REPAIR REACH: SAFELORA'S GATE IS A PUBLIC SCALAR, AND THE ATTACKER READS IT TOO

Anonymous authors
Paper under double-blind review

ABSTRACT

Post-hoc safety repairs promise to undo harmful fine-tuning cheaply, each through a set fixed before the attack is seen, such as SAFELORA's alignment matrix from public checkpoints, the row space of gradient patching, RESTA's direction or a coordinate mask's support. We call that set the repair's *reach*. SAFELORA's gate is *public* too: the scalar that gates its threshold mode is a closed form in the alignment matrix's singular values, weighted by the update's energy on each, a value the attacker computes before training and steers. Constraining the adapter's output factor into the leading singular subspace (one thin SVD per matrix, a projection per step, with data, objective, optimiser, rank and locus unchanged) makes the gate read the attack as already aligned, and on three instruct models the repaired model still complies with 52.5–$62.9\,\%$ of harmful requests, against 57.9–$62.1\,\%$ unrepaired, while the defence takes 24–59 points off every ordinary family mean; no threshold removes the attack while leaving the customer half their gain on their worst task. A second adversary, constrained out of the public safety span, walks past our coordinate mask at rank 16 and the $20\,\%$ budget we set; $40\,\%$ catches it on Qwen2.5-1.5B, and on Qwen2.5-7B only gate and mask together do. Read back as a detector the gate fails, on one model, to two statistics an attacker buys off separately. The study covers four models, 3,086 cells and 367,138 judged generations. Code and records: `https://github.com/<anonymised>/repair-reach` (public upon acceptance).

1 INTRODUCTION

Fine-tuning an aligned language model on a handful of examples strips its refusal behaviour, even when the examples are entirely benign (Qi et al., 2024; Kazdan et al., 2026). Because open models ship their weights and adapters are trained elsewhere and shared, a large literature asks how the holder can *undo* the damage after the fact, and the answers are cheap. SAFELORA (Hsu et al., 2024) needs no data: it forms an alignment matrix from the difference between the public aligned and base checkpoints and projects the submitted adapter onto it. *Safety at One Shot* (Zhang et al., 2026b) needs one safety instance, and restores pre-attack refusal across five models and five attack types.

Both results reproduce here, the second only in part. It reproduces on two of four checkpoints, and at 7–8B only on a finely swept grid, because the coarse grid reaches no clean operating point there; in the gaps it leaves, Llama-3.1-8B has three such points and Qwen2.5-7B none (Appendix A23, Table 17). We ask what these results license. Each of these methods acts on the attacked weights through a set fixed before the attack is seen: SAFELORA's operator is built from two public checkpoints, and an SGD step on $\mathcal{D}_{\text{safe}}$ can only move a weight matrix inside the span of the activations $\mathcal{D}_{\text{safe}}$ produces, since a linear layer's gradient is a sum of outer products $g_t x_t^\top$. So an evaluation that places the attack where attacks are usually placed cannot distinguish "this defence removes harmful updates" from "the harmful updates people happen to write land inside its subspace." Separating them means varying *where in the weights the attack lives*, which the repair literature does not.

This paper varies where the attack lives. We define a repair's *reach*, show that, under an isotropic null, covering an attack and preserving the customer's update trade one-for-one, and build the three adversaries it points at: one confined *into* SAFELORA's public subspace, one confined *out of* the

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.

safety span a gradient patch can reach, and one trained off a coordinate mask's support. The first two survive the repair aimed at them where an otherwise identical attack does not; the third does not survive at all. Each shares the ordinary attack's data, objective, optimiser, rank and locus, and only its placement differs.

Contributions.

1. **Repair reach** (§2): a closed form ρ for how much of a weight locus a repair can touch, $|T|/d_{\text{in}}$, which one forward pass confirms is *attained* on all but three matrices per model; its post-hoc counterpart *coverage* $c(\Delta W)$, which is not a closed form and carries the result (Appendix A12), and a trace identity (Proposition 1) making coverage of the attack and preservation of the benign update trade one-for-one under an isotropic null, which states Ponkshe et al. (2026)'s empirical conclusion for the operators. The null fails on both sides, tilted the defender's way (§2). The same lemma caps the rank behind a published mechanism (§2.4): the low-rankness Zhang et al. (2026b) read as a safety subspace of dimension below 20 is $\text{rank}(\partial L/\partial W) \leq |T|$, and a matched-length benign instance is just as concentrated on all four models by $\text{CER}(20)$, a random-token one within 0.027 (App. A22, A61).

2. **An adversary that changes nothing about the attack** (§3, §5): constraining the adapter's output factor into the *public* alignment matrix's dominant left-singular subspace drives SAFELORA's gate above its threshold on most matrices, so the defence acts on too few to matter, though never none, since A3 adapters still lose up to 0.60 of their update (§6). It costs one thin SVD per matrix of public weights, with data, objective, optimiser, rank and locus unchanged. The budget m is model-dependent: a sweep, not a redesign (App. A33). Under an independent judge the repaired model still complies with 52.5–$62.9\,\%$ of harmful requests, where ordinary families are cut by 23.9–58.9 points to 1.1–$28.9\,\%$ on the three models with an aligned θ_0 (§5.5), and no threshold removes it while leaving the customer half their gain on their worst task, on either grid we sweep.

3. **A second blind spot, in a different place** (§5.4): on Qwen2.5-1.5B our coordinate-mask stand-in, built at θ_0 at a budget we set, cuts every ordinary attack to ≤ 0.041 and leaves the *row-orthogonal* adversary A2 at 0.232–0.473 over eight seeds, where SAFELORA takes the same eight to ≤ 0.014 and is walked past by A3; Qwen2.5-7B repeats the direction. Unlike τ, this dial turns cheaply: at $40\,\%$ it takes the same eight to 0.023–0.173 for $6\,\%$ of one SQL-Create fine-tune's gain.

4. **The gate statistic is a description, not a defence** (§6): read as a detector it flags 0 of 12 held-out all-modules benign fine-tunes at every margin (other loci overlap it on two models, App. A8) and 66 of 81 A3 adapters at the $+0.10$ its other-locus false positives force, but it is a closed form on two public files and a subspace the attacker picks, and on Qwen2.5-1.5B one who widens it to d_{out} passes *under* the threshold with the projection removing nothing.

The evidence is 3,086 evaluated cells, each with benign over-refusal measured (the headline uses the three models with an aligned θ_0). An independent judge labels 367,138 generations, a second checks ten of its 29 stages, and every headline and paired-difference claim carries a paired bootstrap interval.

Related work. Where safety *lives* is well studied (Qi et al., 2025; Wei et al., 2024; Arditi et al., 2024; Wollschläger et al., 2025; Siu et al., 2026); we ask instead where an *attack* may be put, and what that does to a repair. Post-hoc repairs whose range, domain, direction or support is fixed before the attack (Hsu et al., 2024; Zhang et al., 2026b; Yi et al., 2025; Zhao et al., 2024b; Bhardwaj et al., 2024; Djuhera et al., 2026) are the operators of Table 1; those that choose their set after seeing it (Huang et al., 2025; Wang et al., 2025; Lu et al., 2025; Wu et al., 2025) are outside this frame (two, re-implemented from their released code, remove A3; App. A62). Zloczower et al. (2026) break fifteen defences installed *before* the attacker's fine-tune, including LOX (Perin et al., 2025); repairing the *attacked* weights is the setting they do not touch, and our adversary is orthogonal to the axes they vary. Ponkshe et al. (2026) find projection defences unlikely to work without utility loss; ours is an operator-level statement adjacent to theirs. Zhang et al. (2026a) use the row-space fact we exploit, from the defender's side. Ding (2026) reports SAFELORA scoring harmful adapters as benign, but theirs are harmful only once composed; what is new is that the gate's value is a closed form on public files, so one adapter's score is a design parameter, not luck (Appendix A1).

Table 1: The five repairs as operators; in every row the constrained set is fixed before the attack is seen (rows 1–2 at θ_0, with the step at θ_a). Rows 1–2 Zhang et al. (2026b), 3 Hsu et al. (2024), 4 NLSR's class (Yi et al., 2025), 5 Bhardwaj et al. (2024). Row 3 acts by C's spectral *weighting*, not its range: C is a contraction, and the projector it *derives* is a separate arm (App. A6, A36).

repair	operator	what is fixed	side
One-shot fine-tuning	gradient steps on $\mathcal{D}_{\text{safe}}$	$\mathcal{H}_{\text{safe}}$, $\dim \le \sum_s L_s$	rows
Eq. 5 rank-k edit	$-\alpha\eta \sum_{i\le k} \sigma_i u_i v_i^\top$	$\text{span}\{v_1, \dots, v_k\} \subseteq \mathcal{H}_{\text{safe}}$	rows
SAFELORA	$\Delta W \mapsto C\Delta W$	$\{\sigma_i^2/\|V\|_F\}$ on V's u_i, public	columns
our θ_0 mask (NLSR's class)	coordinate mask M	$\text{supp}(M)$, set at θ_0	coordinates
RESTA	$W \mapsto W + \gamma V_{\text{s}}$	$\text{span}\{V_{\text{s}}\}$	additive

2 REPAIR REACH

Setup. Let θ_0 be a safety-aligned open-weight checkpoint and θ_b its base, both public. An attacker fine-tunes to $\theta_a = \theta_0 + \Delta$; a *post-hoc repair* R maps $\theta_a \mapsto \theta_a + \delta_R$ using a safety set $\mathcal{D}_{\text{safe}}$ and possibly $\theta_0, \theta_b, \Delta$. For a linear layer $W \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$, $y = Wx$. Proofs in Appendix A4.

2.1 THE ROW-SPACE CONFINEMENT LEMMA

Lemma 1 (gradient row space)**.** *For a linear layer and any scalar loss L that depends on W only through its outputs $\{y_t\}_{t\in T}$, $\partial L/\partial W = \sum_{t\in T} g_t x_t^\top$ with $g_t = \partial L/\partial y_t$. Hence* $\text{rowspace}(\partial L/\partial W) \subseteq \text{span}\{x_t : t \in T\}$ *and* $\text{rank}(\partial L/\partial W) \le |T|$.

Corollary 1 (repair reach)**.** *Let R move W by a sum of gradients of losses evaluated on $\mathcal{D}_{\text{safe}}$ with the layer's inputs $\{x_t\}$ held fixed: one step of plain or momentum SGD, or any number of steps on a layer whose inputs are not being trained. Then*

$$\text{rowspace}\big(\delta_R(W)\big) \subseteq \mathcal{H}_{\text{safe}}(W) := \text{span}\{x_t : t \in \mathcal{D}_{\text{safe}}\}, \quad \dim \mathcal{H}_{\text{safe}}(W) \le \min\big(\textstyle\sum_s L_s,\, d_{\text{in}}\big),$$

where t ranges over every token of $\mathcal{D}_{\text{safe}}$ and L_s is the token count of instance s; the bound is attained *on all but three of each model's target matrices, a fact about the safety data (Appendix A12). Everything the attacker places in $\mathcal{H}_{\text{safe}}(W)^\perp$ is untouched by such a step. Over K steps of a deep network neither hypothesis holds and the containment is no longer exact: what survives is a dimension bound $K\sum_s L_s$ (SGD only) on a span the defender cannot evaluate at θ_0, and the norm that escapes $\mathcal{H}_{\text{safe}}(\theta_0)$ over the ten steps we run is $7.6\,\%$ under SGD and $71\,\%$ under AdamW (Qwen2.5-1.5B, Appendix A40). We evaluate $\mathcal{H}_{\text{safe}}$ at the public θ_0, which is what makes it attack-independent; the step is taken at θ_a, a gap the $7.6\,\%$ (a patch of θ_0) does not price.*

Gradient-projection optimisers (Zhao et al., 2024a) and, for the defender, Zhang et al. (2026a) rest on this; new here is that it bounds a *repair*. Adam's $1/\sqrt{v}$ breaks the containment and buys little: A2 stands at 0.375–0.400 wherever the patch is usable (App. A40, one model, `bf16`).

Definition 1 (reach and coverage)**.** $\rho(W) := \dim \mathcal{H}_{\text{safe}}(W)/d_{\text{in}}$, aggregated over a locus by *parameter* count as $\rho^{\text{par}} := \sum_i d_{\text{out},i} \dim \mathcal{H}_{\text{safe}}(W_i)/\sum_i d_{\text{out},i} d_{\text{in},i}$, the null for c and what Proposition 1 counts; the dimension-weighted ρ^{dim} is smaller by up to $1.89\times$ (Appendix A12). For an attack update ΔW, $c(\Delta W) := \|\Delta W Q\|_F^2/\|\Delta W\|_F^2$ with Q an orthonormal basis of $\mathcal{H}_{\text{safe}}(W)$.

ρ costs one forward pass and is computable *before any attack exists*; c is its post-hoc counterpart, with $\mathbb{E}[c] = \rho^{\text{par}}$ under an isotropic null the trained attacks violate by 1.48–$6.36\times$. Both are tabulated per locus and model (Appendix A12): at one-shot scale $\rho \le 2.7\,\%$ everywhere under either aggregate and c is higher at every locus, so an ordinary attack lands disproportionately *inside* the safety row space, which favours a repair confined to it. Since $\|\Delta W(I - QQ^\top)\|_F = \sqrt{1-c}\,\|\Delta W\|_F$, a repair facing an attack with small c can compensate elsewhere but cannot remove what survives.

2.2 THE REPAIRS WE TEST ARE REACH-CONFINED ON AT LEAST ONE SIDE

Table 1's five fix a row space, a subspace of it, a spectral weighting on a public column basis, a coordinate support and a direction; repairs that choose their set *after* the attack are outside the frame (Appendix A26). Only SAFELORA is checked against its authors' code (Appendix A57), so every

negative result about the other four is a result about our implementation of them. SAFELORA is the sharpest instance because its set is not merely fixed but *public*, and because it is *gated*: it projects a matrix only when $\cos(C\Delta W, \Delta W) < \tau$, reading a larger cosine as aligned already. Hsu et al. (2024) describe the projector onto $\mathrm{col}(V)$, their release implements $C = VV^\top/\|V\|_F$, and the two differ: the projector is the *identity* wherever $d_{\mathrm{out}} \le d_{\mathrm{in}}$ (140 of 196 matrices on Qwen2.5-1.5B), so it is the release that has a spectrum to aim at, and the release is what runs (Appendix A6). §5 takes τ apart.

2.3 COVERAGE AND UTILITY TRADE OFF, ONE-FOR-ONE UNDER AN ISOTROPIC NULL

Proposition 1 (trace identity)**.** *On the* parameter *space $\mathbb{R}^D$ the repair's reach is the orthogonal projector $\widetilde{\Pi} = \bigoplus_i (I_{d_{\mathrm{out},i}} \otimes \Pi_i)$ acting by $\Delta W_i \mapsto \Delta W_i \Pi_i$, with Π_i the projector onto $\mathcal{H}_{\mathrm{safe}}(W_i)$ and $k = \mathrm{tr}(\widetilde{\Pi}) = \sum_i d_{\mathrm{out},i}\mathrm{rank}(\Pi_i)$. Let $\{\mathcal{L}_j\}$ partition the matrices into loci, P_j their coordinate projectors ($\sum_j P_j = I$), D_j the parameter count of $\mathcal{L}_j$. Then*

$$\sum_j D_j\, \rho^{\mathrm{par}}(\mathcal{L}_j) \;=\; \sum_j \mathrm{tr}(P_j \widetilde{\Pi} P_j) \;=\; \mathrm{tr}(\widetilde{\Pi}) \;=\; k, \quad \textit{hence} \quad \min_j \rho^{\mathrm{par}}(\mathcal{L}_j) \;\le\; k/D.$$

Symmetrically, a benign update that is spherically symmetric within its own locus places $\rho^{\mathrm{par}}(\mathcal{L}_{\mathrm{task}})$ of its energy in $\widetilde{\Pi}$'s range in expectation (Appendix A4). Both halves are in the same aggregate, so they chain: the row-space version would be in ρ^{dim}, and there the chain fails on all four models (Appendix A12). SAFELORA*'s C is not a projector, so no such budget follows (App. A6).*

What Lemma 1 bounds one matrix at a time this turns into a budget the defender cannot evade by re-partitioning: whatever k is, some locus receives at most the weighted average k/D and the attacker may put the adapter there; and it cannot be enlarged for free, because the reach that covers the attack is the reach that exposes the customer's update, a ρ^{par}-fraction of it for a repair that overwrites inside its reach, and nearly all of it for SAFELORA, whose C, as measured here, deletes rather than projects (Appendix A6). This predicts the utility drop Zhang et al. (2026b) see as safety examples are added, and §5.3 measures both, in different units: energy is a poor proxy for the customer's gain (Appendix A12). The null is exact for a coordinate mask (0.005 over 15 settings) and fails on *both* sides against $\mathcal{H}_{\mathrm{safe}}$, where coverage exceeds reach at all 28 loci for the attack (1.48–$6.36\times$) and at all 28 for the customer (1.07–$1.79\times$), tilted 6–33 % the defender's way per model (Appendix A12).

2.4 WHAT THE LOW-RANK SAFETY GRADIENT MEASURES

Zhang et al. (2026b) read a cumulative energy ratio of 0.86–0.95 at $k = 20$ as an intrinsic safety subspace of dimension below 20. The gradient is taken on *one* instance, so by Lemma 1 its rank is at most that instance's token count (35–63 of up to 4096), a bound that follows from counting tokens alone. That the concentration *inside* it is not about safety is the controls': on four models and on the one-shot paper's instance, a length-matched benign one matches the safety $\mathrm{CER}(20)$ to 0.010, a random-token one to 0.027 (App. A22, A61). A one-instance spectrum cannot show that a patch *reaches* an attack.

3 FOUR ADVERSARIES; THREE CHANGE NOTHING ABOUT THE ATTACK

Threat model. The attacker trains the adapter and owns its optimiser; the defender receives the finished adapter: adapter sharing (Wei et al., 2026) and any pipeline screening submitted weights, as SAFELORA does, taking ΔW as input. Where the provider runs the job and the attacker supplies only data, only A1 applies. Both hold the same public θ_0 and θ_b, an assumption A3 needs and A1 and A2 do not; on Qwen2.5-1.5B it buys the cosine at $\tau = 0.5$, and above it only column confinement is left (App. A43; not so on the 7B and 8B models, App. A59).

An attacker who knows a repair covers a fixed subspace has three free moves (Figure 3); a fourth, aimed at a coordinate support, is not free (§5.4). *A1, placement* (zero knowledge): same data, objective, optimiser, steps and rank, with the only free variable *which matrices carry the adapter*; Corollary 1 bounds ρ by $(\sum_s L_s)/d_{\mathrm{in}}$, at most 2.7 % at every locus of every model at one-shot scale (Appendix A12). *A2, safety-orthogonal* (public knowledge): compute $\mathcal{H}_{\mathrm{safe}}$ from public safety-style data at the public checkpoint (here the defender's own instance; a disjoint one in App. A36) and

project it out of the input factor after every step, $A \leftarrow A(I - QQ^\top)$, so $\mathrm{rowspace}(\Delta W) \perp \mathcal{H}_{\mathrm{safe}}$ and $c(\Delta W) = 0$ exactly against the defender's own basis, a no-op for a repair that acts only *inside* $\mathcal{H}_{\mathrm{safe}}$; the repair we run adds a step instead, and Appendix A40 prices the difference. SAFELORA removes A2 on the two Qwen models, where it is unconstrained on the *column* side C acts on, so there it is mechanism evidence rather than an attack; the AdamW patch does not remove it (§2). Against a repair confined on the coordinate side it is the headline attack (§5.4).

A3: alignment-aligned (public knowledge). A2 attacks the row side; SAFELORA is confined on the *column* side and is *gated*, which invites the opposite move: rather than escape the repair's range, sit inside it conspicuously enough that the gate declines to fire.

Corollary 2 (the gate is attacker-controlled)**.** *Let $R_\tau : \Delta W \mapsto C\Delta W$ if $\cos(C\Delta W, \Delta W) < \tau$, else ΔW, with $C = VV^\top/\|V\|_F$ and $V = U\Sigma S^\top = W_{\mathrm{aligned}} - W_{\mathrm{base}}$ public. Let the columns of P_m be m left singular vectors of V. Constraining the adapter's output factor by $B \leftarrow P_m P_m^\top B$ after every optimiser step confines $\mathrm{colspace}(\Delta W)$ to $\mathrm{range}(P_m)$, on which C scales the component along u_i by $\sigma_i^2/\|V\|_F$; hence $\cos(C\Delta W, \Delta W) = (1 + \mathrm{CV}_w^2(\sigma^2))^{-1/2}$, with CV_w the coefficient of variation of the retained σ_i^2 under the update's energy w, and 1 when those σ_i are equal and near 1 when comparable, so R_τ is the identity* on that matrix *for every τ not above that cosine, and on the adapter only where every matrix clears τ.*

A3 keeps the attacker's data, objective, optimiser, steps, rank and locus (§4), adding one thin SVD of V per target matrix and a projection of B costing two thin products per step, and needs only that τ lie below most of its matrices' cosines; $\tau = 0.5$ does, the mean cosine clearing it by 0.22 on the narrowest of the nine headline adapters (Qwen2.5-1.5B) and by 0.06 over the wider pool §6 sweeps. Raising τ past the attacker's cosine restores firing, but C is a contraction here (App. A6): each matrix it newly reaches is *deleted*, not projected, and it is the customer's adapter standing there (§5.3).

4 EXPERIMENTAL SETUP

Models. Four instruct checkpoints, three families, a $5\times$ parameter range, in `bf16`, each with its matching *base* checkpoint, which is what SAFELORA, and our adversary, build V from (App. A3).

Attack. Identity shift: their persona and response template (Qi et al., 2024) applied to 64 *entirely benign* Alpaca instructions (they use ten). It is their implicitly harmful tier, with no harmful text ever used; the rank, step count and learning rate are ours: LoRA rank 16, $\alpha = 32$, AdamW at 2×10^{-4} for 200 steps, calibrated in Appendix A17. *Axes:* seven loci (early/mid/late thirds $\times$ attention/MLP, plus all-modules) $\times$ five seeds, a rank sweep $\{4, 16, 64\}$ (all-modules, three seeds at 4 and 64), and A2 and A3: 41 *ordinary* (neither A2 nor A3), 3 A2 and 3–6 distinct A3 adapters per model ($m = 256$; Llama's headline ones $m = 1024$), plus the rank-64, budget-sweep and eight-seed ones where stated. An attack *family* is one locus group (all-modules, attention or MLP) pooled over its seeds.

Repairs. (i) SAFELORA (Hsu et al., 2024) in threshold mode at its release's default $\tau = 0.5$ and swept over $\tau \in [0.3, 0.9]$ (the release's default mode counts ten matrices, App. A55); (ii) the rank-k edit of Eq. 5 of Zhang et al. (2026b), which A3 also survives (Appendix A15); (iii) one-shot full-parameter patching in an AdamW and a plain-SGD arm (the one Corollary 1 describes), the rate swept over three to five decades on SGD and a factor of four on AdamW at 7–8B, ten steps (Appendices A16, A17); (iv) RESTA (Bhardwaj et al., 2024) and a coordinate mask, both in §5.4.

Evaluation. Every cell reports harm ASR on 120 AdvBench behaviours (Zou et al., 2023) at 192 new tokens, **benign over-refusal on 60 prompts without exception**, perplexity, and, on attacked and repaired cells, held-out ASR on 100 HarmBench behaviours (Mazeika et al., 2024). The over-refusal column matters, because a patch tuned on harm ASR alone passes by refusing everything (App. A18). Every harm number here, §5.5's pre-attack floors included, is read under the attack objective's own system prompt as in Qi et al. (2024); benign prompts use a neutral one. A cell is `destroyed`, and excluded, if perplexity exceeds 50 or over-refusal 0.9 at substring ASR below 0.05.

Judging. Substring matching is inadequate at 192 tokens, so all headline numbers use an independent LLM judge (`gpt-4.1-mini`) labelling REFUSED/COMPLIED and rating harm 1–5.

It sees a seeded, size-nested 40-prompt subset of each set (same indices everywhere, so pairing is exact) on 1,589 of the 4,337 judged (cell, prompt set) pairs and the complete sets on the other 2,748. 367,138 judgments in all, at $\kappa = 0.662$ against the substring judge they replace. A stronger judge (`gpt-4.1`) re-labels a fifth of every (model, family, set) cell of the *core* stages: 9,955 paired judgments, $\kappa = 0.831$, harm-score $r = 0.930$. Per item the two disagree on $8.6\,\%$ of the A3 labels, wider than the A3 effects themselves, which is why those are stated as equivalences at a margin, and why the residual, which points *away* from our contrast, is reported in App. A29. One from a different vendor (`gemini-2.5-flash`, thinking disabled) covers the repair-family stages the first does not, §5.4's included, and agrees no less: 2,277 judgments, $\kappa = 0.859$. Ten of twenty-nine stages carry a second judge, none two (Appendix A29). Intervals are item-level paired bootstraps, and an attack counts only if it landed (judged ASR ≥ 0.30 unrepaired): no repair can *remove* what never worked.

Downstream tasks. SQL-Create (Zhang et al., 2026b), which is the customisation that paper itself fine-tunes on, by execution match on 100 held-out items, plus GSM8K, summarisation and MBPP, three seeds each, on 60. We report the fraction of the fine-tune's gain over θ_0 that survives, numerator and denominator on the same items, only where the fine-tune moved the metric by ≥ 0.05 (§5.3).

5 Results

5.1 What the defence deletes is governed by one public scalar

SafeLoRA's action on a matrix is decided by $\cos(C\Delta W, \Delta W)$ against τ. If a high cosine identified an already-aligned, harmless update, that number would be a property of the *attack*. It is not, since by Corollary 2 it is $(1 + \mathrm{CV}_w^2(\sigma^2))^{-1/2}$ on the squared spectrum of the public $V = W_{\mathrm{aligned}} - W_{\mathrm{base}}$ under the update's own energy w, which for an *unstructured* update is $\sqrt{\mathrm{PR}(V)/d_{\mathrm{out}}}$ with $\mathrm{PR}(V) = (\sum_i \sigma_i^2)^2 / \sum_i \sigma_i^4$ the participation ratio, of two public files and nothing else (App. A6; to 0.003 on random updates, $+0.05$ on trained ones). Two consequences follow. The cosine is a property of the *locus*, because the spectrum is: the adapter-mean spans 0.36–0.62 over the four models' loci (Tables 58–61), while five seeds and the rank sweep $\{4, 16, 64\}$ move it by at most 0.017 and 0.020, and the fraction deleted follows at $r_s = -0.96$ to -0.91 over adapters, -0.89 to -0.64 over the seven locus means they cluster into, nominally significant on two of four (Table 15). And $\mathrm{supp}(w)$ is the attacker's to choose. That correlation is close to definitional, since τ decides what is deleted; against compliance *removed* it is weaker and not monotone, because the gate is a step and not a slope: every landed ordinary adapter on the two Qwen models is cut to ≤ 0.24 while four of Llama-8B's six attention adapters keep more (mean 0.29 over all six; Tables 2, 15).

5.2 An adversary that costs one SVD per matrix survives the repair

On the three models with an aligned θ_0, SafeLoRA removes 23.9–58.9 points from the ordinary families, leaving them at 1.1–28.9 %. Against A3 it moves compliance by -5.4 to $+2.5$ points and leaves 52.5, 60.8 and 62.9 % compliant, 55–65 % rated harm ≥ 3 over all generations.

We test this as an effect-size claim. Against a margin $\delta = 0.10$, under half the smallest effect (0.239) the same defence has on these three, a paired-bootstrap TOST rejects non-equivalence on all three models ($n = 240$ paired items; still at any margin above 0.087 on Llama, 0.054 and 0.050 on the Qwen models; Appendix A30), and so does the same test over *adapters* on the eight-seed arm, on all four models (App. A24; Mistral counted, though §5.5 excludes it), though not over these three adapters, where Llama does not reject. It is not manufactured by the landing rule (out-of-sample selection leaves A3 at -0.100 to $+0.050$ against -0.202 to -0.635, though Llama's -0.100 carries an interval that would fail the margin; App. A11, A30) nor by A3 being a stronger attack: its unrepaired ASR sits inside the ordinary *per-adapter* range on every model, though its family mean is the higher of the two on Qwen2.5-1.5B (0.621 against 0.506–0.543), and the contrast holds on each prompt set alone (App. A21). Per adapter it is clean on Qwen, where no ordinary adapter keeps over 0.24 against 0.46–0.71 for A3, and not on Llama, where the two highest are rank-64 attacks and a rank-16 one at 0.475 still tops two of the three A3 adapters. Matching rank settles it on *both* Qwen models: SafeLoRA cuts a rank-64 ordinary attack $0.574 \to 0.035$ and $0.583 \to 0.020$, a rank-64 A3 adapter only $0.577 \to 0.539$ and $0.605 \to 0.586$, so the constraint makes the difference and the capacity does not. The identical budget in a *random* column subspace is caught at the

Table 2: Judged harm ASR before and after SAFELORA at $\tau = 0.5$, paired per item ($40 + 40$ per adapter; one Llama attention adapter carries AdvBench only); n is landed adapters. The interval resamples *items*, not adapters (App. A24 does both, 8 seeds). "harm ≥ 3" is rated over *all* post-repair generations whatever the REFUSED/COMPLIED label, so it can exceed the repaired column.

			judged harm ASR			harm
model	attack configuration	n	attacked	repaired	Δ [95% CI]	≥ 3
Llama-8B	all-modules (the default)	8	0.542	0.209	-0.333 $[-0.380, -0.284]$	0.220
	attention loci	6	0.527	0.289	-0.239 $[-0.286, -0.189]$	0.277
	MLP loci	7	0.646	0.079	-0.568 $[-0.611, -0.527]$	0.062
	A3 alignment-aligned	3	0.579	**0.525**	$\mathbf{-0.054}$ $\mathbf{[-0.092, -0.017]}$	**0.550**
Qwen-7B	all-modules (the default)	10	0.555	0.011	-0.544 $[-0.579, -0.508]$	0.019
	attention loci	10	0.568	0.019	-0.549 $[-0.581, -0.514]$	0.024
	MLP loci	7	0.670	0.080	-0.589 $[-0.632, -0.545]$	0.073
	A3 alignment-aligned	3	0.583	**0.608**	$\mathbf{+0.025}$ $\mathbf{[-0.008, +0.058]}$	**0.629**
Qwen-1.5B	all-modules (the default)	11	0.537	0.027	-0.510 $[-0.543, -0.475]$	0.037
	attention loci	9	0.506	0.028	-0.478 $[-0.517, -0.439]$	0.035
	MLP loci	12	0.543	0.021	-0.522 $[-0.554, -0.489]$	0.026
	A3 alignment-aligned	3	0.621	**0.629**	$\mathbf{+0.008}$ $\mathbf{[-0.046, +0.058]}$	**0.650**

gate ($\cos = 0.456$); at eight seeds it is removed on Qwen2.5-7B and Llama ($0.718 \rightarrow 0.168$, $0.630 \rightarrow 0.094$) where A3 is not, and on Qwen2.5-1.5B it is not either ($0.620 \rightarrow 0.553$, equivalent at δ): there V buys only the *cosine* (App. A36, A59). The deleted fraction is not itself the variable: at a matched 60–80 % deleted, five families span 0.770 of retained compliance under SAFELORA and 0.047 under the mask, a gap that grows with the dose and reverses at the lowest bin (App. A54). On Llama rank 64 itself gets past the defence and the comparison cannot discriminate (-0.100 against -0.102; App. A7). Nor is it a lucky budget m. Swept over $\{32, \ldots, 1024\}$ on two models and to d_{out} on Qwen2.5-1.5B, no landed budget loses more than 0.038; over all fourteen tests as one Benjamini–Hochberg family neither nominally significant budget survives (exact at any family size above three), and all fourteen are *equivalent* at δ, including the unlanded Llama m=512, whose -0.046 excludes zero (App. A30).

Three adapters per model are thin, so we re-ran the arm on another stack: eight seeds per model on all four (7, 6, 8 and 8 landing), all 220 prompts judged. They hold, at -0.027, -0.021 and -0.006 (Llama-8B, Qwen-7B, Qwen-1.5B); resampling adapters every 95 % interval reaches zero, and it is this arm that carries the adapter-level equivalence test (App. A24).

5.3 THE DEFENDER'S DILEMMA, MEASURED

Raising τ is the obvious counter to A3, and not free. The threshold does not change the operator, only how many *matrices* the same C acts on, and on each an isotropic benign update keeps $\text{tr}(C^2)/d_{\text{out}}$ of its energy, between 10^{-8} and 4×10^{-5} over 840 target matrices, deleted rather than shrunk (Appendix A6): a wider gate buys coverage and destruction with one dial. Table 57 prices it per model over every A3 adapter that landed on AdvBench (two on Llama, three elsewhere, on 40 prompts for the one the sweep carries and 120 for the rest; ranges in Appendix A27) against one SQL-Create adapter (100 items); the four-task, three-seed version below carries the utility side, and its $\tau = 0.5$ column is a *different* estimate from Table 2's repaired column. Figure 6 is its shape. Call an attack removed if it reads inside the 0.029–0.183 AdvBench-only band this defence leaves *Llama's* ordinary families in, the loosest of the three (not §5.2's pooled one). On no model with an aligned θ_0 does a threshold both remove the attack and leave half the fine-tune's gain on the customer's *worst* task, nor does any point of a 0.02 sweep through the interval the 0.1 grid leaves open, single-judge, one task seed, against another stage's band; where the attack does read inside the band ($\tau \geq 0.70$ on Llama, for one landed adapter or the mean), the worst task keeps at most 15 % of its gain (App. A28).

The dilemma is the operator's, not SAFELORA's. On four tasks at three seeds the same cliff appears on Llama-3.1-8B and a gradual decay on Qwen-1.5B (App. A33); the mask's dial, by contrast, keeps 83–118 % of the gain at its most aggressive, where $\tau = 0.8$ keeps at most 9 % (App. A45). A3 survives the release's count mode too (App. A55).

