At a vanishing w the derivative is zero. The trace term is estimated the same way, with unit-normalized isotropic probes in the range of Π and a second-order central difference of the scalar loss:

$$\frac{\operatorname{Tr}\big(\Pi\, \nabla^2 \mathcal{L}_f(\theta)\, \Pi\big)}{d_\Pi} \;\approx\; \frac{1}{m} \sum_{j=1}^{m} \frac{\mathcal{L}_f(\theta + \epsilon z_j) - 2\,\mathcal{L}_f(\theta) + \mathcal{L}_f(\theta - \epsilon z_j)}{\epsilon^2}, \tag{11}$$

where $z_1, \ldots, z_m$ are m independent probes drawn uniformly on the unit sphere of the range of Π, whose second moment Π/d_Π explains the normalization in Eq. (8). The implementation differentiates scalar-loss central differences with the projector and probes frozen per update; an HVP alone does not give the trace's training gradient. These approximations require higher smoothness, and their cost depends on the probe count.

C.5.2 Validated Neighborhood Bound

where $[a]_+$ denotes the positive part $\max\{a, 0\}$ and $\mathbb{B}_\rho(\theta_u)$ the ball of radius ρ around the release, so the quantity is the negative second directional derivative of the loss along v. A valid upper bound may replace this supremum and must cover the prescribed ball. One such bound combines a release-point reading with a Hessian Lipschitz constant:

$$M_s(\rho) \;\le\; \bar{m}_s + K\rho, \qquad \bar{m}_s \;=\; \sup_{\substack{\mathcal{L} \in \mathcal{G},\ v \in \mathcal{A}_s \\ \|v\| = 1}} \big[-v^\top \nabla^2 \mathcal{L}(\theta_u)\, v\big]_+, \tag{12}$$

where $\bar{m}_s$ reads the same quantity at the release point alone and K is the Hessian Lipschitz constant on the prescribed ball in operator norm, verified rather than estimated. A few Hessian-vector products in gradient directions yield local estimates, which Section 4 distinguishes from certified bounds.

C.6 ALGORITHM

Algorithm 1 PLATEAU: Certified Durable Unlearning at the Landing Point

Input: released parameters θ_u; forget set $\mathcal{D}_f$; recovery family $\mathcal{G}$; threshold γ; neighborhood radius ρ; cumulative displacement rank s; optimizer-family constants (η, G); requested budget B; subspace size k; refresh period T; regularizer weights (α, β, λ).

Assemble the recovery family $\mathcal{G}$ from $\mathcal{D}_f$: paraphrase, format and language variants
▷ *Phase I: Directional Measure Estimation (DME)*
Margin Δ from the forget-set losses of θ_u over $\mathcal{G}$ ▷ Eq. (3)
Truncated SVD of each layer's forget gradient; directional gradient norm g_s ▷ Eq. (4)
Uniform negative-curvature bound $M_s(\rho)$ on the declared ball and rank-s set ▷ Eq. (5)
▷ *Phase II: Barrier Certification (BC)*
Certified radius R_s^{c} from the measures (Δ, g_s, M_s) ▷ Eq. (7)
Recovery budget B^{c} by pricing the radius at the declared step bound ηG
if $\lfloor B^{\mathrm{c}} \rfloor$ falls short of the requested budget B **then**
    Name the binding measure and continue to Phase III
end if
Issue the certificate $(\Delta, g_s, M_s, R_s^{\mathrm{c}}, B^{\mathrm{c}})$
▷ *Phase III: Certified Durability Training (CDT)*
repeat
    Update θ on the durability objective at weights (α, β, λ) ▷ Eq. (8)
    Re-estimate the recovery subspace Π every T steps ▷ Eq. (9)
until the requested budget is certified or the utility floor is reached
Re-run Phases I and II at the new landing point
return certificate $(\Delta, g_s, M_s, R_s^{\mathrm{c}}, B^{\mathrm{c}})$; released parameters θ_u when Phase III runs

C.7 BROADER IMPACTS

PLATEAU lowers the cost of showing that a deletion request stays honored once the released model continues to be trained, supporting compliance with data-protection and copyright obligations. We release no new dataset or pretrained model, so direct deployment risk is limited. Two indirect concerns remain: a budget is issued for a declared optimizer family, displacement set and recovery family, and a deployer who reads it outside that declaration may over-trust it, which auditing practice can mitigate by recording the declaration alongside the budget; and a durability objective raises a quantity a release is also scored on, so each trained release is re-evaluated against the benchmark's own forgetting and utility gate before it is issued.

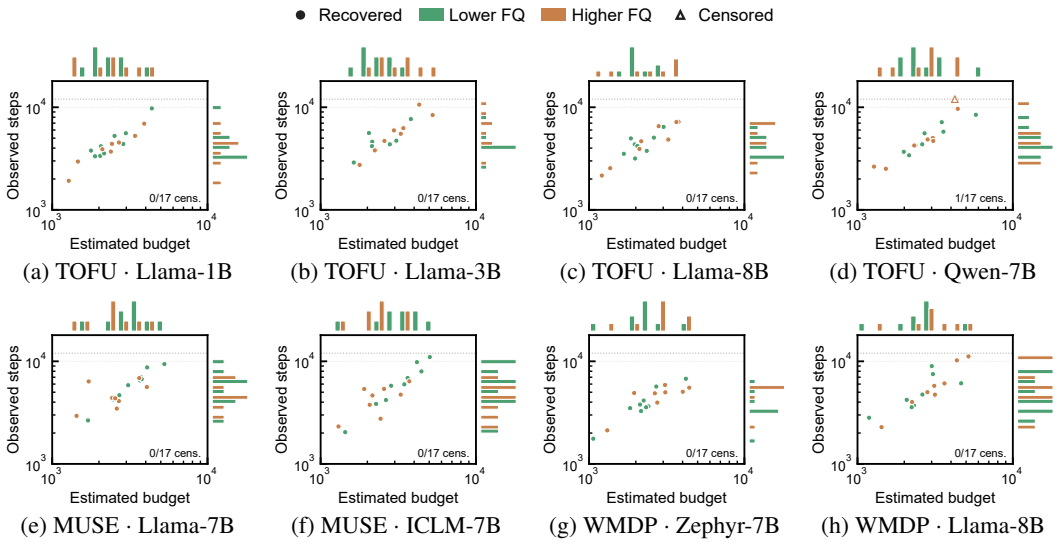
C.8 EXTENDED EXPERIMENTAL RESULTS

Margin at the budget across attackers. Figure 8a covers five attacker configurations and eight deployment cells. For each seed, it evaluates $m(t) = \Delta[1 - (t/T)^2]$ at CB from the capped crossing time T. Each cell shows the mean and standard deviation across the five seeds. A $\geq$ marks a lower bound on the mean when a crossing time is censored; the spread uses capped inputs.

Recovery relative to the estimate. Figure 8b summarizes the same crossing times divided by CB, with rows and columns in the shared audit ordering. Each cell gives a mean and sample standard deviation over five seeds, with $\geq$ marking a lower bound on an uncapped mean and the spread describing capped values. The two panels share their crossing times and read as two views of one record set. Table 2 uses a recovery median instead of these means; V counts individual crossings within CB.

D ADDITIONAL EXPERIMENTS ON DURABLE UNLEARNING

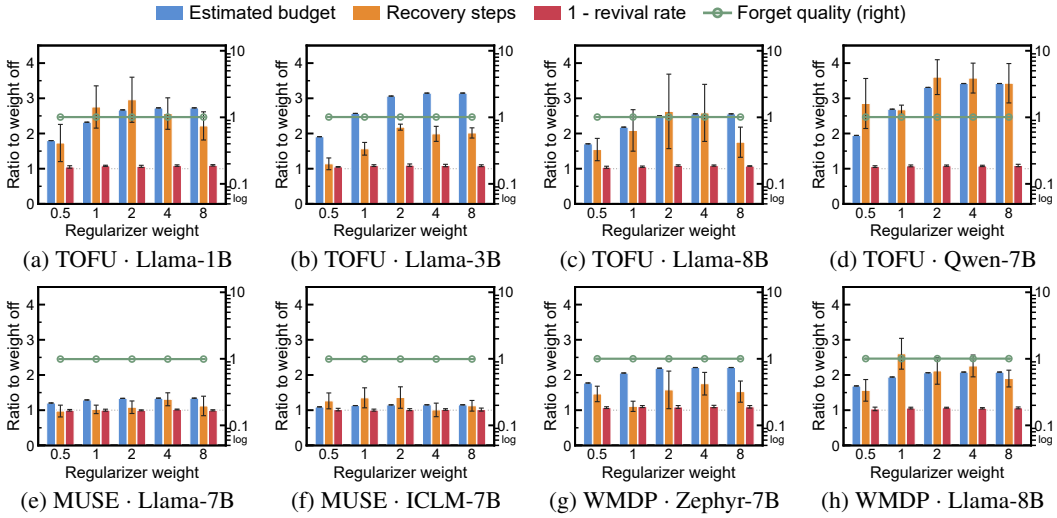
These experiments test how the geometry shaped by PLATEAU governs recovery under controlled updates and changes during continued training. We reuse the TOFU Llama-3.2-1B releases of PLATEAU, its NPO starting checkpoint, and PRISM, with seed 0 for each additional configuration.