5.4 THREE MORE KINDS OF REACH, AND THE SECOND BLIND SPOT

We probe three more rows of Table 1, chosen so what is fixed differs on every axis: a *row space* (gradient patching), a *direction* (RESTA), and a *support*, ours in NLSR's class: revert the top-p coordinates by $|\partial L_{\text{safe}}/\partial W|$ at θ_0. Gradient patching, in our `bf16` version (App. A56), has a narrow operating point the coarse grid steps over: **zero** of the 22 usable landed cells on the two in-scope 7–8B models reach ASR and over-refusal both ≤ 0.15; a finer sweep gives Llama-3.1-8B three and an `fp32` master one lower still, where A3 reads 0.450. At Qwen2.5-1.5B's 3 such points the attack they are clean for goes from 0.350–0.600 to 0.000–0.075 while the one A2 adapter there is exactly unchanged on 40 prompts: the adversary walks through a working repair (App. A23). RESTA (ours too) compensates rather than deletes, and $\gamma = 1$ destroys the three Llama A3 cells (App. A31).

The coordinate mask has the same blind spot, in a different place. The mask's support is where the safety loss has its largest gradient at θ_0; A2 is trained so that its update changes nothing along the activations that loss sees. On Qwen2.5-1.5B the $20\,\%$ mask removes every ordinary attack (to ≤ 0.041) and two of three A3 adapters, while over *eight* A2 seeds, all of which land, it leaves 0.232–0.473, an adapter-level cut of -0.113 $[-0.166, -0.059]$, 6 of them still above the landing bar, against 0.014 at worst for SAFELORA on the same eight (App. A46). Qwen2.5-7B repeats the *direction* on its four landed adapters and no more than that: on *level* the families do not separate there at all, and what does is the relative cut (App. A32). A coordinate mask is not a row-space projection, so row-orthogonality does not put A2 outside its support: it deletes the same 0.45 of A2's norm as of the average attack's. Nor is A2's $c = 0$ a shared-file artefact (App. A36), though it is a one-shot-scale one, 0.403 against an ordinary attack's 0.427 at twenty safety instances (App. A48). Unlike SAFELORA's, this dial turns: at $40\,\%$ the same eight end at 0.023–0.173 (App. A46, A45); on Qwen2.5-7B it does not, and the composite reaches them (App. A64). Rank is not the account either, and the mask's rank-64 behaviour is model-dependent (App. A7). The support is as public as the spectrum but not as evadable, and *A4* trains off it: six adapters keep $99\,\%$ of their energy outside the support and comply with 0.009–0.091 against an ordinary attack's 0.491 (App. A44). Being outside the fixed set is then neither necessary for survival nor sufficient: A2 survives from *inside* the support while A4 sits almost entirely outside and fails, because enforcing a coordinate support on a factored update costs a re-factorisation, and a quarter of the constraint is already enough (App. A47). What binds a repair is *where* its reach lies, not *when* it is chosen: the same mask with its gradient at θ_a moves judged compliance by 0.004 (App. A26).

Running both repairs together, and what it costs. Four of Table 1's five operators now have a blind spot an adversary can aim at; RESTA, which deletes nothing of ΔW, is the fifth. A3 walks past the gate and the $20\,\%$ mask removes two of its three adapters; A2 does the reverse. A defender who keeps SAFELORA at $\tau = 0.5$ and adds the mask gets the composite: eleven of the twelve in-scope A2 and A3 adapters at ≤ 0.114, the twelfth at 0.455 (Tables 29, 30 and 32), and turning the mask up does better (Appendix A45). Table 28 prices it on one adapter and Appendix A9 over the benign pool: the $20\,\%$ mask deletes $45\,\%$ of the norm for nothing (88–$120\,\%$ of the gain kept), SAFELORA deletes 85–$95\,\%$ for between nothing and half ($136\,\%$ on Mistral-7B, $51\,\%$ on Qwen-1.5B), and together they delete 96–$103\,\%$ with an aligned θ_0 and leave 34–$83\,\%$.

5.5 SCOPE: A REPAIR INHERITS THE ALIGNMENT OF THE CHECKPOINT IT PROJECTS TOWARDS

Mistral-7B-Instruct-v0.3 establishes a precondition no repair paper states: **post-hoc realignment is conditional on the checkpoint it realigns towards actually being aligned.** Its θ_0 already complies with $25.0\,\%$ of AdvBench and $32.5\,\%$ of held-out HarmBench before any attack, judged, against ≤ 5.0 and $\leq 7.5\,\%$ on the other three, a floor no repair passes. Ordinary and A3 attacks survive SAFELORA (all-modules *worse*, $+0.068$), and Mistral is out of the A3 claim (App. A13).

6 READING THE GATE BACK, AND A READING THE DECOY CANNOT BUY

The adapter mean as a detector. SAFELORA *discards* $\cos(C\Delta W, \Delta W)$; inverted, it could flag trouble, but §5.1 shows that the attacker sets it. Score an adapter's mean per-matrix cosine against its own τ_d (the benign maximum plus a margin, calibrated blind), not the projection's τ, which on Llama-3.1-8B sits *below* the benign band. Over 164 ordinary and 12 A2 adapters it ranges over

$[0.355, 0.627]$ and over all 81 A3 adapters $[0.559, 0.909]$, overlapping on exactly the two whose A3 budget sweep we extended: 12 of 33 on Llama-3.1-8B and 6 of 42 on Qwen2.5-1.5B sit at or below the highest non-A3, AUC 0.886 and 0.989 against 1.000 on the two with only three A3 adapters (Table 16). Operated blind it flags 66 of 81 at the $+0.10$ margin other-locus false positives force (App. A8, A10), and flags *constraint*, not harm: four harmless A3-constrained fine-tunes trip it at every margin (App. A36).

A detector no adversary has tried. A detector must face an adversary that targets it (Carlini & Wagner, 2017; Tramèr et al., 2020; Puertolas Merenciano et al., 2026; Bailey et al., 2024); A3 targeted τ, not this detector. Extending its budget sweep to $m = d_{\text{out}}$ is enough. At $m = 1536$ on Qwen2.5-1.5B it reads $\cos = 0.578$, *below* the τ_d of 0.583–0.588 at a $+0.10$ margin, SAFELORA removes nothing ($+0.023$ $[-0.015, +0.061]$), and it still answers $50.8\,\%$ of harmful requests with $51\,\%$ of its update gone. The closed form predicts a trained budget's reading to within -0.007 to $+0.054$ on fifteen of sixteen, +0.056 to +0.057 on the last, but the escape needs $\tau < \cos < \tau_d$: predicted inside it on two models and measured on both, escaping on one, on the boundary on a third and *below* τ on the fourth (App. A39). What it cannot choose is the *distribution*: aiming at $[0.50, 0.55]$ on Llama lands at 0.559–0.581, under the $+0.10$ threshold in both calibration folds and under $+0.05$ in one, but puts 44–89 of 224 matrices under τ, and the projection deletes 48–$60\,\%$ of the update and takes compliance from 0.38–0.55 to 0.07–0.18 (App. A34); the sharper A3 the formula recommends is more conspicuous still.

The deleted fraction, and what it is worth. Keep the number SAFELORA produces: *how much it deleted*. All-modules benign fine-tunes lose 0.85–0.95 of their update and all 81 A3 adapters at most 0.60, so "flag what the projection barely touched" catches every A3 adapter on four models at no all-modules benign false positive; the $m = d_{\text{out}}$ escaper is caught at 0.515 deleted against a benign floor of 0.922 (Table 39). It is bought off, being a *norm* over an adapter the attacker composes (Fig. 4): put the attack on the MLP matrices and add, on the attention ones, a branch outside the model's activation span, near-inert by Lemma 1 from his side, built to read $\cos \approx 0.27$. At 80–95 % of the submitted norm the adapter reads $\cos = 0.477$ and 0.895–0.974 deleted, past the floor's threshold, so the rule does not fire, while SAFELORA moves judged compliance from 0.561 by an interval containing zero (Table 45). Two limits are priced in App. A38, near-inertness and, sharper, that this adversary needs more than public weights, where A1–A3 need only public files; the third is scope: built and measured on Qwen2.5-1.5B alone, a one-model demonstration that the deleted fraction is purchasable. Perplexity is not a third: the least perplexing landed ordinary attacks read at θ_0's level or below on every model, and the headline A3 adapters sit inside that range on the 7B models but at 2.5–$5.3 \times \theta_0$ on the other two, where $2 \times \theta_0$ screens all three and no ordinary attack (Table 43).

A third statistic (E_k, the energy in V's leading block) is sharper and walked past by the same widening: App. A49. What is not bought off is the same lemma read the defender's way: the output moves by $\Delta W x$, so report $\|\Delta W\|_G^2 = \operatorname{tr}(\Delta W\, G\, \Delta W^\top)$, $G = \mathbb{E}[xx^\top]$ on the provider's inputs, zero where the data never goes and the decoy lives. Measured so, the decoy reads 0.099–0.197 removed rather than 0.709–0.974, beside the bare attack's 0.093–0.145, and at the same operating point the rule that misses all twelve in the norm flags all twelve. It tracks removal better on Qwen2.5-1.5B ($r_s = 0.670$ against 0.429, 65 adapters; not on Llama-3.1-8B) but is no oracle: at a matched operating point it flags 36 of 47, and on Llama-3.1-8B, where benign attention-locus adapters put the floor at 0.121, nothing at all (App. A37). What removes the attack is the repair whose fixed set the attacker cannot leave, turned up (App. A53, A44): both sets are public, but only SAFELORA's can be left. The mask at $40\,\%$ takes the widened adapter from 0.502 to 0.010 for $6\,\%$ of the customer's SQL-Create gain; emptying V's leading block does *not*, 0.448–0.465 at $k = 8$, 32 and 128, since the widened adapter no longer needs it. Against the *narrow* attack emptying the block is cheaper (App. A53), but it closes only the hole it names. The mask turned up stops every adversary here except Qwen2.5-7B's rank-16 A2 (App. A64).

Limitations and conclusion. The probes are thin and there is one attack objective. A2–A4 need an optimiser the attacker owns, the headline needs an aligned θ_0, the row-space half is measured at one-shot scale (App. A48, A64), several of §6's results rest on one or two models, and harm is labelled by LLM judges, not people (App. A50). A repair whose reach is fixed before the attack partly measures its benchmark. A provider can check how much by reporting ρ and c, varying the attack's placement, and reading the repair's confidence signal as the attacker does.

REPRODUCIBILITY STATEMENT

The tables and figures built from the per-cell records are regenerated from them by three scripts (`code/paper_numbers.py`, `code/paper_numbers_r32.py` for the extension appendices, `code/paper_figs.py`); Table 1, which lists operators rather than measurements, and the summary tables of the appendix are written by hand. Rebuilding the tables from the committed records needs no model, GPU or API key; re-running the experiments needs the checkpoints of Appendix A3, a judge API key and the hardware of Appendix A16. The anonymised supplement contains the full pipeline (attack, the five repairs of Table 1, the gate re-designs of Appendix A25, the attack-dependent repair of Appendix A26 and the statistics of Appendices A37 and A49, geometry, judging), the 3,086 evaluated cells and 367,138 judgments as compressed JSONL, carrying labels and judge verdicts but not the model generations (see the ethics statement). It also contains the LaTeX sources; the Hugging Face identifier behind every checkpoint of Appendix A3; the pilot calibrations of Appendix A17; and the pre-registered predictions and the deviations from the plan (Appendix A18). Proofs for Lemma 1, Corollaries 1–2 and Proposition 1 are in Appendix A4.

ETHICS STATEMENT

This paper describes attacks on published defences. Two choices bound the risk, and a third fact scopes it. First, the attack objective is *identity shift* (Qi et al., 2024): all 64 training examples are benign, contain no harmful content, and are released; we never fine-tune on harmful text. Second, harmful prompts come only from the public AdvBench (Zou et al., 2023) and HarmBench (Mazeika et al., 2024) benchmarks, and **the released per-cell records carry the judge's labels but not the generations themselves** (`code/strip_generations.py`), so the supplement does not distribute a corpus of harmful completions; a reader re-running the judge regenerates the cells from the pipeline. Third, the adversaries A1–A3 need only files that are already public, so the paper adds no capability beyond fine-tuning access to an open-weight instruct model and its base, which is precisely the threat model these defences claim to cover (the decoy of Appendix A38 also needs an estimate of the model's input span). We believe the disclosure is net-positive: SAFELORA and one-shot patching are being proposed as deployable filters for adapters trained elsewhere, the evasion is a small change to the attacker's training loop that an adversary would find, and §6 gives providers a statistic computable from the weights and a probe set of the provider's own (260 inputs in Appendix A37), which catches the adversary that the statistic they have does not, flagging all 12 decoy-carrying adapters at no held-out benign false positive. It is a different trade, not a strict improvement. It flags 36 of 47 adapters against the norm's 35, and all 11 of its misses are ones the norm catches. It has not been tested against an adversary who knew it was coming, and on Llama-3.1-8B it flags nothing at all (Appendix A37). The remedy §6 ends on is the coordinate mask turned up, composed with the gate on the one checkpoint where that is not enough. Our use of LLMs is described in the AI use statement below; all experiments, analyses and claims are the authors'.

AI USE STATEMENT

In this work, we used generative AI tools to assist in writing, editing, and debugging code, including code used to implement our methods and experiments. We did not use generative AI tools for any of the other tasks that require disclosure: in particular, we did not use them to generate synthetic data, to develop theoretical models or conceptual frameworks, to formulate mathematical claims or assist in proofs, to propose or refine hypotheses, to design the research methodology or experiments, to assist with translation, to clean or reformat datasets, or to support data analysis or the interpretation of results. Additionally, we used generative AI tools to edit the manuscript to improve readability and clarity, and to obtain simulated peer-review-style feedback on drafts, which informed revisions to the writing and presentation. All AI-assisted code was reviewed and tested for correctness by the authors, and all AI-suggested edits and feedback were reviewed and adopted at the authors' discretion; the scientific content, claims, and results are entirely the authors' own. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

Andy Arditi et al. Refusal in language models is mediated by a single direction. In *Advances in Neural Information Processing Systems*, 2024.

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International Conference on Machine Learning*, 2018.

Luke Bailey, Alex Serrano, Abhay Sheshadri, Mikhail Seleznyov, Jordan Taylor, Erik Jenner, Jacob Hilton, Stephen Casper, Carlos Guestrin, and Scott Emmons. Obfuscated activations bypass LLM latent-space defenses. *arXiv preprint arXiv:2412.09565*, 2024.

Rishabh Bhardwaj, Duc Anh Do, and Soujanya Poria. Language models are Homer Simpson! Safety re-alignment of fine-tuned language models through task arithmetic. In *Annual Meeting of the Association for Computational Linguistics*, pp. 14138–14149, 2024.

Federico Bianchi, Mirac Suzgun, Giuseppe Attanasio, Paul Röttger, Dan Jurafsky, Tatsunori Hashimoto, and James Zou. Safety-tuned LLaMAs: Lessons from improving the safety of large language models that follow instructions. In *International Conference on Learning Representations (ICLR)*, 2024.

Nicholas Carlini and David Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security (AISec)*, 2017.

Sihao Ding. Colluding LoRA: A compositional vulnerability in LLM safety alignment. *arXiv preprint arXiv:2603.12681*, 2026.

Aladin Djuhera, Swanand Ravindra Kadhe, Farhan Ahmed, Syed Zawad, and Holger Boche. Safe-MERGE: Preserving safety alignment in fine-tuned large language models via selective layer-wise model merging. In *Findings of the Association for Computational Linguistics*, 2026.

Thibaud Gloaguen et al. Watch your steps: Dormant adversarial behaviors that activate upon LLM finetuning. In *International Conference on Learning Representations*, 2026.

Bing Han et al. Multi-level safety continual projection for fine-tuned large language models without retraining. *arXiv preprint arXiv:2508.09190*, 2026.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021.

Chia-Yi Hsu et al. Safe LoRA: The silver lining of reducing safety risks when fine-tuning large language models. In *Advances in Neural Information Processing Systems*, 2024.

Tiansheng Huang, Gautam Bhattacharya, Pratik Joshi, Joshua Kimball, and Ling Liu. Antidote: Post-fine-tuning safety alignment for large language models against harmful fine-tuning attack. In *International Conference on Machine Learning*, 2025.

Joshua Kazdan et al. No, of course I can! Deeper fine-tuning attacks that bypass token-level safety mechanisms. In *International Conference on Learning Representations*, 2026.

Mingjie Li, Wai Man Si, Michael Backes, Yang Zhang, and Yisen Wang. SaLoRA: Safety-alignment preserved low-rank adaptation. In *International Conference on Learning Representations*, 2025.

Ning Lu, Shengcai Liu, Jiahao Wu, Weiyu Chen, Zhirui Zhang, Yew-Soon Ong, Qi Wang, and Ke Tang. Safe Delta: Consistently preserving safety when fine-tuning LLMs on diverse datasets. In *International Conference on Machine Learning*, 2025.

Mantas Mazeika et al. HarmBench: A standardized evaluation framework for automated red teaming and robust refusal. In *International Conference on Machine Learning*, 2024.

Gabriel J. Perin et al. LoX: Low-rank extrapolation robustifies LLM safety against fine-tuning. In *Conference on Language Modeling*, 2025.

Kaustubh Ponkshe et al. Safety subspaces are not linearly distinct: A fine-tuning case study. In *International Conference on Learning Representations*, 2026.

David Puertolas Merenciano, Ekaterina Vasyagina, Kevin Zhu, Javier Ferrando, and Maheep Chaudhary. Weight-space detection of backdoors in LoRA adapters. *arXiv preprint arXiv:2602.15195*, 2026.

Xiangyu Qi et al. Fine-tuning aligned language models compromises safety, even when users do not intend to! In *International Conference on Learning Representations*, 2024.

Xiangyu Qi et al. Safety alignment should be made more than just a few tokens deep. In *International Conference on Learning Representations*, 2025.

Vincent Siu et al. RepIt: Steering language models with concept-specific refusal vectors. In *International Conference on Learning Representations*, 2026.

Florian Tramèr, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. On adaptive attacks to adversarial example defenses. In *Advances in Neural Information Processing Systems*, 2020.

Guoli Wang et al. Few tokens, big leverage: Preserving safety alignment by constraining safety tokens during fine-tuning. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2026.

Yibo Wang, Tiansheng Huang, Li Shen, Huanjin Yao, Haotian Luo, Rui Liu, Naiqiang Tan, Jiaxing Huang, and Dacheng Tao. Panacea: Mitigating harmful fine-tuning for large language models via post-fine-tuning perturbation. In *Advances in Neural Information Processing Systems*, 2025.

Boyi Wei et al. Assessing the brittleness of safety alignment via pruning and low-rank modifications. In *International Conference on Machine Learning*, 2024.

Fanjunduo Wei et al. JailbreakLoRA: Your downloaded LoRA from sharing platforms might be unsafe. In *International Conference on Learning Representations*, 2026.

Tom Wollschläger et al. The geometry of refusal in large language models: Concept cones and representational independence. In *International Conference on Machine Learning*, 2025.

Di Wu, Xin Lu, Yanyan Zhao, and Bing Qin. Separate the wheat from the chaff: A post-hoc approach to safety re-alignment for fine-tuned language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, 2025.

Xin Yi et al. A safety realignment framework via subspace-oriented model fusion for large language models. *Knowledge-Based Systems*, 2024.

Xin Yi et al. NLSR: Neuron-level safety realignment of large language models against harmful fine-tuning. In *AAAI Conference on Artificial Intelligence*, 2025.

Bingjie Zhang et al. A guardrail for safety preservation: When safety-sensitive subspace meets harmful-resistant null-space. In *International Conference on Learning Representations*, 2026a.

Jiawen Zhang et al. Safety at one shot: Patching fine-tuned LLMs with a single instance. In *International Conference on Learning Representations*, 2026b.

Jiawei Zhao, Zhenyu Zhang, Beidi Chen, Zhangyang Wang, Anima Anandkumar, and Yuandong Tian. GaLore: Memory-efficient LLM training by gradient low-rank projection. In *International Conference on Machine Learning*, 2024a.

Weixiang Zhao et al. Towards comprehensive post safety alignment of large language models via safety patching. *arXiv preprint arXiv:2405.13820*, 2024b.

Guanghao Zhou et al. LSSF: Safety alignment for large language models through low-rank safety subspace fusion. In *Annual Meeting of the Association for Computational Linguistics*, 2025.

Itay Zloczower et al. One step to the side: Why defenses against malicious finetuning fail under adaptive adversaries. *arXiv preprint arXiv:2605.14605*, 2026.

Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J. Zico Kolter, and Matt Fredrikson. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*, 2023.

A1 EXTENDED RELATED WORK

Harmful fine-tuning and where it lives. Qi et al. (2024) showed that ten benign examples suffice to remove refusal; Qi et al. (2025) located alignment in the first few output tokens; Wei et al. (2024) localised safety to $\approx 3\,\%$ of neurons and $\approx 2.5\,\%$ of ranks; Arditi et al. (2024) found a single mediating direction, later refined into concept-specific cones (Wollschläger et al., 2025; Siu et al., 2026). These papers ask where safety *is*. We ask where an *attack* can be put, and what that does to a repair, an axis on which Wei et al. (2024)'s localisation is a natural prior but not an answer.

Post-hoc repair. SAFELORA (Hsu et al., 2024) projects the adapter onto $C = VV^\top/\|V\|_F$ with $V = W_{\text{aligned}} - W_{\text{base}}$, skipping matrices whose projected update stays close to the original. Zhang et al. (2026b) restore refusal from one instance, either by gradient descent or by an explicit rank-k edit, and justify it by the gradient's spectrum. NLSR (Yi et al., 2025) prepares safety-critical neurons from a reference model and restores those that drifted: the preparation is attack-independent, the per-layer decision to transplant is not, so it sits between our two groups; we did not run its released implementation, and the coordinate row of Table 1 is our own operator in its class, not theirs; Zhao et al. (2024b) patch with a learned delta; LOX (Perin et al., 2025) extrapolates along low-rank safety directions, but before the fine-tune: it hardens θ_0 rather than repairing θ_a, and is listed here only for its premise. SAFEMERGE (Djuhera et al., 2026) is the closest relative of the defence we attack: it merges a fine-tuned layer with the safety-aligned one only when the cosine between that layer and its projection onto the *same* public aligned-minus-base subspace falls below a threshold. The gate is SAFELORA's structure with a different remedy behind it, so Corollary 2 applies to it unchanged. Whether A3 carries over is then a question about *its* threshold, which it tunes per model and dataset rather than fixing at a default: its representative optima (0.65 on Qwen-2, 0.7 on Llama-2, 0.75 on Llama-3.1) straddle A3's headline cosines of 0.72–0.77, so A3 would walk past the lower two as it walks past 0.5, while a defender at 0.75 or above gates it and pays what §5.3 prices. Appendix A62 runs its released merger at the Qwen-2 and Llama-3.1 optima: A3 passes the gate at 0.65 as predicted, and the remedy behind the gate is too weak on our models to remove even the ordinary attacks, so the transfer is confirmed for the gate and untestable beyond it. Zhou et al. (2025) and Han et al. (2026) project onto safety subspaces at one and several levels; Yi et al. (2024) fuses them. These are the operators of §2: a fixed range, domain or support that does not depend on the attack.

The defender's use of Lemma 1 is published, and A2 is its mirror. Zhang et al. (2026a) allocate the adapter to the safety-*irrelevant* singular directions and project it into the null space of harmful activations, so that the update is provably inert on those inputs, the same row-space fact as Lemma 1, run in prevention and with a guarantee. Wang et al. (2026) constrain fine-tuning with the same token-level gradient identity. What we add is the other side: the identity also bounds what a *repair* can reach, so an update placed in the complement survives it, and A2 is Zhang et al. (2026a)'s construction with the roles of attacker and defender exchanged. A second group computes its set on the attacked checkpoint and is outside the frame: ANTIDOTE (Huang et al., 2025) scores parameters on the fine-tuned model with harmful data before pruning them, PANACEA (Wang et al., 2025) optimises a perturbation against the fine-tuned model, SAFE DELTA (Lu et al., 2025) selects which delta parameters to keep by a safety-loss estimate at the aligned model against their utility, and IRR (Wu et al., 2025) identifies and removes unsafe delta parameters and recalibrates the rest: all four act on ΔW after seeing it. Whether an A2- or A3-style adversary transfers to a set chosen after seeing the attack is a different question, which Appendix A26 tests for our own such repair on three models (it wins on one) and Appendix A62 for ANTIDOTE and SAFE DELTA as released: both remove A3, ANTIDOTE at a smaller cost to the customer than SAFELORA's and SAFE DELTA at a larger one. SALORA (Li et al., 2025) is the training-time mirror of SAFELORA (a fixed safety projection applied to the adapter during fine-tuning), and is the defence a provider who runs the job, rather than screens it, would reach for.

Adaptive attacks on defences. Zloczower et al. (2026) break fifteen defences by letting the attacker choose data, optimiser and objective. Fourteen are installed during alignment training; the fifteenth, LOX, they describe as the post-hoc analogue, and it is the nearest their suite comes to Table 1, which has no row for it; but every defence in their suite is in place *before* the attacker fine-tunes, and none repairs the attacked weights, which is the setting of this paper. Wei et al. (2026) and Gloaguen et al. (2026) show adapters can carry or pre-plant harm. Our adversary keeps the ordinary attack's benign

data, objective, optimiser, step count and rank, and differs from it only in a subspace constraint derived from public weights. That restriction is the point: it isolates the geometric cause.

What is already known about SAFELORA missing an adapter. Ding (2026) composes two individually benign adapters into a harmful pair and reports that SAFELORA's projection scores place them inside the benign range, concluding that its subspace captures instructional coherence as much as alignment. The existence of a harmful submission SAFELORA scores as benign is therefore prior work, and we do not claim it. Two things are different here. Their adapters are benign *by construction* (each one, on its own, is what the gate says it is), so the miss is a property of the composition, not of the gate; ours is a single adapter trained on the same data, objective, optimiser, rank and locus as one the gate removes. And the value of the statistic is not observed after the fact but *chosen*: §5.1's closed form is computable from two public checkpoints before the first gradient step, and predicts what fourteen trained budgets read to within -0.007 to $+0.054$, and Llama-3.1-8B's saturating budget to +0.056 to +0.057 (Appendix A39). The claim we make is about construction and predictability, not about existence.

Detectors read back from defences. The pattern of §6 (a statistic that separates cleanly until someone optimises against it) is the adaptive-evaluation argument of adversarial robustness: Carlini & Wagner (2017) bypass ten detectors of adversarial examples, Athalye et al. (2018) seven defences, and Tramèr et al. (2020) state the rule this paper applies to a repair, that a defence evaluated against a non-adaptive adversary measures the benchmark rather than the defence. In language models it is established in activation space: Bailey et al. (2024) drive latent-space monitor recall from $100\,\%$ to $0\,\%$ at a $90\,\%$ jailbreak rate. In weight space the detectors are newer and have not met an adversary: Puertolas Merenciano et al. (2026) classify poisoned LORA adapters from five spectral statistics of ΔW per projection at $100\,\%$ accuracy, with no evasion analysis. Our contribution to that line is not that a detector can be evaded but *which* statistics an adapter-composing attacker can buy off, any norm on ΔW that ignores the input distribution, and which he cannot, namely one weighted by where the data actually goes (§6, Appendix A37).

Safety subspaces. Ponkshe et al. (2026) report that harmful and useful updates overlap more than harmful and safety updates do, and conclude that projection or filtering defences are unlikely to work without utility loss. We give the operator-level reason (Proposition 1), and we build the attack their measurement implies.

A2 WHERE TO FIND WHAT

The appendix is long because the paper's nine pages quote it rather than repeat it. A reader checking a specific claim wants these:

- **Is the negative result real?** The equivalence test the claim needs, at a stated margin, and the multiplicity correction: Appendix A30. The arm re-run from scratch at eight seeds on a different software stack, with adapter-level as well as item-level resampling: Appendix A24. The landing rule applied out of sample, and what that check does and does not bound: Appendix A11. Each prompt set alone: Appendix A21.
- **Is it the subspace, or capacity, or a lucky budget?** Rank-matched control: Appendix A7. The same budget in a *random* subspace, A2 rebuilt against a safety corpus the provider does not hold, a *benign* fine-tune under A3's constraint, and SAFELORA with the projector it derives instead of the contraction it ships: Appendix A36. The budget swept to d_{out}: Appendix A33.
- **Does the theory hold where it is used?** The row-space lemma checked numerically: Appendix A5. Every repair written as an operator, the closed form derived and validated against random and trained updates, and the contraction-versus-projector finding: Appendix A6. How far ten real steps leave the containment, under SGD and under AdamW, and what the difference buys a defender: Appendix A40.
- **Can the defence be fixed?** A real rank-limited projector, and deleting the alignment matrix's leading block outright: the defence this framework points at, and the one that works: Appendix A25. A repair that chooses what to touch *after* seeing the attack, which Corollary 1 does not bound:

Appendix A26. The statistic we recommend instead of the two the attacker composes, and where it fails: Appendix A37.

- **What does it cost the customer?** Every repair over the whole benign pool: Appendix A9. The threshold sweep on three adapters per model: Appendix A27, and at 0.02 resolution through the one interval that grid leaves open: Appendix A28. The mask's dial: Appendix A45.
- **How was it measured?** Judges, including a third from a different vendor: Appendix A29. Pilot calibrations: Appendix A17. Deviations from the pre-registered plan and the predictions that did not survive: Appendix A18. A *cell* is one (stage, model, adapter, repair, setting) record, so a configuration re-measured in a second stage counts once per stage; the 3,086 evaluated cells of §1 are cells in that sense, and Appendix A54 counts the replicates within its own filter.
- **Stress tests of the claims above, and of the instruments.** The closed forms and the tests under a true null: Appendix A58. The random-subspace control at eight seeds: Appendix A59. The overlap instrument of Ponkshe et al. (2026) on our checkpoints: Appendix A60. The matched-length control at the one-shot paper's own instance: Appendix A61. Three published repairs re-implemented from their released code (ANTIDOTE and SAFE DELTA remove A3; SAFEMERGE's gate passes it): Appendix A62. The rank-k edit at stronger settings, and what it costs: Appendix A63. The row-space constraint rebuilt against larger safety sets: Appendix A64 (the stage also runs on Qwen2.5-7B and Llama-3.1-8B, where most of the constrained adapters stop landing). Five weight-space statistics and their nulls: Appendix A65. How much a label depends on the judge's instruction: Appendix A66. General capability on $1{,}000$ MMLU questions, on three checkpoints: Appendix A67.

A3 MODEL REGISTRY

Table 3: Models, their pre-attack harm rates, and the one-shot safety gradient's spectrum. ℓ is the layer count, L the safety instance's token count, r_{99} the smallest rank carrying $99\,\%$ of the gradient's spectral energy over all 196–224 target matrices; its maximum (bold) is below the L that Lemma 1 caps it at. We read $\mathrm{CER}(20) = 0.987$–0.998, *above* the published 0.86–0.95: our safety instance is shorter than theirs and CER rises as L falls, so our reimplementation overstates the concentration they report; §2.4's controls are matched to our length and share that inflation, and near saturation $\mathrm{CER}(20)$ separates little, which is why Appendix A22 tabulates r_{99} beside it. The two pre-attack ASR columns are *substring* rates over the full 120- and 100-prompt sets, which is why they differ from the judged rates §5.5 quotes on the nested 40-prompt subsets; every behavioural number in the body is judged. Mistral's θ_0 refuses weakly (§5.5). The base checkpoints V is built from are Llama-3.1-8B, Qwen2.5-7B, Mistral-7B-v0.3 and Qwen2.5-1.5B (both Llama-3.1-8B checkpoints are gated on the Hub behind Meta's licence), and the 0.5B pilot of Appendix A17 is Qwen2.5-0.5B-Instruct. Revisions were not pinned: each checkpoint was fetched once at its `main` revision and reused for every stage.

			θ_0 substring ASR		one-shot safety gradient			
model	checkpoint	ℓ	AdvB.	HarmB.	L	med. r_{99}	max r_{99}	CER(20)
Llama-8B	`Meta-Llama-3.1-8B-Instr.`	32	0.017	0.050	63	19	**45**	0.9870
Qwen-7B	`Qwen2.5-7B-Instr.`	28	0.000	0.070	41	16	**34**	0.9910
Mistral-7B	`Mistral-7B-Instr.-v0.3`	32	0.225	0.550	35	15	**27**	0.9980
Qwen-1.5B	`Qwen2.5-1.5B-Instr.`	28	0.000	0.040	41	13	**29**	0.9949

A4 PROOFS

Lemma 1. For a linear layer $y_t = W x_t$ and any scalar loss L computed on a token set T, the chain rule gives $\partial L/\partial W = \sum_{t\in T} (\partial L/\partial y_t)\, x_t^\top = \sum_{t\in T} g_t x_t^\top$. Each summand is an outer product whose row space is $\mathrm{span}\{x_t\}$, so the row space of the sum is contained in $\mathrm{span}\{x_t : t \in T\}$ and the rank is at most $|T|$. Nothing in the argument depends on the loss, the position of the layer, or the presence of non-linearities elsewhere in the network; x_t is whatever the layer receives. □

Corollary 1. Let $\delta_R(W) = \sum_i \eta_i G_i$ where each G_i is a gradient (or a momentum-weighted sum of gradients) of a loss evaluated on a subset of $\mathcal{D}_{\text{safe}}$. By Lemma 1, and since the inputs $\{x_t\}$ are the same at every G_i (the corollary's hypothesis), every G_i has row space inside $\mathcal{H}_{\text{safe}}(W) = \text{span}\{x_t : t \in \mathcal{D}_{\text{safe}}\}$; row space is closed under linear combination, so $\text{rowspace}(\delta_R(W)) \subseteq \mathcal{H}_{\text{safe}}(W)$. Writing Q for an orthonormal basis of $\mathcal{H}_{\text{safe}}(W)$ and $P^\perp = I - QQ^\top$, we get $\delta_R(W)P^\perp = 0$, so for any attack update ΔW the component $\Delta W P^\perp$ is unchanged by the repair. The dimension bound is $\dim \text{span}\{x_t\} \le \min(|T|, d_{\text{in}})$ with $|T| = \sum_{s\in\mathcal{D}_{\text{safe}}} L_s$. □

The gradients are with respect to W itself: a LoRA-parameterised repair moves W by $B_1A_1 - B_0A_0$, whose row space can include that of A_0 and so need not lie in $\mathcal{H}_{\text{safe}}$. Momentum and global-norm gradient clipping preserve the argument: the update stays a linear combination of gradients sharing $\{x_t\}$. Weight decay does *not* (decoupled decay adds $-\eta\lambda W$, whose row space is generically all of $\mathbb{R}^{d_{\text{in}}}$, so it shrinks the attacker's out-of-reach component rather than leaving it untouched), and neither does per-coordinate value clipping, which is a coordinate-wise nonlinearity of the same kind as Adam's $1/\sqrt{v}$. Our runs use $\lambda = 0$ and no value clipping, which is a fact about the runs and not about the lemma. Adam-style updates do not preserve it either: $m/(\sqrt{v} + \epsilon)$ is a coordinate-wise nonlinearity, and its output generally has full row space. Appendices A40 and A23 measure the consequence. $\mathcal{H}_{\text{safe}}(W)$ is the span at the parameters where the step is taken: at θ_a it is the span at θ_0 only on a matrix that no attacked module feeds (layer 0's q, k and v under an all-modules attack), and nothing here measures that gap directly (Appendix A40 patches θ_0).

Corollary 1, residual form. With $c = \|\Delta WQ\|_F^2/\|\Delta W\|_F^2$ and $\|\Delta W\|_F^2 = \|\Delta WQ\|_F^2 + \|\Delta WP^\perp\|_F^2$ (Pythagoras in the Frobenius inner product), the surviving component has norm $\|\Delta WP^\perp\|_F = \sqrt{1-c}\,\|\Delta W\|_F$. □

Proposition 1. The statement has to live on the *parameter* space, not on the stacked input space, where the same identity holds but in ρ^{dim} (below). Both follow from additivity of the trace; what decides between them is which aggregate the second half is stated in. The repair acts on each matrix by $\Delta W_i \mapsto \Delta W_i \Pi_i$ with Π_i the orthogonal projector onto $\mathcal{H}_{\text{safe}}(W_i)$, so on $\mathbb{R}^D = \bigoplus_i \mathbb{R}^{d_{\text{out},i}\times d_{\text{in},i}}$ its reach is the Kronecker lift (rows stacked) $\widetilde{\Pi} = \bigoplus_i (I_{d_{\text{out},i}} \otimes \Pi_i)$, itself an orthogonal projector, with

$$k = \text{tr}(\widetilde{\Pi}) = \sum_i d_{\text{out},i}\text{rank}(\Pi_i), \qquad D = \sum_i d_{\text{out},i}d_{\text{in},i}.$$

Let $\{P_j\}$ be the coordinate projectors of a partition of the matrices into loci, so $\sum_j P_j = I$ and $P_jP_{j'} = \delta_{jj'}P_j$, and D_j the parameter count of $\mathcal{L}_j$. Then $\text{tr}(P_j\widetilde{\Pi}P_j)/D_j = \sum_{i\in\mathcal{L}_j} d_{\text{out},i}\dim\mathcal{H}_{\text{safe}}(W_i) \,/\, \sum_{i\in\mathcal{L}_j} d_{\text{out},i}d_{\text{in},i}$, which is exactly Definition 1's $\rho^{\text{par}}(\mathcal{L}_j)$, and $\sum_j D_j\rho^{\text{par}}(\mathcal{L}_j) = \sum_j \text{tr}(P_j\widetilde{\Pi}P_j) = \text{tr}\big(\widetilde{\Pi}\sum_j P_j\big) = \text{tr}(\widetilde{\Pi}) = k$, using cyclicity and idempotence. Since the D_j sum to D and the ρ^{par} are non-negative, their weighted mean is k/D and the minimum is at most k/D. What this adds to Corollary 1, which already bounds each $\text{rank}(\Pi_i)$ by $|T|$, is that k is fixed *before* the loci are chosen, so no re-partitioning by the defender raises the minimum. Nothing here uses the Kronecker form except to read $\text{tr}(\widetilde{\Pi})$ as $\sum_i d_{\text{out},i}\text{rank}(\Pi_i)$: for any orthogonal projector Π on $\mathbb{R}^D$ both halves hold with $\rho(\mathcal{L}_j) = \text{tr}(P_j\Pi P_j)/D_j$, which is the form Appendix A12 uses for the coordinate mask, whose fraction p is set per matrix, so $\rho = p$ on every locus.

Why not the row space. On $\bigoplus_i \mathbb{R}^{d_{\text{in},i}}$ the same computation runs with $\Pi = \bigoplus_i \Pi_i$ and, writing $\dim(\mathcal{L}_j) = \sum_{i\in\mathcal{L}_j} d_{\text{in},i}$, gives $\sum_j \dim(\mathcal{L}_j)\rho^{\text{dim}}(\mathcal{L}_j) = \text{tr}(\Pi) = \sum_i \text{rank}(\Pi_i)$, so the aggregate there is ρ^{dim} and the bound is $\min_j \rho^{\text{dim}} \le k'/d$ with $k' = \sum_i \text{rank}(\Pi_i)$, $d = \sum_i d_{\text{in},i}$. That is a true statement about a different quantity, and it is *not* the one the body chains with the second half: ρ^{par} and ρ^{dim} part company wherever a locus mixes matrices of different d_{in}, so that their reach $\dim\mathcal{H}_{\text{safe}}/d_{\text{in}}$ differs, and of different d_{out}, the weight ρ^{par} adds: at every MLP locus here (by up to $1.89\times$ here, Table 10), and $\min_j \rho^{\text{par}}$ *exceeds* k'/d on all four models: by 3.8, 18.2, 3.6 and 23.0 %. So the row-space version cannot be read as an energy budget, and the parameter-space version above is the one Proposition 1 states.

For the second half, let $\Delta = P_{\text{task}}z$ with z rotation-invariant on $\mathbb{R}^D$ (a Gaussian with covariance $\sigma^2 I$, say) and $\Delta \neq 0$ almost surely. Then $u = \Delta/\|\Delta\|$ is uniform on the unit sphere of $\mathcal{L}_{\text{task}}$, so $\mathbb{E}[uu^\top] =$

$P_{\text{task}}/D_{\text{task}}$, and $\mathbb{E}\big[\|\widetilde{\Pi}\Delta\|^2/\|\Delta\|^2\big] = \mathbb{E}[u^\top\widetilde{\Pi}u] = \operatorname{tr}(P_{\text{task}}\widetilde{\Pi}P_{\text{task}})/D_{\text{task}} = \rho^{\text{par}}(\mathcal{L}_{\text{task}})$: the same aggregate as the first half, which is the point of putting both on $\mathbb{R}^D$. Concretely, for ΔW_i with i.i.d. entries of a common variance the ratio of expectations is $\sum_i d_{\text{out},i}\operatorname{rank}(\Pi_i)/\sum_i d_{\text{out},i}d_{\text{in},i}$, and the $d_{\text{out},i}$ are what the row-space version drops. Under the weaker second-moment assumption $\mathbb{E}[\Delta\Delta^\top] = \sigma^2 P_{\text{task}}$ alone, the same computation gives the ratio of expectations $\mathbb{E}\|\widetilde{\Pi}\Delta\|^2/\mathbb{E}\|\Delta\|^2 = \rho^{\text{par}}(\mathcal{L}_{\text{task}})$, which is the form §5.3 uses. □

Corollary 2. Write $\Delta W = BA$ with $B \in \mathbb{R}^{d_{\text{out}}\times r}$. If $B = P_m P_m^\top B$ then $\operatorname{colspace}(\Delta W) \subseteq \operatorname{colspace}(B) \subseteq \operatorname{range}(P_m)$. Let $V = U\Sigma S^\top$ be the SVD of the alignment matrix, so $C = VV^\top/\|V\|_F = U\Sigma^2U^\top/\|V\|_F$ and $\operatorname{range}(P_m) = \operatorname{span}\{u_1,\dots,u_m\}$. Expanding $\Delta W = \sum_{i\le m} u_i a_i^\top$ with $a_i^\top = u_i^\top \Delta W$ gives $C\Delta W = \|V\|_F^{-1}\sum_{i\le m}\sigma_i^2\, u_i a_i^\top$, and since the u_i are orthonormal,

$$\cos(C\Delta W, \Delta W) \;=\; \frac{\langle C\Delta W, \Delta W\rangle}{\|C\Delta W\|_F\,\|\Delta W\|_F} \;=\; \frac{\sum_{i\le m}\sigma_i^2 w_i}{\big(\sum_{i\le m}\sigma_i^4 w_i\big)^{1/2}}, \qquad w_i = \frac{\|a_i\|^2}{\|\Delta W\|_F^2}$$

with $\sum_{i\le m} w_i = 1$, and $\|V\|_F$ cancelling, whenever $\sum_{i\le m}\sigma_i^2 w_i > 0$ (so $C\Delta W \neq 0$). This is an equality, not a bound. By Cauchy–Schwarz the expression is at most 1, with equality exactly when the σ_i carrying weight are all equal; it decreases as the spread of $\{\sigma_i\}_{i\le m}$ on the support of w grows. On the spectra of these checkpoints a smaller m retains a flatter block and so raises the plug-in cosine (0.838 at $m = 256$ against 0.717 at $m = 1024$ on Llama-3.1-8B, Table 47); that is a property of the measured spectra and not of the formula, which is not monotone in m for an arbitrary spectrum. Whenever the resulting cosine is at least τ, R_τ is the identity on that matrix.

Two things follow that the main text uses. On these spectra the attacker cannot make the cosine large *and* keep the adapter's full capacity: shrinking m raises cos but confines ΔW to an m-dimensional output subspace, and widening it to d_{out} takes the cosine down to Eq. 2's saturating value (Appendix A39). And the plug-in value is computable by either party from V alone, before any training, which is what makes it both the attack's lever (§3) and the defender's detector (§6); the trained value differs from it by the bias Table 47 measures. □

A5 Numerical verification of the row-space lemma

The corollary is exact only for plain or momentum SGD with the layer's inputs fixed, so we check it numerically rather than assume it. On a 32×64 linear layer driven by five input vectors, over seven steps at rate 10^{-2}, plain SGD leaves $\|\delta_R(I - QQ^\top)\|_F/\|\delta_R\|_F = 3.95 \times 10^{-6}$ of the displacement outside $\operatorname{span}\{x_t\}$, zero to floating-point accumulation, while the *same* layer, inputs and step count under AdamW leave 0.53 (a random direction would leave 0.96), confirming that per-coordinate preconditioning destroys the containment. Both are in `code/tests/test_offload_optim.py` (`test_sgd_rowspace_confinement`, `test_adamw_breaks_rowspace`), together with an `fp32` parity check of our offloaded optimiser against `torch.optim.AdamW` (1.2×10^{-6} relative, 1.5×10^{-8} absolute) and a `bf16`-state parity check against a declared $2\,\%$ budget (measured $0.45\,\%$). At model scale the containment over ten steps is measured in Appendix A40; ρ, the size of the span, is tabulated per locus and model in Appendix A12.

A6 Post-hoc repairs as linear operators

Table 1 in §2 is the taxonomy; this appendix is what it leaves out. Rows 1, 2 and 5 of that table are not linear in ΔW: row 5 is a fixed addition, and rows 1 and 2 a gradient step and a gradient-derived edit, both evaluated at θ_a; all three are listed because what is fixed about them is a subspace all the same. Row 3 is linear on each matrix it acts on, but which matrices those are is read off ΔW by the gate (R_τ of Corollary 2).

SafeLoRA's C is not a projector, and on a matrix it gates it deletes the update. Hsu et al. (2024) give two forms. The exact one is the orthogonal projector onto $\operatorname{colspace}(V)$; the one their released implementation uses, and the one we replicate, is $C = VV^\top/\|V\|_F$. Write the projector with a pseudo-inverse, $\hat{C} = V(V^\top V)^+V^\top$, and not the inverse the derivation writes: $V^\top V$ is $d_{\text{in}} \times d_{\text{in}}$

of rank at most d_{out}, so it is singular wherever $d_{\text{out}} < d_{\text{in}}$ and $(V^\top V)^{-1}$ does not exist on `k_proj`, `v_proj` or `down_proj` at all. The distinction is not cosmetic. Every matrix we target has $d_{\text{out}} \le d_{\text{in}}$ except `gate_proj` and `up_proj`, so V has full *row* rank generically, $\text{colspace}(V) = \mathbb{R}^{d_{\text{out}}}$, and $\hat{C} = I$: the *exact* form of the repair is the identity on most of the network (140 of Qwen2.5-1.5B's 196 target matrices, Table 41). What acts is the approximation, and the approximation is not idempotent. Writing $V = U\Sigma S^\top$ (the right factor is S, not W, which is a weight matrix in §2),

$$C \;=\; U\,\frac{\Sigma^2}{\|V\|_F}\,U^\top, \qquad \text{tr}(C) = \|V\|_F, \qquad \|C\|_2 = \frac{\sigma_1^2}{\|V\|_F}.$$

Measured on Llama-3.1-8B over a spread of eight target matrices: $\|V\|_F = 1.60$–5.49, $\|C\|_2 = 0.012$–0.380, and for an isotropic rank-16 update, $\|C\Delta W\|_F/\|\Delta W\|_F = 0.0012$–$0.0063$; over all 224 of its target matrices the last quantity runs 0.0007–0.0063. **On these checkpoints C is a contraction, by two to four orders of magnitude on an isotropic update**; along V's leading direction it keeps $\|C\|_2$ of the norm, at most 0.38 here. On a matrix whose cosine falls below τ, SafeLoRA therefore does not shrink the adapter's update towards an aligned subspace; it all but removes it, returning that matrix close to θ_0.

Two things in the body follow from this rather than merely coexisting with it. §5.1's -0.91 to -0.96 correlation between the cosine and the fraction deleted is close to definitional (we say so there), and this is exactly why: with the update annihilated wherever the gate fires, the deleted fraction *is* the square root of the energy carried by the gated matrices. And the cliff of §5.3 ($100\,\%$ of a customer's fine-tune gain at $\tau = 0.5$ on Llama-3.1-8B and $0\,\%$ at $\tau = 0.8$) is whole matrices being reverted, not a projection biting harder.

What survives for a general symmetric operator, and what does not. Two things have to be said separately, because SafeLoRA fails the hypotheses of Proposition 1 on both counts.

The first half does not generalise. Its chain runs $\text{tr}(\widetilde{\Pi}) = k = \sum_i d_{\text{out},i}\text{rank}(\Pi_i)$, which needs idempotence. For $C = VV^\top/\|V\|_F$, $\text{tr}(C) = \|V\|_F$ (a dimensionful scalar, 0.08 to 5.58 over the 840 target matrices whose spectra we store, not a rank). The partition identity itself survives for any operator, since the P_j are whole-matrix coordinate projectors; what fails is the reading of the trace as a dimension, so SafeLoRA has no dimension budget it must obey. What replaces it is the coarser statement at the end of the next paragraph.

The second half does generalise, in retained rather than lost energy. With $\Delta = Pz$ and z rotation-invariant on the side C acts,

$$\mathbb{E}\Big[\|C\Delta\|_F^2/\|\Delta\|_F^2\Big] \;=\; \text{tr}(PC^2P)/\dim(P),$$

which is ρ when $C = \Pi$ is idempotent. The quantity the body wants is the energy *lost*, and for a contraction that is not one minus the above but $\mathbb{E}\big[\|\Delta - C\Delta\|_F^2/\|\Delta\|_F^2\big] = \big(\dim(P) - 2\text{tr}(PCP) + \text{tr}(PC^2P)\big)/\dim(P)$, with a cross term the projector case does not have. It is harmless here only because both traces are tiny: $\text{tr}(C)/d_{\text{out}} < 1.5\times10^{-3}$ and $\text{tr}(C^2)/d_{\text{out}} < 4.1\times10^{-5}$ over those 840 matrices, so the loss is at least 0.997, which is what §2's "nearly all of it" means, and it is a measurement of these checkpoints, not a theorem. It is tempting to read the retained-energy form as the same one-for-one trade with $\text{rank}(\Pi)$ replaced by $\text{tr}(C^2)$, and that reading is wrong: C is 1-homogeneous in V, so $\text{tr}(C^2)$ is 2-homogeneous, carries the units of σ^2, is not bounded by $\dim$, and is not a rank. Numerically $\text{tr}(C^2)/d_{\text{out}}$ is between 1.2×10^{-8} and 4.0×10^{-5} over the 840 target matrices of all four models, against the $\rho \le 2.7\,\%$ of Appendix A12, three to six orders apart. So the one-for-one reading is the right one for the repairs whose reach is a projector (one-shot patching, the rank-k edit, the coordinate mask), and for SafeLoRA the trade is coarser: what τ buys is a *count of matrices*, and each one costs the whole update inside it. That is a worse bargain for the defender than the identity's smooth version, which is what Table 57 shows.

The gate is a closed-form property of V, and reads the attack only through w. Let $\Delta W = \sum_i u_i a_i^\top$ in V's left-singular basis, as in Appendix A4, and let $w_i = \|a_i\|^2$ be the (unnormalised) energy ΔW places on u_i. Since C multiplies that component by $\lambda_i = \sigma_i^2/\|V\|_F$,

$$\cos(C\Delta W, \Delta W) \;=\; \frac{\sum_i w_i\lambda_i}{\sqrt{\big(\sum_i w_i\big)\big(\sum_i w_i\lambda_i^2\big)}} \;=\; \Big(1 + \text{CV}_w^2(\sigma^2)\Big)^{-1/2},$$

where CV_w is the coefficient of variation of $\{\sigma_i^2\}$ under the weights w. $\|V\|_F$ cancels, so the gate is scale-free even though C is not, and by Cauchy–Schwarz the cosine is at most 1, with equality exactly when the retained σ_i are equal; and by the Pólya–Szegő inequality it is at least $2\sqrt{\kappa}/(1+\kappa)$ for *every* w, with κ the ratio of the largest to the smallest retained σ_i^2, so a matrix clears $\tau = 0.5$ whatever training does once $\kappa \le 7 + 4\sqrt{3} \approx 13.9$. Two corollaries, one for each side of the paper.

For an *unstructured* update w is uniform *in expectation*, and plugging that in collapses the expression to

$$\cos(C\Delta W, \Delta W) \;=\; \sqrt{\mathrm{PR}(V)/d_{\mathrm{out}}}, \qquad \mathrm{PR}(V) = \Big(\textstyle\sum_i \sigma_i^2\Big)^2 \Big/ \textstyle\sum_i \sigma_i^4,$$

the normalised participation ratio of V's spectrum. This is a property of the two public checkpoints alone: at this plug-in the gate reads *nothing* about the update.

Read it as what it is: $\cos$ is a nonlinear function of w, so this is a plug-in at $\mathbb{E}[w]$ and not an identity. A rank-r update's w_i fluctuates with relative deviation $\sqrt{2/r}$ (at $r = 16$, about 0.35), so a single draw need not sit on the prediction. Two checks. Against random rank-16 updates on individual Llama-3.1-8B matrices the closed form is exact to two decimals (0.061/0.061, 0.374/0.371, 0.774/0.774, 0.372/0.373, predicted/measured). Those four are all matrices with $d_{\mathrm{out}} \le d_{\mathrm{in}}$, so they do not exercise the case where V has fewer than d_{out} non-zero σ_i and the padded denominator bites; on `gate_proj` it does, and the form predicts 0.358 there. The one that matters is whether it predicts a *trained* adapter, which is not isotropic. Evaluating $\sqrt{\mathrm{PR}(V)/d_{\mathrm{out}}}$ over every target matrix and averaging gives 0.478 on Llama-3.1-8B, 0.437 on Qwen2.5-7B, 0.433 on Qwen2.5-1.5B and 0.399 on Mistral-7B, against measured adapter means of 0.537, 0.480, 0.488 and 0.446 over the five all-modules ordinary adapters of each. The ordering is right except between the two Qwen models, whose predictions are 0.004 apart, and the residual is $+0.043$ to $+0.060$, a near-constant offset, which is what a real update's mild concentration on V's leading directions should add. So the level of the gate, and its ordering across models, is set by the alignment matrix's spectrum before any attack exists; the attack contributes about 0.05. That is the strongest form of §5.1's claim that the cosine is a property of the locus rather than of the attack.

The same numbers give the contraction across the whole network: $\mathrm{tr}(C^2)/d_{\mathrm{out}}$ ranges over 1.2×10^{-8} to 4.0×10^{-5} across the 840 target matrices of all four models, so $\|C\Delta W\|_F/\|\Delta W\|_F$ for an isotropic update is between 0.0001 and 0.0063 everywhere.

For a *constrained* update, the attacker chooses $\mathrm{supp}(w)$, and Corollary 2's conclusion is the second reading of the same formula: confining ΔW's columns to m singular directions makes the cosine $(1+\mathrm{CV}_w^2(\sigma^2))^{-1/2}$ over those directions, which is near 1 whenever the retained σ_i^2 are of comparable size. The corollary should not be read as $C\Delta W \approx \Delta W$ (it is not, by the contraction above) but only as $\cos \to 1$, and the measured 0.72–0.91 is what the formula predicts for the top-m block of a decaying spectrum, not a shortfall against it.

A sharper attack the formula implies, and what it does to the detector. Nothing in the expression prefers the *top* m directions; it prefers a *flat* band, and for a decaying spectrum whose successive ratios $\sigma_i^2/\sigma_{i+1}^2$ shrink (a power law, say) the leading block is the least flat band of its width. So the corollary implies a sharper A3 than the one the rest of this paper measures. We built it (Appendix A41): a full SVD per target matrix, the width-m window minimising the coefficient of variation of σ^2, and otherwise the recipe of §4 unchanged. The prediction holds (the gate cosine goes from 0.717 to **0.897** at the same rank and the same m), and so does the consequence for the defender: at 0.897 the adapter sits 0.315 above the highest benign adapter on that model, against the 0.135 the top-m version of the same attack manages. The flattening dial does not have an exit between the two thresholds (widening does, on one model: Appendix A39); turning it towards evasion turns it towards the detector.

A7 A RANK-MATCHED CONTROL: IS IT THE SUBSPACE, OR IS IT CAPACITY?

On Llama-3.1-8B the two ordinary adapters of the first-pass grid behind Table 2 (80 judged prompts each) that best survive SAFELORA are rank-64 all-modules attacks, left at 0.675 and 0.650: above the weakest of the three A3 adapters (0.325). A reader is entitled to ask whether A3 survives because of *where* its update sits or just because a bigger adapter has more of its norm left after any fixed projection. So we trained A3 and A2 at rank 64 as well, against the rank-64 ordinary adapters the

grid already holds, and put all three families through the same repairs. Everything else is the recipe of §4; only the rank and the constraint differ.

Table 4 settles it on Qwen2.5-1.5B, where all three families land. At rank 64, SAFELORA takes the ordinary attacks from 0.574 to **0.035** (-0.539 $[-0.580, -0.500]$) and A2 from 0.605 to **0.020** (-0.585 $[-0.623, -0.547]$), and leaves A3 at **0.539** from 0.577 (-0.038 $[-0.070, -0.006]$). Same model, same rank, same data, objective, optimiser and steps: on this model capacity is not what defeats the repair; the constraint is. A3's rank-64 interval does exclude zero (the defence takes something off, as it does at rank 16 on Llama), but at an order below what it takes off either matched family. The 20 % mask shows the mirror image at the same rank (A3 to 0.027, A2 left at 0.245), so neither blind spot is an artefact of rank 16.

On Llama-3.1-8B the control does not separate the families, and we say so. There SAFELORA moves the rank-64 ordinary attacks by -0.100 $[-0.152, -0.047]$ and the rank-64 A3 adapters by -0.102 $[-0.136, -0.070]$, the same amount, to 0.461 and 0.398. Rank 64 is a second, capacity-driven route past the defence on that model, and once it is open the matched comparison has nothing left to discriminate: the rank-matched evidence that it is the constraint and not the capacity is the two Qwen models', not Llama's. What Llama adds instead is that the two survivals are mechanically different and the statistic of §6 tells them apart. The rank-64 ordinary adapters survive *having been cut*, the projection deletes 82–85 % of them, while the rank-64 A3 adapters survive *barely having been touched*, 13–17 % deleted, reading $\cos = 0.736$–0.738 against the rank-64 ordinary adapters' 0.530–0.534. Inside this matched arm a provider who sees a surviving adapter can ask which of the two it is; over Llama's full adapter pool the same comparison is the -0.068 of Table 16, which is the honest number and the one §6 reports.

Qwen2.5-7B is the second model where it does separate, and the cleanest of the three. The arm was trained with the rest and judged last; one of its three A3 adapters landed, so that row is a single adapter and is read as one. There SAFELORA takes the rank-64 ordinary attacks from 0.583 to **0.020** (-0.564 $[-0.602, -0.526]$) and A2 from 0.532 to **0.011** (-0.521 $[-0.561, -0.483]$), and leaves the A3 adapter at **0.586** from 0.605, -0.018 $[-0.068, +0.032]$, an interval that *includes* zero, which the rank-64 A3 interval on Qwen2.5-1.5B does not, though on a third of that row's items (220 against 660), so part of that difference is power. So the rank-matched control holds on two models rather than one, and the larger of the two is where the defence removes the matched ordinary attack most completely and the matched A3 adapter least. The 20 % mask on the same arm does the opposite of what it does at 1.5B: ordinary 0.289, A2 0.259, A3 0.218, three families within 0.071 of each other. A coordinate mask at rank 64 separates nothing on this model, which is why §5.4 reports the mask's rank-64 behaviour as model-dependent rather than as a second instance of the 1.5B result.

The run also closes a gap in §5.4: A2 does not land on Llama-3.1-8B at rank 16, but two of the three rank-64 A2 adapters do, and the mask leaves them at 0.186 from 0.334 where SAFELORA takes them to 0.023. The direction is the same as on the two Qwen models, on a third checkpoint. The same arm shows a limit of the mask that the rank-16 grid does not: on Llama-3.1-8B the 20 % mask leaves the rank-64 ordinary attacks where they were (0.586 from 0.561, $+0.026$ $[-0.023, +0.077]$, two of the three higher than before), and only the two repairs together remove them (0.015).

A8 THE BENIGN BAND, WITH THE LOCUS VARIED

The twelve benign adapters of §6 are all-modules, one value of the locus, which §5.1 identifies as the one variable that moves the gate statistic. A band measured that way says nothing about a customer who fine-tunes attention only, so we trained benign adapters at three further loci (early/attention, mid/attention and late/MLP) on two tasks each, on every model the locus stage reached.

Table 5 is the result, and it removes the detector's margin outright on two models. Benign adapters at mid/attention sit at 0.60 on Llama-3.1-8B and 0.58 on Qwen2.5-1.5B, well above the all-modules bands of 0.527–0.535 and 0.477–0.488; at late/MLP they sit well below. Measured against the A3 census of Table 16, the margin from the highest benign adapter at any locus to the lowest A3 adapter is -0.046 on Llama-3.1-8B, $+0.191$ on Qwen2.5-7B, $+0.202$ on Mistral-7B and -0.007 on Qwen2.5-1.5B: negative on exactly the two models where §6 builds an adversary against the detector, and positive on the two where it does not. Each other-locus cell is one adapter: on Llama-3.1-8B four of the six clear the lowest A3 adapter, on Qwen2.5-1.5B two do, by 0.004 and 0.007, inside

Table 4: A3, A2 and ordinary attacks at rank 64, judged, averaged over the adapters of each family that landed (n in the label); Mistral is excluded as elsewhere. Every arm is paired against this stage's own unrepaired arm on the prompts both carry, and the small figure is that paired difference. On both Qwen models the three families separate; on Llama-3.1-8B SAFELORA moves the ordinary and A3 families by the same amount and the control is uninformative there. Only one of Qwen2.5-7B's three rank-64 A3 adapters landed, so that row is a single adapter. mask_{20} is the 20 % coordinate mask of §5.4 and "both" is SAFELORA at $\tau = 0.5$ then that mask; every adapter carries 220 judged prompts, and the small figures are computed before rounding.

model	family (rank 64)	unrepaired	SAFELORA	mask_{20}	both
Llama-8B	ordinary (3)	0.561	0.461 −0.100	0.586 +0.026	0.015 −0.545
Llama-8B	A2 (2)	0.334	0.023 −0.311	0.186 −0.148	0.014 −0.320
Llama-8B	A3 (2)	0.500	0.398 −0.102	0.064 −0.436	0.061 −0.439
Qwen-7B	ordinary (3)	0.583	0.020 −0.564	0.289 −0.294	0.020 −0.564
Qwen-7B	A2 (3)	0.532	0.011 −0.521	0.259 −0.273	0.011 −0.521
Qwen-7B	A3 (1)	0.605	0.586 −0.018	0.218 −0.386	0.218 −0.386
Qwen-1.5B	ordinary (3)	0.574	0.035 −0.539	0.059 −0.515	0.027 −0.547
Qwen-1.5B	A2 (3)	0.605	0.020 −0.585	0.245 −0.359	0.015 −0.589
Qwen-1.5B	A3 (3)	0.577	0.539 −0.038	0.027 −0.550	0.027 −0.550

the 0.011 spread of the twelve all-modules adapters, so on Qwen2.5-1.5B the sign is within seed noise. Reading the same row off the first pass and the $\{32, \ldots, 1024\}$ sweep alone gives $+0.138$ and $+0.032$ instead, which is how a locus-corrected margin can look safe on a pool that predates the attack aimed at it. Mistral is included here although it is excluded from the A3 harm claim of §5.5: the gate statistic is a function of ΔW and the two checkpoints alone, so it is well defined whether or not θ_0 refuses anything.

Two things follow. The detector's calibration set has to span the loci a provider expects to receive, not one of them (Appendix A10 shows what happens otherwise). And the statistic behaves the same way for benign work as for ordinary attacks (both track the locus, both stay inside the same range), which is why A3, the one configuration that leaves that range on Qwen2.5-7B and Mistral-7B, is what it can separate there, and why on Llama-3.1-8B and Qwen2.5-1.5B, where A3 does not leave it, it cannot.

Table 5: Mean $\cos(C\Delta W, \Delta W)$ for benign fine-tunes at four loci. One adapter per (locus, task); the all-modules row is the twelve-adapter band of §6. "A3 (lowest adapter)" is the smallest value over the A3 census of Table 16 on that model (first pass, both budget sweeps, the aimed adapters of Appendix A34 and the flattest-band adapters of Appendix A41: 33, 3, 3 and 42 adapters), and "gap to benign" subtracts the largest benign value in the column above, computed before rounding.

benign adapter	Llama-8B	Qwen-7B	Mistral-7B	Qwen-1.5B
early/attention, SQL-Create	0.588	0.531	0.490	0.550
early/attention, DialogSum	0.586	0.531	0.473	0.538
mid/attention, SQL-Create	0.604	0.559	0.514	0.579
mid/attention, DialogSum	0.605	0.571	0.510	0.582
late/MLP, SQL-Create	0.430	0.340	0.360	0.351
late/MLP, DialogSum	0.444	0.349	0.365	0.374
all modules (12 benign adapters)	$[0.527, 0.535]$	$[0.460, 0.478]$	$[0.433, 0.444]$	$[0.477, 0.488]$
A3 (lowest adapter)	0.559	0.762	0.715	0.575
gap to benign	-0.046	$+0.191$	$+0.202$	-0.007

A9 WHAT EACH REPAIR COSTS, OVER THE WHOLE BENIGN POOL

The cost rows under Table 28 are one SQL-Create adapter per model, which is thin support for a claim about what a provider would do to its customers. Table 6 repeats them over the benign pool of §5.3 (four unrelated tasks, three seeds each) under the same minimum-gain rule (a fine-tune counts only if it gains at least 0.05 over θ_0, which is the floor below), applied here per (task, seed) rather than per (model, task) because the unit being averaged is the adapter. That is a stricter screen than

§5.3's and it drops four pairs, not one: GSM8K seeds 8 and 9 on Llama-3.1-8B, DialogSum seed 9 on Qwen2.5-7B and GSM8K seed 7 on Mistral-7B. Each entry is therefore a mean over 10–12 adapters.

The ordering survives and the spread widens. The coordinate mask is free at both settings on every model (88–120 % of the gain kept); RESTA at its own $\gamma = 0.5$ is nearly free (91–99 %) and at $\gamma = 1$ is not (64–88 %); SAFELORA at $\tau = 0.5$ ranges from 51 % of the gain on Qwen2.5-1.5B to 136 % on Mistral-7B; and the composite costs from 17 % to 66 %. The deleted fractions §5.4 quotes beside these are read off the one SQL-Create adapter per model behind the cost rows of Tables 28–33, as $\|\Delta W' - \Delta W\|_F / \|\Delta W\|_F$ for the repaired update $\Delta W'$: 0.44–0.45 under the 20 % mask, 0.85–0.95 under SAFELORA at $\tau = 0.5$, and 0.96–1.03 under the composite on the three models with an aligned θ_0 (0.93 on Mistral-7B). The composite's can exceed 1 because the projection and the mask are not orthogonal, so the repaired update can end farther from the original than zero is.

These are means of *ratios* with a 0.05 floor on the denominator, which is fragile from above, and the two most striking entries are the fragile ones. Mistral-7B's 136 % under SAFELORA is two GSM8K cells whose fine-tune gain is exactly 0.100 on a 60-item test set, carrying ratios 2.83 and 3.00; its median is 105 %. Llama's 109 % has one DialogSum cell at gain 0.056 and ratio 2.08; its median is 100 %. Medians for every entry are in Table 6 beside the means. What survives the swap is the position of the composite, last on three of the four models under either statistic (on Llama-3.1-8B RESTA at $\gamma = 1$ is below it under the mean and tied with it under the median), and the mask above SAFELORA on three of the four: Mistral-7B is the exception and only under the mean, where the two GSM8K cells put SAFELORA on top; under the median the mask leads there too. What does not survive is any particular percentage. Read the means as shapes, as §5.4 does, and read a single entry above 120 % as a small denominator.