(a) TOFU · Llama-1B (b) TOFU · Llama-3B (c) TOFU · Llama-8B (d) TOFU · Qwen-7B

(e) MUSE · Llama-7B (f) MUSE · ICLM-7B (g) WMDP · Zephyr-7B (h) WMDP · Llama-8B

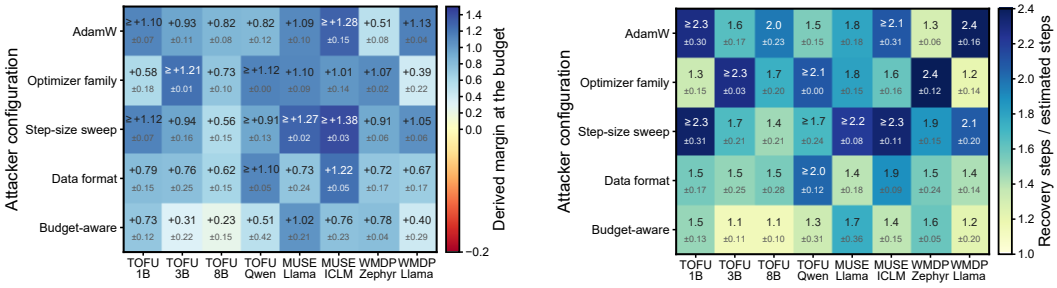
Figure 6: Budget estimates and recovery medians under the same declaration.



(a) TOFU · Llama-1B (b) TOFU · Llama-3B (c) TOFU · Llama-8B (d) TOFU · Qwen-7B

(e) MUSE · Llama-7B (f) MUSE · ICLM-7B (g) WMDP · Zephyr-7B (h) WMDP · Llama-8B

Figure 7: Regularization sweep, normalized to weight off; recovery steps are capped.



(a) Margin at the budget. (b) Recovery-to-estimate ratio.

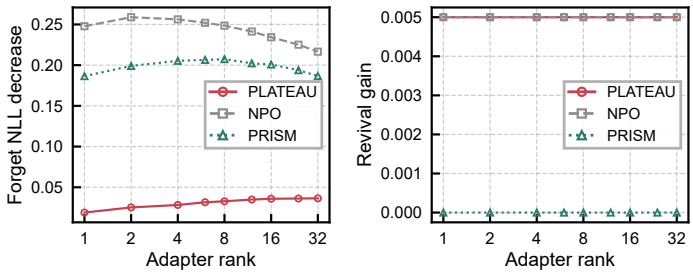
Figure 8: Attacker audit heatmaps; mean and SD across five seeds.

Within-release changes account for their different initial forgetting and utility levels. Record-level recovery uses the same held-out forget records and pre-unlearning reference throughout.

Attack training and evaluation use disjoint pools of two hundred records each. Every attack update obeys the same weight-space norm bound. Revival denotes the fraction of records whose length-normalized answer probability reaches half of its pre-unlearning value. Revival gain subtracts the fraction already above that threshold at the relevant starting state. Each sweep contains nine measured settings or observation points; the distribution summaries are computed from the recorded per-item probabilities.

D.1 Recovery under Fixed-Basis Low-Rank Updates

Figure 9 tests whether the release resists increasingly expressive low-rank updates. The attacker modifies query and value projections through a fixed left basis constructed from its own training data. Orthonormality makes coefficient displacement equal to induced weight displacement, allowing the same step bound to apply at every rank. Each adapter starts from zero and receives the same recovery budget.



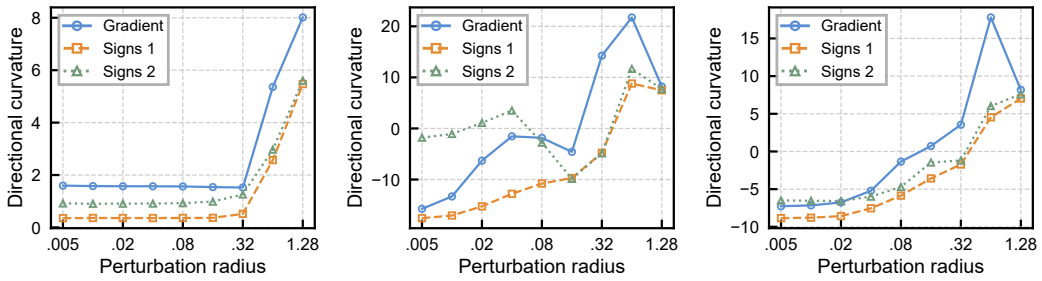
(a) Loss reduction. (b) Revival gain.

Figure 9: Recovery under low-rank updates with a common weight-space step bound. Each attack runs for the same number of updates.

PLATEAU permits substantially less forget-loss reduction across this rank range. Its response levels off as the basis widens, connecting resistance to the geometry shaped at release. The NPO and PRISM trajectories show that capacity and the update constraint act together: widening the basis does not produce monotonically stronger recovery at a fixed budget. Loss reduction also resolves progress below the record-level revival threshold. The result supports resistance within the tested fixed bases, with the basis choice forming an additional degree of freedom in the full rank class.

D.2 Curvature across the Release Neighborhood

Figure 10 examines the spatial scale over which a local directional reading remains representative. Central differences evaluate the same three directions at increasing perturbation radii: the attack-side query/value gradient and two fixed layer-sign mixtures. The directions are constructed on attack records, while separate held-out records evaluate curvature.



(a) PLATEAU. (b) NPO. (c) PRISM.

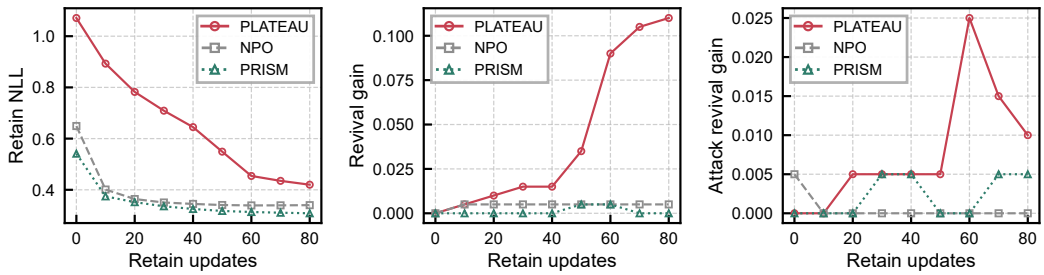
Figure 10: Directional curvature at increasing distances from the release. Each curve follows one fixed direction.

The PLATEAU readings in Figure 10 remain comparatively stable over the small-radius range and change as the neighborhood expands. Together with the sign changes in the other releases, this behavior makes the role of distance and direction explicit. It complements Figure 9: the geometry that resists nearby recovery must be assessed over the region an update sequence can reach. The

plotted directions provide local diagnostics; a certified neighborhood uses bounds covering its full declared displacement set.

D.3 Continued Fine-Tuning on Retained Data

Figure 11 follows each release through further training on retained records. At nine checkpoints, we evaluate retain loss and revival, then apply the same short recovery attack. The attack state is restored before benign training resumes, so the two update sequences remain separate. The middle panel measures revival relative to the initial release; the right panel measures the additional revival produced by the attack from the current fine-tuned state.



(a) Retained-task fit. (b) Fine-tuning revival. (c) Additional attack revival.

Figure 11: Retain-only fine-tuning changes the release. Revival from continued training and additional revival from a fixed recovery attack are measured separately.

Figure 11 pairs improving retained-task fit with changes in forgotten-answer probabilities. PLATEAU exhibits greater revival growth along this retain-only trajectory, while the short recovery attack produces a smaller additional change. Retain training updates shared parameters without explicitly preserving the forgetting margin. Its objective therefore allows forgotten-answer probabilities to change even when the forget records are absent from training. The separation between the two revival curves locates most of the observed change along the continued-training trajectory.

A certificate is attached to a particular release, displacement neighborhood, and update budget. Subsequent training must be accounted for from that release, or the resulting model re-evaluated as a new state. The certified average-loss condition and the fraction of records crossing a probability threshold also measure different properties. Figure 11 therefore motivates tracking both during further training and refreshing the release assessment when the model changes.