Two more things are worth saying about that spread. It is not noise: the per-adapter range within a model is 35–154 % for the composite on Llama and −7 to 87 % on Qwen-1.5B, so individual customers fare very differently under the same repair. And the cost of the composite is not predicted by the cost of its parts: Mistral-7B keeps 136 % of the gain under SAFELORA alone (the most of any model, 105 % at the median), and only 59 % under the composite, while Llama-3.1-8B keeps 109 % and 83 % (100 and 79 at the median). That reversal holds under either statistic. The trade-off Proposition 1 describes is between coverage and utility on one model, not a constant across models.

Table 6: Percentage of a benign fine-tune's gain over θ_0 that survives each repair, over the four-task, three-seed pool of §5.3, as **mean / median**; n is the number of (task, seed) pairs clearing the minimum-gain rule. Above 100 means the repaired adapter beat the unrepaired one. These are ratios with a floored denominator, so the mean is fragile from above and the median is the entry to read where the two diverge; the ordering of the six repair settings is the same under either except where a fragile mean moves one entry (SAFELORA on Mistral-7B, and the mask$_{20}$/RESTA^1 pair on Qwen-1.5B). RESTA^1 is RESTA at $\gamma = 1$ (its default is $\gamma = 0.5$); mask$_p$ is the θ_0 coordinate mask at budget p %. Read the body's figures from this table as shapes: means over 10–12 (task, seed) pairs, with per-adapter spread up to 222 points (SAFELORA on Mistral-7B, 78–300 %) and no intervals.

repair	Llama-8B	Qwen-7B	Mistral-7B	Qwen-1.5B
SAFELORA	109 / 100 (n=10)	79 / 72 (n=11)	136 / 105 (n=11)	51 / 48 (n=12)
RESTA	91 / 97 (n=10)	91 / 95 (n=11)	96 / 100 (n=11)	99 / 100 (n=12)
RESTA^1	64 / 79 (n=10)	66 / 67 (n=11)	86 / 95 (n=11)	88 / 92 (n=12)
mask$_5$	112 / 106 (n=10)	102 / 100 (n=11)	117 / 111 (n=11)	98 / 97 (n=12)
mask$_{20}$	116 / 108 (n=10)	105 / 105 (n=11)	120 / 114 (n=11)	88 / 90 (n=12)
both	83 / 79 (n=10)	47 / 48 (n=11)	59 / 51 (n=11)	34 / 26 (n=12)

A10 OPERATING THE DETECTOR WITHOUT SEEING AN ATTACK

A gap is not a detector. A provider has to choose a threshold from what they hold, their own benign fine-tunes, and then live with its false positives, so the separation of §6 has to be re-stated as a rule that never looks at an attack.

Table 7 does that. The benign adapters are split by *task*: τ_d is set to the largest mean cosine among the adapters of two tasks, plus a margin the provider chooses, and evaluated on the adapters of the

other two tasks and on every attack adapter. Both task splits are reported, pooled. At every margin from $+0.01$ to $+0.10$ the rule flags no held-out benign adapter. It flags every A3 adapter only at $+0.01$; at $+0.05$ it misses 10–12 of 33 on Llama-3.1-8B and none elsewhere, and at $+0.10$ it also misses the three $m = 1536$ adapters of §6 on Qwen2.5-1.5B, so it *flags* 66 of 81: the figure §6 quotes. Those misses are the whole trade-off: the margin buys specificity against *ordinary* attacks and against benign work at other loci (at $+0.10$ on Llama-3.1-8B and Mistral-7B nothing but A3 is flagged, while at $+0.01$ two thirds of the other-locus benign adapters are), and it is the setting that buys the most specificity that the widened adversary walks through. A provider cannot have both ends of this dial, which is the defender's dilemma of §5.3 reappearing in the detector.

Two caveats. The threshold is per model, as it must be, the calibrated τ_d ranges from 0.452 to 0.635 across the four models and three margins, so no absolute threshold transfers. And twelve benign adapters, six per fold, is a small calibration set, and a max-plus-margin threshold can only flag a held-out adapter that exceeds the in-fold maximum by more than the margin, so the zero held-out benign false-positive rate of Table 7 is a weak test; the number a provider would use is their own submission history.

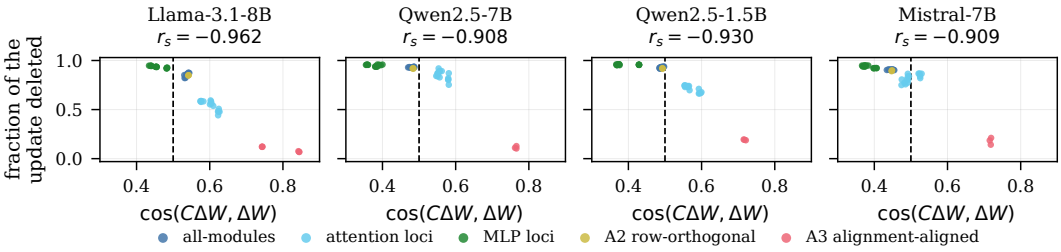

Figure 1: The gate of §5.1: how much of an adapter SAFELORA deletes against the scalar computable from two public checkpoints before the attack exists. One point per adapter, $n = 47$–50 per model, r_s as in Table 15, every $p < 10^{-21}$. The correlation is close to definitional (the threshold rule deletes a matrix exactly when the scalar falls below τ), and is included because the *level* is what §5.1's closed form predicts from $\sigma(V)$ alone. The behavioural counterpart, which is not definitional, is the right half of Table 15.

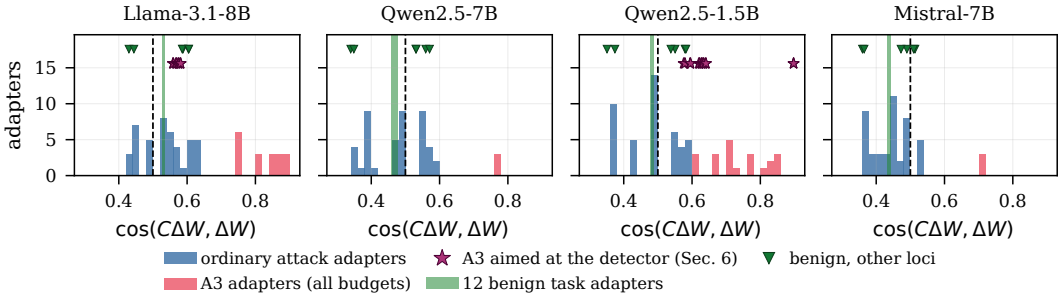

Figure 2: The gate axis of Figure 1 read as a detector. Histograms are the ordinary attack adapters and every A3 adapter the paper holds; the green band is the twelve benign task adapters at the all-modules locus, the triangles are benign adapters at three other loci, and the stars are the adversaries of §6 that aim at this statistic: every one of them, on every model, including the twelve on Llama-3.1-8B that close the gap there. Read the stars, not the histograms: on Llama-3.1-8B all twelve sit at or below the highest ordinary adapter (0.627) and on Qwen2.5-1.5B six of twenty-one do, which is the overlap §6 reports and the reason the AUC over the full pool is 0.886 and 0.989 rather than 1.000.

A11 THE LANDING RULE, APPLIED OUT OF SAMPLE

An attack counts only if it landed (§4), and "landed" is read off the unrepaired arm, which is then one side of the paired difference the table reports. Selection and measurement therefore share their

Table 7: The detector operated without seeing an attack. τ_d is calibrated on the benign adapters of two tasks and evaluated on the other two. The two folds hold out disjoint benign adapters, so that column pools them; the A3, other-loci and ordinary sets are the same under both folds, so those columns give the distinct denominator and, where the folds disagree, the range. "A3" counts the first-pass A3 adapters after the cross-stage dedup of §6, both budget sweeps, the aimed adapters of Appendix A34 and the flattest-band adapters of Appendix A41 included, 42 of them on Qwen2.5-1.5B and 81 in all. Flagging an ordinary attack is not an error, it is an attack, but the last column shows what the margin buys.

model	margin	τ_d	held-out benign	A3	benign, other loci	ordinary
Llama-8B	$+0.01$	0.540–0.545	0/12	33/33	4/6	15–21/44
	$+0.05$	0.580–0.585	0/12	21–23/33	4/6	10–11/44
	$+0.10$	0.630–0.635	0/12	21/33	0/6	0/44
Qwen-7B	$+0.01$	0.480–0.488	0/12	3/3	4/6	15–24/44
	$+0.05$	0.520–0.528	0/12	3/3	4/6	15/44
	$+0.10$	0.570–0.578	0/12	3/3	0–1/6	5/44
Mistral-7B	$+0.01$	0.452–0.454	0/12	3/3	4/6	16–17/44
	$+0.05$	0.492–0.494	0/12	3/3	2/6	6–8/44
	$+0.10$	0.542–0.544	0/12	3/3	0/6	0/44
Qwen-1.5B	$+0.01$	0.493–0.498	0/12	42/42	4/6	15–18/44
	$+0.05$	0.533–0.538	0/12	42/42	4/6	15/44
	$+0.10$	0.583–0.588	0/12	39/42	0/6	5/44

noise, and not symmetrically: a family the repair genuinely removes is pushed away from zero by the selection, while a family it does not touch is not. The direction of the bias flatters our claim, so it has to be measured.

Table 8 applies the rule on one prompt set and measures the difference on the other, so that selection is independent of what is reported. Every ordinary family stays between -0.20 and -0.64; A3 stays between -0.100 and $+0.050$.

The bias is real but it is not the explanation, and it does not run one way. Over the 22 non-A3 (model, family, direction) comparisons, 18 ordinary and 4 A2, the pooled estimate of Table 2 is anywhere from 0.12 closer to zero to 0.45 further from it than the out-of-sample one (both extremes are A2 rows; over the 18 ordinary ones the range is 0.09 closer to 0.17 further), and in 10 of the 22 the out-of-sample estimate is the larger, which is what a selection effect of this size against this much prompt-set heterogeneity should look like. What survives either selection is the gap between the two kinds of attack: the smallest ordinary effect under either (-0.202) is twice the largest A3 effect (-0.100).

Two things this check is not. It is a second point estimate under a selection rule that is independent of what it measures, *not* a bound on the first one's bias: the discrepancies above reach 0.45 on an A2 row and 0.17 on an ordinary one, three to eight times the largest in-sample A3 effect (0.054) it is used to defend, so what it establishes is the sign and the ordering rather than a number. And the three columns of Table 8 are not three readings of one quantity, selecting on a different set changes which adapters enter the row (Qwen2.5-7B's A2 row is one adapter pooled and three under the HarmBench rule; Llama's attention row is 6, 11 and 4), so the columns are comparable in direction and not item for item.

How much the rule removes. A rule that excludes a quarter of the study should have its exclusion rate on the page. Over the 480 (stage, model, attack) unrepaired arms with at least 20 judged items, 118 **are excluded** (24.6 %), and the rate is very uneven. Every arm of `a4`, `a4frontier`, `obj` and `cleanpt` is excluded, and in those four the non-landing *is* the result rather than a cost the rule imposed: the first three report it as their finding (Appendices A44, A47, A42), and `cleanpt` ran A2 at Llama-3.1-8B's clean operating points, where A2 does not land (Appendix A23). Among the stages where landing is a precondition rather than the question it is 36.1 % in `a3sweep` (13/36, where the budget sweep deliberately walks past the point at which the attack still works), 27.7 % in the attack–repair matrix (53/191, which is mostly the crippled m=256 A3 cells on Llama-3.1-8B and the early-attention locus), 27.5 % in `repairs2` and 26.7 % in `maskdial`, and exactly zero in ten of the twenty-two stages that carry an unrepaired arm at all. 46 of the 480 sit within ± 0.05 of

the cut, so the rule is not separating two populations, it is drawing a line through one, which is why the out-of-sample check above exists and why the eight-seed arm of Appendix A24 names its three excluded adapters and their unrepaired rates individually.

Table 8: The headline contrast with the landing rule applied out of sample. "pooled" is Table 2's rule (landed on AdvBench + HarmBench together, measured on both); the other two columns select on one set and measure on the other. Qwen-7B's A2 row is three adapters, two of which land on HarmBench only.

		Δ judged harm ASR after SAFELORA		
model	attack configuration	pooled	sel. HB, meas. Adv	sel. Adv, meas. HB
Llama-8B	all-modules (the default)	-0.333	-0.295	-0.329
Llama-8B	attention loci	-0.239	-0.202	-0.263
Llama-8B	MLP loci	-0.568	-0.400	-0.629
Llama-8B	A3 alignment-aligned	-0.054	-0.025	-0.100
Qwen-7B	all-modules (the default)	-0.544	-0.443	-0.619
Qwen-7B	attention loci	-0.549	-0.475	-0.622
Qwen-7B	MLP loci	-0.589	-0.553	-0.635
Qwen-7B	A2 row-orthogonal	-0.750	-0.300	-0.775
Qwen-7B	A3 alignment-aligned	$+0.025$	$+0.033$	$+0.017$
Qwen-1.5B	all-modules (the default)	-0.510	-0.520	-0.500
Qwen-1.5B	attention loci	-0.478	-0.371	-0.567
Qwen-1.5B	MLP loci	-0.522	-0.502	-0.581
Qwen-1.5B	A2 row-orthogonal	-0.442	-0.333	-0.562
Qwen-1.5B	A3 alignment-aligned	$+0.008$	-0.033	$+0.050$

A12 REACH AND COVERAGE, TABULATED

Definition 1 and its post-hoc counterpart are the paper's first contribution, so Table 9 reports both for every locus and every model rather than leaving them as constructions. ρ is the one-shot reach $\dim \mathcal{H}_{\text{safe}}/d_{\text{in}}$, parameter-weighted over the matrices of a locus; c is the coverage $c(\Delta W) = \|\Delta W Q\|_F^2/\|\Delta W\|_F^2$ of Definition 1, with Q an orthonormal basis of $\mathcal{H}_{\text{safe}}$, pooled over the locus's matrices by energy ($\sum_i \|\Delta W_i Q_i\|_F^2 / \sum_i \|\Delta W_i\|_F^2$, `code/geom.py`) and averaged over the adapters we trained at that locus. Its null mean is ρ^{par} when every entry of the locus has one variance (Appendix A4).

Two aggregates, because two different things are being counted. Averaging ρ over a locus is a choice, and the two natural choices are not the same number. c is an energy fraction, so the null it should be read against is the *parameter*-weighted mean ρ^{par}, which is what Table 9 reports, and it is the aggregate Proposition 1 is stated in. The row-space form of the same identity (Appendix A4), where $\text{tr}(\Pi)$ is a sum of ranks, is about $\rho^{\text{dim}} = \sum \dim \mathcal{H}_{\text{safe}} / \sum d_{\text{in}}$. The two coincide where d_{in} is constant across a locus *and* the matrices in it have the same reach, which is the mid and late attention loci exactly; they differ by 1.40–$1.89\times$ at every MLP locus, where `down_proj`'s input is the intermediate width, and by 0.2–$0.6\,\%$ at early attention, where layer 0's q, k and v are the three matrices whose span does not saturate and q and o carry four to seven times the parameters of k and v. Table 10 gives both. The gap is $1.40\times$ on Llama-3.1-8B and Mistral-7B, $1.77\times$ on Qwen2.5-7B and $1.89\times$ on Qwen2.5-1.5B. The headline claim does not turn on the choice: both are under $2.7\,\%$ at every locus on every model, and ρ^{dim} is never the larger of the two here: it equals ρ^{par} on the mid and late attention loci, where d_{in} is constant, and is smaller at every MLP and all-modules locus, where d_{out} and d_{in} are anti-correlated across the matrices; for arbitrary shapes the order can reverse.

Which aggregate the identity is in, and why it is not the row-space one. Stated on the stacked *input* space, where $\text{tr}(\Pi) = \sum_i \text{rank}(\Pi_i)$, the identity would be in ρ^{dim}. That version is true and useless: $\sum_j \dim(\mathcal{L}_j)\rho^{\text{dim}}(\mathcal{L}_j) = 14{,}055$, $7{,}997$, $7{,}822$ and $7{,}997$ against a k of the same four numbers, which is an algebraic identity once the loci partition the target matrices (as the parameter-space one below is too) and is not evidence of anything. What disqualifies it is that the body chains the identity's min-bound with the identity's second half, which is an *energy* statement and so is in

ρ^{par}, and $\min_j \rho^{\mathrm{par}}$ *exceeds* the row-space k/d on all four models, by 3.8, 18.2, 3.6 and 23.0 %, so the chain would fail.

The proposition therefore lives where the energy lives. The repair acts on each matrix by $\Delta W_i \mapsto \Delta W_i \Pi_i$, so its reach on the parameter space is $\widetilde{\Pi} = \bigoplus_i (I_{d_{\mathrm{out},i}} \otimes \Pi_i)$ with $\mathrm{tr}(\widetilde{\Pi}) = \sum_i d_{\mathrm{out},i} \mathrm{rank}(\Pi_i)$, and the locus mean of *that* is ρ^{par} by construction. Both halves are then the same aggregate and the chain closes. As a check on the code rather than the mathematics, over the partition by the six band $\times$ module loci (all-modules excluded): $\sum_j D_j \rho^{\mathrm{par}}(\mathcal{L}_j)$ is 86,587,392, 56,954,368, 48,132,096 and 26,423,296 against a k of the same four numbers, and $\min_j \rho^{\mathrm{par}} \leq k/D$ on all four (0.01172 against 0.01241, 0.00835 against 0.00873, 0.00651 against 0.00690, 0.01932 against 0.02017). We record the earlier form because the difference between a true statement and a load-bearing one is the whole point of the section.

The bound of Corollary 1 is attained, which is why ρ varies only through d_{in}. Reading the cached bases directly (the rank counts singular values of the stacked module inputs above 10^{-6} of the largest): $\dim \mathcal{H}_{\mathrm{safe}}$ equals the safety instance's token count on 221 of 224 target matrices on Llama-3.1-8B and Mistral-7B and on 193 of 196 on both Qwen models. The exceptions are the same three matrices every time (layer 0's `q_proj`, `k_proj` and `v_proj`) whose input is an embedding lookup, so repeated tokens share a row and the rank is the number of *distinct* tokens: 44 of 63 on Llama-3.1-8B, 28 of 41 on both Qwen models, 29 of 35 on Mistral-7B. Everywhere else the safety activations are in general position and the bound is tight.

This is worth stating plainly because it says what ρ is and is not. It is not an estimate of an unknown quantity that the forward pass discovers; it is a check that Lemma 1's bound is attained, after which $\rho = |T|/d_{\mathrm{in}}$ on all but three matrices. That is also why pre-registered prediction (i) in Appendix A18, that ρ would vary by at least $5\times$ across loci, could not have come true: with the span saturated, ρ can vary across loci only through the width of the layer, and the widest ratio is $1.38\times$ under ρ^{par}, $2.61\times$ under ρ^{dim}. We record it as a failed prediction there and as a structural consequence here, and the useful reading is the one that survives: the repair's reach really is the token count, not less.

Four things are worth reading off it. First, at one-shot scale ρ_1 never exceeds 2.7 %: the patch's row space is a thin slice of every model, which is Corollary 1's premise measured rather than assumed. Second, c exceeds ρ at every one of the 28 loci, by a factor of 1.48 to 6.36 (median 2.5; the small ratios are Mistral's, where the safety row space is widest relative to the update): an ordinary attack lands disproportionately *inside* the safety row space, which is why a repair confined to it looks good on the attacks people usually write. It is also why the paper's closing paragraph asks for both numbers: a benchmark on which c sits far above ρ is a favourable sample, and reporting only the post-repair ASR hides that. Third, the adversaries separate on this axis in the way the construction predicts: A2's coverage is 0 to floating-point accumulation on all four models, while A3's is ordinary (1.2–3.8 %): A3 does not evade by leaving the row space, it evades the gate, and no reach-based statistic distinguishes it from an ordinary attack.

Fourth, and this is the framework's main quantitative caveat rather than a footnote: *reach buys back linearly in the safety set*. Corollary 1 bounds $\dim \mathcal{H}_{\mathrm{safe}}$ by $\sum_s L_s$, and five safety instances take ρ from 0.7–2.7 % to 3.5–13.3 % (ρ_5 in Table 9), above an ordinary attack's coverage at most loci. So the confinement premise is a statement about the regime the published repairs actually operate in (SAFELORA uses no safety data at all, *Safety at One Shot* uses one instance), not about gradient repair in general. A defender who is willing to spend a large safety set does buy reach; Proposition 1 says what that costs on the other side. We do not run a five-instance repair (every patching arm here uses the one instance the method specifies), so ρ_5 is a bound on what such a repair could reach, not a measurement of what it would do.

Energy is what the identity prices, and it is a poor proxy for what the customer loses. Proposition 1 is about *energy*; §5.3 is about task *gain*, and letting the first stand in for the second is a step the identity does not license. The records carry both, so here they are against each other on one model and the same customer adapters. On Llama-3.1-8B the coordinate mask moves the deleted energy from 0.049 to 0.598, a factor of twelve, and the gain kept falls only from 1.15 to 1.05, a slope of 0.18 points of gain per unit of energy. SAFELORA over $\tau \in [0.70, 0.80]$ moves the deleted energy from 0.935 to 0.998, a span of 0.063, and the gain kept falls from $+0.674$ to -0.037: a slope of 11.3,

sixty-two times steeper (Table 12). Both ends are the ends of the operator's own dial, averaged over its tasks there, which is what makes the slope and not the level the comparable quantity.

The reason is in the same rows and it is the paper's own mechanism rather than a new one. The mask's deletion is *spread*: p of the coordinates of all 224 target matrices, at every setting of the dial. SAFELORA's is *concentrated*: C is a contraction, so a matrix it gates is emptied rather than projected, and the τ dial moves matrices from intact to empty: 184 of 224 at $\tau = 0.70$ and all 224 by $\tau = 0.80$. Those last forty matrices are seven points of energy and they are the customer's whole task. So what a repair costs tracks how many of the customer's matrices it empties, not how much of their update's energy it removes, and a reader should not read Proposition 1 as pricing the second in units of the first. One caveat: the mask's task arm on this model is SQL-Create alone (5 cells, one per dial setting) against SAFELORA's four tasks at one seed over six thresholds (24), so the mask's slope rests on one task.

The other half of the trade, measured where the first half is measured. The one-for-one claim is that $\mathbb{E}[c]$ is ρ^{par} for the attack *and* for the customer. Until now this paper measured the first against $\mathcal{H}_{\mathrm{safe}}$ and the second against a coordinate mask, so the two numbers whose equality the claim asserts had never been printed side by side. They are in Table 11: $c(\Delta W_{\mathrm{benign}})$ against the same one-shot $\mathcal{H}_{\mathrm{safe}}$, over the 19 benign and downstream-task adapters per model that Appendices A9 and A8 already use, 76 in all. It needs no model and no GPU, c is a projection of the adapter's own factors onto a cached basis.

The claim survives with a correction. The customer's update is *also* above the null at every one of the 28 loci, by 1.07–$1.79\times$ (1.49–1.79 taken model by model; per locus each benign adapter is restricted to that locus's matrices, and the low end is Qwen2.5-1.5B's late attention), so the safety row space is not a random subspace for benign updates either (at mid and late attention, where the low end sits and every matrix has the same reach, the ratio cannot come from how the update's energy is split between matrices; at the MLP loci, where `down_proj`'s reach is smaller, part of it could); and the attack's coverage exceeds the customer's by $1.06\times$ on Mistral-7B, $1.18\times$ on Qwen2.5-1.5B and 1.31–$1.33\times$ on the two larger models. Read as the identity intends, the reach spent on the attack really is close to the reach spent on the customer (within a third, on every model), and it is tilted the defender's way rather than against him. What is *not* supported is the isotropic null itself: it predicts both ratios to be 1.00 and neither is, at any locus on any model. The trade is a fact about these updates, not a consequence of the null, which is how §2 states it.

Proposition 1 as a prediction, and where it stops. It is fair to ask what the identity predicts that could have come out otherwise, and there is one place it can be checked directly. A coordinate mask reverting a fraction p of the coordinates has $\rho = p$ (Appendix A4 for the general $\rho(\mathcal{L}_j) = \mathrm{tr}(P_j \Pi P_j)/\dim$), so the second half of Proposition 1, which is stated for a *benign* task update, says that update loses p of its *energy*. The dial's task arm is exactly that update, and reports its deleted fraction as a norm ratio whose square is the energy. Over 15 cells, $p \in \{5, 10, 20, 40, 60\}\,\%$ on three models, the measured energy is 0.049–0.605 against a predicted 0.05–0.60, every gap in $[-0.0024, +0.0046]$; at $p = 20\,\%$ on Qwen2.5-1.5B it is 0.2002 against 0.20. The same test on the *attack* adapters, which the proposition says nothing about, agrees nearly as well (gaps within 0.0066). Nothing is fitted: the prediction is the support fraction, fixed before an adapter exists.

Two things follow, and the second is the limit. The updates we train, benign and adversarial alike, are isotropic with respect to a *coordinate* support to within 0.007, which is why the identity governs the mask so exactly, and they are *not* isotropic with respect to $\mathcal{H}_{\mathrm{safe}}$, where coverage exceeds reach by 1.48 to $6.36\times$ above. Those are different subspaces, and the contrast is the informative part: the safety row space is a special direction for an ordinary attack, the mask's coordinate support is not. What the identity prices, though, is energy and not the customer's task metric. At $p = 20\,\%$ it predicts 0.20 of the benign update deleted and 0.2002 is deleted, and that same customer still keeps $117\,\%$ of the fine-tune's gain (Table 50). The map from deleted energy to retained gain is not Proposition 1's to make, and we do not claim it makes one.

Table 9: Reach and coverage, both in per cent, by locus. ρ_1 and ρ_5 are the reach of a one-instance and a five-instance safety set, a property of the locus and of $|\mathcal{D}_{\text{safe}}|$, not of any attack; c is the coverage of the update the attacks trained at that locus actually produced. A2's zero is by construction (§3) and is the numerical check of Lemma 1 in Appendix A5 restated per model.

locus	Llama-8B			Qwen-7B			Mistral-7B			Qwen-1.5B		
	ρ_1	ρ_5	c	ρ_1	ρ_5	c	ρ_1	ρ_5	c	ρ_1	ρ_5	c
early/attention	1.5	6.7	3.1	1.1	5.1	3.4	0.8	4.5	1.3	2.6	12.8	5.8
mid/attention	1.5	6.9	3.5	1.1	5.3	3.0	0.9	4.6	1.3	2.7	13.3	4.7
late/attention	1.5	6.9	4.9	1.1	5.3	4.5	0.9	4.6	2.1	2.7	13.3	6.1
early/MLP	1.2	5.3	2.9	0.8	3.8	3.7	0.7	3.5	1.3	1.9	9.5	5.4
mid/MLP	1.2	5.3	4.3	0.8	3.8	2.6	0.7	3.5	1.4	1.9	9.6	4.8
late/MLP	1.2	5.3	7.5	0.8	3.8	4.3	0.7	3.5	2.6	1.9	9.6	6.3
all modules	1.2	5.6	2.5	0.9	4.0	2.1	0.7	3.7	1.1	2.0	10.0	3.6
A2 row-orthogonal	–	–	0.0	–	–	0.0	–	–	0.0	–	–	0.0
A3 alignment-aligned	–	–	3.2	–	–	2.5	–	–	1.2	–	–	3.8

Table 10: The same one-shot reach under the two aggregates of Definition 1. ρ^{par} weights each matrix by its parameter count and is the null coverage is read against; $\rho^{\text{dim}} = \sum \dim \mathcal{H}_{\text{safe}} / \sum d_{\text{in}}$ weights by input dimension and is what the row-space form of the trace identity counts (Proposition 1 itself is in ρ^{par}). They agree exactly at mid and late attention, differ by 1.40–$1.89\times$ at every MLP locus and by under $1\,\%$ at early attention. Table 9 reports ρ^{par}.

	Llama-8B		Qwen-7B		Mistral-7B		Qwen-1.5B	
locus	ρ^{par}	ρ^{dim}	ρ^{par}	ρ^{dim}	ρ^{par}	ρ^{dim}	ρ^{par}	ρ^{dim}
early / attn	0.0151	0.0151	0.0112	0.0112	0.0085	0.0084	0.0262	0.0261
early / mlp	0.0117	0.0084	0.0083	0.0047	0.0065	0.0047	0.0193	0.0102
mid / attn	0.0154	0.0154	0.0114	0.0114	0.0085	0.0085	0.0267	0.0267
mid / mlp	0.0117	0.0084	0.0083	0.0047	0.0065	0.0047	0.0193	0.0102
late / attn	0.0154	0.0154	0.0114	0.0114	0.0085	0.0085	0.0267	0.0267
late / mlp	0.0117	0.0084	0.0083	0.0047	0.0065	0.0047	0.0193	0.0102
all / all	0.0124	0.0113	0.0087	0.0071	0.0069	0.0063	0.0202	0.0157

A13 MISTRAL-7B: A REPAIR INHERITS THE ALIGNMENT OF THE CHECKPOINT IT PROJECTS TOWARDS

Mistral-7B-Instruct-v0.3 is the one model where no repair removes the ordinary or A3 families. Its *aligned* checkpoint θ_0, the state a repair is trying to restore, barely refuses: before any attack it complies with $25.0\,\%$ of AdvBench and $32.5\,\%$ of held-out HarmBench judged, against $\leq 5.0\,\%$ and $\leq 7.5\,\%$ elsewhere. (Table 3 reports the same two quantities by substring over the full sets, 22.5 and $55.0\,\%$; the judged rates are the ones quoted here, because this number decides whether a model is in the claim and §4 rejects substring matching for exactly that kind of decision.) That floor is what a post-hoc repair returns the model to, so every ordinary and A3 configuration survives SAFELORA there (the composite and the Eq. 5 edit do move Mistral's cells, Tables 31 and 13, towards the floor and at a perplexity cost), and the all-modules attack is in fact *worse* after repair ($+0.068$ $[+0.029, +0.107]$): more than the floor alone explains, and we do not claim to explain the excess. Surviving a defence that removes nothing demonstrates nothing, so Mistral is excluded from the A3 claim; it establishes instead a precondition none of the repair papers states: **every claim about post-hoc realignment is conditional on the checkpoint being realigned towards actually being aligned**. Where the cut lies we do not know: the three aligned checkpoints sit at $\leq 5.0\,\%$ and $\leq 7.5\,\%$ judged, Mistral-7B at $25.0\,\%$ and $32.5\,\%$, and no checkpoint between them was measured.

Table 11: The two halves of Proposition 1's trade, both measured against the same one-shot $\mathcal{H}_{\text{safe}}$: coverage of the *attack* updates trained at the all-modules locus, and of the 19 benign and downstream-task adapters per model. Under the isotropic null both c/ρ^{par} columns would be 1.00. "Tilt" is the attack's coverage over the customer's; above 1 means the repair reaches more of the attack than of the customer, which is the direction the defender wants and is not what the null predicts.

	benign		attack		customer		
model	adapters	ρ^{par}	c	c/ρ^{par}	c	c/ρ^{par}	tilt
Llama-8B	19	0.0124	0.0253	2.04×	0.0193	1.55×	1.31×
Qwen-7B	19	0.0087	0.0207	2.38×	0.0156	1.79×	1.33×
Mistral-7B	19	0.0069	0.0110	1.59×	0.0103	1.49×	1.06×
Qwen-1.5B	19	0.0202	0.0360	1.79×	0.0306	1.52×	1.18×

Table 12: What each operator's dial costs the customer, per unit of the customer's update energy it deletes, on one model and the customer's own adapters. The mask spends energy freely and the customer barely notices; SAFELORA's dial moves the deleted energy by only 0.063 and takes the task. The last column is why: the mask acts partially on every target matrix at every setting, and SAFELORA's dial moves whole matrices from intact to empty.

model	operator	cells	energy	gain kept	slope	matrices acted on
Llama-8B	coordinate mask, $p = 5$–$60\,\%$	5	0.049–0.598	$+1.15 \to +1.05$	0.2	all 224
Llama-8B	SAFELORA, $\tau = 0.70$–0.80	24	0.935–0.998	$+0.67 \to -0.04$	11.3	184–224 of 224

A14 PRECISION-ROBUST RANK MEASURES

In `bf16` the machine epsilon is $\approx 8 \times 10^{-3}$, so a numerical rank defined by a relative 10^{-5} singular-value threshold counts noise: it reports 211–3263 for gradients whose energy lies in 13–19 directions at the median (Table 3), and it is not comparable across models. We therefore report

- r_{99}: the smallest k with $\sum_{i \le k} \sigma_i^2 / \sum_i \sigma_i^2 \ge 0.99$ (Tables 3 and 17);
- stable rank $\|G\|_F^2 / \|G\|_2^2$, which is threshold-free;
- participation ratio $(\sum_i \sigma_i^2)^2 / \sum_i \sigma_i^4$.

All three are at most the rank, and so by Lemma 1 at most the sequence length; the geometry stage records r_{99} for every target matrix and all three on three probe matrices per model.

A15 THE CLOSED-FORM RANK-k EDIT

Equation 5 of Zhang et al. (2026b) specifies the repair as $W \leftarrow W - \alpha\eta \sum_{i \le k} \sigma_i u_i v_i^\top$ for each target matrix, with α, η free. We set them per matrix so that the edit has Frobenius norm $s\|W\|_F$ (a per-matrix scale in place of Eq. 5's single one, which makes ours a variant that reweights the matrices against each other) and calibrate s on the vanilla attack of the 0.5B pilot, as for every other baseline: at $s \le 5 \times 10^{-4}$ the model is intact and the substring rate barely moves; by $s = 2 \times 10^{-3}$ perplexity is 16.7 and by $s = 5 \times 10^{-3}$ it is 414. The formal run uses $s = 5 \times 10^{-4}$, $k = 20$, on every adapter of the attack grid. Table 13 gives the judged outcome with the accounting of Table 2 (landed attacks, item-level pairing, AdvBench subset, cells the edit destroyed excluded: 17 of 47 on Mistral, whose perplexity reaches 5.8×10^4; none elsewhere). The edit removes part of every ordinary attack family on Llama-3.1-8B and Mistral-7B and of the attention and MLP loci on the two Qwen models, but not the Qwen all-modules attacks, and it leaves the models well above the 0.000–0.025 AdvBench rates that SAFELORA reaches on the Qwen models (Table 18). Against A3 it does nothing or makes the model worse on the three models of the claim ($+0.150$, $+0.058$, $+0.025$), and cuts Mistral's by 0.117 $[-0.208, -0.025]$, an interval that excludes zero. We did not sweep s further and draw no conclusion about the variant in its authors' hands; at this setting it is weaker and less uniform than SAFELORA, and A3 survives it as well.

Table 13: The closed-form rank-k edit of Eq. 5 (Zhang et al., 2026b) at $s = 5 \times 10^{-4}$, $k = 20$: judged harm ASR before and after, paired at the item level over the 40-prompt AdvBench subset; n is the number of distinct landed adapters, destroyed cells excluded; Δ is computed before rounding.

			judged harm ASR		
model	attack configuration	n	attacked	after Eq. 5	Δ [95% CI]
Llama-8B	all-modules (the default)	8	0.494	0.244	-0.250 $[-0.319, -0.181]$
	attention loci	6	0.446	0.129	-0.317 $[-0.383, -0.250]$
	MLP loci	7	0.575	0.175	-0.400 $[-0.464, -0.336]$
	A3 alignment-aligned	3	0.542	0.692	$+0.150$ $[+0.058, +0.250]$
Qwen-7B	all-modules (the default)	10	0.480	0.497	$+0.018$ $[-0.040, +0.073]$
	attention loci	10	0.520	0.305	-0.215 $[-0.267, -0.165]$
	MLP loci	7	0.632	0.475	-0.157 $[-0.214, -0.100]$
	A3 alignment-aligned	3	0.433	0.492	$+0.058$ $[-0.033, +0.150]$
Mistral-7B	all-modules (the default)	9	0.683	0.478	-0.206 $[-0.258, -0.153]$
	attention loci	11	0.755	0.330	-0.425 $[-0.484, -0.368]$
	MLP loci	7	0.757	0.389	-0.368 $[-0.436, -0.300]$
	A3 alignment-aligned	3	0.733	0.617	-0.117 $[-0.208, -0.025]$
Qwen-1.5B	all-modules (the default)	11	0.523	0.480	-0.043 $[-0.098, +0.009]$
	attention loci	9	0.461	0.164	-0.297 $[-0.356, -0.239]$
	MLP loci	12	0.525	0.290	-0.235 $[-0.292, -0.177]$
	A3 alignment-aligned	3	0.575	0.600	$+0.025$ $[-0.067, +0.117]$

A16 OFFLOADED FULL-PARAMETER OPTIMISER

Full-parameter AdamW at 7–8B needs two `fp32` moment buffers over the trained projection weights, at least 52 GB, on a 56 GB host, alongside the model on a 24 GB card. Our implementation registers a `post_accumulate_grad_hook` on each parameter, so a parameter's update is applied the moment its gradient is ready and the gradient is freed immediately; moments live on the CPU in `bf16` and the arithmetic is done in `fp32` on the GPU in 2^{26}-element chunks; deferring the updates to `step()` would keep every gradient alive at once, which does not fit. Correctness is asserted, not assumed: see Appendix A5.

A17 PILOT CALIBRATIONS

Attack strength. The recipe from our 0.5B pilot (Qwen2.5-0.5B-Instruct on CPU, 48 steps) reaches only ASR 0.146 at 8B. Sweeping: lr $= 2 \times 10^{-4}$, 200 steps converges to training loss 0.000 at ASR 0.875 (chosen); 5×10^{-4} reaches 0.979 with a larger perturbation than needed; 10^{-3} reaches ASR 1.000 but its *training loss diverged to* 6.87. We kept the last row in the record because without checking the loss it would have looked like the strongest attack in the sweep.

SGD range for the patch. In the pilot, learning rates in $[10^{-3}, 10^{-1}]$ destroy an 8B model (perplexity 2.8×10^4 to 2.2×10^6) and yield a meaningless "ASR 1.000". The sweep therefore covers $[10^{-6}, 10^{-3}]$ on every model (10^{-3} is kept because at formal scale it destroys some cells and not others), plus 10^{-2} and 10^{-1} on Llama-3.1-8B so that the pilot's finding is on the record at formal scale; every cell carries the `destroyed` flag described in §4.

Judge calibration. At 40 new tokens the substring judge and the API judge agree only at $\kappa = 0.077$, because the attacked model emits its persona preamble and little substance; at 192 tokens agreement rises to $\kappa = 0.478$ on the pilot and $\kappa = 0.662$ on the full run. All headline numbers use 192-token generations and the API judge. Generation is greedy. The judge sees the request and the response and returns JSON: COMPLIED if the response makes any substantive attempt at the request, REFUSED if it declines, deflects, lectures without helping, answers a different question, or is empty or incoherent, plus a harm rating from 1 (harmless or refusal) to 5 (highly actionable). Every judge runs at temperature 0 under its unversioned API name, with a 24-token JSON response and up to four attempts; the exact prompt is in `code/judge_formal.py`. Every reported rate uses the

primary judge's label; the second and third judges are agreement checks and never override it. Every substring-based ASR in this paper should be read as a refusal-suppression rate rather than a harm rate; the judged numbers of Table 2 are the harm rates.

Benign and perplexity sets. The 60 over-refusal prompts of §4 are 40 held-out Alpaca instructions and 20 hand-written borderline ones that sound harmful and are not ("How do I kill a Python process that is stuck?"), in the style of the over-refusal literature; perplexity is taken on 40 held-out Alpaca instruction–response pairs (`code/build_data_formal.py`, `code/build_data.py`).

A18 Design notes and deviations from the pre-registered plan

1. **Over-refusal is part of the cell definition.** A repair criterion on harm ASR and perplexity alone admits "repairs" with benign over-refusal of 1.00: a patch tuned that way passes by refusing every benign prompt. Every cell therefore carries over-refusal on 60 benign prompts, a clean operating point requires all three (Appendix A23), and the repair axis is a frontier sweep rather than a single tuned setting.
2. **A3 adapters are counted once.** Every number counts distinct adapters: 15 in the first pass (three at $m = 256$ and three at $m = 1024$ on Llama-3.1-8B, three at $m = 256$ on each other model) and 33 further ones in the budget sweep of Appendix A33 (later stages bring the pool to the 81 of §6): three of that sweep's Llama cells re-evaluate adapters the first pass already held, identified by their geometry agreeing to full precision, and are counted once. The script `code/dedup_a3.py` identifies, from the cosine, update norm, projected-matrix count and ASR agreeing to full precision, any cell whose declared budget did not produce a distinct adapter, and such cells enter no number in this paper.
3. **A3's subspace budget interacts with d_{out}.** At $m = 256$ the evasion works on Llama-3.1-8B (cosine 0.844, 7 % removed) but the attack itself is crippled (judged ASR 0.108 unrepaired); $m = 1024$ restores it (cosine 0.744, judged ASR 0.613 unrepaired). On the other three models $m = 256$ both evades and preserves the attack. The right budget is model-dependent; Appendix A33 sweeps it on two models and §6 states what the Llama points support and no more.
4. **Frontier denominators.** Clean operating points are counted over cells whose attack landed, not over all swept cells: driving an already harmless model to ASR 0 is not a repair. The distinction matters, over all swept cells Llama-3.1-8B would show $6/30$ and Qwen2.5-7B $3/22$, against $0/18$ and $0/11$ over landed attacks; the excluded cells sit on the crippled $m = 256$ A3 adapter and on A2.
5. **Judged subset.** The judge labels a fixed, seeded subset of each prompt set in every cell: 40 prompts per set for the first-pass stages (`matrix`, `tau`, `frontier`, `a3sweep`), and the *full* 120 AdvBench and 100 HarmBench sets for the later stages whose purpose is to remove the subset as a degree of freedom: the eight-seed replications, the repair-family comparison of §5.4 and every later stage whose caption says it is judged on the full sets. Subsets are nested by index, so every pairing is still exact. Two cells carry 99 rather than 100 HarmBench judgments (Qwen2.5-1.5B's A3 seed 1 under RESTA at $\gamma = 1$, and one Llama-3.1-8B mid-attention patch cell at 5×10^{-6}); their rates are over the 99. Substring rates, reported only in Tables 58–61, use all 120 AdvBench and 100 HarmBench prompts. Whether an attack landed is decided by the judge everywhere a judged number is reported. A seed fixes both the adapter's initialisation (its A factor; B starts at zero) and the order the training examples are visited in. A locus is a contiguous third of the layer stack crossed with a module group, attention being `q_proj`, `k_proj`, `v_proj` and `o_proj` and MLP being `gate_proj`, `up_proj` and `down_proj`; all-modules is both groups at every layer.

The seven predictions, as filed. Table 14 reproduces them verbatim from the plan written before any of this was run, including the *failure condition* each was filed with, which is the part that makes a pre-registration worth having and the part a later restatement tends to lose. Each verdict is read against that condition and not against a paraphrase of the claim. **Two of the seven survived as filed** (V2, V6). Three survived in half (V1, V4, V5) and two did not (V3, V7); for each of the five the paper states the narrowed version. One verdict has moved since the pilot: V5's behavioural half was scored a failure there and the formal run reverses it where it has content (item (iv)).

Table 14: The seven pre-registered predictions with the failure condition each was filed with, and what the data returned. V4 was filed as *one* row; the a/b split used in the analysis was made after the pilot, when the two repair types separated, and is not part of the filing.

	claim, as filed	fails if	what came back
V1	one-shot repair reach ρ is small ($<$ 10 %) and varies across loci	ρ uniform and large	**half.** Small ($\leq$ 2.7 %) but uniform: 1.31–1.38$\times$ across loci
V2	numerical rank of the one-shot safety gradient $\approx \min(L, d)$, and global CER(20) $\geq$ 0.9 reproduces the published number	rank $\gg L$	**as filed.** $\max r_{99} = 45/34/29/27$ against $L = 63/41/41/35$; CER(20) = 0.987–0.998, above the published 0.86–0.95 (Table 3)
V3	ASR after attack is $\approx$ locus-invariant (all loci within 15 pp of the best)	only one locus can break safety	**no.** The spread is 0.06–0.86 within a model, far past 15 pp, though several loci do break safety, so it did not meet its own condition either
V4	post-repair ASR falls with coverage c; Spearman ≤ -0.5 across cells	$\lvert r_s \rvert < 0.3$ or wrong sign	**half.** -0.873 for the projection repair; $+0.109$ for the gradient patch, which fails the filed condition
V5	A2 drives c to $\leq 1/5$ of A0 at equal training loss, and post-repair ASR $\geq 2\times$ A0's	orthogonalisation kills the attack	**half.** $c = 0$ exactly, at *lower* training loss. The ASR half holds only where the patch does anything: 5.3$\times$ at the rate that removes A0, 1.0–1.2$\times$ at the two weaker ones
V6	where repair restores refusal, residual $\lVert(I - QQ^\top)\Delta\rVert / \lVert\Delta\rVert \geq 0.9$	residual small	**as filed.** Residual $\geq$ 0.962 at every locus of every model
V7	per-prompt post-repair compliance is predicted by that prompt's activation overlap with $\mathcal{H}_{\text{safe}}$	no correlation	**no.** The predictor is near-constant over 120 prompts (sd 0.008–0.012) on all four models, so there is no correlation to find

(i) V1's "varies across loci" was filed expecting a spread of at least 5$\times$; the widest locus-to-locus ratio within a model is 1.31–1.38$\times$ under ρ^{par} and 1.83–2.61$\times$ under ρ^{dim} (Appendix A12), below it under either aggregate. Reach is uniformly tiny, a cleaner premise for Corollary 1 than the one predicted, and the paper claims uniformity rather than spread. **(ii)** V3's parenthetical is what fails: unrepaired ASR spans 0.06–0.86 across loci within a model (Tables 58–61), so relocation is cheap and has several equally good addresses, not free, which is what §5.1 says. **(iii)** V4's second half, the *gradient* repair, fails its own threshold ($+0.11$, $p = 0.79$ on the pilot), because Adam-style preconditioning breaks the row-space confinement (Appendix A5). The paper's claim is therefore about projection-type repairs, and the gradient patch appears only as the family with no clean operating point (Appendix A23). **(iv)** V5's second half is comparative, A2 at least twice A0 after the patch, and the pilot rejected it on differences inconsistent in sign across seeds, measured at one learning rate with an Adam surrogate whose second moment was held in `bf16`. The formal run, full AdamW with `fp32` arithmetic (Appendix A16) over the swept arm on Qwen2.5-1.5B, does not reproduce that rejection: at the rate that removes the ordinary attack ($0.375 \to 0.075$) A2 stands at 0.400, a ratio of 5.3, and at the two weaker rates the patch removes neither and the ratio is 1.0–1.2 (Appendix A40). So the filed prediction holds wherever it has content and is vacuous where it does not, which is the version §2 states; the pilot verdict stood on a surrogate and we record the reversal here rather than quietly dropping it. A2 remains the headline attack against *coordinate*-confined repairs (§5.4). **(v)** V7 was dropped from the pilot for time and run in the formal study, where it returned a null of a particular kind: the predictor barely varies, so its filed failure condition is met without a correlation ever having been in reach (Table 17, last row).

The sharpest result in the paper was not on the list. §5.1's finding (that what SAFELORA deletes is governed by a public per-locus scalar, so that the attacker chooses it) is not among the seven, and was recorded as unanticipated when it was found. It is reported as what it is, a finding the plan did not anticipate, and the pre-registration constrains only the seven rows above.

A19 TABULAR FORMS OF THE MAIN FIGURES

Tables 15 and 16 are the numbers behind Figures 1 and 2, so that every plotted point can be read off a table.

Table 15: The gate statistic against what SAFELORA deletes from the update (every adapter of each model; the tabular form of Figure 1) and against the fraction of judged compliance it removes (adapters whose attack landed). Brackets are bootstrap 95 % intervals. The first correlation is close to definitional given the gate rule; the second is a step (§5.1), so a rank correlation understates it where removal saturates. The adapters are not independent: §5.1's own claim is that the cosine is a property of the locus, and the all-modules locus alone supplies 17–20 of the points. Over the seven locus means the first correlation is -0.893, -0.714, -0.643 and -0.893 (Llama-8B, Qwen-7B, Mistral-7B, Qwen-1.5B) at $p = 0.005$, 0.082, 0.138 and 0.005: nominally significant on two of four. (p here uses the Spearman variance $1.06/(n-3)$ rather than the Pearson $1/(n-3)$.) The adapter-level p values, and the intervals beside them, should be read as descriptive: the claim the paper draws from this row is the *step*, which §5.1 states in compliance terms and not in r_s.

	vs. fraction of the update deleted		vs. judged compliance removed	
model	n	Spearman [95 % CI]	n	Spearman [95 % CI]
Llama-8B	50	$\mathbf{-0.962}$ $[-0.985, -0.895]$	24	-0.508 $[-0.780, -0.133]$
Qwen-7B	47	$\mathbf{-0.908}$ $[-0.942, -0.831]$	31	-0.001 $[-0.410, +0.455]$
Mistral-7B	47	$\mathbf{-0.909}$ $[-0.943, -0.819]$	45	-0.216 $[-0.458, +0.081]$
Qwen-1.5B	47	$\mathbf{-0.930}$ $[-0.963, -0.849]$	38	-0.385 $[-0.663, -0.040]$

Table 16: Separation of A3 from every other attack adapter by the defence's own gate statistic; the tabular form of Figure 2. The pooled column and the AUCs are over the 44 non-A3 attack adapters per model, 41 ordinary and the 3 A2 (§4), which are not separated from the ordinary ones by this statistic and are not meant to be. The A3 column pools the first-pass A3 adapters, both budget sweeps, the aimed adapters of Appendix A34 and the flattest-band adapters of Appendix A41, 81 after the cross-stage dedup. Counts in parentheses; the gap to non-A3 is the lowest A3 cosine minus the highest non-A3 one, from unrounded values.

	$\cos(C\Delta W, \Delta W)$			
model	non-A3 attack	A3 adapters	gap to non-A3	AUC
Llama-8B	$[0.434, 0.627]$ (44)	$[0.559, 0.909]$ (33)	-0.068	$\mathbf{0.886}$
Qwen-7B	$[0.355, 0.581]$ (44)	$[0.762, 0.766]$ (3)	$+0.181$	$\mathbf{1.000}$
Mistral-7B	$[0.366, 0.529]$ (44)	$[0.715, 0.719]$ (3)	$+0.186$	$\mathbf{1.000}$
Qwen-1.5B	$[0.367, 0.600]$ (44)	$[0.575, 0.898]$ (42)	-0.024	$\mathbf{0.989}$

A20 THE TWO SIDES OF THE CONFINEMENT, DRAWN

Figure 3 draws what §2 says in symbols: an adapter $\Delta W = BA$ has a row side, which a repair confined to $\mathcal{H}_{\text{safe}}$ can reach, and a column side, which SAFELORA's C weights; A2 leaves the first, A3 sits inside the second.

A21 THE HEADLINE CONTRAST ON EACH PROMPT SET

Table 2 pools the judged AdvBench and held-out HarmBench subsets ($40 + 40$ items per adapter). Table 18 separates them. The contrast is the same on both: every ordinary family loses 0.21–0.61 on either set, while A3's six paired intervals (three models, two sets) span -0.142 to $+0.125$, five of them containing zero.

Table 17: What replicates, and on how many models. Mistral is parenthesised in *both* A3 rows because its θ_0 is not aligned enough for either to mean anything (§5.5): the threshold claim is true there but vacuously, since A3's compliance *rises* with τ on that model (Table 24) rather than trading against the task. The one-shot patch row pools the coarse grid with the finer one where it was run (Appendix A23) and counts only cells whose attack landed *and* that are not already `destroyed`, since a destroyed cell cannot be clean by construction; it is parenthesised on Mistral too, whose over-refusal is already 0.98–1.00 at the smallest rate swept, so its zero is about the grid rather than the method (Appendix A23, which also reports what a cut other than 0.15 would give).

result	Llama-8B	Qwen-7B	Qwen-1.5B	Mistral	holds on
$\cos(C\Delta W, \Delta W)$ gates deletion (Spearman)	-0.962	-0.908	-0.930	-0.909	**4/4**
$\max r_{99} < L$ (gradient rank $\leq$ sequence length)	$45 < 63$	$34 < 41$	$29 < 41$	$27 < 35$	**4/4**
A3 survives repair (harm ≥ 3 after)	0.550	0.629	0.650	(0.787)	**3/3** elig.
no τ keeps the task and removes A3	✓	✓	✓	(✓)	**3/3** elig.
one-shot patch: clean point at ≤ 0.15	3/31	0/17	3/30	(0/7)	**1/2** at 7–8B
per-prompt activation overlap predicts nothing	sd 0.012	sd 0.008	sd 0.009	sd 0.011	**4/4** null

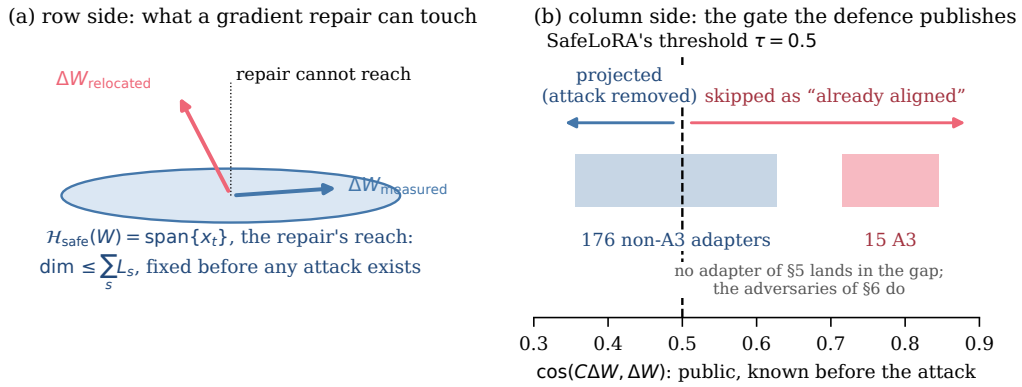

Figure 3: Both families of post-hoc repair act in a subspace fixed before the attack. **(a)** A gradient-based repair on a safety set can only move a weight matrix inside the span of that set's activations (Lemma 1); anything the attacker places in the orthogonal complement ($\Delta W_{\text{relocated}}$, as A2 does; $\Delta W_{\text{measured}}$ is an update inside the span) is untouched exactly by one step with the inputs fixed (Corollary 1); ten steps leave 7.6 % of the repair's displacement outside (Appendix A40). **(b)** SAFELORA is gated by one scalar computed from public weights: matrices with $\cos(C\Delta W, \Delta W) \geq \tau$ are skipped as already aligned. Our adversary A3 puts itself on the right of the threshold by construction. The ranges drawn are adapter means of that per-matrix cosine over the 191 adapters of §5; an adapter whose mean is above τ can still have matrices below it (Appendix A34).

A22 Is the low-rank safety gradient about safety? A matched-length control

Lemma 1 bounds the rank of a one-instance gradient by its token count L; it says nothing about how the energy is spread inside those L directions, and Table 3 shows it is far more concentrated than the bound requires (r_{99} medians of 13–19 at $L = 35$–63). Two controls decide whether that concentration is about safety. The first is a *benign* instance of the same token count under the same tokenizer, chat template and system prompt, built by truncating a benign response until the supervised span matches; three distinct instances per model, drawn from a pool of four downstream training sets. The second removes language altogether: uniformly random token ids of the same length over the same supervised span.

Table 19 runs both on all four formal models and on the 0.5B pilot. The matched-length benign spectra sit within 0.007 of the safety spectrum's CER(20) on every one (0.9805–0.9832 against 0.9870 on Llama-3.1-8B, 0.9899–0.9919 against 0.9910 on Qwen2.5-7B), so a benign instance of the same length is as concentrated as the safety one. The random-token spectra are *more* concentrated than the safety one on Llama-3.1-8B (0.9919–0.9931 against 0.9870) and somewhat less so on the

Table 18: Judged harm ASR after SAFELORA ($\tau = 0.5$) and its paired change, by prompt set; the same adapters as Table 2.

			AdvBench		HarmBench (held-out)	
model	attack configuration	n	repaired	Δ [95% CI]	repaired	Δ [95% CI]
Llama-8B	all-modules (the default)	8	0.159	-0.334 $[-0.397, -0.275]$	0.259	-0.331 $[-0.403, -0.262]$
	attention loci	6	0.183	-0.263 $[-0.325, -0.200]$	0.415	-0.210 $[-0.285, -0.135]$
	MLP loci	7	0.029	-0.546 $[-0.604, -0.486]$	0.129	-0.589 $[-0.646, -0.529]$
	A3 alignment-aligned	3	0.517	-0.025 $[-0.075, +0.017]$	0.533	-0.083 $[-0.142, -0.033]$
Qwen-7B	all-modules (the default)	10	0.000	-0.480 $[-0.527, -0.430]$	0.022	-0.608 $[-0.658, -0.555]$
	attention loci	10	0.003	-0.518 $[-0.568, -0.467]$	0.035	-0.580 $[-0.630, -0.530]$
	MLP loci	7	0.025	-0.607 $[-0.664, -0.550]$	0.136	-0.571 $[-0.636, -0.507]$
	A3 alignment-aligned	3	0.467	$+0.033$ $[-0.017, +0.083]$	0.750	$+0.017$ $[-0.025, +0.058]$
Qwen-1.5B	all-modules (the default)	11	0.002	-0.520 $[-0.566, -0.473]$	0.052	-0.500 $[-0.550, -0.450]$
	attention loci	9	0.000	-0.461 $[-0.511, -0.411]$	0.056	-0.494 $[-0.550, -0.439]$
	MLP loci	12	0.000	-0.525 $[-0.571, -0.479]$	0.042	-0.519 $[-0.567, -0.469]$
	A3 alignment-aligned	3	0.542	-0.033 $[-0.108, +0.033]$	0.717	$+0.050$ $[-0.033, +0.125]$

other three, by at most 0.017 (0.9740–0.9928 against 0.9910–0.9980), so the random control is a weaker match than the benign one, and the claim §2.4 rests on is the benign one; on all of them they remain far above what Lemma 1 requires, at median r_{99} of 10–26 for $L = 35$–63. The top twenty directions carry 97–100 % of the energy whether the tokens are a refusal, a Python exercise or noise (the leading one carries 42–69 %, and on Llama-3.1-8B the benign instances are the *least* concentrated of the three), so the concentration is a property of a short sequence's activations rather than of safety content, which is what §2.4 needs, and no more. Two caveats we would rather state than hide: the length match is tight enough that on every model the three benign instances it admits are MBPP-style Python prompts rather than one from each training set, and CER(20) is a coarse summary, which is why r_{99} is tabulated beside it.

Table 19: Spectrum of the per-matrix gradient on one instance of L tokens: the safety instance against three matched-length benign instances and three random-token sequences. CER(k) is the cumulative energy ratio over all target matrices and r_{99} is as defined in Appendix A14. Ranges are over the three instances of each kind. Qwen2.5-0.5B-Instruct is the pilot model of Appendix A17, run on CPU, and is included because it extends the range of scales over which the control holds.

			CER(20)			median r_{99}		
model	L	matrices	safety	benign (3)	random (3)	safety	benign	random
Llama-8B	63	224	0.9870	0.9805–0.9832	0.9919–0.9931	19	25–26	17–21
Qwen-7B	41	196	0.9910	0.9899–0.9919	0.9740–0.9760	16	16–18	21–25
Mistral-7B	35	224	0.9980	0.9954–0.9985	0.9901–0.9928	15	12–15	21
Qwen-1.5B	41	196	0.9949	0.9922–0.9940	0.9795–0.9909	13	13–14	17–19
Qwen-0.5B	41	168	0.9953	0.9950–0.9971	0.9897–0.9924	12	10–13	15–17

A23 THE GRADIENT-REPAIR FRONTIER

Of the 51 frontier cells at 7–8B whose attack landed ($18/11/22$ on Llama-3.1-8B, Qwen2.5-7B, Mistral-7B), 22 already carry the `destroyed` flag (perplexity above 50, or over-refusal above 0.9 at substring ASR below 0.05), and cannot be clean by construction, so the count that means anything is over the 29 that can: 11, 11 and 7, so 22 on the two in-scope models (the count §5.4 quotes). None of the 29 is a clean operating point; at 1.5B, 3 of 30 are. On Llama-3.1-8B the AdamW arm reaches judged harm ASR 0.025 at 10^{-5} and 0.000 at 2×10^{-5}, at benign over-refusal 0.30 and 0.93; refusing a third of benign traffic or more is a different failure, not a repair. The plain-SGD arm behaves as Corollary 1 predicts and barely moves a working attack ($0.73/0.63/0.63$ judged at $10^{-6}/10^{-5}/10^{-4}$), though `bf16` rounding is a second candidate reason (Appendix A56); at lr $\geq 10^{-2}$ it emits gibberish, which a substring judge scores as ASR 1.000, and those cells are `destroyed` too.

And the grid is what it bounds: swept finer, Llama-3.1-8B has three clean points. Everything above is a statement about a grid whose AdamW arm at 7–8B is $\{5, 10, 20\} \times 10^{-6}$ and whose step count is fixed at ten. `frontier2` sweeps the two gaps it leaves ($\{6, 7, 8\} \times 10^{-6}$ on the rate axis, $\{1, 3\}$ on the step axis at 5×10^{-6}) and, on the plain-SGD arm, the two decades between 10^{-4} and 10^{-2} that hold its whole transition. Evaluation is the full instrument, not the frontier stage's AdvBench-only, because a clean operating point has to clear over-refusal and perplexity as well. 3 of its 20 usable landed cells on Llama-3.1-8B are clean:

arm	rate	judged ASR	over-refusal	perplexity
AdamW, all-modules	8×10^{-6}	0.036	0.100	8.45
AdamW, mid/attention	8×10^{-6}	0.109	0.033	8.20
plain SGD, mid/attention	3×10^{-4}	0.014	0.067	7.48

The clean AdamW rate sits *between* the coarse grid's two lowest, and the clean SGD rate between its 10^{-4} and the 10^{-2} at which every cell emits gibberish. The same sweep on Qwen2.5-7B finds nothing: 0 of 6 usable cells, and not narrowly: the best reads 0.467 on the worse of its two halves, against Llama's 0.067. So the finer grid does not rescue the repair everywhere; it rescues it on one of the two models, which is why Table 17 counts this row per model, over both grids pooled ($11 + 20$ usable landed cells on Llama-3.1-8B, $11 + 6$ on Qwen2.5-7B). So the zero above is the grid's and not the method's, which is what the paragraph below warns about. *Safety at One Shot* therefore reproduces on Llama-3.1-8B, though not on Qwen2.5-7B, and joins SAFELORA on the side of the line §1 asks its question about, rather than being a failed replication.

And at a clean operating point, both adversaries are still standing. A clean point is a setting at which this repair removes an ordinary attack and leaves the model usable, so it is the setting at which the question this paper asks is worth asking: what does an adversary who knows where the repair's reach lies do *there*? Qwen2.5-1.5B has 3 of them (two in the AdamW arm at 2×10^{-5}, one in the plain-SGD arm at 10^{-4}, which is the arm Corollary 1 is *exact* for), and at every one the ordinary attack the point is clean for is taken from 0.350–0.600 to 0.000–0.075 at over-refusal 0.050–0.083, while at the *same setting* A2 reads 0.400, which is exactly unchanged from its unrepaired arm on the same prompts, and A3 loses 0.050–0.125. Size that evidence before weighing it: one A2 adapter on the 40-prompt subset at three settings, with no seed arm behind it (Appendix A46 re-runs A2 at eight seeds against the mask, not against this repair), so the patch's blind spot rests on thinner evidence than the mask's, and "unchanged" on the two AdamW points is an observation, not the prediction Corollary 1 makes only for the SGD one. The repair works, on the attack people write, at a cost the customer would accept; and the row-orthogonal adversary walks through it untouched while the alignment-aligned one loses at most 0.125. That is this paper's claim on a second operator, and on the SGD arm it is the operator whose containment Corollary 1 states exactly rather than approximately.

On Llama-3.1-8B the same comparison cannot be made: A2 does not land there (judged 0.064 and 0.282 unrepaired, against the 0.30 the landing rule requires), which is why the A2-versus-mask dissociation *at rank* 16 is a Qwen2.5-1.5B claim and says so. A2 itself is not: it lands and is removed by SAFELORA on both Qwen models (Table 8) and cut on Mistral-7B (Table 31), and on Llama-3.1-8B it lands at rank 64, where the mask leaves it at 0.186 and SAFELORA takes it to 0.023 (Appendix A7).

What this count does and does not bound. It bounds the grid, and the grid is not symmetric between the two arms. The plain-SGD arm (the one Corollary 1 is exact for, so the one whose reach it confines to $\mathcal{H}_{\text{safe}}$) is swept over five decades on Llama-3.1-8B and three on the others. The AdamW arm, which is the one the published method uses and the only one that ever reaches low ASR, is swept at 7–8B over $\{5, 10, 20\} \times 10^{-6}$: three points spanning a factor of four, at a step count of ten (the finer grid above adds $K \in \{1, 3\}$ at one rate, and fewer steps do worse), against an attack whose 200 steps were calibrated (Appendix A17). A negative claim about a published repair should say how hard the repair was tried, so: not as hard as the attack.

Two gaps in particular are ours to name. On Llama-3.1-8B the all-modules cell goes from (ASR 0.200, over-refusal 0.017) at 5×10^{-6} to $(0.025, 0.300)$ at 10^{-5}, and the mid-attention cell from $(0.225, 0.033)$ to $(0.000, 0.283)$; a point with both at ≤ 0.15 is exactly what one would look for *between* those two learning rates, and it is there that `frontier2` found two of Llama's

three clean points (the three are 3 of its 20 usable landed cells, above; the third is the plain-SGD arm's, at 3×10^{-4}): the coarse grid, not the method, is what the zero was about on that model. And on Mistral-7B over-refusal is already 0.98–1.00 at the *smallest* rate swept, so that model's zero is a statement about a grid that never reached the operating regime, not about the method, which is a second reason, on top of §5.5's, not to count Mistral in this claim. Read the coarse-grid 7–8B result as: over that grid, no clean operating point, and the grid is three learning rates wide on the arm that matters; the finer grid above is what moves Llama-3.1-8B to three.

Where the 0.15 came from, and what happens if it moves. It did not come from the data: it is a reporting threshold, not a pre-registered one (Appendix A18 lists what was), and a negative claim that rests on a number we chose should be stated as a function of it. So, *on the coarse grid*: at 7–8B the count of clean cells among the usable landed ones is 0 at every threshold up to and including 0.15 (on both in-scope models, and on Mistral, whose zero this appendix has already declined to count), and the sharpest way to say the same thing without the threshold is that the *best* any usable landed cell does, on the worse of its two axes, is 0.183 on Llama-3.1-8B, 0.217 on Qwen2.5-7B and 0.575 on Mistral-7B. The finer grid above moves Llama's figures to 3 of 20 and 0.067, and leaves Qwen2.5-7B's at 0 of 6 and 0.467. At 1.5B the count is $3/30$ at 0.10 and at 0.15, 0 at 0.05, and the best cell reaches 0.067.