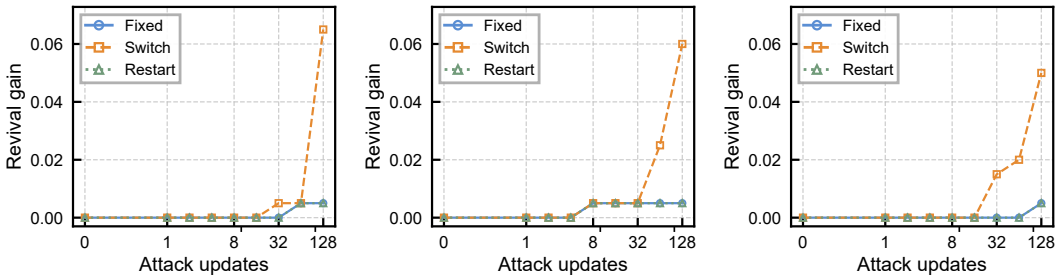
D.4 Switched and Restarted Recovery Optimizers

Figure 12 compares fixed AdamW, a prespecified AdamW/SGD alternation, and AdamW with periodic optimizer-state resets. All updates count toward the same total budget and obey the same weight-space step bound. The alternating schedule changes direction without changing that bound; a restart clears optimizer history while retaining the current model parameters. The plots use the same observation grid and subtract revival already present at release.

The alternating trajectory in Figure 12 produces more recovery across the tested releases, while resetting AdamW state closely tracks the fixed optimizer. Optimizer choice thus affects how efficiently an attacker uses the allowed displacement. This result supports measuring durability across multiple feasible trajectories. These attacks update the full two-dimensional weights and use the corresponding full-space displacement class.

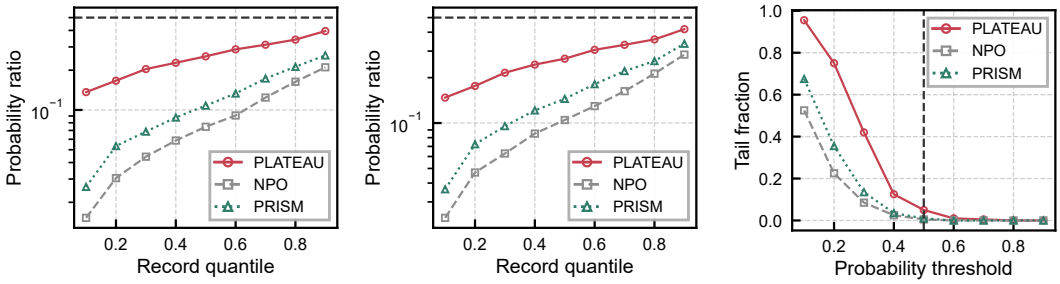
D.5 The Distribution of Record-Level Recovery

Figure 13 resolves recovery across individual records. The first two panels compare empirical quantiles of answer probability relative to the pre-unlearning model, before and after the same fixed-budget attack. The third panel reports the fraction of records above each probability threshold, exposing the upper tail directly. All three panels use the same recorded examples and reference probabilities.



(a) PLATEAU. (b) NPO. (c) PRISM.

Figure 12: Recovery depends on the optimizer trajectory. Revival gain is measured from each release under a common update budget.



(a) At release. (b) After recovery updates. (c) Recoverable tail.

Figure 13: Record-level recovery separates the released state from subsequent change. Dashed lines mark the half-origin probability threshold.

The distributions in Figure 13 show that the aggregate revival rate summarizes one threshold within a broader spectrum of retained answer probabilities. Reading each release beside its attacked state distinguishes residual knowledge from recovery induced by the update sequence. The tail plot makes this distinction visible for records closest to the recovery criterion, complementing average forget loss and the within-release gains above. Initial distributions differ across these releases, so this diagnostic preserves their starting states when interpreting the attack response.

D.6 Additional Protocol Details

The rank sweep uses $1, 2, 4, 6, 8, 12, 16, 24, 32$ and runs each fixed-basis attack for 64 steps. Query/value bases are constructed from 16 attack-side records. The continued-training sweep uses $0, 10, \ldots, 80$ retain updates and a 32-step attack at each checkpoint; retained-data evaluation and training use disjoint pools of 100 and 200 records. Switching and restart attacks are observed at $0, 1, 2, 4, 8, 16, 32, 64, 128$ steps. Optimizer alternation occurs every 16 steps, and state resets occur before steps 33, 65, and 97.

Curvature uses 32 held-out audit records and radii $0.005, 0.01, 0.02, 0.04, 0.08, 0.16, 0.32, 0.64, 1.28$. Distribution plots report quantiles and probability thresholds from 0.1 to 0.9. Cost measurements run on an NVIDIA H100 with 80GB memory. The CDT step benchmark uses batch size 1 and a rank-16 subspace prepared from 16 records. All model computations use float32, with the same tokenization and answer-only evaluation as the recorded releases.

E Certificate Comparison and Reproducibility

Scope. This section complements Tables 1, 2 and 4 and Figures 5–8b with the certified radii and budgets of the same releases, their sensitivity to the declaration and variation across independently trained releases. Appendix D reports the continuation and cost diagnostics.

E.1 Estimated and Certified Budgets

Table 6 uses the same TOFU releases, quality scores, and five attack-seed records as Table 1. The certificate comparison pairs the local estimate $\widehat{B}$ with the validated radius, the integer complete-update budget, and recovery under the same declaration. An asterisk marks the certificate slots: R^* uses the validated margin lower bound, slope upper bound, and neighborhood-curvature upper bound; $B^* = \lfloor R^*/h \rfloor$. They instantiate R_s^{c} and $\lfloor B^{\mathrm{c}} \rfloor$ for each release.

Common declaration. Within each benchmark the protocol fixes $\rho = 0.20$, cumulative rank $s = 16$, $h = 2 \times 10^{-5}$, and the original/paraphrase/format/language loss family. The thresholds are 0.68, 1.15, and 0.90 for TOFU, MUSE, and WMDP-bio. FQ/MU reuse Table 1.

Recovery distance. Each run logs $d_t = \|\theta_t - \theta_u\|$, the complete-update norms, path length, and the released-weight norm in the same parameterization. Comparing R_s^{c} with d_{rec} separates protected distance from the step-size convention; both are also reported divided by $\|\theta_u\|$ for relative scale. The weight norm is measured on the released checkpoint.

Table 6: **TOFU: certified budget and recovery comparison.** R^*, B^* use validated bounds. RS is the five-seed median; d_{rec} is median recovery displacement. Rose/blue identify the largest/second-largest B^*, excluding the retain reference.

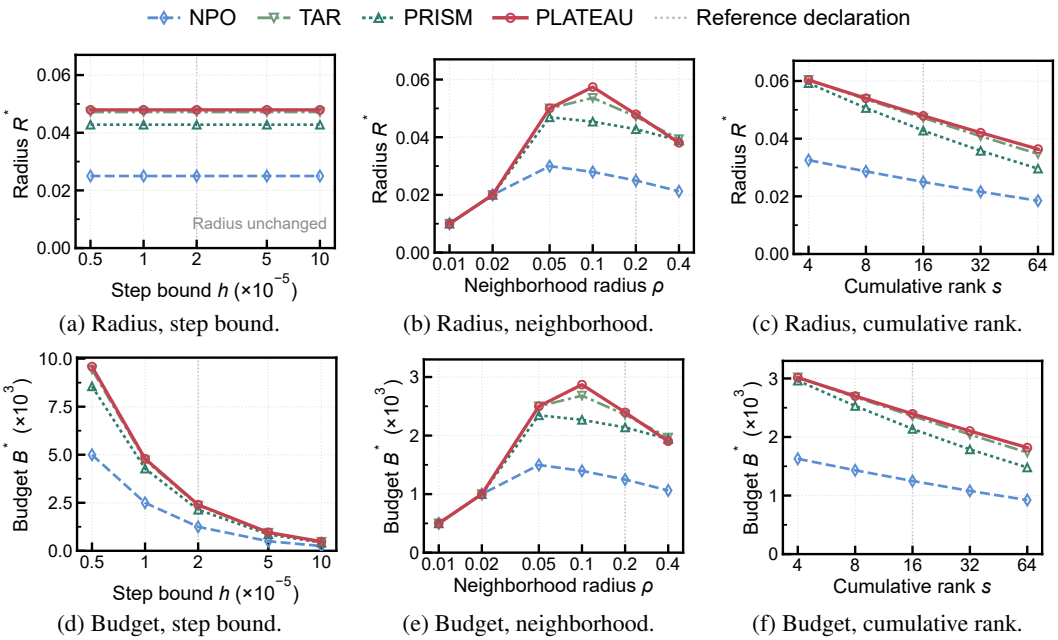
	Local estimate	Certificate slots (validated bounds)			Recovery slots (five seeds)			
Method	$\widehat{B}$	$\overline{M}_s^*$	R^*	$B^* \uparrow$	RS	d_{rec}	V/5	Cens./5
Unlearning objectives								
GradDiff	2519	966.9	0.03535	1767	5262	0.0608	0/5	0/5
NPO	1887	1513.3	0.02499	1249	3329	0.0448	0/5	0/5
SimNPO	1777	3059.4	0.02013	1006	3781	0.0582	0/5	0/5
AltPO	2386	890.4	0.03595	1797	3697	0.0471	0/5	0/5
IdkDPO	2103	2048.4	0.02341	1170	3899	0.0518	0/5	0/5
IdkNLL	2039	1223.4	0.02621	1310	3360	0.0503	0/5	0/5
RMU	1278	1367.3	0.01737	868	1921	0.0295	0/5	0/5
UNDIAL	1464	1707.7	0.02018	1009	2956	0.0276	0/5	0/5
Durability defenses								
Vaccine	2158	1608.3	0.02967	1483	3554	0.0504	0/5	0/5
RepNoise	2060	2632.8	0.02616	1307	4170	0.0404	0/5	0/5
Booster	2425	783.6	0.03259	1629	4403	0.0619	0/5	0/5
TAR	3913	652.2	0.04718	2358	6922	0.0936	0/5	0/5
Unlearn-Smooth	2868	788.6	0.04025	2012	4384	0.0638	0/5	0/5
ILU	2691	1163.4	0.03538	1769	4534	0.0626	0/5	0/5
PRISM	2980	350.3	0.04280	2139	5582	0.0659	0/5	0/5
SSIUU	3417	359.2	0.04848	**2424**	5263	0.0853	0/5	0/5
PLATEAU and retain reference								
PLATEAU	4379	802.7	0.04793	**2396**	9749	0.1492	0/5	1/5
Retain reference	2268	1523.3	0.02819	1409	4137	0.0566	0/5	0/5

E.2 Sensitivity to the Deployment Declaration

Figure 14 separates three changes on the same TOFU release. Changing h leaves the radius fixed and changes only the number of complete updates. Changing ρ changes both the neighborhood cap and the curvature envelope; enlarging the admissible rank changes the directional envelopes. Bounds are recomputed at every declaration, each declaration is fixed before the recovery run, and attack cost and displacement are reported under that declaration. Persistent adapters are maintained relative to the frozen release, with cumulative ranks checked after complete updates.

E.3 Variation across Independently Trained Releases

Table 7 uses five independently trained releases per method and five attacks per release. This separates variation in landing-point geometry from attack randomness around a fixed release. Each reported deviation is across the five release-level values; RS first takes the median across five attacks within a release. The 60 releases and 300 attacks use a 30,000-step horizon.



(a) Radius, step bound. (b) Radius, neighborhood. (c) Radius, cumulative rank.

(d) Budget, step bound. (e) Budget, neighborhood. (f) Budget, cumulative rank.

Figure 14: Declaration sensitivity of the certified radius and budget on the same TOFU release. Dotted lines mark the reference declaration.

Table 7: **Independent-release variation.** Mean and sample standard deviation across five releases, with five attacks per release. Rose/blue rank mean B^* within each benchmark.

		Release quality		Budget slots		Recovery
Benchmark	**Method**	FQ	MU	$\widehat{B}$	B^*	RS
TOFU	NPO	0.719(±0.018)	1.006(±0.035)	1893(±268)	1299(±181)	2363(±378)
	TAR	0.720(±0.013)	1.015(±0.014)	4107(±894)	**2432(±516)**	7210(±1632)
	PRISM	0.723(±0.014)	0.993(±0.020)	2655(±923)	1851(±435)	3797(±1283)
	PLATEAU	0.716(±0.010)	0.984(±0.029)	4992(±651)	**2994(±619)**	7042(±1623)
MUSE	NPO	0.652(±0.012)	0.990(±0.008)	4952(±1067)	**2974(±638)**	7327(±1949)
	TAR	0.648(±0.014)	0.999(±0.031)	3452(±754)	2086(±426)	5405(±593)
	PRISM	0.652(±0.010)	0.990(±0.030)	3766(±286)	2097(±281)	5805(±885)
	PLATEAU	0.653(±0.013)	0.981(±0.016)	5301(±921)	**3165(±875)**	6751(±1832)
WMDP-bio	NPO	0.737(±0.007)	0.984(±0.011)	2079(±337)	1349(±289)	2541(±721)
	TAR	0.742(±0.013)	0.996(±0.034)	4443(±505)	**2512(±147)**	6223(±1143)
	PRISM	0.737(±0.013)	1.000(±0.016)	2512(±297)	1832(±222)	3244(±694)
	PLATEAU	0.741(±0.014)	1.005(±0.028)	4050(±360)	**2408(±229)**	8013(±1055)

E.4 Artifact and Verification Package

The reproducibility package includes checkpoint identifiers and permitted adapter deltas; fixed evaluation prompts and splits; declaration manifests; release-quality measurements; and complete-update, cumulative-rank, and displacement logs. A numerical certificate additionally needs the verifier configuration, arithmetic precision and rounding policy, bound witnesses or reproducible enclosures, and an independent checker. Training seeds and attack seeds are stored separately. CSV records and plotting/table scripts reproduce every reported aggregate.

CDT is a candidate-construction procedure: the gradient penalty acts on $\mathcal{L}_f$, and the signed trace is a training surrogate. The certification pass evaluates the margin, slope and neighborhood bound over every member of $\mathcal{G}$ for the updated release. This separation places the guarantee on the checked output and gives the artifact package a concrete object to verify.

F Proofs of Theoretical Results

F.1 Assumptions and Scope

All norms refer to the released weight parameterization. A displacement consists of matrix blocks $v_\ell \in \mathbb{R}^{m_\ell \times n_\ell}$, with inner product and norm

$$\langle v, w\rangle = \sum_\ell \operatorname{Tr}(v_\ell^\top w_\ell), \qquad \|v\|^2 = \sum_\ell \|v_\ell\|_F^2, \qquad \mathcal{A}_s = \{v : \operatorname{rank}(v_\ell) \le s \text{ for every } \ell\}. \tag{13}$$

We take $s \ge 1$ and extend each singular-value sequence by zeros beyond the block dimensions. The set $\mathcal{A}_s$ is closed under scalar multiplication and sign reversal, but generally not under addition. Its intersection with the unit sphere is compact. The family $\mathcal{G}$ is finite, and every $\mathcal{L} \in \mathcal{G}$ is C^2 on an open neighborhood of the closed ball $\mathbb{B}_\rho(\theta_u)$, where $\rho > 0$ is fixed before certification. Thus the suprema defining g_s and $M_s(\rho)$ are finite. We abbreviate $M_s(\rho)$ by M_s when the neighborhood is fixed.

The threshold γ is fixed, the release margin $\Delta = \min_{\mathcal{L}\in\mathcal{G}} \mathcal{L}(\theta_u) - \gamma$ is positive, and recovery means strict inequality $\mathcal{L} < \gamma$ for at least one protected loss. The step bound $h = \eta G > 0$ concerns each complete optimizer displacement. The proofs allow adaptive data and optimizer choices, provided that every cumulative displacement lies in the declared set. Any validated lower bound on Δ and upper bounds on g_s, M_s can be substituted: decreasing the margin or increasing either descent bound only weakens the resulting certificate. Local probes alone do not supply these uniform bounds. Higher smoothness is invoked explicitly only for the finite-difference error estimates below.

F.2 Proof of Lemma 1 (Directional Geometry)

Lemma 1 (Directional gradient norm)**.** *For $W_\ell = \nabla_{\theta_\ell}\mathcal{L}(\theta_u)$, the supremum of $|\langle W, v\rangle|$ over unit $v \in \mathcal{A}_s$ equals $(\sum_\ell \sum_{i=1}^s \sigma_i(W_\ell)^2)^{1/2}$.*

Proof. Let P_ℓ be the orthogonal projector onto the column space of v_ℓ. Since $P_\ell v_\ell = v_\ell$ and $\operatorname{rank}(P_\ell) \le s$, the Frobenius inner product gives

$$\begin{aligned} |\langle W_\ell, v_\ell\rangle| &= |\operatorname{Tr}(W_\ell^\top P_\ell v_\ell)| = |\langle P_\ell W_\ell, v_\ell\rangle| \le \|P_\ell W_\ell\|_F \|v_\ell\|_F, \\ \|P_\ell W_\ell\|_F^2 &= \operatorname{Tr}(W_\ell^\top P_\ell W_\ell) = \operatorname{Tr}(P_\ell W_\ell W_\ell^\top) \le \sum_{i=1}^s \sigma_i(W_\ell)^2. \end{aligned} \tag{14}$$

For the last inequality, diagonalize $W_\ell W_\ell^\top = \sum_i \sigma_i^2 u_i u_i^\top$. The trace is $\sum_i \sigma_i^2 \|P_\ell u_i\|^2$, where every weight lies in $[0,1]$ and their sum is at most s. Assigning unit weight to the s largest eigenvalues maximizes this sum. Writing $(W_\ell)_s$ for the truncated SVD and applying Cauchy–Schwarz across blocks now yields

$$\begin{aligned} |\langle W, v\rangle| &\le \sum_\ell |\langle W_\ell, v_\ell\rangle| \le \sum_\ell \|(W_\ell)_s\|_F \|v_\ell\|_F \\ &\le \left(\sum_\ell \|(W_\ell)_s\|_F^2\right)^{1/2} \left(\sum_\ell \|v_\ell\|_F^2\right)^{1/2} = \left(\sum_\ell \sum_{i=1}^s \sigma_i(W_\ell)^2\right)^{1/2}. \end{aligned} \tag{15}$$