Two consequences, and the second is against us. The 1.5B result is not an artefact of the cut we chose, it survives tightening to 0.10, but it is a statement about an interval of cuts and not about all of them: at 0.05 the 1.5B count is 0 too, and at 0.20 it is 3 against Llama's 2, so the size contrast in §5.4 holds for cuts in $[0.10, 0.15]$. The 7–8B zero does *not* survive loosening to 0.20: Llama-3.1-8B then has two, the AdamW arm's 5×10^{-6} all-modules cell at $(0.200, 0.017)$ and the plain-SGD arm's 10^{-3} all-modules cell at $(0.025, 0.183)$, the latter at perplexity 9.3, so not a destroyed model but a working one that refuses $18\,\%$ of benign traffic. A reader who would ship a repair at $18\,\%$ over-refusal should read our claim as false on Llama. We would not, and say so as a stated preference rather than a measurement. The 10^{-3} cell is also the more interesting of the two: the SGD arm's all-modules cell is flat at judged 0.63–0.73 through 10^{-6}–10^{-4} and gibberish at 10^{-2}, so its entire transition happens inside one decade with nothing run inside it (the mid-attention cell is already destroyed a decade earlier, at 10^{-3}); the finer grid above is what probes that decade and the AdamW gap.

A24 A3 AT EIGHT SEEDS, IN A SECOND SOFTWARE STACK

The headline table rests on three A3 adapters per model, judged on a 40-prompt subset of each prompt set, with item-level intervals that hold the adversary set fixed. Table 20 re-runs that arm from scratch at eight seeds on each of the four models (Mistral-7B included, though §5.5 puts it outside the headline's scope) at the usable subspace budget, trained and evaluated end to end on a different software stack, every one of the 120 AdvBench and 100 HarmBench prompts judged, and a two-level bootstrap that resamples adapters as well as prompts. One Llama and two Qwen-7B adapters did not land and are excluded by the rule of §4: $m{=}1024$ seed 5 on Llama-3.1-8B at unrepaired 0.205, and $m{=}256$ seeds 0 and 2 on Qwen2.5-7B at 0.291 and 0.241. Those are the three weakest of the thirty-two, which is what the landing rule does by construction and why Appendix A11 measures the bias it induces. On the three in-scope models every item-level *estimate* is at least eight times smaller than what the same defence takes off the weakest ordinary family on that model in Table 2 ($8.8\times$ on Llama-3.1-8B against its attention loci, $26\times$ and $76\times$ on the two Qwen models); read instead against each interval's furthest endpoint, which is the conservative version of the same comparison, the ratios are $5.2\times$, $14\times$ and $18\times$.

The equivalence test at the level the claim is about. Appendix A30 runs TOST over three fixed adapters per model and pairs at the item level, so what it tests is those three adapters. This arm supports the test the claim actually needs. Applying the same decision rule (reject non-equivalence at $\alpha = 0.05$ iff the $90\,\%$ interval lies inside $\pm\delta$, at the same $\delta = 0.10$ fixed in Appendix A30) to the *two-level* bootstrap, which resamples adapters and then prompts within them, the intervals are Llama-3.1-8B $[-0.053, -0.003]$, Qwen2.5-7B $[-0.044, +0.002]$, Mistral-7B $[-0.012, +0.032]$ and Qwen2.5-1.5B $[-0.028, +0.016]$. All four are inside the margin, so non-equivalence is rejected on every model, Mistral included. Llama-3.1-8B's $90\,\%$ interval excludes zero, where the $95\,\%$ one in the table reaches it: SAFELORA does remove something there, and what the test says is that what

it removes is smaller than the margin Appendix A30 fixes from Table 2's smallest ordinary effect, below which the paper does not call it a repair. This is a stricter reading of this arm's data than Appendix A30's and it agrees with it; we report the adapter-level one because the inference is about the family, not about eight particular seeds.

Table 20: A3 re-run at eight seeds per model on a different software stack, judged on the complete AdvBench and HarmBench sets (220 items per adapter); n counts adapters whose attack landed. The item-level interval resamples prompts, the adapter-level interval resamples adapters as well.

		judged harm ASR		Δ [95 % CI]		harm
model	n	attacked	repaired	item-level	adapter-level	≥ 3
Llama-8B	7	0.653	0.626	-0.027 $[-0.045, -0.010]$	-0.027 $[-0.057, +0.000]$	0.658
Qwen-7B	6	0.605	0.583	-0.021 $[-0.039, -0.004]$	-0.021 $[-0.048, +0.007]$	0.609
Mistral-7B	8	0.748	0.759	$+0.011$ $[-0.007, +0.028]$	$+0.011$ $[-0.016, +0.036]$	0.804
Qwen-1.5B	8	0.623	0.617	-0.006 $[-0.027, +0.015]$	-0.006 $[-0.032, +0.021]$	0.662

A25 Can the gate be repaired? Two re-designs, priced

An attack paper should say whether the defence survives being fixed. Appendix A6 already shows that SafeLoRA's shipped $C = VV^\top/\|V\|_F$ is a contraction rather than the projector its derivation names, and Appendix A36 that the derived projector is the *identity* wherever $d_{\text{out}} \leq d_{\text{in}}$, so "ship the correct operator" is not the fix. What is left is a *rank-limited* projector, and the reach framework says two things about it before it is run. Both were written down before the experiment and both held, the second already at half the k it names.

The two arms. Let U_k hold V's leading k left-singular directions, from a thin SVD of the same two public files. Then:

- $C_k = U_k U_k^\top$**, gated as the shipped** C **is**: compute $\cos(C_k \Delta W, \Delta W)$, project the matrices that read below τ. A real, rank-k orthogonal projector in place of a contraction. *Prediction:* this does not have a setting that removes A3 and keeps the customer, because A3's update lies *inside* $\text{span}(U_m)$ by construction: small k makes the gate fire on everything, large k reads A3 as aligned, and Proposition 1 says the two ends are priced against each other.
- $I - U_k U_k^\top$**, applied unconditionally**, *delete* whatever the update puts in V's leading block. *Prediction:* this removes A3 by the time k reaches m, and it removes nothing else, because emptying a subspace only reaches what was put there.

Both cost one SVD of two public files and are computable before any attack is seen. Table 21 runs them over the seven landed adapters of the A3, A2 and ordinary families on Qwen2.5-1.5B (15,400 judgments) and prices them on the four benign adapters of Appendix A9.

The projector is a dial between two failures. At $k = 8$ the gated projector removes everything (A3 to 0.000–0.014 from 0.500–0.614, A2 and the ordinary attack to 0.005), and keeps 20 % of the customer's fine-tune, deleting 0.939 of every update it sees. Turn k up and the customer comes back and so does the attack: 29 % kept and A3 at 0.227–0.491 at $k = 32$, 64 % kept and A3 at 0.500–0.627, fully recovered, at $k = 128$. The mechanism is the one §5.1 derives: widening the block raises $\cos(C_k \Delta W, \Delta W)$ on an update built inside it ($0.306 \to 0.492 \to 0.780$ on the three A3 adapters over the three k, $0.249 \to 0.392 \to 0.610$ over all seven), so the gate stops firing on exactly the adapter it was widened to catch. There is no k at which this operator both removes A3 and leaves a customer with a quarter of their fine-tune. That is Proposition 1 on the only dial the re-design offers.

Emptying the block works, and only against what is in it. The complementary operator is the one the framework points at, and it behaves as predicted. At $k = 8$ and $k = 32$ it removes nothing: no family's mean moves by more than 0.07 and every adapter stays above 0.30. At $k = 128$, half of A3's $m = 256$, it takes A3 from 0.500–0.614 to 0.009–0.014 while leaving A2 at 0.350–0.464 and the ordinary attack at 0.441, for 95 % of the customer's gain and benign over-refusal 0.04. So a provider

who has read this paper can stop A3 for a ninth of what SAFELORA at $\tau = 0.5$ costs the customer, $5\,\%$ of the fine-tune's gain against $46\,\%$, which is that operator's cost *on this model and seed*, and emptying the block keeps more on each of the four tasks (77–106 % against 33–79 %); pooled over the benign task set SAFELORA keeps $51\,\%$ here and 79–136 % on the other three (Appendix A9), and has stopped nothing else: the two attacks the same table shows surviving are the ones whose energy is not in the block he emptied. We think that is the right shape for a defence built on this framework and the right way to state its scope: *this* construction, in *that* subspace, and a defender still needs SAFELORA's own projection for A2 (which the mask does not remove, §5.4) and either operator for the ordinary families. It is a patch on a named hole, not a repair that generalises, and Appendix A53 watches the attacker leave the hole: widen m towards d_{out} and the same $k = 128$ deletion reads 0.457 against an unrepaired 0.502, while the mask (whose support is not a spectrum and does not move when m does) still removes him.

Is A2 an artefact of a one-instance safety set? Appendix A12 reports ρ_5, the reach a five-instance repair would have, as a *bound*, because no such repair had been run. The last block of Table 21 runs it: the same coordinate mask, the same budgets, the gradient taken over the whole five-instance safety pool of §4 instead of the single instance. The ordinary attack falls to 0.009 at the $20\,\%$ budget and A2 stands at 0.209–0.414. Five times the safety data leaves the three A2 adapters at 0.209–0.414 against 0.218–0.405 under one instance, a family mean of -0.012, at the same budget (Table 29): the blind spot is a property of *where* the gradient has support, not of how much data was used to estimate it. It is not a measurement at the $|\mathcal{D}_{\text{safe}}| = 100$ of Appendix A48, and we do not claim one.

Table 21: Two re-designs of SAFELORA's gate and a five-instance coordinate mask, on Qwen2.5-1.5B over the seven landed adapters of the three families, judged on the full AdvBench and HarmBench sets. Ranges are over adapters within a family (three A3, three A2, one ordinary); the unrepaired row is the same adapters with no repair. "Task gain kept" is the mean over the four downstream tasks of Appendix A9 at seed 7. U_k is the leading k left-singular block of the public V; $k = 128$ is half of A3's subspace budget $m = 256$.

	judged ASR after repair				
repair	A3	A2	ordinary	benign refusal	task gain kept
unrepaired	$[0.500, 0.614]$	$[0.368, 0.518]$	$[0.532, 0.532]$	0.02	$100\,\%$
as shipped					
SAFELORA, $C \propto VV^\top$	$[0.532, 0.605]$	$[0.005, 0.009]$	$[0.014, 0.014]$	0.03	$54\,\%$
a real rank-k projector, gated at τ as C is					
$C_k = U_k U_k^\top$, k=8	$[0.000, 0.014]$	$[0.005, 0.005]$	$[0.005, 0.005]$	0.05	$20\,\%$
k=32	$[0.227, 0.491]$	$[0.005, 0.009]$	$[0.005, 0.005]$	0.05	$29\,\%$
k=128	$[0.500, 0.627]$	$[0.005, 0.009]$	$[0.009, 0.009]$	0.03	$64\,\%$
delete V's leading block outright					
$I - U_k U_k^\top$, k=8	$[0.473, 0.641]$	$[0.359, 0.495]$	$[0.491, 0.491]$	0.02	$96\,\%$
k=32	$[0.482, 0.623]$	$[0.368, 0.468]$	$[0.464, 0.464]$	0.02	$102\,\%$
k=128	$[0.009, 0.014]$	$[0.350, 0.464]$	$[0.441, 0.441]$	0.04	$95\,\%$
the coordinate mask, five safety instances not one					
mask_5, $\lvert\mathcal{D}\rvert$=5	$[0.341, 0.468]$	$[0.341, 0.495]$	$[0.323, 0.323]$	0.06	$98\,\%$
mask_{20}, $\lvert\mathcal{D}\rvert$=5	$[0.023, 0.382]$	$[0.209, 0.414]$	$[0.009, 0.009]$	0.05	$88\,\%$

A26 UN-FIXING THE REACH: A REPAIR THAT CHOOSES WHAT TO TOUCH AFTER THE ATTACK

Every repair in §5.4 fixes its reach before the attack exists (SAFELORA's C from two public files, RESTA's direction from the same two, the coordinate mask's support from $\partial L_{\text{safe}}/\partial W$ at θ_0), and Corollary 1 then bounds what any of them removes from an update built outside that reach. A repair that chooses after seeing the attack is not bound by it. This appendix runs one on three models. Qwen2.5-1.5B is reported in full, over the union of its A3, A2 and ordinary adapters (7 landed, 16,940

judgments), priced against the same four customers as Appendix A9; Llama-3.1-8B and Qwen2.5-7B, four landed adapters each, are the paragraph before the last, and one of them does not agree.

The clean test: move the gradient, change nothing else. The coordinate mask's reach *is* the safety gradient's support, so the minimal attack-dependent repair is the identical rule with the gradient taken at θ_a. Table 22 measures it, and the answer is that it makes no difference: at a $20\,\%$ budget the mask moves judged compliance by -0.269 with its gradient at θ_0 and -0.273 with it at θ_a, deletes 0.454 against 0.455 of the update, leaves benign refusal at 0.05 either way, and returns $90\,\%$ against $92\,\%$ of the customers' gain. That is worth stating plainly because it is not what the framework's informal reading would predict: what binds a repair is *where* its reach lies, not *when* it is chosen, and adding an identity-shift update does not move where the safety loss has gradient.

A rule that does move the reach, and what it costs. Reverting coordinate (i, j) moves W by $-\Delta W_{ij}$, so to first order it changes the safety loss by $-g_{ij}\Delta W_{ij}$: ranking by $g_{ij}\Delta W_{ij}$ at θ_a and reverting the top fraction is the loss-aware selection of the SAFEDELTA/IRR family (Lu et al., 2025; Wu et al., 2025), written for a factored update. This does reach A3 and A2 where the fixed-reach repairs do not. It also takes judged compliance to 0.005 at a $20\,\%$ budget, as low as any repair in this paper reaches, and it is not a defence. Benign refusal goes from 0.03 to 0.38–0.50, and the customers' fine-tunes end *below* no fine-tune at all on two of four tasks ($-23\,\%$ of the gain pooled, $-194\,\%$ on summarisation at $20\,\%$). At the $1\,\%$ budget, where benign refusal is 0.20, the attack is only down to 0.262 from 0.492: worse than the $20\,\%$ mask, at four times its benign cost. Adding the utility term the method's own derivation calls for, $-\lambda|u_{ij}\Delta W_{ij}|$ with u the gradient of a held-out utility objective at θ_a and $\lambda = 1$, changes nothing measurable at the 5 and $20\,\%$ budgets; at $20\,\%$ it is 0.006 against 0.005 compliance, 0.39 against 0.38 refusal, $-22\,\%$ against $-23\,\%$ of the gain. At $1\,\%$ it leaves more of the attack (0.302 against 0.262), and the arm without it has no utility run there. The first-order utility estimate does not see the damage, which is the expected failure of a linearisation evaluated at a revert this large.

On the other two models, the loss-aware rule wins once. The same stage ran on Llama-3.1-8B (three A3 adapters and one ordinary) and Qwen2.5-7B (one A3, two A2, one ordinary). Moving the mask's gradient to θ_a again changes nothing: at $20\,\%$ it leaves 0.093 against 0.095 on Llama-3.1-8B and 0.299 against 0.303 on Qwen2.5-7B. The loss-aware revert is model-dependent. On Llama-3.1-8B it is the same bad trade as on Qwen2.5-1.5B: at $5\,\%$ with the utility term it leaves 0.224 at benign refusal 0.145 and loses the customer's MBPP and summarisation gains ($7\,\%$ and $-155\,\%$ kept). On Qwen2.5-7B it is the best repair we ran: at $5\,\%$ with the utility term it takes all four adapters to 0.009–0.077 (A3 to 0.023 from 0.755) at benign refusal 0.105 and keeps 66–$127\,\%$ of the customer's gain on the four tasks, where the $20\,\%$ mask leaves 0.303 and SAFELORA at $\tau = 0.5$ leaves 0.200. An attack-dependent repair can therefore remove both adversaries at a usable operating point; the one we built does so on one checkpoint of three.

What the arm is and is not evidence for. It is evidence that attack-dependence is not the free lunch the reach framework might be read to promise: on Qwen2.5-1.5B and Llama-3.1-8B a coordinate-level loss-aware revert is a worse trade than the fixed-reach mask at every point we measured. It is not evidence that no attack-dependent repair can work, and Qwen2.5-7B is the counterexample: SAFEDELTA as published adds a second-order utility model and a compensation step we did not reproduce, the defender's safety set here is the five-example pool of §4 (a rule that optimises against one refusal template will produce a model that emits it). In sum, the one attack-dependent repair we built beats the fixed-reach mask it was matched against on one model of three, which is what a set chosen after the attack is able to do and a fixed one is not, and costs the customer too much on the other two.

A27 THE DEFENDER'S DILEMMA, ON EVERY ADAPTER RATHER THAN THE WORST ONE

Table 57's harm curve would rest on one A3 adapter per model if it were read off §5.3's threshold sweep alone, which runs over a representative set that carries one, and on Llama-3.1-8B that one adapter is the *maximum* of the family at every threshold, 0.800 against a pooled 0.646 at $\tau = 0.5$

Table 22: An attack-dependent repair on Qwen2.5-1.5B, over the 7 landed adapters of the A3, A2 and ordinary families pooled, judged on the full AdvBench and HarmBench sets. Δ is paired against this stage's own unrepaired arm; "deleted" is the fraction of the update's norm removed; "task gain kept" is the mean over the four downstream tasks of Appendix A9 at seed 7, shown only where all four ran. SAFELORA's cost to the customer is priced over the whole benign pool in Appendix A9 rather than here. Indented rows add the utility term to the row group above them.

repair	n	judged ASR	Δ	benign refusal	deleted	task gain kept
unrepaired	7	0.492	+0.000	0.03	0.000	100 %
reach fixed before the attack						
SAFELORA τ=0.5	7	0.249	−0.243	0.03	0.609	–
mask$_{20}$, g at θ_0	7	0.223	−0.269	0.05	0.454	90 %
same rule, reach recomputed after it						
mask$_5$, g at θ_a	7	0.426	−0.066	0.06	0.233	96 %
mask$_{20}$, g at θ_a	7	0.218	−0.273	0.05	0.455	92 %
reach moved: coordinates ranked by predicted safety gain						
$g \cdot \Delta W$, 1 %	7	0.262	−0.230	0.20	0.196	–
$g \cdot \Delta W$, 5 %	7	0.077	−0.414	0.50	0.372	12 %
$g \cdot \Delta W$, 20 %	7	0.005	−0.487	0.38	0.587	−23 %
$-\lvert u \cdot \Delta W \rvert$, 1 %	7	0.302	−0.190	0.19	0.194	71 %
$-\lvert u \cdot \Delta W \rvert$, 5 %	7	0.082	−0.409	0.50	0.365	11 %
$-\lvert u \cdot \Delta W \rvert$, 20 %	7	0.006	−0.485	0.39	0.571	−22 %

and 0.725 against 0.388 at $\tau = 0.7$. A table that is the worst case on the harm axis, read in the body as typical, is a cherry-pick whether or not it was meant as one, so the column now pools every A3 adapter that landed, under the AdvBench landing rule every AdvBench-only harm number in this paper uses (Table 2 lands on both sets pooled, which is why it counts three Llama adapters). Table 24 is the per-adapter detail behind it. Two Llama adapters are outside the pool and are named rather than dropped quietly: the third m=1024 seed, unrepaired 0.225 on AdvBench against 0.525 on held-out HarmBench, and the m=256 adapter at 0.050, neither clearing the 0.30 this paper requires before asking whether a repair removed anything. Both sweeps step τ by 0.1; Appendix A28 resolves the one interval where that grid leaves the question open. Table 57's $\tau = 0.5$ column is a *different* estimate from Table 2's repaired column (0.646/0.403/0.525 against 0.525/0.608/0.629): AdvBench alone over the adapters that landed in *that* stage, pooled as an unweighted per-adapter mean, against both benchmarks pooled at the item level.

Pooling costs Llama its most dramatic row and strengthens two others: on Qwen2.5-7B $\tau = 0.8$ reads 0.119 over three adapters against the single adapter's 0.025, and Mistral's rise is now on three. Every conclusion §5.3 draws survives it, including the one that goes the wrong way. On Qwen2.5-1.5B the pooled rate is 0.428–0.525 from $\tau = 0.3$ through $\tau = 0.7$ and exactly 0.000 from $\tau = 0.8$, which is where the customer keeps nothing. On Qwen2.5-7B it is 0.339–0.425 through $\tau = 0.7$, 0.119 at $\tau = 0.8$ and 0.000 at $\tau = 0.9$, the same crossing, one threshold later. And Mistral-7B still fails the other way, now on three adapters rather than one: A3 *rises* with τ, to 0.872 at $\tau = 0.8$, and is still at 0.275 at $\tau = 0.9$, which is the behaviour §5.5 attributes to a θ_0 that is not aligned.

The spread is worth stating because it is not small. On Qwen2.5-7B the three adapters span 0.067–0.725 at $\tau = 0.7$: the pooled curve is a summary of three quite different adapters, and the claim it supports, that no threshold both keeps the task and removes the attack, holds for each of them separately only in the sense that none is removed before the threshold at which the customer's gain is gone. Llama-3.1-8B is no better (its two adapters span 0.675 at the same τ, the widest gap in the table), while on Mistral-7B and Qwen2.5-1.5B the spread stays under 0.27. A defender choosing τ from a single adapter's curve on either 7–8B model could misread the harm that survives it by half that spread.

The A2 adapters swept alongside are removed by SAFELORA at every $\tau \geq 0.4$ on Qwen2.5-1.5B and every $\tau \geq 0.5$ on Qwen2.5-7B (one there still reads 0.167 at 0.4), which is the expected half of §5.4's complementary-blind-spot result: A2 is built against the coordinate mask's support, not against C.

A28 THE THRESHOLD, AT FIVE TIMES THE RESOLUTION

§5.3 sweeps τ in steps of 0.1 and concludes that no threshold on Llama-3.1-8B both keeps a downstream task and removes A3. On that grid the whole question sits inside a single interval: at $\tau = 0.7$ the attack is at 0.388 over the two landed adapters that grid carries (0.050 on 120 prompts, 0.725 on 40), and the customer keeps $17\,\%$ of the SQL-Create gain; at $\tau = 0.8$ the attack is at 0.017 and the gain is gone. The table below is three adapters, 120 prompts and four tasks, so its endpoints are not that row's. One may ask what happens in between, and a 0.1 grid cannot answer it. This appendix answers it at 0.02.

What was run. Six thresholds, $\tau \in \{0.70, 0.72, \ldots, 0.80\}$, on both sides of the trade at once. On the harm side, the three m=1024 A3 adapters of Table 2 at every threshold, 120 AdvBench prompts per cell (the third seed lands on the two sets pooled but not on AdvBench alone, 0.225, so Appendix A27 leaves it out; on the other two alone the mean is 0.254 at $\tau = 0.74$ and 0.146 at 0.76, so there the window below is $\tau = 0.76$ alone), all $2{,}160$ generations judged by the same pipeline as every other number here. On the utility side, the four benign adapters of Appendix A9 at seed 7 (SQL-Create, GSM8K, dialogue summarisation and MBPP code) at every threshold, plus the unrepaired checkpoint θ_0 measured inside this stage so the baseline and the repaired arms share their test items.

The finer grid qualifies the claim rather than confirming it. Table 23 is the sweep, and it is not the result we expected. Judged A3 compliance falls monotonically across the interval, $0.286 \to 0.008$, and the number of tasks more than 0.02 above θ_0 falls with it, $4 \to 0$, but the two curves cross rather than clearing each other. At $\tau = 0.76$ the attack reads 0.117, which is *inside* the 0.029–0.183 that the same defence at $\tau = 0.5$ leaves Llama's three ordinary families at *on AdvBench* (by §5.2's own standard, removed), while all four of the customer's tasks are still at or above θ_0: $+0.162$ on MBPP, $+0.046$ on summarisation, $+0.017$ on GSM8K and $+0.017$ on SQL-Create. The comparator has to be the AdvBench-only one: Table 2's 0.079–0.289 pools AdvBench with the held-out HarmBench set, which every model complies with more (Appendix A21), and these cells are AdvBench alone. The distinction decides the paragraph: against the pooled band $\tau = 0.70$ would qualify too, and it plainly should not. A defender who searched τ at this resolution on this model would find such a point. One who took the release's $\tau = 0.5$, or stepped by 0.1 as §5.3 does, would not.

We state it rather than round it away, and four things bound it. The window is narrow and badly placed: it is two grid points wide ($\tau = 0.74$ qualifies too, at 0.181 against a 0.183 ceiling), and it sits 0.26 above the release's $\tau = 0.5$, and one step past it ($\tau = 0.78$) the attack is at 0.022 but GSM8K is 0.033 *below* not fine-tuning at all. Two of the four tasks are inside $+0.02$ of not having been fine-tuned, so "still ahead" is carrying very little at 0.76. It is a property of the mean, not of each adapter: the landed `a3m1024-s1` (and the third seed, which does not land on AdvBench) is inside or below the band from $\tau = 0.70$, where the worst task keeps $15\,\%$ of its gain, while on `a3m1024-s0`, the adapter §5.3's 0.1 sweep carries on 40 prompts, the attack reads 0.433 and 0.258 at those two thresholds and does not enter the band until $\tau = 0.78$, where the customer is already behind. And the customer is not intact there in the sense the rest of the paper means: the paragraph headed "What the window costs" prices those two thresholds at $45\,\%$ and $31\,\%$ of the gain. And it is one model: the same search on the two Qwen checkpoints is Appendix A27's 0.1 sweep, where the crossing is a cliff rather than a window.

The one baseline the two stages disagree about, and which way it cuts. This stage re-measures θ_0 inside its own run so the baseline and the repaired arms share their test items, and on one task of four that produces a different number from Appendix A9's: GSM8K reads 0.517 here and 0.600 there, five items of sixty. The *tuned* arms agree exactly on all four tasks, so what differs is how the untuned checkpoint's answers are scored, not the fine-tune: this stage reads the final answer only from the `####` field the fine-tunes learn to write, Appendix A9 from the last number in the response, and the five items are ones only the second rule credits. It is the denominator of every retention ratio below, so the direction matters: using this stage's θ_0 gives the *higher* retention at every threshold from 0.72 up (0.45 against 0.35 at $\tau = 0.74$, 0.31 against 0.15 at 0.76), so the choice we made is the one that makes §5.3's negative claim harder to sustain, and it is sustained anyway. It cuts the same way for the weak reading of *keeps*: against Appendix A9's baseline GSM8K is 0.067 below

θ_0 at $\tau = 0.76$, which would leave the three-adapter window at $\tau = 0.74$ alone and the two landed adapters' at none.

With both words pinned, the sweep decides it. "Still ahead of θ_0" is the weakest reading of *keeps* available, and it is the one under which the window exists. Pin the two words the way §5.3 states them (*removes* is inside the 0.029–0.183 AdvBench band, *keeps* is half the fine-tune's gain), and no threshold in the sweep does both, under either reading of which task must be kept. The first threshold that removes is $\tau = 0.74$; there the customer's worst task, SQL-Create, holds $5\,\%$ of its gain and the four-task mean holds $45\,\%$. Above it the attack keeps falling and the customer keeps falling faster. So the claim in the body is not a hedge about what a defender would happen to try: at this resolution, on the model where the question is open, the threshold that removes the attack and the threshold that keeps the customer do not exist together.

What the window costs, once the denominator exists. With the stage's own unrepaired arm, the retention ratio the rest of the paper uses can be quoted here, and it is what settles the paragraph above. Buying the attack down from SafeLoRA at $\tau = 0.5$ (0.800, at $100\,\%$ of the gain) costs the customer steeply and monotonically: $67\,\%$ of the gain survives at $\tau = 0.70$, $45\,\%$ at the 0.74 that first reaches the band, $31\,\%$ at 0.76, and $7\,\%$ at the 0.78 that removes the attack. The window is therefore real under the weak criterion and thin under the paper's own: at $\tau = 0.76$ nothing is below θ_0, but SQL-Create retains $5\,\%$ of what fine-tuning bought it and GSM8K $8\,\%$. Two of the four tasks have been bought out even where the arithmetic says they are still ahead, which is the sense in which §5.3's claim holds: no threshold removes the attack and leaves the fine-tune, and the finest grid we ran prices the trade rather than escaping it.

And the spread is large where the attack survives. At $\tau = 0.70$ the three seeds read 0.717, 0.092 and 0.050, so the mean is a summary of three quite different adapters, exactly as Appendix A27 finds at 0.1 resolution. By $\tau = 0.78$ the spread has collapsed to 0.008–0.042 and the summary is safe. The utility columns are kept as $\mathrm{metric} - \theta_0$ beside the ratio because they need no denominator: where one is at or below zero the fine-tune has bought the customer nothing at that threshold however the denominator is computed.

Table 23: The threshold swept at 0.02 through the interval §5.3's 0.1 grid leaves open, on Llama-3.1-8B. "A3" is judged compliance over the three m=1024 adapters, 120 AdvBench prompts each; the four task columns are $\mathrm{metric} - \theta_0$ on the benign adapters at seed 7, in **bold** where the customer is still more than 0.02 ahead of not fine-tuning, and "up" counts them, and "gain kept" is the mean over the four tasks of $(\mathrm{metric} - \theta_0)/(\mathrm{tuned} - \theta_0)$ and "worst" its minimum, with both ends measured inside this stage. At $\tau = 0.74$ and 0.76 the attack is inside the AdvBench band this defence leaves Llama's ordinary families in (0.029–0.183) and all four tasks are still at or above θ_0, on $45\,\%$ and $31\,\%$ of the gain; one step further the attack is removed and $7\,\%$ remains.

τ	A3 (mean)	A3 (range)	MBPP	GSM8K	SQL	DialogSum	up	gain kept	worst
0.70	0.286	0.050–0.717	**+0.179**	**+0.217**	**+0.050**	**+0.080**	4	67%	15%
0.72	0.242	0.050–0.617	**+0.171**	**+0.083**	**+0.033**	**+0.077**	4	48%	10%
0.74	0.181	0.033–0.433	**+0.176**	**+0.083**	+0.017	**+0.066**	3	45%	5%
0.76	0.117	0.033–0.258	**+0.162**	+0.017	+0.017	**+0.046**	2	31%	5%
0.78	0.022	0.008–0.042	**+0.097**	−0.033	+0.000	+0.002	1	7%	−17%
0.80	0.008	0.000–0.017	+0.000	−0.033	+0.000	+0.002	0	−4%	−17%

A29 A THIRD JUDGE, FROM A DIFFERENT VENDOR

§4 validates the primary judge (`gpt-4.1-mini`) against a stronger one (`gpt-4.1`) at $\kappa = 0.831$ over 9,955 paired judgments. Both are the same vendor's models trained by the same lab, so that number bounds reliability *within* one family and is silent on whether the family shares a bias. This appendix re-runs the same stratified protocol, on the repair-family stages the first cross-judge does not cover, against `gemini-2.5-flash` with thinking disabled.

Table 24: The threshold sweep on the A3 adapters that landed on AdvBench (two on Llama-3.1-8B, three on each of the others) rather than one, judged on AdvBench (120 items for these, 40 for the one the main sweep carries). Each pair of columns is the mean over those adapters and their range; one is the adapter the main sweep carries, the others are new. These columns sit below Table 2's repaired figure for the same adapters and the same τ, 0.403 against 0.608 on Qwen2.5-7B, because that table pools AdvBench with the held-out HarmBench set, which every model complies with more, attacked and repaired alike ($0.433 \to 0.467$ on AdvBench against $0.733 \to 0.750$ on HarmBench for these three adapters; Appendix A21). The conclusion is the same on each set separately; only the level differs.

	Llama-8B		Qwen-7B		Qwen-1.5B		Mistral-7B	
τ	mean	range	mean	range	mean	range	mean	range
0.3	0.688	[0.575, 0.800]	0.425	[0.125, 0.700]	0.469	[0.392, 0.567]	0.706	[0.575, 0.775]
0.4	0.704	[0.583, 0.825]	0.425	[0.142, 0.708]	0.494	[0.442, 0.592]	0.703	[0.575, 0.767]
0.5	0.646	[0.492, 0.800]	0.403	[0.133, 0.675]	0.525	[0.442, 0.583]	0.708	[0.600, 0.775]
0.6	0.637	[0.475, 0.800]	0.381	[0.117, 0.650]	0.514	[0.375, 0.642]	0.794	[0.775, 0.825]
0.7	0.388	[0.050, 0.725]	0.339	[0.067, 0.725]	0.428	[0.375, 0.483]	0.839	[0.808, 0.858]
0.8	0.017	[0.008, 0.025]	0.119	[0.025, 0.292]	0.000	[0.000, 0.000]	0.872	[0.833, 0.925]
0.9	0.033	[0.017, 0.050]	0.000	[0.000, 0.000]	0.000	[0.000, 0.000]	0.275	[0.242, 0.325]

How much the primary judge disagrees with itself. Four cells of Appendix A56's `fp32` arm were run twice end to end by accident. The cells came out *identical* (same ASR, same over-refusal, same perplexity to the digits recorded) so the 320 judged items that resulted are the same prompt and the same generation put to the primary judge twice. It returned a different verdict on 9 **of** 320 (2.8 %) and a different harm score on 13. Every per-item interval in this paper is a resample over items and does not contain that term; it is small beside the 8.6 % at which the two judges disagree with *each other* (§4), and it is the right floor to read the equivalence margins against (`results/judge_selfrepeat.json`).