Let $S^2 = \sum_\ell \|(W_\ell)_s\|_F^2$. If $S > 0$, the choice $v_\ell = (W_\ell)_s / S$ is feasible and attains the bound:

$$\|v\|^2 = \frac{\sum_\ell \|(W_\ell)_s\|_F^2}{S^2} = 1, \qquad \langle W, v\rangle = \frac{\sum_\ell \langle W_\ell, (W_\ell)_s\rangle}{S} = \frac{S^2}{S} = S. \tag{16}$$

If $S = 0$, Eq. (15) makes every feasible inner product zero. Sign symmetry implies that the supremum without the absolute value is the same. Taking the maximum over $\mathcal{G}$ proves Eq. (4). □

The full-space value is $\|\nabla\mathcal{L}(\theta_u)\|$, and equality with the rank-restricted value holds if every block gradient has rank at most s. For a fixed linear subspace with orthogonal projector P, the corresponding calculation is

$$\sup_{v\in\operatorname{range}(P),\, \|v\|=1} |\langle W, v\rangle| = \sup_{v\in\operatorname{range}(P),\, \|v\|=1} |\langle PW, v\rangle| = \|PW\|. \tag{17}$$

These are different attack declarations. In particular, $\mathcal{A}_s \subseteq \mathcal{A}_{s'}$ for $s \le s'$ implies $g_s \le g_{s'}$ and $M_s(\rho) \le M_{s'}(\rho)$; enlarging the set cannot improve the radius when the margin and neighborhood are unchanged.

From a release-point bound to a neighborhood bound. Suppose $H_{\mathcal{L}}(\theta) = \nabla^2\mathcal{L}(\theta)$ satisfies $\|H_{\mathcal{L}}(\theta) - H_{\mathcal{L}}(\theta_u)\|_{\mathrm{op}} \le K\|\theta - \theta_u\|$ uniformly over the family and ball. For any unit $d \in \mathcal{A}_s$, the positive-part inequality $[a+b]_+ \le [a]_+ + |b|$ gives the complete bound

$$\begin{aligned}[-d^\top H_{\mathcal{L}}(\theta)d]_+ &= \left[-d^\top H_{\mathcal{L}}(\theta_u)d - d^\top(H_{\mathcal{L}}(\theta) - H_{\mathcal{L}}(\theta_u))d\right]_+ \\ &\le [-d^\top H_{\mathcal{L}}(\theta_u)d]_+ + |d^\top(H_{\mathcal{L}}(\theta) - H_{\mathcal{L}}(\theta_u))d| \\ &\le \bar{m}_s + \|H_{\mathcal{L}}(\theta) - H_{\mathcal{L}}(\theta_u)\|_{\mathrm{op}} \le \bar{m}_s + K\|\theta - \theta_u\| \le \bar{m}_s + K\rho.\end{aligned} \tag{18}$$

Taking the supremum proves Eq. (12). The constant K and the release-point upper bound must themselves be valid. Sampling a few directions or points only lower-bounds the corresponding supremum; it does not justify replacing $M_s(\rho)$ by the largest sampled value. This distinction is why the empirical budget $\widehat{B}$ is separate from a certified budget.

The curvature calculation includes interactions between trainable blocks. Although the gradient support function separates by layer, the Hessian quadratic form expands as

$$\begin{aligned}v^\top H_{\mathcal{L}}(\theta)v &= \sum_{\ell,\ell'}\langle v_\ell, H_{\ell\ell'}(\theta)v_{\ell'}\rangle \\ &= \sum_{\ell}\langle v_\ell, H_{\ell\ell}(\theta)v_\ell\rangle + 2\sum_{\ell<\ell'}\langle v_\ell, H_{\ell\ell'}(\theta)v_{\ell'}\rangle.\end{aligned} \tag{19}$$

Ignoring the cross terms can invalidate a putative upper bound on negative curvature. For two scalar blocks, take $H = \left(\begin{smallmatrix} 0 & -K \\ -K & 0 \end{smallmatrix}\right)$ with $K > 0$. Each diagonal block has zero curvature, but the admissible unit direction $v = (1,1)/\sqrt{2}$ gives $v^\top Hv = -K$. Thus summing independently measured diagonal-block curvatures is insufficient without an additional bound on their interactions. Equation (18) uses the full Hessian operator and covers these interactions.

F.3 Proof of Theorem 3 (Certified Update Budget)

Lemma 2 (Barrier)**.** *For every $\mathcal{L} \in \mathcal{G}$ and $v \in \mathcal{A}_s$ with $r = \|v\| \le \rho$, the loss satisfies $\mathcal{L}(\theta_u+v) - \gamma \ge \Delta - g_s r - M_s r^2/2$.*

Proof. The case $r = 0$ is the release condition. Otherwise let $d = v/r$, $\theta(t) = \theta_u + tv$, and $f(t) = \mathcal{L}(\theta(t))$. The segment stays in the ball because $\|\theta(t) - \theta_u\| = tr \le \rho$ for $0 \le t \le 1$. The chain rule and the directional bounds give

$$\begin{aligned}f'(0) &= \langle \nabla\mathcal{L}(\theta_u), v\rangle = r\langle \nabla\mathcal{L}(\theta_u), d\rangle \ge -g_s r, \\ f''(t) &= v^\top H_{\mathcal{L}}(\theta(t))v = r^2 d^\top H_{\mathcal{L}}(\theta(t))d \ge -M_s r^2.\end{aligned} \tag{20}$$

Integrating twice and reversing the order over $0 \le u \le t \le 1$ expands the Taylor remainder without dropping intermediate terms:

$$\begin{aligned}\mathcal{L}(\theta_u + v) - \mathcal{L}(\theta_u) = f(1) - f(0) &= \int_0^1 f'(t)\,dt \\ &= \int_0^1 \left(f'(0) + \int_0^t f''(u)\,du\right)dt = f'(0) + \int_0^1\int_u^1 f''(u)\,dt\,du \\ &= \langle \nabla\mathcal{L}(\theta_u), v\rangle + \int_0^1 (1-u)v^\top H_{\mathcal{L}}(\theta_u + uv)v\,du \\ &\ge -g_s r - M_s r^2 \int_0^1 (1-u)\,du = -g_s r - \frac{1}{2}M_s r^2.\end{aligned} \tag{21}$$

Consequently the same constants protect the whole evaluation family:

$$\begin{aligned}\min_{\mathcal{L}\in\mathcal{G}}[\mathcal{L}(\theta_u + v) - \gamma] &\ge \min_{\mathcal{L}\in\mathcal{G}}[\mathcal{L}(\theta_u) - \gamma - g_s r - \tfrac{1}{2}M_s r^2] \\ &= \min_{\mathcal{L}\in\mathcal{G}}[\mathcal{L}(\theta_u) - \gamma] - g_s r - \tfrac{1}{2}M_s r^2 = \Delta - g_s r - \tfrac{1}{2}M_s r^2.\end{aligned} \tag{22}$$

The radial segment is a device for bounding the endpoint loss; the actual optimizer trajectory need not follow it. Its validity uses the cumulative displacement direction, not the rank of an individual update. □

Theorem 3 (Certified radius and recovery budget)**.** *Suppose every cumulative displacement $\theta[t]-\theta_u$ lies in $\mathcal{A}_s$ and every complete update has norm at most h. Then Eq.* (7) *certifies $\mathcal{L}(\theta[t]) \geq \gamma$ for every $\mathcal{L} \in \mathcal{G}$ and integer $0 \leq t \leq \lfloor B^c \rfloor$.*

Proof. For $M_s > 0$, the barrier is nonnegative precisely when

$$\begin{aligned}\Delta - g_s r - \tfrac{1}{2}M_s r^2 \geq 0 &\iff M_s r^2 + 2g_s r \leq 2\Delta \\ &\iff (M_s r + g_s)^2 \leq g_s^2 + 2M_s\Delta \\ &\iff 0 \leq r \leq r_* \equiv \frac{\sqrt{g_s^2+2M_s\Delta}-g_s}{M_s} \\ &\qquad = \frac{(g_s^2+2M_s\Delta)-g_s^2}{M_s(\sqrt{g_s^2+2M_s\Delta}+g_s)} = \frac{2\Delta}{g_s+\sqrt{g_s^2+2M_s\Delta}}.\end{aligned} \tag{23}$$

Here $r \geq 0$ and $g_s \geq 0$ justify taking the nonnegative square root. The rationalized expression also avoids subtracting nearly equal positive numbers. If $M_s = 0 < g_s$, then $r_* = \Delta/g_s$; if $g_s = 0 < M_s$, then $r_* = \sqrt{2\Delta/M_s}$; if both vanish, set $r_* = +\infty$. The finite neighborhood always caps the result: $R_s^c = \min\{\rho, r_*\}$.

Write $v[j] = \theta[j+1]-\theta[j]$ and $B_{\text{safe}} = \lfloor R_s^c/h \rfloor$. Telescoping the iterates and using the complete-update bound gives, for every $t \leq B_{\text{safe}}$,

$$\begin{aligned}\|\theta[t]-\theta_u\| &= \left\|\sum_{j=0}^{t-1}(\theta[j+1]-\theta[j])\right\| = \left\|\sum_{j=0}^{t-1} v[j]\right\| \\ &\leq \sum_{j=0}^{t-1}\|v[j]\| \leq \sum_{j=0}^{t-1} h = th \leq h\left\lfloor \frac{R_s^c}{h}\right\rfloor \leq R_s^c \leq \rho.\end{aligned} \tag{24}$$

Applying the barrier at this cumulative displacement proves the claim. Equality with γ remains safe, so a first strict recovery needs at least $B_{\text{safe}}+1$ complete updates, including when R_s^c/h is an integer. The proof is deterministic for each admissible trajectory and therefore also covers adaptive or randomized choices satisfying the bounds on every realization. □

Dependence on the three measures. For a finite, uncapped root, differentiate $g_s r_* + M_s r_*^2/2 = \Delta$. Since $g_s + M_s r_* > 0$, collecting the terms yields

$$\begin{aligned}(g_s + M_s r_*)\, dr_* &= d\Delta - r_*\, dg_s - \tfrac{1}{2}r_*^2\, dM_s, \\ \frac{\partial r_*}{\partial \Delta} &= \frac{1}{g_s+M_s r_*}, \qquad \frac{\partial r_*}{\partial g_s} = -\frac{r_*}{g_s+M_s r_*}, \qquad \frac{\partial r_*}{\partial M_s} = -\frac{r_*^2}{2(g_s+M_s r_*)}.\end{aligned} \tag{25}$$

Thus increasing the margin or decreasing either descent measure enlarges the uncapped radius. The capped radius is locally constant when $r_* > \rho$, and the integer budget changes only at multiples of h. These qualifications matter when interpreting a training ablation.

For $g_s > 0$, let $a = M_s\Delta/g_s^2$. In the slope-dominated regime $a \to 0$, the rationalized root expands as

$$r_* = \frac{\Delta}{g_s}\frac{2}{1+\sqrt{1+2a}} = \frac{\Delta}{g_s}\left(1-\frac{a}{2}+O(a^2)\right) = \frac{\Delta}{g_s} - \frac{M_s\Delta^2}{2g_s^3} + O\left(\frac{M_s^2\Delta^3}{g_s^5}\right). \tag{26}$$

At $g_s = 0$ the square-root expression above applies exactly. Equation (25) explains Figure 2: at fixed margin and curvature bound, reducing the slope moves the threshold of the uniform lower bound outward. It does not identify the exact threshold contour of an arbitrary loss from three scalar bounds alone.

Conversely, with $M_s > 0$ and $b = g_s/\sqrt{2M_s\Delta} \to 0$, the curvature-dominated expansion is

$$\begin{aligned} r_* &= \sqrt{\frac{2\Delta}{M_s}}\,\frac{1}{b+\sqrt{1+b^2}} = \sqrt{\frac{2\Delta}{M_s}}\big(\sqrt{1+b^2}-b\big) \\ &= \sqrt{\frac{2\Delta}{M_s}}\left(1-b+\frac{b^2}{2}+O(b^4)\right). \end{aligned} \tag{27}$$

For positive g_s, M_s, the relative sensitivities follow directly from the same derivatives:

$$-\frac{\partial \log r_*}{\partial \log g_s} = \frac{g_s}{g_s + M_s r_*}, \qquad -\frac{\partial \log r_*}{\partial \log M_s} = \frac{M_s r_*}{2(g_s + M_s r_*)}. \tag{28}$$

The slope sensitivity approaches one when $M_s r_* \ll g_s$, while the curvature sensitivity approaches one half when $g_s \ll M_s r_*$. These are statements about changing one valid scalar bound while holding the other inputs fixed; they do not imply that a training regularizer changes only its intended measure.

F.4 Derivation of the Cumulative-Displacement Conditions

If only individual increments have rank at most s, their cumulative rank can grow. For each layer,

$$\operatorname{rank}(\theta_\ell[t] - \theta_{u,\ell}) = \operatorname{rank}\left(\sum_{j=0}^{t-1} v_\ell[j]\right) \le \sum_{j=0}^{t-1} \operatorname{rank}(v_\ell[j]) \le ts. \tag{29}$$

The two rank-one increments $\operatorname{diag}(1,0)$ and $\operatorname{diag}(0,1)$ sum to a rank-two matrix. A horizon B of independent rank-s increments therefore requires the bounds g_{Bs}, M_{Bs}, with ranks capped at each block's dimension. For such a declaration, the sufficient horizon condition is $Bh \le R^{\mathrm{c}}_{Bs}$; inserting R^{c}_s would leave some reachable directions uncovered.

A single unmerged adapter instead represents the current weight as $W_t = W_u + B_t A_t$, relative to the frozen release. Its cumulative displacement has rank at most s, although $B_{t+1}A_{t+1} - B_t A_t$ can have rank $2s$. The radius theorem needs the former property and a norm bound on the latter. Factor-space clipping by itself does not establish the required weight-space norm bound, since

$$\begin{aligned} \delta W &= (B+\delta B)(A+\delta A) - BA = B\delta A + \delta B A + \delta B \delta A, \\ \|\delta W\|_F &\le \|B\|_{\mathrm{op}}\|\delta A\|_F + \|\delta B\|_F \|A\|_{\mathrm{op}} + \|\delta B\|_F \|\delta A\|_{\mathrm{op}}. \end{aligned} \tag{30}$$

One must validate these factor-dependent quantities or enforce the bound on the resulting weight update while preserving the cumulative adapter constraint. Clipping and admissibility are separate requirements; a weight-space clipping rule must not silently enlarge the displacement set.

For a proposed complete update $q[t]$, radial clipping gives $v[t] = \min\{1, h/\|q[t]\|\}q[t]$ when $q[t] \neq 0$, and $v[t] = 0$ otherwise. Hence

$$\|v[t]\| = \min\{1, h/\|q[t]\|\}\|q[t]\| = \min\{\|q[t]\|, h\} \le h. \tag{31}$$

This bounds the complete displacement, including preconditioning, momentum, and weight decay. A percentile used to calibrate h is not a hard bound without enforcement. More generally, valid varying step bounds h_j give $\|\theta[t] - \theta_u\| \le \sum_{j<t} h_j$ and certify every prefix whose sum is at most R^{c}_s, under the same cumulative-set condition.

F.5 Derivation of Tightness and Recovery Bounds

Set $g = g_s$, $M = M_s$ and suppose $g + M > 0$. Along a unit admissible ray, the quadratic $q(t) = \gamma + \Delta - gt - Mt^2/2$ has

$$q(0) - \gamma = \Delta, \qquad q'(t) = -g - Mt, \qquad q''(t) = -M, \qquad q(t) - \gamma = \Delta - gt - \tfrac{1}{2}Mt^2. \tag{32}$$

It attains the barrier and reaches γ exactly at r_*. If $\rho > r_*$, it is strictly below γ immediately beyond that root, so no larger radius follows from these constants alone. This is a local construction: when a nonnegative loss is required and $\gamma > 0$, choose the neighborhood just beyond the root so the quadratic

remains positive. No globally negative log-likelihood is needed. The radius cap also remains essential when the loss outside the prescribed neighborhood is unconstrained.

Worst-case attainability supplies no recovery upper bound for every instance. For example, the constant loss $\gamma + \Delta$ obeys any nonnegative upper bounds g, M but never recovers. Even exact release slope and worst negative curvature need not be attained by the same direction and objective. An instance-specific upper bound therefore requires a feasible descent ray in addition to the certificate measures.

To make this requirement explicit, assume a loss $\mathcal{L}_*$ attains the release margin and a unit d has $td \in \mathcal{A}_s$ and $\frac{d}{dt}\mathcal{L}_*(\theta_u + td) \le -cg - cMt$ on $0 \le t \le r_*/c \le \rho$, where $0 < c \le 1$. Then

$$\begin{aligned}\mathcal{L}_*(\theta_u + td) - \gamma &= \mathcal{L}_*(\theta_u) - \gamma + \int_0^t \frac{d}{du}\mathcal{L}_*(\theta_u + ud)\, du \\ &\le \Delta - \int_0^t (cg + cMu)\, du = \Delta - cgt - \tfrac{1}{2}cMt^2, \qquad (33)\\ \mathcal{L}_*(\theta_u + (r_*/c)d) - \gamma &\le \Delta - gr_* - \frac{Mr_*^2}{2c} = -\frac{Mr_*^2}{2}\left(\frac{1}{c} - 1\right) \le 0.\end{aligned}$$

The global lower bound excludes threshold contact before r_*, while continuity and Eq. (33) ensure contact by r_*/c. This brackets the first threshold distance. Strict recovery is already implied when the last inequality is strict; otherwise the descent condition must extend beyond the touching point. A discrete recovery upper bound further needs a realizable update path to that point. The geometric ray condition alone does not establish that an arbitrary data-driven optimizer can realize it.