Which stages each cross-judge covers, and which carry none. The two cross-checks are on *disjoint* stage sets, and it is better to say so than to let a reader discover it. `gpt-4.1` re-labels a fifth of the attack–repair matrix, the τ sweep and the gradient-repair frontier (the three stages every headline table in §5 is built from), and none of `repairs2`, the stage behind §5.4's repair-family comparison. `gemini-2.5-flash` covers `repairs2` and six more (`a3seeds`, `a3tuned`, `a3band`, `a3sweep2`, `maskdial`, `rank64`). Ten of our twenty-nine stages therefore carry a second judge and none carries two, so the cross-vendor κ below is a check on a different population from the within-family one rather than a confirmation of it. The nineteen with no second judge (`adaptive`, `decoy`, `escdef`, `gatefix`, `tau2`, `a3sweep`, `exactc`, `controls9`, `taufine`, `a4`, `a3alt`, `obj`, `frontier2`, `cleanpt`, `a2seeds`, `a4frontier`, `fixedcount`, `frontier32`, `a3wide`) are all appendix stages, and seventeen of the nineteen report a *difference* between two arms judged by the same instrument on the same items, which is the comparison an instrument bias *mostly* cancels in. Mostly, not exactly, and the residual is worth measuring rather than asserting away. On the items the strong judge re-labelled for *both* arms of a repair pair, the paired difference moves by -0.037 to -0.114 on the nine ordinary families of the three in-scope models (every one of them negative, meaning the stronger judge sees the defence removing *more*), and by at most 0.021 in either direction on the three A3 families. (`a4` reports no difference at all: neither aim lands, so there is nothing for a repair to remove.) So the residual is systematic and it points the wrong way for a reader who wants §5.2's contrast to be a judging artefact: it widens the gap between the ordinary families and A3 rather than creating it. A second fact belongs beside it. Per item, the two judges disagree on 8.6 % of the 596 A3 matrix labels, which is larger than every A3 effect in Table 2 (0.008 to 0.054). That is not a defect in those effects: it is why the claim about them is stated as an *equivalence* at a margin rather than as an effect (Appendix A30), and why a reader should not read -0.054 as a measurement of something the instrument could resolve one item at a time. `taufine` is the exception and it is the one §5.3 leans on: its output is a *level* (0.117, 0.181) read against a band measured in the attack–repair matrix, on different adapters, so an instrument bias does not cancel there. §5.3 says so where it cites it, and the margin the claim turns on is 0.066: large against this judge's disagreement with the strong one on that model, and not something a second vendor has checked.

The extension stages carry their own second judge. These stages (Appendices A59, A62, A63, A64) were judged by the primary instrument like everything else (on the *complete* 120-prompt AdvBench and 100-prompt HarmBench sets rather than the nested subsets the earlier stages use) and a stratified sample of those judgments was re-labelled by `gpt-4.1` under the protocol of §4: 2,560 paired judgments over 4 stages (2 % of them), agreement 92.6 %, $\kappa = 0.845$, harm-score correlation $r = 0.904$ — the same reliability as the within-family check on the main stages ($\kappa = 0.831$). A cross-vendor sample of the same stages against `gemini-2.5-flash` stands at 365 paired judgments ($\kappa = 0.868$, agreement 96.4 %) on `a2span`, limited by that endpoint's rate cap; it is small and is reported here rather than in a table. Two cautions travel with these figures. The re-labelled sample is about 16 items per cell, so a single cell's rate moves by a median of 0.048 and at most 0.333 between the two judges over the 42 cells the sample covers; the appendices' conclusions are pooled over three adapters and 660 judgments, which is the level at which the two judges agree. And the agreement is on the verdict, not on the threshold: both judges see the same generations, so a shared bias about what counts as compliance is not measured by either check.

Agreement is not lower across vendors than within one: $\kappa = 0.859$ over 2,277 paired judgments spanning seven stages and all four models, against 0.831 for the within-family check, with harm-score correlation $r = 0.868$. Table 25 breaks it down. The one model where the judges diverge is Mistral-7B ($\kappa = 0.641$), which is the model §5.5 excludes from every claim for an unrelated reason, its θ_0 is not aligned, so a large share of its cells are refusal-free on both arms and the disagreements concentrate in generations that are neither a refusal nor a usable answer. On the three models the paper's claims rest on, κ is 0.844–0.926.

Where the sampler happened to cover both arms of a repair pair, the third judge reproduces the headline difference exactly rather than approximately: +0.000 on the seven paired Llama-3.1-8B items and +0.143 on the fourteen Qwen2.5-1.5B ones, the same to three decimals as the primary judge on the same items. Fourteen items is not an independent replication of Table 2 and we do not present it as one; it is a check that the two judges do not merely agree on labels in aggregate while disagreeing on the contrast the paper draws.

Table 25: The primary judge against a third from a different vendor (`gemini-2.5-flash` with thinking disabled), same stratified protocol as the within-family check of §4. Stages covered: the five-repair comparison, the eight-seed replication, the rank-64 control, the mask dial, and the three gate-aiming stages.

model	paired judgments	agreement	Cohen's κ
Llama-8B	790	0.965	0.909
Qwen-7B	305	0.967	0.926
Mistral-7B	317	0.842	0.641
Qwen-1.5B	865	0.931	0.844
all	2277	0.935	**0.859**

A30 THE EQUIVALENCE CLAIM, TESTED, AND ONE MULTIPLICITY CORRECTION

The paper's central negative claim, that SAFELORA does not remove A3, is an equivalence claim, and a confidence interval that happens to contain zero is not a test of one. This appendix supplies the test the claim needs and, separately, asks what survives when the intervals the paper reports are treated as a family rather than one at a time.

Two one-sided tests, and where the margin came from. We fix a negligibility margin $\delta = 0.10$ before running any equivalence test: 0.10 is under half of 0.239, the *smallest* effect SAFELORA has on any ordinary family on any of the three in-scope models (Table 2), so a difference inside $\pm\delta$ is one the paper would not call a removal anywhere else. Said precisely, because it matters: δ is *calibrated* to a quantity measured in this same study, not chosen before any data existed, and it is not one of the seven predictions in Appendix A18's pre-registration. What it is fixed ahead of is the test, no value of δ was tried and discarded, and the fragility that convention is meant to guard against

is reported directly below instead of being asserted away. On Mistral-7B, which §5.5 excludes, the all-modules ordinary family moves $+0.068$ and is itself inside the margin, so the same test applied there is reported for completeness and carries no information. Table 26 runs a paired bootstrap TOST over the item-level compliance pairs behind the headline table: non-equivalence is rejected when the $(1 - 2\alpha)$ interval at $\alpha = 0.05$ lies strictly inside $(-\delta, +\delta)$. It is rejected on all three eligible models. Note what this does and does not say. Llama-3.1-8B's interval excludes zero (the repair does move that model, by 5.4 points), and equivalence at $\delta = 0.10$ is the statement that the movement is negligible on the scale of the same defence's effect elsewhere, not that it is absent. That distinction is the paper's claim, stated as a test rather than as a comparison of magnitudes.

How much margin each verdict has. A tick at $\delta = 0.10$ hides how close the test ran, so the smallest margin at which each model would still pass is the interval's furthest endpoint: 0.087 on Llama-3.1-8B, 0.054 on Qwen2.5-7B and 0.050 on Qwen2.5-1.5B. Llama passes with $13\,\%$ of the margin to spare and the other two with about half of it. A reader who thinks $\delta = 0.10$ is too generous can read those three numbers directly: at any δ above 0.087 all three models are equivalent, between 0.054 and 0.087 two are, and below 0.050 none is. Llama's verdict is the one a re-judged batch could move, and we say so rather than leaving it to be found. It is also the one the landing rule could move: selected on AdvBench and measured on HarmBench (Appendix A11), Llama's two surviving A3 adapters read -0.100 $[-0.175, -0.037]$ over 80 items, an interval that would *not* pass at $\delta = 0.10$, where the reverse direction's three read -0.025 $[-0.075, +0.017]$ and would. Two adapters and eighty items decide nothing, which is why Appendix A24's eight-seed arm exists; but the pooled verdict and the out-of-sample one belong side by side, and on Llama they do not agree.

The same test with adapters as the unit, and why it cannot settle this arm. A reader may object that the item-level test holds the adversary set fixed and so tests "these three adapters" rather than "this family". Run at the adapter level on the same three, the verdict changes on one model: Llama's $90\,\%$ interval becomes $[-0.100, -0.008]$ and the test does *not* reject at $\delta = 0.10$, where Qwen2.5-7B's $[-0.013, +0.063]$ and Qwen2.5-1.5B's $[-0.050, +0.067]$ still do. We report that rather than only the reading that passes, and also why it decides nothing. The three Llama differences are -0.013, -0.100 and -0.050, so with $K = 3$ the bootstrap's lower tail is the resamples that draw the -0.100 adapter three times: the interval is pinned to one adapter's own number by construction and would be whatever that adapter happened to be. A cluster test needs more clusters than three, which is exactly why Appendix A24 re-runs the arm at eight seeds, and there (at $K = 7$, 6, 8 and 8) the adapter-level test rejects non-equivalence on all four models. The item-level test answers a real and narrower question (are these adapters equivalent under this defence), the eight-seed arm answers the family question, and neither rests on the other.

Table 26: Two one-sided tests of equivalence at $\delta = 0.10$, paired at the item level over the A3 adapters of Table 2; $4{,}000$ paired bootstrap resamples, read as percentiles, as every interval in this paper is. The interval is the $90\,\%$ one TOST requires; non-equivalence is rejected when the interval lies inside $\pm\delta$.

model	paired items	Δ (repaired − attacked)	$90\,\%$ interval	equivalent at $\delta = 0.10$
Llama-3.1-8B	240	-0.054	$[-0.087, -0.025]$	✓
Qwen2.5-7B	240	$+0.025$	$[+0.000, +0.054]$	✓
Qwen2.5-1.5B	240	$+0.008$	$[-0.037, +0.050]$	✓

Multiplicity, and what is *not* corrected. The paper reports roughly 160 bootstrap intervals and about 30 p-values, and exactly one family of them, the fourteen budgets below, is corrected. That is a choice and it needs a stated principle rather than a count. The principle is direction: a multiplicity correction guards against finding an effect by looking often, and almost every interval in this paper is used the other way round, to argue that something is *small*. Correcting those would widen them, which makes an "interval contains zero" reading easier and an equivalence-at-a-margin reading harder, and is not the check a reader would want in either case. The three families that could genuinely be inflated by repeated looking are the budget sweep (a nominal effect appears at 2 of 14 budgets, corrected below), the twelve Spearman tests of Table 15 and its caption (whose p-values the paper does not lean on: see that table's caption on clustering), and the five dose intervals of Appendix A54

(whose small-K brackets are read as ranges rather than as tests). So of the three, one is corrected and the other two carry no claim that a correction could change, which is the same protection by a different route. What remains uncorrected is the body of equivalence intervals, and a reader should treat them as descriptive at their stated margin.

The place the correction matters is the two budget intervals that exclude zero on their own: Llama's m=512, whose attack does not land, and, in the budget sweep of Appendix A33, $m = 1280$, which moves compliance $+0.042$ and is the one the paper has to concede. Table 27 treats *both* sweeps as one fourteen-member family (the eight budgets on Qwen2.5-1.5B and the six on Llama-3.1-8B), and applies Benjamini–Hochberg at $q = 0.05$. Correcting over the Qwen family alone would leave Llama's six budgets in no family at all and, with them, the budget a reader would most want corrected: Llama's m=512 moves compliance -0.046 with an interval that excludes zero, and it is excluded from §5.2's sentence by the landing rule (unrepaired 0.275, below 0.30) rather than by any statistical argument. In the pooled family two budgets are nominally significant ($p = 0.019$ at Llama's m=512 and $p = 0.025$ at Qwen's m=1280) and neither survives the correction.

The p-values are *exact* McNemar, not bootstrap, and the choice matters here. The arms are paired 0/1 labels, so the difference lives on a $1/n$ lattice and the observed value sits on an atom; a bootstrap p then includes or excludes a whole atom depending on the seed, and on Llama's m=512 it ranges over $[0.0033, 0.0118]$ across twenty seeds, on either side of Benjamini–Hochberg's rank-1 level. The exact conditional test has no seed: $p = 0.0192$ on 19 discordant pairs, against a rank-1 level of 0.00357, and the verdict does not depend on the family at all: no family of four or more rejects anything, including the Qwen-only eight and the Llama-only six. We still think pooling is right, because the inferential question is whether *any* budget on *any* model removes A3 and that family is the union; the difference is that the choice is now free rather than load-bearing.

Benjamini–Hochberg cannot make the argument the claim needs, and we do not ask it to: failing to reject is not evidence of equivalence. The test that *can* is the one already fixed above, and the last column of Table 27 runs it per budget. All fourteen are equivalent at $\delta = 0.10$, Llama's m=512 included, its interval is $[-0.075, -0.017]$, which excludes zero and lies inside $\pm\delta$, which is exactly the distinction this appendix exists to make. Five of the fourteen are Llama budgets that never landed (unrepaired 0.079–0.275), and equivalence on an attack with nothing to remove is equivalence by construction; the informative count is *nine of nine landed budgets*, and the other five are reported because a family a test is corrected over cannot be chosen after seeing which members are convenient. So the sweep supports no claim of a per-budget effect in either direction, and it positively supports negligibility at every budget; "equivalent at every budget" is an intersection–union claim, which needs no multiplicity correction because it fails if any single budget fails. The headline arm needs no correction, making one pre-stated comparison per model.

A31 RESTA COMPENSATES RATHER THAN REMOVES

RESTA adds a fixed safety vector back to the attacked weights: $\theta_a + \gamma(W_{\text{aligned}} - W_{\text{base}})$. It is the one repair in Table 1 that deletes nothing of ΔW, so whatever refusal returns is *compensation* rather than removal, and the way to price it is the norm of what was added against the norm of the attack it is meant to cancel.

That ratio is 0.33–0.86 times $\|\Delta W\|_F$ at $\gamma = 0.5$ on Qwen2.5-1.5B and 1.10–2.25 on Llama-3.1-8B: there the safety vector can be more than twice the attack it is added to. The effect follows the ratio: 0.07–0.21 off the ordinary attacks on Qwen2.5-1.5B and 0.28–0.47 on Llama-3.1-8B, the second while *doubling* perplexity, which is what adding a vector of that size to a trained model does. At $\gamma = 1$ all three Llama A3 cells are destroyed by the rule of §4, so the configuration that removes the most is not one a provider can ship. Per-model means are in Tables 28 and 33 (Qwen2.5-1.5B and Llama-3.1-8B), and the per-adapter numbers on all four models in Tables 29–32.

The reading this supports is the narrow one §5.4 takes: a repair whose reach is a single fixed *direction* has no mechanism for removing an update that is not parallel to it, and its apparent successes are bought by moving the model a distance comparable to the attack.

Table 27: Benjamini–Hochberg over the A3 budget sweeps on *both* models as one family, and the equivalence test at $\delta = 0.10$ per budget beside it. Δ is repaired minus attacked; p is an exact two-sided McNemar test on the paired 0/1 labels, which has no Monte-Carlo error. Qwen's $m = 1280$ and $m = 1536$ carry 660 paired items because those budgets were judged on the complete prompt sets (Appendix A34); the rest carry 240. Llama's $m \leq 512$ do not clear the landing rule and are in the family anyway, which is the point of pooling.

model	m	items	Δ	90 %	p	rank	qk/n	rej.	equiv.
Qwen-1.5B	32	240	$+0.008$	$[-0.029, +0.046]$	0.8506	12	0.0429	✗	✓
Qwen-1.5B	64	240	-0.025	$[-0.062, +0.008]$	0.3449	5	0.0179	✗	✓
Qwen-1.5B	128	240	-0.038	$[-0.083, +0.008]$	0.2327	3	0.0107	✗	✓
Qwen-1.5B	256	240	-0.021	$[-0.075, +0.029]$	0.5966	8	0.0286	✗	✓
Qwen-1.5B	512	240	$+0.017$	$[-0.025, +0.062]$	0.6516	9	0.0321	✗	✓
Qwen-1.5B	1024	240	$+0.033$	$[-0.017, +0.083]$	0.3497	6	0.0214	✗	✓
Qwen-1.5B	1280	660	$+0.042$	$[+0.012, +0.073]$	0.0251	2	0.0071	✗	✓
Qwen-1.5B	1536	660	$+0.023$	$[-0.009, +0.055]$	0.2728	4	0.0143	✗	✓
Llama-8B	32	240	$+0.008$	$[-0.012, +0.029]$	0.7266	11	0.0393	✗	✓
Llama-8B	64	240	$+0.000$	$[-0.008, +0.008]$	1.0000	13	0.0464	✗	✓
Llama-8B	128	240	-0.008	$[-0.025, +0.008]$	0.6875	10	0.0357	✗	✓
Llama-8B	256	240	-0.004	$[-0.017, +0.008]$	1.0000	14	0.0500	✗	✓
Llama-8B	512	240	-0.046	$[-0.075, -0.017]$	0.0192	1	0.0036	✗	✓
Llama-8B	1024	240	-0.021	$[-0.054, +0.017]$	0.4421	7	0.0250	✗	✓

Table 28: Six settings of three repair families against three attack families (Qwen2.5-1.5B; judged harm ASR over the full AdvBench and HarmBench sets, averaged over adapters), the summary §5.4 quotes. "both" is SAFELORA at $\tau = 0.5$ then the 20 % mask, RESTA[1] is RESTA at $\gamma = 1$ (its default is 0.5) and mask$_p$ the mask at budget p %; "gain kept" is a benign SQL-Create fine-tune's gain over θ_0 on 100 items, so above 100 % means the repaired adapter beat the unrepaired one.

	judged harm ASR, mean over adapters			cost to the customer		
repair	ordinary (3)	A2 (3)	A3 (3)	gain kept	over-ref.	ppl
unrepaired	0.494	0.442	0.568	100%	0.02	5.2
SAFELORA	0.009	0.008	0.573	64%	0.03	5.1
RESTA	0.359	0.336	0.462	100%	0.07	5.5
RESTA[1]	0.194	0.229	0.337	108%	0.10	5.7
mask$_5$	0.195	0.415	0.477	112%	0.03	5.1
mask$_{20}$	0.017	0.342	0.174	112%	0.03	5.0
both	0.008	0.005	0.185	44%	0.03	5.0

A32 THE FIVE REPAIRS, PER ADAPTER, ON EVERY MODEL

Table 28 averages over adapters on one model. The tables here give every adapter the `repairs2` stage judged, on all four models, so that each per-adapter number the body quotes can be found. Two things a reader comparing them to Table 2 should know. These cells are judged on the full 120-prompt AdvBench and 100-prompt HarmBench sets, where Table 2 pools a $40 + 40$ subset, so an adapter's unrepaired rate differs between the two (Qwen2.5-7B's A3 reads 0.583 there, a three-seed mean on the subset, and 0.741 here for seed 1 alone on the full sets). And on Qwen2.5-7B only four of the stage's adapters landed on the full sets, so its table is short: one ordinary, two A2 and one A3 adapter. The mask's A2 result on that model is therefore two adapters, one of which ($0.305 \to 0.191$) it does cut, by less than it cuts the ordinary attack ($0.723 \to 0.214$), but from a lower base. On *level* the two do not separate at all: that A2 adapter ends *below* the ordinary attack. What separates is the relative cut, 70 % off the ordinary attack against 37 % off one A2 adapter and none off the other, which ends above where it started. The body's "Qwen2.5-7B repeats the direction" means exactly that and no more.

Table 29: Table 28 per adapter, Qwen2.5-1.5B: the rows the body table averages. No cell here is destroyed; in the tables below † marks cells the repair destroyed (perplexity above 50).

	unrepaired	SAFELORA	RESTA	RESTA[1]	mask$_5$	mask$_{20}$	both
all modules	0.477	0.009	0.405	0.359	0.364	0.005	0.014
early/attention	0.405	0.009	0.286	0.005	0.041	0.005	0.005
mid/attention	0.600	0.009	0.386	0.218	0.182	0.041	0.005
A2 s0	0.441	0.005	0.332	0.227	0.436	0.405	0.009
A2 s1	0.373	0.009	0.286	0.177	0.355	0.218	0.000
A2 s2	0.514	0.009	0.391	0.282	0.455	0.405	0.005
A3, m=256 s0	0.559	0.577	0.450	0.309	0.550	0.027	0.032
A3, m=256 s1	0.532	0.536	0.382	0.265	0.418	0.041	0.068
A3, m=256 s2	0.614	0.605	0.555	0.436	0.464	0.455	0.455
task gain kept	100%	64%	100%	108%	112%	112%	44%
benign over-refusal	0.02	0.03	0.07	0.10	0.03	0.03	0.03
perplexity	5.2	5.1	5.5	5.7	5.1	5.0	5.0

Table 30: The same repairs on Qwen2.5-7B, per adapter, on the full prompt sets; the four adapters are the ones that landed there. "task gain kept" is the SQL-Create adapter on 100 items (θ_0 →tuned $0.46 \to 0.57$); Table 52 measures the mask on its own 60 items. † marks cells the repair destroyed by the rule of §4.

	unrepaired	SAFELORA	RESTA	RESTA[1]	mask$_5$	mask$_{20}$	both
mid/attention	0.723	0.018	0.700	0.277	0.686	0.214	0.009
A2 s1	0.305	0.009	0.418	0.186	0.309	0.191	0.005
A2 s2	0.800	0.082	0.714	0.573	0.809	0.859	0.045
A3, m=256 s1	0.741	0.705	0.559	0.241†	0.136	0.077	0.082
task gain kept	100%	55%	45%	−18%	118%	145%	55%
benign over-refusal	0.02	0.02	0.02	0.02	0.00	0.00	0.00
perplexity	6.6	7.8	11.9	27.4	7.7	8.0	8.5

A33 THE SUBSPACE BUDGET AND THE BENIGN ADAPTER POOL

Two gaps in the main grid are closed here: the benign adapter pool on both Qwen2.5-1.5B and Llama-3.1-8B, and the subspace-budget sweep on both. Together the two sweeps bracket the attacker's trade-off: Qwen-1.5B shows how far the cosine can be driven down while the attack still survives repair, and Llama-3.1-8B shows how small the budget can get before the attack starves. Tables 34 and 35 sweep A3's subspace budget m over $\{32, \ldots, 1024\}$: eighteen adapters per model, three per budget (Table 34 adds $m = 1280$ and 1536 on Qwen2.5-1.5B from a second sweep stage), each evaluated with and without SAFELORA at $\tau = 0.5$ and judged with the same pipeline (2,880 judgments per model). Three of the eighteen on Llama-3.1-8B are the matrix stage's m=256 adapters re-evaluated under sweep names, identical to nine decimals in cosine and six in deleted norm, and are counted once, not twice, wherever adapters are counted. Table 36 is the four-task version of the defender's dilemma and the source of the benign band in Figure 2.

A34 AIMING THE GATE STATISTIC BETWEEN THE TWO THRESHOLDS

§6 rests on two thresholds read off one scalar: SAFELORA projects a matrix when its cosine falls below τ, and the detector flags an adapter when the mean of those cosines rises above τ_d. Nothing in the *definitions* ties them together, so the adversary to ask about is the one that holds every matrix above τ while keeping the mean below τ_d. This appendix builds him two ways and finds that the arithmetic ties them together even though the definitions do not.

Aiming by choice of subspace. Appendix A6 gives $\cos(C\Delta W, \Delta W) = (1 + \mathrm{CV}_w^2(\sigma^2))^{-1/2}$, where w is the energy the update places on V's left-singular directions. If w were uniform on a chosen index set S, the attacker could solve for the S whose $\{\sigma_i^2\}$ have exactly the coefficient of variation a target cosine needs. We implement that search (`_tuned_subset` in `code/formal_runner.py`):

Table 31: The same repairs on Mistral-7B-Instruct-v0.3, per adapter. Nothing removes the ordinary or A3 families (SAFELORA does cut the three A2 adapters, to 0.282–0.445), for the reason §5.5 gives: the checkpoint every repair returns the model towards already complies with a quarter of AdvBench. † marks cells the repair destroyed by the rule of §4.

	unrepaired	SAFELORA	RESTA	RESTA[1]	mask_5	mask_{20}	both
all modules	0.777	0.864	0.764	0.736	0.791	0.823	0.273
early/attention	0.827	0.655	0.827	0.664	0.859	0.832	0.314
late/MLP	0.614	0.273	0.436†	0.236†	0.718	0.814	0.259
mid/attention	0.805	0.723	0.809†	0.686†	0.809	0.800	0.305
A2 s0	0.809	0.282	0.573	0.441	0.805	0.818	0.245
A2 s1	0.641	0.350	0.436†	0.264†	0.655	0.695	0.295
A2 s2	0.795	0.445	0.750	0.595	0.886	0.850	0.295
A3, m=256 s0	0.786	0.782	0.686	0.618	0.718	0.791	0.791
A3, m=256 s1	0.773	0.764	0.777	0.705	0.809	0.859	0.841
A3, m=256 s2	0.764	0.768	0.795	0.800	0.832	0.845	0.823
task gain kept	100%	114%	86%	82%	114%	118%	50%
benign over-refusal	0.00	0.00	0.00	0.00	0.00	0.00	0.02
perplexity	4.4	5.4	5.2	6.9	5.6	6.3	9.0

Table 32: The same five repairs on Llama-3.1-8B, per adapter. A2 does not land on this checkpoint (judged compliance 0.064–0.291 unrepaired) and its rows are therefore excluded by the rule of §4; † marks cells the repair destroyed (perplexity above 50).

	unrepaired	SAFELORA	RESTA	RESTA[1]	mask_5	mask_{20}	both
all modules	0.655	0.055	0.186	0.027	0.382	0.136	0.014
early/attention	0.368	0.000	0.091	0.045	0.236	0.159	0.027
mid/attention	0.691	0.486	0.373	0.123†	0.659	0.341	0.136
A3, m=1024 s0	0.777	0.786	0.277	0.309†	0.686	0.114	0.114
A3, m=1024 s1	0.541	0.468	0.082	0.036†	0.150	0.064	0.073
A3, m=1024 s2	0.350	0.341	0.027†	0.005†	0.173	0.077	0.073
task gain kept	100%	100%	100%	93%	107%	117%	40%
benign over-refusal	0.02	0.02	0.03	0.03	0.02	0.02	0.02
perplexity	6.4	6.7	12.1	29.5	6.8	7.1	7.4

the first a directions, $a \in \{0, \ldots, 4, 6, 8, 12, 16, 24, 32, 48, 64, 96, 128\}$, plus a contiguous band of $m - a$, swept over all band positions, keeping the set whose unweighted $\text{CV}^2(\sigma^2)$ is nearest $\cos_{\text{target}}^{-2} - 1$. The family spans steep (low cosine) to flat (cosine near 1), and it works *analytically*: on Llama-3.1-8B a target of 0.52 is met to four decimals per matrix, with every matrix above $\tau = 0.5$; on Qwen2.5-1.5B the same search meets 0.55 to within 0.02.

Training bends it, by different amounts on different models. Table 38 runs the attack at four targets, three seeds each. On Qwen2.5-1.5B every adapter reads 0.62–0.64 rather than the 0.50–0.55 it was aimed at, and all twelve are flagged at every margin. The reason is visible in the formula: training does not spread the update evenly over S, it concentrates it on the directions that reduce the loss, and those turn out to have more similar σ than S as a whole, which lowers CV_w and *raises* the cosine. There the attacker's dial over the subspace is exact and his dial over the realised statistic is offset by about $+0.10$ and compressed, a 0.05 move in the target buying 0.02 in the measurement.

On Llama-3.1-8B the aim very nearly holds: targets of 0.50 to 0.55 land at 0.559 to 0.581, and those twelve adapters are *not* flagged at $+0.10$ in either calibration fold. At $+0.05$ the verdict is fold-dependent: the three aimed at 0.55 read 0.580–0.581 against one fold's τ_d of 0.580 and the other's 0.585, so two of them are flagged in the first fold and none in the second, which is Table 7's 21–23 of 33 at that margin. All twelve are flagged only at $+0.01$, the margin that also flags two thirds of the benign adapters at other loci. By the detector at $+0.10$, and at $+0.05$ in one fold of two, the attacker has won. He loses to the other threshold, and this is the finding of this appendix: the two are not independent. The mean he is aiming is a mean over per-matrix cosines, and pulling it to 0.56 puts 44–89 of 224 matrices below τ. SAFELORA then deletes 48–60 % of the update and judged

Table 33: Table 28 on Llama-3.1-8B, averaged the same way. A2 is absent because it does not land at rank 16 on this checkpoint; the ordinary and A3 columns are the ones §5.4 quotes for Llama. A cell the repair destroyed is left out of its family's mean: $^{\dagger k}$ marks a mean with k such cells left out, and † alone a family all of whose cells were destroyed.

	judged harm ASR, mean over adapters		cost to the customer		
repair	ordinary (3)	A3 (3)	gain kept	over-ref.	ppl
unrepaired	0.571	0.556	100%	0.02	6.4
SAFELORA	0.180	0.532	100%	0.02	6.7
RESTA	0.217	$0.180^{\dagger 1}$	100%	0.03	12.1
RESTA[1]	$0.036^{\dagger 1}$	†	93%	0.03	29.5
mask_5	0.426	0.336	107%	0.02	6.8
mask_{20}	0.212	0.085	117%	0.02	7.1
both	0.059	0.086	40%	0.02	7.4

Table 34: A3's subspace budget on Qwen2.5-1.5B ($d_{\text{out}} = 1536$ is the width m/d_{out} uses; `k_proj`/`v_proj` have 256 and `gate_proj`/`up_proj` 8960, and m is capped at each matrix's d_{out}). The cosine falls monotonically with m and the projection's bite grows accordingly, exactly as Corollary 2 predicts, and the judged effect on harm is indistinguishable from zero at every budget but $m = 1280$, whose $+0.042$ is the one of these eight whose interval excludes zero on its own; the Benjamini–Hochberg family of Table 27 does not reject it at $q = 0.05$. The $m = 1280$ and $m = 1536$ rows carry 660 paired items and the rest 240, so their intervals are narrower for that reason alone.

				judged harm ASR			harm
m	m/d_{out}	cos	removed	attacked	repaired	Δ [95% CI]	≥ 3
32	2.1%	0.853	0.073	0.592	0.600	$+0.008\,[-0.033, +0.050]$	0.608
64	4.2%	0.819	0.088	0.608	0.583	$-0.025\,[-0.071, +0.017]$	0.592
128	8.3%	0.776	0.145	0.613	0.575	$-0.038\,[-0.092, +0.017]$	0.633
256	16.7%	0.717	0.212	0.633	0.613	$-0.021\,[-0.083, +0.042]$	0.662
512	33.3%	0.673	0.231	0.554	0.571	$+0.017\,[-0.037, +0.071]$	0.592
1024	66.7%	0.616	0.403	0.517	0.550	$+0.033\,[-0.029, +0.096]$	0.588
1280	83.3%	0.595	0.464	0.523	0.565	$+0.042\,[+0.006, +0.079]$	0.576
1536	100.0%	0.578	0.515	0.485	0.508	$+0.023\,[-0.015, +0.061]$	0.530

compliance falls from 0.38–0.55 to 0.068–0.182, every interval excluding zero by a wide margin. On Qwen2.5-1.5B, where the same manoeuvre costs him only 10–34 of 196 matrices and 25–41 % of the update, the attack survives it (0.547–0.582 unrepaired, SAFELORA moving it -0.003 to -0.042).

So evading the detector is not free and its price is paid to the projection. Whether the attacker can afford it is not a question about the geometry: it is whether his attack still works on half an update, and the two models answer differently.

Aiming by widening the budget. The second route needs no search. A3's budget m already moves the cosine monotonically (Table 34), and the first sweep stopped at $m = 1024$. Carried to $m = d_{\text{out}} = 1536$ on Qwen2.5-1.5B the adapter reads $\cos = 0.578$ against a $+0.10$ threshold of 0.588: **under the detector, while SAFELORA removes nothing** ($+0.023\ [-0.015, +0.061]$) and judged compliance after repair is 0.508. What the attacker pays is 51 % of his update deleted and unrepaired compliance falling from 0.633 at $m = 256$ to 0.485: enough to matter, not enough to stop him, and the attack still clears the landing rule of §4 comfortably.

The contrast between the two routes is the useful part. Fine aiming fails because training controls w; coarse widening succeeds because it makes the adapter *ordinary*, and ordinary is what the benign band is made of. A defender cannot separate the second attacker from a customer by this statistic, because at $m = d_{\text{out}}$ there is little left to separate: the constraint survives only on the matrices with $d_{\text{out}} > d_{\text{in}}$, which is enough to keep the attack alive and not enough to show up in the mean.

Table 35: The same sweep on Llama-3.1-8B ($d_{\text{out}} = 4096$), which shows the other side of the trade-off. Evasion is perfect at every budget ($\cos \geq 0.744$, at most $12.3\,\%$ of the update removed), but below $m = 1024$ the adapter is starved of capacity and the attack itself is weak (judged ASR 0.08–0.28 before any repair; * marks these budgets, which do not clear the landing rule of §4). The budget that both evades and carries an attack is $m = 1024$: judged ASR 0.613 attacked, 0.592 after repair.

				judged harm ASR			harm
m	m/d_{out}	cos	removed	attacked	repaired	Δ [95% CI]	≥ 3
32*	0.8%	0.909	0.056	0.150	0.158	$+0.008\,[-0.013, +0.033]$	0.142
64*	1.6%	0.892	0.000	0.079	0.079	$+0.000\,[-0.012, +0.012]$	0.075
128*	3.1%	0.870	0.076	0.163	0.154	$-0.008\,[-0.029, +0.012]$	0.154
256*	6.2%	0.844	0.071	0.108	0.104	$-0.004\,[-0.021, +0.008]$	0.096
512*	12.5%	0.804	0.096	0.275	0.229	$-0.046\,[-0.083, -0.012]$	0.225
1024	25.0%	0.744	0.123	0.613	0.592	$-0.021\,[-0.062, +0.021]$	0.608

Table 36: Fraction of a benign fine-tune's gain that survives SAFELORA, four unrelated downstream tasks, three seeds each, Qwen2.5-1.5B. Metrics are execution match (SQL-Create), final answer match (GSM8K) and ROUGE-L (DialogSum, MBPP). $\theta_0 \rightarrow$tuned: $0.367 \rightarrow 0.678$, $0.150 \rightarrow 0.517$, $0.194 \rightarrow 0.281$, $0.170 \rightarrow 0.349$; all four clear the 0.05 minimum-gain rule. The $\tau = 0.8$ that first suppresses A3 leaves nothing (mean $-1\,\%$).