F.6 Proof of Theorem 4 (Durability Separation)

Theorem 4 (Separation)**.** *For every width $w \ge 2$, there exists a fixed smooth input-dependent classifier architecture and parameterization with identical nonconstant release predictions whose minimum recovery costs under unit-bounded updates differ exponentially in w.*

Proof. Let $x \in [-1, 1]$ and make all coordinates $(a, b) \in \mathbb{R}^w \times \mathbb{R}^w$ trainable. A fixed identity skip supplies the input dependence; its coefficient is part of the architecture. Define the residual, logit, and probability by

$$u(a, b) = \sum_{i=1}^{w} a_i e^{-b_i^2}, \qquad z_{a,b}(x) = x[1 + u(a, b)], \qquad (34)$$

$$p_{a,b}(1 \mid x) = (1 + e^{-x[1+u(a,b)]})^{-1}. \qquad (35)$$

Use the forget example $(x, y) = (1, 1)$, whose loss is

$$\mathcal{L}(a, b) = -\log p_{a,b}(1 \mid 1) = \log(1 + e^{-[1+u(a,b)]}). \qquad (36)$$

Set $D = 2^w$, $\gamma = \log(1 + e^{-2})$, $\theta_a = (0, 0)$, and $\theta_b = (0, D\mathbf{1})$. At both releases $u = 0$, so both predict $(1 + e^{-x})^{-1}$ on every input, a nonconstant function with decision boundary $x = 0$. Their common positive margin is $\log(1 + e^{-1}) - \gamma$. Monotonicity gives the recovery condition

$$\mathcal{L} < \gamma \iff \log(1 + e^{-(1+u)}) < \log(1 + e^{-2}) \iff e^{-(1+u)} < e^{-2} \iff u > 1. \qquad (37)$$

Starting from θ_a, two unit increases of a_1 produce $a_1 = 2$, $b = 0$, and $u = 2 > 1$. Thus its minimum recovery cost T_a is at most two; this path is sufficient and is not asserted to be optimal.

For the second release, consider any point at distance at most $D/2$. Since $\|a\|^2 + \|b - D\mathbf{1}\|^2 \le D^2/4$, we have $\|a\| \le D/2$ and $b_i \ge D - |b_i - D| \ge D/2$. Therefore

$$\begin{aligned}u(a, b) = \sum_i a_i e^{-b_i^2} &\le \sum_i |a_i| e^{-b_i^2} \le e^{-D^2/4} \sum_i |a_i| \\ &\le e^{-D^2/4}\sqrt{w}\left(\sum_i a_i^2\right)^{1/2} \le \sqrt{w}\,\frac{D}{2} e^{-D^2/4} < 1.\end{aligned} \qquad (38)$$

For completeness, $4^{w-1} \geq 2w$ for integer $w \geq 2$, with equality at $w = 2$ and induction thereafter. Also $\log w \leq w - 1$ and $\log 2 < 1$, so

$$\log(\sqrt{w}D/2) = \tfrac{1}{2}\log w + (w-1)\log 2 < \tfrac{3}{2}(w-1) < 2w \leq 4^{w-1} = D^2/4. \tag{39}$$

Exponentiating after subtracting $D^2/4$ proves the strict inequality in Eq. (38). With unit-bounded updates, the telescoping argument of Eq. (24) keeps every iterate through step $D/2$ inside this ball, independently of the attack's choices. Hence $T_b > D/2$.

A finite recovery path first increases a_1 to two in two unit steps, then decreases b_1 from D to zero in D unit steps. At the endpoint $u = 2$. These moves are also normalized coordinate-gradient descent steps in exact arithmetic, because

$$\begin{aligned}
\frac{d\mathcal{L}}{du} &= -\frac{e^{-(1+u)}}{1+e^{-(1+u)}} = -\frac{1}{1+e^{1+u}}, \\
\frac{\partial \mathcal{L}}{\partial a_i} &= \frac{d\mathcal{L}}{du}\, e^{-b_i^2} = -\frac{e^{-b_i^2}}{1+e^{1+u}} < 0, \\
\frac{\partial \mathcal{L}}{\partial b_i} &= \frac{d\mathcal{L}}{du}(-2a_i b_i e^{-b_i^2}) = \frac{2a_i b_i e^{-b_i^2}}{1+e^{1+u}} > 0 \quad (a_i, b_i > 0).
\end{aligned} \tag{40}$$

Consequently $1 \leq T_a \leq 2$ and $2^{w-1} < T_b \leq 2^w + 2$, implying $T_b/T_a > 2^{w-2}$. The lower bound covers all unit-bounded updates, and the upper paths lie within the coordinate first-order subclass. □

The architecture and weight coordinates are fixed within each pair. Parameter magnitudes may grow as 2^w, and normalization of the exponentially small gradients uses exact arithmetic. The construction separates output-only checks from durability; it does not establish exponential separation under a uniform weight bound, finite-precision normalization, or every model architecture.

F.7 Proof of Theorem 5 (Directional Curvature)

Theorem 5 (Anisotropy)**.** *The largest positive Hessian eigenvalue alone does not determine the certified radius.*

Proof. At the origin, for $K, g, \Delta > 0$, consider

$$\begin{aligned}
\mathcal{L}_1(x) &= \gamma + \Delta - Kx, & \nabla\mathcal{L}_1(0) &= -K, & \nabla^2\mathcal{L}_1 &= 0, \\
\mathcal{L}_2(x,y) &= \gamma + \Delta - gx + \tfrac{K}{2}y^2, & \nabla\mathcal{L}_2(0,0) &= (-g, 0), & \nabla^2\mathcal{L}_2 &= \mathrm{diag}(0, K).
\end{aligned} \tag{41}$$

For full-space displacements of norm at most r, their smallest losses are exactly

$$\begin{aligned}
\min_{|x|\leq r}[\mathcal{L}_1(x) - \gamma] &= \min_{|x|\leq r}(\Delta - Kx) = \Delta - Kr, \\
\min_{x^2+y^2\leq r^2}[\mathcal{L}_2(x,y) - \gamma] &= \Delta - gr, \qquad \text{since } -gx + \tfrac{K}{2}y^2 \geq -gr \text{ with equality at } (r, 0).
\end{aligned} \tag{42}$$

Their exact threshold radii are Δ/K and Δ/g. Both have $M_s = 0$, so the uncapped certified roots coincide with these values. The first radius tends to zero despite a zero Hessian; the second stays fixed as the largest positive eigenvalue K grows. Choose a neighborhood containing the respective threshold to avoid an active cap. □

Negative curvature has the opposite role. Replacing $+Ky^2/2$ by $-Ky^2/2$ gives, on the gradient-orthogonal ray $(0, t)$, a loss drop $Kt^2/2$ and threshold distance $\sqrt{2\Delta/K}$. Thus orthogonality to the release gradient does not remove a direction from the curvature bound. Likewise, $H = \mathrm{diag}(-K, K)$ has $\mathrm{Tr}(H) = 0$ but $\sup_{\|d\|=1}[-d^\top H d]_+ = K$. A signed trace cannot certify the worst negative curvature, even when its value is known exactly.

F.8 Derivation of CDT Gradients and Probe Estimators

Projector and gradient penalty. Fix the projector during one update, with $U_{\ell,k}^\top U_{\ell,k} = I_{k_\ell}$. Under column-wise vectorization, the block action in Eq. (9) has matrix $P_\ell = I_{n_\ell} \otimes U_{\ell,k} U_{\ell,k}^\top$. Thus

$$\begin{aligned} P_\ell^2 &= I_{n_\ell} \otimes U_{\ell,k}(U_{\ell,k}^\top U_{\ell,k}) U_{\ell,k}^\top = P_\ell = P_\ell^\top, \\ d_\Pi = \operatorname{rank}(\Pi) = \operatorname{Tr}(\Pi) &= \sum_\ell \operatorname{Tr}(I_{n_\ell}) \operatorname{Tr}(U_{\ell,k} U_{\ell,k}^\top) = \sum_\ell n_\ell \operatorname{Tr}(U_{\ell,k}^\top U_{\ell,k}) = \sum_\ell n_\ell k_\ell. \end{aligned} \tag{43}$$

This is a fixed subspace for the current training update, rather than the union $\mathcal{A}_s$. The trace estimator requires $d_\Pi > 0$. For $W_\ell = \nabla_{\theta_\ell} \mathcal{L}_f(\theta)$ and $w = \nabla \mathcal{L}_f(\theta)$, the penalty bounds the squared directional norm for this objective:

$$\sum_\ell \sum_{i=1}^{s} \sigma_i(W_\ell)^2 \le \sum_\ell \sum_i \sigma_i(W_\ell)^2 = \sum_\ell \operatorname{Tr}(W_\ell^\top W_\ell) = \sum_\ell \|W_\ell\|_F^2 = \|w\|^2. \tag{44}$$

This controls $\mathcal{L}_f$ only; the maximum over $\mathcal{G}$ and the minimum margin must be recomputed after training. Writing $H = \nabla^2 \mathcal{L}_f(\theta)$ and differentiating coordinate by coordinate yields

$$\nabla_\theta \|w\|^2 = \nabla_\theta \sum_i w_i^2 = 2 \sum_i w_i \nabla_\theta w_i = 2(Dw)^\top w = 2H^\top w = 2Hw. \tag{45}$$

The symmetry follows from C^2 smoothness. If $w = 0$, the derivative is zero. Otherwise hold $u = w/\|w\|$ fixed inside the probe calculation. The fundamental theorem of calculus gives

$$\begin{aligned} \frac{\nabla \mathcal{L}_f(\theta + \epsilon u) - \nabla \mathcal{L}_f(\theta - \epsilon u)}{2\epsilon} &= \frac{1}{2\epsilon} \int_{-\epsilon}^{\epsilon} H(\theta + tu) u \, dt, \\ \|w\| \frac{\nabla \mathcal{L}_f(\theta + \epsilon u) - \nabla \mathcal{L}_f(\theta - \epsilon u)}{\epsilon} &\longrightarrow 2\|w\| Hu = 2Hw. \end{aligned} \tag{46}$$

This proves the normalization in Eq. (10). Continuity of the Hessian gives convergence; a quadratic error rate needs more. Let $\psi(t) = \nabla \mathcal{L}_f(\theta + tu)$ and suppose $\|\psi'''(t)\| \le C_4$ on the probe interval. Taylor expansion at zero makes the cancellation and error coefficient explicit:

$$\begin{aligned} \psi(\pm\epsilon) &= \psi(0) \pm \epsilon Hu + \tfrac{1}{2}\epsilon^2 \psi''(0) + R_\pm, \qquad \|R_\pm\| \le C_4 \epsilon^3/6, \\ \left\| \frac{\psi(\epsilon) - \psi(-\epsilon)}{2\epsilon} - Hu \right\| &= \frac{\|R_+ - R_-\|}{2\epsilon} \le \frac{\|R_+\| + \|R_-\|}{2\epsilon} \le \frac{C_4 \epsilon^2}{6}. \end{aligned} \tag{47}$$

Multiplication by $2\|w\|$ bounds the penalty-gradient error by $C_4 \|w\| \epsilon^2/3$. This condition concerns fourth derivatives of the loss and is stronger than the C^2 assumption used for certification.

Trace normalization and finite differences. Let $Q \in \mathbb{R}^{p \times d_\Pi}$ have orthonormal columns spanning $\operatorname{range}(\Pi)$, so $QQ^\top = \Pi$. Draw q uniformly from the unit sphere in $\mathbb{R}^{d_\Pi}$ and set $z = Qq$. Sign symmetry makes $\mathbb{E}[q_i q_j] = 0$ for $i \ne j$; permutation symmetry and $\sum_i q_i^2 = 1$ give $\mathbb{E}[q_i^2] = 1/d_\Pi$. It follows that

$$\begin{aligned} \mathbb{E}[zz^\top] &= Q \mathbb{E}[qq^\top] Q^\top = Q \frac{I_{d_\Pi}}{d_\Pi} Q^\top = \frac{\Pi}{d_\Pi}, \\ \mathbb{E}[z^\top H z] = \mathbb{E}[\operatorname{Tr}(Hzz^\top)] &= \operatorname{Tr}(H \mathbb{E}[zz^\top]) = \frac{\operatorname{Tr}(H\Pi)}{d_\Pi} = \frac{\operatorname{Tr}(\Pi H \Pi)}{d_\Pi}. \end{aligned} \tag{48}$$

The last equality uses cyclicity of the trace and $\Pi^2 = \Pi$. Unit normalization is essential: probes with second moment Π estimate the unnormalized trace instead.

Sampling error of the quadratic probe. The variance can also be calculated without identifying an empirical average with a certificate. Put $d = d_\Pi$ and $J = Q^\top H Q$. For $d \ge 2$, write $a = \mathbb{E}[q_1^4]$ and $b = \mathbb{E}[q_1^2 q_2^2]$. Rotational symmetry makes q_1 and $(q_1 + q_2)/\sqrt{2}$ identically distributed; sign symmetry removes the odd terms. Therefore

$$\begin{aligned} a = \frac{1}{4}\mathbb{E}[(q_1 + q_2)^4] &= \frac{2a + 6b}{4}, \qquad a = 3b, \\ 1 = \mathbb{E}\left[\left(\sum_{i=1}^{d} q_i^2 \right)^2 \right] &= da + d(d-1)b = d(d+2)b. \end{aligned} \tag{49}$$

Diagonalize the symmetric J with eigenvalues λ_i. The probe distribution is unchanged under this rotation, so

$$\begin{aligned}\mathbb{E}[(q^\top Jq)^2] &= \sum_i \lambda_i^2 \mathbb{E}[q_i^4] + 2\sum_{i<j} \lambda_i\lambda_j \mathbb{E}[q_i^2 q_j^2] \\ &= \frac{3\sum_i \lambda_i^2 + 2\sum_{i<j}\lambda_i\lambda_j}{d(d+2)} = \frac{2\,\mathrm{Tr}(J^2) + (\mathrm{Tr}\,J)^2}{d(d+2)}, \\ \mathrm{Var}\left(\frac{1}{m}\sum_{j=1}^m z_j^\top H z_j\right) &= \frac{1}{m}\left(\mathbb{E}[(q^\top Jq)^2] - \frac{(\mathrm{Tr}\,J)^2}{d^2}\right) = \frac{2}{md(d+2)}\left(\mathrm{Tr}(J^2) - \frac{(\mathrm{Tr}\,J)^2}{d}\right).\end{aligned} \tag{50}$$

For $d = 1$, the quadratic form is constant on the two-point sphere and the same variance formula equals zero. Independent probes thus reduce variance by a factor of m, but this controls estimation of the normalized trace only. It does not remove negative-eigenvalue cancellation or establish a uniform bound over the neighborhood.

For $\phi(t) = \mathcal{L}_f(\theta + tz)$, expand the positive and negative displacements and add their remainders. The linear terms cancel, leaving the exact identity

$$\begin{aligned}D_\epsilon(z) &\equiv \frac{\phi(\epsilon) - 2\phi(0) + \phi(-\epsilon)}{\epsilon^2} \\ &= \frac{1}{\epsilon^2}\int_0^\epsilon (\epsilon - t)[\phi''(t) + \phi''(-t)]\,dt \\ &= \int_{-1}^1 (1 - |u|) z^\top H(\theta + \epsilon u z) z\,du \longrightarrow z^\top H z.\end{aligned} \tag{51}$$

If $|\phi^{(4)}(t)| \le C_4$ on the probe interval, expand through third order. The odd terms cancel and the remainders $\widetilde{R}_\pm$ satisfy $|\widetilde{R}_\pm| \le C_4\epsilon^4/24$, giving

$$\begin{aligned}\phi(\epsilon) - 2\phi(0) + \phi(-\epsilon) &= \epsilon^2\phi''(0) + \widetilde{R}_+ + \widetilde{R}_-, \\ |D_\epsilon(z) - z^\top H z| &= \frac{|\widetilde{R}_+ + \widetilde{R}_-|}{\epsilon^2} \le \frac{C_4\epsilon^4/24 + C_4\epsilon^4/24}{\epsilon^2} = \frac{C_4\epsilon^2}{12}.\end{aligned} \tag{52}$$

With a uniform bound over probe directions, this also bounds the bias of the mean estimator in Eq. (11). Finite sampling remains a separate source of error; neither a small finite-difference bias nor many probes turns an average curvature into the worst-direction bound M_s.

Finally, the trace regularizer is optimized by differentiating scalar finite differences with Π and the sampled z_j frozen. For one probe the training gradient is exactly

$$\nabla_\theta D_\epsilon(z) = \frac{\nabla\mathcal{L}_f(\theta + \epsilon z) - 2\nabla\mathcal{L}_f(\theta) + \nabla\mathcal{L}_f(\theta - \epsilon z)}{\epsilon^2}. \tag{53}$$

An HVP estimates Hz, whereas Eq. (53) differentiates the scalar curvature surrogate; they are not interchangeable. Refreshing Π every T steps changes the surrogate, and no derivative through that refresh is assumed. Together with the signed-trace counterexample above, this explains why CDT motivates a candidate release but supplies no monotone certificate-improvement guarantee. Each candidate must pass the forgetting and utility checks and undergo DME and BC again with bounds valid for its new neighborhood.