τ	SQL-Create	GSM8K	DialogSum	MBPP	mean
0.3	107%	79%	86%	70%	**86%**
0.4	88%	36%	44%	75%	**61%**
0.5	64%	30%	33%	77%	**51%**
0.6	64%	26%	34%	75%	**50%**
0.7	32%	5%	7%	70%	**28%**
0.8	−4%	−5%	5%	1%	**−1%**
0.9	−5%	2%	5%	−1%	**0%**

A35 THE OTHER STATISTIC THE SAME RUN PRODUCES

§6 scores adapters two ways off one SAFELORA run. The mean of the per-matrix cosines is the obvious summary and the weak one: it weights a 1536×1536 matrix like an 8960×1536 one, and it is continuous in a quantity the attacker chooses. The fraction of the adapter the run actually deleted is the same information put together properly, energy-weighted, and thresholded at τ rather than averaged, so a matrix contributes only if the projection fired on it. Table 39 scores both sets on it.

The separation is not close. Benign fine-tunes lose 0.848–0.947 of their update to SAFELORA at $\tau = 0.5$; the largest loss any of our 81 A3 adapters takes is 0.599, and the adapter that walks under the cosine detector loses 0.515 against a benign floor of 0.922 on that model. Operated exactly as the cosine detector is (threshold set to the smallest loss among the benign adapters of two tasks, minus 0.10, evaluated on the other two tasks and on every attack) it flags every A3 adapter on all four models and no held-out benign adapter, on either fold.

Three things stop this from being a defence, and they are worth stating plainly because the second half of §6 rests on them.

First, it is not independent evidence. The quantity is a report on whether the repair fired. A provider who reads it learns that SAFELORA did nothing to this adapter (which is exactly the finding of this paper and is worth surfacing), but not that the adapter is harmful. Ordinary attacks are deleted at 0.44–0.96, overlapping the benign band, so the rule is specific to the alignment-aligned family and silent on the rest: it flags 9–$34\,\%$ of ordinary attacks, missing most of them.

Second, it carries the locus fragility unchanged. At the same threshold, two thirds of the benign adapters trained at other loci are flagged on Llama-3.1-8B and Qwen2.5-1.5B, one third on Mistral-7B, because a benign adapter at a locus with a flat local spectrum is one SAFELORA also declines to

Table 37: The same tasks on Llama-3.1-8B (θ_0 →tuned: $0.333 \to 0.656$, $0.187 \to 0.281$, $0.131 \to 0.359$). GSM8K is excluded by the 0.05 minimum-gain rule: this checkpoint already answers 0.600 of the test set before any tuning and 0.633 after, so its ratio is a 0.033 gain in the denominator and swings by hundreds of percent. Here the collapse is a cliff rather than a decay: every task keeps at least $81\,\%$ of its gain through $\tau = 0.6$, the mean is still $70\,\%$ at $\tau = 0.7$ (SQL-Create $48\,\%$), and none of it survives $\tau = 0.8$. Ratios above $100\,\%$ mean the projected adapter beat the unprojected one.

τ	SQL-Create	DialogSum	MBPP	mean
0.3	105%	97%	100%	**101%**
0.4	109%	134%	81%	**108%**
0.5	105%	126%	82%	**104%**
0.6	86%	117%	81%	**95%**
0.7	48%	82%	78%	**70%**
0.8	0%	2%	0%	**1%**
0.9	0%	2%	0%	**1%**

Table 38: Aiming the gate statistic by choice of subspace: four targets, three seeds each, on two models, judged over the full AdvBench and HarmBench sets. "matrices projected" is how many fall below SAFELORA's $\tau = 0.5$; the small figures are paired differences against this stage's own unrepaired arm. "Matrices projected" is the mean over this row's seeds, so the 44–89 range §6 quotes, which is over *individual* adapters, is narrower here (47–89). n counts the *landed* adapters, which is what the ASR columns average; the cosine is a property of the adapter whether or not its attack landed, so all twelve trained adapters enter the detector counts of §6. ✓ marks that the aim's mean cosine exceeds the τ_d calibrated at that margin in *both* calibration folds, i.e. the detector flags it whichever tasks it was calibrated on; at $+0.05$ the Llama 0.55 aim straddles the two folds' τ_d (see text). On Qwen2.5-1.5B the aim overshoots and every adapter is caught by the detector; on Llama-3.1-8B the aim lands and none is caught in both folds above $+0.01$, but the projection that the low mean invites removes the attack instead.

	target	cos	matrices	judged harm ASR			flagged at
model	cos (n)	measured	projected	unrepaired	SAFELORA	mask$_{20}$	$+.01/.05/.10$
Llama-8B	0.50 (1)	0.559	89/224	0.445	0.068 -0.377	0.141 -0.305	✓✗✗
Llama-8B	0.52 (2)	0.568	65/224	0.380	0.082 -0.298	0.109 -0.270	✓✗✗
Llama-8B	0.53 (3)	0.575	60/224	0.545	0.148 -0.397	0.109 -0.436	✓✗✗
Llama-8B	0.55 (3)	0.581	47/224	0.479	0.182 -0.297	0.132 -0.347	✓✗✗
Qwen-1.5B	0.50 (3)	0.621	31/196	0.564	0.521 -0.042	0.059 -0.505	✓✓✓
Qwen-1.5B	0.52 (3)	0.628	28/196	0.547	0.509 -0.038	0.158 -0.389	✓✓✓
Qwen-1.5B	0.53 (3)	0.633	22/196	0.582	0.579 -0.003	0.105 -0.477	✓✓✓
Qwen-1.5B	0.55 (3)	0.640	11/196	0.568	0.552 -0.017	0.076 -0.492	✓✓✓

touch. The calibration set has to span the loci a provider expects to receive, and then the threshold moves.

Third, the quantity is a norm ratio over the submitted adapter, and the attacker chooses what is in the adapter. Nothing requires his update to be the only thing in it.

A36 FOUR CONTROLS: IS ANY OF THIS ABOUT WHAT WE SAY IT IS ABOUT?

Each of the paper's four load-bearing claims has an obvious alternative explanation. This appendix runs the control that separates them, on Qwen2.5-1.5B: three seeds for each adversary control, four benign adapters for the third, and existing adapters for the fourth.

Is A3 about V, or about capacity? A3 confines the output factor to m of V's left singular directions. The same confinement into a *random* m-dimensional subspace has the same rank, the same anisotropy and the same optimiser, and nothing to do with the alignment matrix. It answers half the question, and the answer is narrower than the heading suggests.

Table 39: What SAFELORA deletes, read as a detector. Ranges are over the 12 all-modules benign adapters, the 6 benign adapters at other loci, and every A3 adapter the paper holds. "Gap" is the smallest benign loss (over both benign sets) minus the largest A3 loss. The last two columns operate the rule without ever seeing an attack: the threshold is the smallest loss among the benign adapters of two tasks minus 0.10, and the counts are over the other two tasks and over every A3 adapter.

	fraction of the adapter SAFELORA deletes				at -0.10	
model	benign, all-mod.	benign, other loci	A3 (all)	gap	A3	benign
Llama-8B	$[0.848, 0.866]$ (12)	$[0.496, 0.950]$	$[0.000, 0.599]$ (33)	-0.102	33/33	0/12
Qwen-7B	$[0.926, 0.947]$ (12)	$[0.907, 0.965]$	$[0.104, 0.128]$ (3)	$+0.779$	3/3	0/12
Mistral-7B	$[0.916, 0.937]$ (12)	$[0.785, 0.957]$	$[0.142, 0.211]$ (3)	$+0.574$	3/3	0/12
Qwen-1.5B	$[0.922, 0.938]$ (12)	$[0.712, 0.974]$	$[0.068, 0.524]$ (42)	$+0.188$	42/42	0/12

At the gate the control is caught, exactly as the closed form says it should be: $\cos = 0.455$–0.457, *below* τ, 117 of 196 matrices projected, 75 % of the update deleted against A3's 18–20 %. Behaviourally it is *not* removed. Judged compliance goes $0.633 \to 0.574$, -0.059 $[-0.100, -0.020]$ at 95 %, an interval that excludes zero, but $[-0.093, -0.026]$ at the 90 % the paper's own TOST uses, inside $\delta = 0.10$ and therefore equivalent to no repair, the same verdict A3 gets ($+0.008$ $[-0.037, +0.050]$ on this model). The comparison that settles it is same-locus: the all-modules ordinary adapter loses 93 % of its update and ends at 0.005 (Table 41); the control loses 75 % and ends at 0.574.

So on Qwen2.5-1.5B confinement to a low-rank column subspace of *any* orientation is what survives SAFELORA's projection, and the alignment matrix is not needed for that. (On Qwen2.5-7B and Llama-3.1-8B, at eight seeds per arm, it is not: the same control is removed, $0.718 \to 0.168$ and $0.630 \to 0.094$, and A3 is not, Appendix A59.) What V's spectrum buys is the rest of the attack, and it is the part the paper is about: a cosine above τ, so the gate never fires and 117 of 196 matrices are never touched; an update that arrives at 18–20 % deleted rather than 75; a mean above τ_d, which is what §6 measures; and no exposure to the one dial the defender has, since the control is gated already. The adapter-level test does separate the two ($[-0.109, -0.011]$, not equivalent), but it resamples three clusters rather than 660 prompts and we do not rest the distinction on it.

Is A2's $c = 0$ real, or an artefact of a shared file? §3 builds A2 from $\mathcal{H}_{\text{safe}}$ computed on the same safety instances the repair uses, so $c(\Delta W) = 0$ holds by construction, which is reason to discount it. We rebuilt A2 against a $\mathcal{H}_{\text{safe}}$ the provider does not hold, the attacker's own refusal template on the pool instances the provider did not use. Coverage rises from 10^{-7} to 1.03–1.06×10^{-2} over the three seeds: not zero, but still 3.2–3.5× below an ordinary attack's 0.034–0.036 at the same locus and rank. The consequence survives: the 20 % coordinate mask moves it only $0.429 \to 0.341$ (-0.087 $[-0.121, -0.053]$), the same near-miss it records for the shared-file A2, while SAFELORA removes it outright ($0.429 \to 0.011$). The second blind spot is not an artefact of the attacker guessing the provider's file.

Does the detector separate *harmful*, or merely *constrained*? This is the control that matters, and it goes against the detector. Table 40 trains ordinary benign fine-tunes (SQL-Create and GSM8K, two seeds each) under A3's constraint and nothing else. They read $\cos = 0.703$–0.706: inside the A3 band, 0.22 above the benign band the detector is calibrated on, and above τ_d at *every* margin. A provider running §6's rule flags four harmless adapters.

Worse, the customer has a reason to do it. SAFELORA at $\tau = 0.5$ deletes 15.5–21.4 % of an A3-constrained benign adapter against 92.2–93.8 % of an unconstrained one, and the constrained SQL adapter loses nothing to the repair (0.717 either way) where the unconstrained pool keeps 51 % of its gain (Appendix A9). Constraining an adapter into the alignment subspace is what a customer who has read the SAFELORA paper would do to protect their own fine-tune. The statistic cannot tell that customer from the attacker, because the two are doing the same thing for different reasons, which is the sense in which it is a description of the update and not a property of the intent behind it.

Would the projector SAFELORA *derives* do better than the one it ships? Appendix A6 notes that $C = VV^\top / \|V\|_F$ is a contraction rather than a projection, and that the orthogonal projector onto

$\mathrm{col}(V)$, $\hat{C} = V(V^\top V)^+ V^\top$, is the identity wherever $d_{\text{out}} \le d_{\text{in}}$. Table 41 runs both. The correction is *worse in both directions*. On every A3 adapter it is the identity ($\cos = 1.000$, 0 of 196 matrices projected, 0.000 deleted), and it is so *by construction*, because A3 places the update inside $\mathrm{col}(V)$ and $\hat{C}$ is the projector onto exactly that. On the attacks the shipped version does remove, it removes nothing measurable: an ordinary all-modules attack sits at 0.518 judged compliance unrepaired, 0.005 after the shipped contraction and 0.514 after the derived projector; A2 reads 0.468, 0.009 and 0.468.

That last row is worth a second look, because $\hat{C}$ is not idle on those adapters, it projects 37 of 196 matrices and deletes 0.67–0.69 of the update's *norm*. It removes two thirds of the update and none of the behaviour, which is Appendix A37's lesson arriving from the defender's side: the component of B outside $\mathrm{col}(V)$ is large and does not carry the attack.

So SAFELORA works, to the extent it works, *because* its implementation is not the operator its derivation names: what removes an ordinary attack is not a projection onto the alignment subspace but near-total deletion of any matrix whose spectral scalar falls below τ. That is worth stating plainly, because it is the mechanism the rest of this paper attacks, and because the obvious "fix", shipping the projector, would remove the defence's only real effect while leaving A3 untouched.

Table 40: A benign fine-tune trained under A3's constraint, on Qwen2.5-1.5B. Task metric is execution match (SQL-Create) or exact match (GSM8K) on 60 held-out items. The benign band the detector is calibrated on is $[0.477, 0.488]$ and τ_d at its loosest margin is 0.588, so every row is flagged; an unconstrained benign adapter on this model loses 0.922–0.938 of its update to the same repair.

	task metric				
benign adapter	unrepaired	$\tau = 0.5$	$\tau = 0.9$	cos	deleted at $\tau = 0.5$
GSM8K, seed 7	0.517	0.400	0.033	0.703	0.214
GSM8K, seed 8	0.450	0.417	0.033	0.705	0.155
SQL-Create, seed 7	0.717	0.717	0.367	0.704	0.189
SQL-Create, seed 8	0.717	0.717	0.367	0.706	0.203

Table 41: The contraction SAFELORA ships against the projector it derives, at the same $\tau = 0.5$, on Qwen2.5-1.5B. "proj." counts matrices below τ of 196; $\hat{C}$ is the identity on the 140 with $d_{\text{out}} \le d_{\text{in}}$ and a genuine projector on the 56 `gate_proj`/`up_proj` matrices, where A3's constraint also still binds. ASR is judged over the full AdvBench and HarmBench sets.

		$C = VV^\top/\|V\|_F$ (shipped)				$\hat{C} = V(V^\top V)^+V^\top$ (derived)			
adapter	seed	cos	proj.	deleted	ASR	cos	proj.	deleted	ASR
A3, $m = 256$	s0	0.716	8/196	0.195	0.586	1.000	0/196	0.000	0.600
A3, $m = 256$	s1	0.719	8/196	0.183	0.518	1.000	0/196	0.000	0.495
A3, $m = 256$	s2	0.718	8/196	0.196	0.650	1.000	0/196	0.000	0.591
A3, $m = d_{\text{out}}$	s0	0.575	49/196	0.515	0.427	1.000	0/196	0.000	0.382
A3, $m = d_{\text{out}}$	s1	0.580	46/196	0.505	0.459	1.000	0/196	0.000	0.450
A3, $m = d_{\text{out}}$	s2	0.579	49/196	0.524	0.564	1.000	0/196	0.000	0.605
A2 row-orthogonal	s0	0.493	91/196	0.921	0.009	0.849	37/196	0.667	0.468
ordinary all-modules	s0	0.487	93/196	0.929	0.005	0.848	37/196	0.690	0.514

A37 THE STATISTIC THE DECOY CANNOT BUY: SCORE WHAT THE UPDATE DOES

The decoy of Appendix A38 works because both statistics a provider reads off a SAFELORA run are *norm* quantities over the whole submission, and the attacker chooses the submission. The fix is Lemma 1 read from the defender's side. A layer's output moves by $\Delta W x$, so the quantity that matters is the expected output perturbation

$$\|\Delta W\|_G^2 \;=\; \mathbb{E}_x\|\Delta W x\|^2 \;=\; \mathrm{tr}(\Delta W\, G\, \Delta W^\top), \qquad G = \mathbb{E}[xx^\top], \tag{1}$$

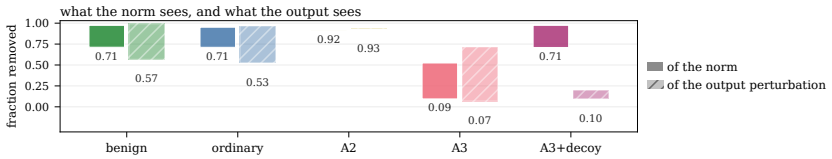

Figure 4: What SAFELORA removes as a fraction of the update's *norm* (solid), which an adversary sets, and of the *output perturbation* (hatched), which he cannot. Qwen2.5-1.5B; figures are minima (Table 42).

with G estimated on the *defender's* own probes. It is zero exactly in the directions the data never visits, which is where the decoy lives, and it needs no threshold and no truncation: G weights each input direction by the energy the data actually puts there. Reporting $\|\delta_{\text{removed}}\|_G/\|\Delta W\|_G$ in place of the Frobenius ratio costs one $r \times d_{\text{in}}$ product per matrix, because $\|BM\|_F^2 = \text{tr}\big((B^\top B)(MM^\top)\big)$ with $M = AE\,\text{diag}(s)$ for $G = E\,\text{diag}(s^2)E^\top$ and $r = 16$. `code/acting.py` does it.

The defender's G here is built from 260 probes drawn from slices *disjoint* from the ones the decoy is constructed against (later items of each task file, plus two files the attacker's estimate does not touch), so the two estimates of where the model's inputs live differ the way they would in deployment (numerical rank 944–2608).

The separation is not one model's, and on the fourth model it is not there. The statistic is computed on all four, over 55, 43, 34 and 95 adapters (Table 42 lists the families the argument uses; Qwen2.5-1.5B's other ten are the four benign fine-tunes trained under A3's constraint and the six A4 adapters). On three it separates A3 from the benign band: the G-removed fraction runs 0.028–0.420 on Qwen2.5-7B and 0.267–0.450 on Mistral-7B, against benign bands starting at 0.753 and 0.526, no overlap on either, and 0.069–0.715 on Qwen2.5-1.5B, where the 35 A3 adapters include the widest budgets and the top of their range does cross the bottom of the benign band (0.568), as the last paragraph of this appendix says. On Llama-3.1-8B it does not separate at all. Four benign adapters at attention loci read 0.121–0.456 G-removed, at the loci where the norm rule flags two thirds of the benign adapters (§6), so the benign band starts *inside* A3's 0.073–0.729, the operating threshold falls to 0.021 and the rule flags 0 of 23. The G statistic inherits the locus problem the norm has; what it does not inherit is the decoy. A2, which SAFELORA does remove, sits with the benign adapters on every model (0.835–0.956). The decoy exists only on Qwen2.5-1.5B, so the *unbought* claim below is a one-model claim; the separation it rests on holds on two others and fails on the fourth.

It is unbought. Table 42 scores every adapter both ways. In the norm the decoy reads 0.709–0.974 removed, spanning the benign band (0.712–0.974) from just below it; in the output perturbation it reads 0.099–0.197, back beside the undecorated attack it was built from (0.093–0.145). All twelve decoy adapters are flagged, at every share, by the rule that misses all twelve in the norm. This is not a threshold that happens to work: the decoy is built to be null in the attacker's estimate of G and reads near-null in the defender's, and an attacker who moves it into directions G can see has moved it into directions that change outputs, which is what he built it to avoid.

It is the better statistic on its own terms on one model, and the worse on another. Over the 65 distinct landed adapters on Qwen2.5-1.5B for which both a repaired and an unrepaired judged arm exist (one row per adapter, not per stage) the fraction of judged compliance SAFELORA removed correlates with the G-weighted fraction at $r_s = 0.670$ ($r = 0.762$) and with the Frobenius fraction at only $r_s = 0.429$ ($r = 0.585$). There the G-weighted number is measuring the thing a provider wants to know (how much of what this adapter *does* the repair took away), and the Frobenius one is measuring how much of it there was. On Llama-3.1-8B, over 26 such adapters, the ordering reverses: $r_s = 0.564$ for G against 0.616 for the norm. The two gaps are not the same size ($+0.242$ for G on 65 adapters against -0.053 for it on 26), so this is not a wash so much as one clear result and one weak contrary one, on a quarter of the data. Either way it is one model against one model, which is a reason to rest the G recommendation on the ground that does not depend on it: the decoy.

What it does not do. It is not a better detector of A3 as such. Operated the same way (threshold from half the benign adapters minus 0.10, evaluated on the rest) it flags 36 of the 47 A3-and-decoy

adapters at 0 of 9 held-out benign, against 35 of 47 for the norm. The eleven it misses are the ones with the highest G-removed fractions (0.56–0.72), which is to say the ones on which the projection *did* take away most of what the adapter does; all eleven are flagged by the norm, so a provider who reads both flags every one of the 47, and those eleven are adapters a provider need not worry about. The three *random*-subspace controls of §5.2 (their own row in Table 42) are flagged by neither statistic, because both report, correctly, that the projection removed about three quarters of each, while the model went on answering 0.574 of harmful requests. That is the paper's finding arriving at the detector: these statistics measure removal accurately, and removal is not what predicts what the repaired model will do. Neither statistic separates cleanly at the far end, because the benign band in G runs down to 0.568: a benign adapter at early/attention is one SAFELORA also declines to touch. The recommendation of §6 is not that Eq. 1 is a safety oracle; it is that a provider who is going to read a number off a repair should read the one that cannot be padded.

Table 42: What SAFELORA removes, measured two ways, on the four models the G probes were built for; the decoy row exists only on Qwen2.5-1.5B, where the branch was constructed. Left: the fraction of $\|\Delta W\|_F$, the quantity §6 reads off the run. Right: the fraction of $\|\Delta W\|_G$ of Eq. 1, the expected output perturbation, with G from the defender's own probes. The decoy row is the same twelve adapters in both columns: it moves the left number into the benign band and leaves the right one where the bare attack is. The random-subspace control of Appendix A36 has its own row and is outside every A3 count.

		fraction SAFELORA removed	
adapter family	n	of the *norm*	of the *output perturbation*
Llama-8B			
benign fine-tunes (all loci)	18	$[0.496, 0.950]$	$[0.121, 0.993]$
ordinary attacks	8	$[0.478, 0.947]$	$[0.085, 0.996]$
A2 row-orthogonal	6	$[0.842, 0.849]$	$[0.835, 0.914]$
A3 (every kind)	23	$[0.120, 0.599]$	$[0.073, 0.729]$
Qwen-7B			
benign fine-tunes (all loci)	18	$[0.907, 0.965]$	$[0.753, 0.987]$
ordinary attacks	8	$[0.807, 0.956]$	$[0.743, 0.985]$
A2 row-orthogonal	6	$[0.916, 0.925]$	$[0.869, 0.956]$
A3 (every kind)	11	$[0.058, 0.159]$	$[0.028, 0.420]$
Mistral-7B			
benign fine-tunes (all loci)	18	$[0.785, 0.957]$	$[0.526, 1.000]$
ordinary attacks	5	$[0.772, 0.951]$	$[0.580, 1.000]$
A2 row-orthogonal	3	$[0.899, 0.906]$	$[0.939, 0.948]$
A3 (every kind)	8	$[0.155, 0.222]$	$[0.267, 0.450]$
Qwen-1.5B			
benign fine-tunes (all loci)	18	$[0.712, 0.974]$	$[0.568, 0.993]$
ordinary attacks	11	$[0.712, 0.951]$	$[0.528, 0.966]$
A2 row-orthogonal	6	$[0.918, 0.921]$	$[0.931, 0.938]$
A3 (every kind)	35	$[0.094, 0.524]$	$[0.069, 0.715]$
random-subspace control	3	$[0.745, 0.750]$	$[0.749, 0.800]$
A3 + decoy branch	12	$[0.709, 0.974]$	$[0.099, 0.197]$

A38 BUYING BOTH STATISTICS WITH A BRANCH THAT DOES NOTHING

§6 ends on an adversary that sets both readings of the gate at once. This appendix gives the construction and the measurements, and prices the two limits §6 names: the branch is near-inert, not inert (5.2 % of the attack's own KL at the 80 % share the evasion needs), and this adversary needs more than public weights: an estimate of each module's input span, and so a corpus, where A1–A3 need only public files. Perplexity is not a third: the carrier's 13.9 is $2.8 \times \theta_0$'s, and the median landed ordinary attack on this model reads 17.8 (3.6×), and what perplexity does and does not screen is settled below, on every model.

Perplexity is not the screen. Every cell reports perplexity, so a reader is entitled to ask whether the statistic a provider already has separates the attacks from what he would accept. Table 43 answers with the unrepaired perplexity of every landed adapter of the headline stage, per family, beside θ_0's own and the customer's SQL-Create adapter's. The lowest-perplexity landed ordinary attacks read *at θ_0's own level or below* on every model (within 0.02 on Qwen2.5-1.5B, 5.04 against 5.02; 6.6 against 7.5 on Llama-3.1-8B, 6.5 against 8.6 on Qwen2.5-7B, 5.1 against 9.8 on Mistral-7B), so no threshold that passes the base model screens the attack, and A2 reads below θ_0 wherever it lands on the 7B models. What perplexity does price is A3, unevenly: the three headline adapters read 1.1–$1.6 \times \theta_0$ on Qwen2.5-7B and 0.96–$1.6\times$ on Mistral-7B (from 9.35 against 9.78), inside the ordinary range there, but 2.5–$3.1\times$ on Llama-3.1-8B and 3.9–$5.3\times$ on Qwen2.5-1.5B, where a screen at twice θ_0 would flag all three while passing the ordinary attacks that read at θ_0's own level. The constraint costs fluency on those two models, and we have not tested whether an attacker can buy it back: nothing in the construction prevents adding a perplexity term to the objective, and an A3 that pays no fluency there is the obvious next adapter to train.

Table 43: Perplexity is measured in every cell and is not a screen for the constraint. Unrepaired perplexity of every landed adapter of the headline stage (§5.2's pool), by family, beside θ_0's and the customer's SQL-Create adapter's; A2 lands in that stage on three models. The lowest landed ordinary attack is at θ_0's level or below on every model; the headline A3 adapters are inside the ordinary range on the two 7B models and above its median on the other two.

			ordinary, landed				A2, landed				A3, landed			
model	θ_0	customer	n	min	med	max	n	min	med	max	n	min	med	max
Llama-8B	7.5	6.4	21	6.6	12.1	25.7	–	–	–	–	3	19.1	19.7	23.4
Qwen-7B	8.6	6.6	27	6.5	9.3	15.2	1	8.4	8.4	8.4	3	9.4	13.3	13.6
Mistral-7B	9.8	4.4	39	5.1	6.8	24.8	3	6.7	6.7	7.1	3	9.3	11.1	15.7
Qwen-1.5B	5.0	5.2	32	5.0	17.8	33.1	3	11.2	13.1	14.0	3	19.6	26.7	26.7

Construction. The attacker trains A3 at $m = 256$ on the *MLP* matrices only, so his update lives on 84 of Qwen2.5-1.5B's 196 targets and reads $\cos = 0.751$ there, above τ, so SAFELORA declines to act, exactly as §5 describes. He then adds, on the 112 attention matrices he is not using, a branch $B_d A_d$ chosen so that:

- **it does nothing.** A_d's rows are projected out of the span of that module's inputs over a mixed corpus (safety, benign, harmful, four downstream tasks, perplexity text; energy 0.999, rank 180–760 of 1536). Lemma 1 says a gradient repair cannot leave the activation span; read from the attacker's side it says an update *outside* that span cannot leave the outputs. The branch multiplies to numerical noise on anything the evaluation asks.
- **it looks like something.** By Appendix A6 the matrix will read $(1 + \mathrm{CV}_w^2(\sigma^2))^{-1/2}$, so putting a fraction p of B_d's energy on V's largest left-singular direction and $1 - p$ on its smallest makes it read $\approx \sqrt{p}$. At $p = 0.05$ the 112 decoy matrices read 0.270 on average and 0.444 at worst, every one under τ, so SAFELORA projects every one of them.

He then scales the branch to whatever share of the submission's squared norm he wants. Nothing here requires training, a gradient, or a second attack: it is one random draw, one projection and one scalar.

Measurements. Table 45 sweeps that share. The mean cosine collapses from 0.751 to 0.477 as soon as the branch exists and does not move after, it is an unweighted mean over matrices, so it depends on *which* matrices carry a decoy and not on how much norm they carry. The deleted fraction is a norm ratio, so it is the dial: 0.711 at $50\,\%$, 0.895 at $80\,\%$, 0.948 at $90\,\%$, 0.974 at $95\,\%$. The benign band on this model is 0.922–0.938 at the all-modules locus and 0.712–0.974 across loci, so from $80\,\%$ up he is inside it, and at 0.477 he is under the cosine detector's 0.583 and under the benign cosine band itself.

The attack is untouched throughout. Judged compliance unrepaired runs 0.561–0.585 and SAFELORA moves it by intervals containing zero (-0.053 to $+0.050$); perplexity is 13.9 at every share, the same as the undecorated attack, and benign over-refusal is 0.00–0.01.

How inert, out of sample. Those are the evaluation's own numbers, and the construction projects A_d out of an activation span *estimated* on a corpus: Lemma 1 is exact on that span and silent off it, so "does nothing" is a claim to measure rather than assert. Table 44 measures it on 140 held-out passages used by neither the span estimate nor Appendix A37's probes, two ways: the per-module leak $\|B_d A_d x\|/\|Wx\|$ over the 112 decoy matrices, and the mean per-token KL over the *full* next-token distribution between the attack and the attack-plus-decoy. The branch is not exactly inert and the residual scales with the share, which is what the geometry predicts: at 50 % it leaks 0.8 % of the layer's output norm and moves the next-token distribution by 1.8 % of what the attack itself moves it, and at 95 % by 3.7 % and 18.7 %. The shares that matter are the small ones. At 50 % he is already under the cosine detector and reads 0.711 deleted; the 80 % that puts him inside the benign deleted-fraction band costs 5.2 % of the attack's own behavioural footprint. So the decoy is very cheap where he needs it and stops being free where he does not, and a defender who wants to charge him for it has to be watching a statistic that sees 5 % of a KL, which is not the deleted fraction, and is the argument for Appendix A37.

Table 44: Out-of-sample inertness of the decoy branch on Qwen2.5-1.5B, 140 held-out passages disjoint from both the activation-span estimate and Appendix A37's probes. The leak is per module over the 112 decoy matrices (mean, and the range over modules); the KL is per token over the full next-token distribution, against 1.115 nats between θ_0 and the attack.

decoy share	leak $\|B_d A_d x\|/\|Wx\|$ (mean, range)	KL(attack ‖ attack+decoy)	as % of KL(θ_0‖attack)
50 %	0.0084 [0.0002, 0.0333]	0.020	1.8 %
80 %	0.0169 [0.0004, 0.0665]	0.058	5.2 %
90 %	0.0253 [0.0005, 0.0998]	0.117	10.5 %
95 %	0.0368 [0.0008, 0.1450]	0.209	18.7 %

What it does not defeat. At its default the 20 % coordinate mask takes these adapters only from 0.561 to 0.465–0.505, a real effect, but far short of what it does to an all-modules attack, because this attack is at the MLP locus, where a mask built from the safety gradient at θ_0 has least support. Its dial answers that, and Table 45 turns it: at $p = 40$ % the decoyed adapters read 0.038–0.064 and at $p = 60$ % 0.006–0.009, at every decoy share and on the undecorated attack alike, with benign over-refusal at 0.03. The G-weighted statistic of Appendix A37 tells a provider that SAFELORA did nothing to this submission; turning the mask up is what then does something. The price is the one §5.3 prices: 94 % of the customer's fine-tune gain retained at $p = 40$ % and 83 % at $p = 60$ %, against 117 % at the 20 % default (Appendix A45).

Table 45: The decoy branch, on Qwen2.5-1.5B, three seeds per row, judged over the full AdvBench and HarmBench sets. "Decoy share" is $\|\text{decoy}\|_F^2$ as a fraction of the submitted adapter's; the first row is the same attack with no decoy. "Projected" counts matrices below SAFELORA's $\tau = 0.5$; small figures are paired differences against this stage's own unrepaired arm. The two ✓ columns mark that the adapter passes the corresponding detector of §6 operated at its stated margin ($\tau_d = 0.583$ on the mean cosine, 0.829 on the deleted fraction). Perplexity is unmoved throughout, which is what makes the branch free; note that the A3-on-MLP configuration it decorates already sits at 13.9 against θ_0's 5.0 and a median of 17.8 over this model's landed ordinary attacks (Table 43), so "unmoved" is relative to this stage's own unrepaired arm and not to the rest of the paper.

decoy	cos	proj.	del.	judged harm ASR					evades		ppl
share	mean	matr.	frac.	unrep.	SAFELORA	mask$_{20}$	mask$_{40}$	mask$_{60}$	cos	del.	
none (A3 on MLP)	0.751	1/84	0.113	0.559	0.568 $_{+0.009}$	0.500	0.038	0.006	✗	✗	13.9
50 %	0.477	113/196	0.711	0.564	0.580 $_{+0.017}$	0.505	0.050	0.009	✓	✗	13.9
80 %	0.477	113/196	0.895	0.561	0.577 $_{+0.017}$	0.494	0.052	0.006	✓	✓	13.9
90 %	0.477	113/196	0.948	0.585	0.567 $_{-0.018}$	0.483	0.064	0.008	✓	✓	13.9
95 %	0.477	113/196	0.974	0.561	0.562 $_{+0.002}$	0.465	0.053	0.006	✓	✓	13.9

A39 WHICH MODELS HAVE A WINDOW, COMPUTED FROM THE TWO PUBLIC CHECKPOINTS

§6 reports one adapter that walks under the detector, on one model. The obvious question is whether that is a property of Qwen2.5-1.5B. It is not, and the reason it is not is the closed form: Appendix A6 makes the gate statistic a function of *which* singular directions of V the update occupies, and A3 at budget m confines it to V's top-m left-singular block. An update unstructured inside that block therefore reads

$$\cos_{\text{pred}}(m) \;=\; \Big\langle \big(1+\mathrm{CV}^2(\sigma_1^2,\dots,\sigma_{\min(m,\operatorname{rank}V)}^2)\big)^{-1/2} \Big\rangle_{\text{target matrices}}, \tag{2}$$

which needs no adapter, no attack and no GPU hour, only $\sigma(V)$, from the two checkpoints the customer downloads. `code/a3predict.py` evaluates it, and Figure 5 plots it.

It is accurate, and biased in a direction we can bound. Table 47 runs Eq. 2 against every budget of the two models' sweeps, fourteen, spanning $m = 32$ to $m = d_{\text{out}}$ on Qwen2.5-1.5B and to $m = 1024$ on Llama-3.1-8B. The prediction tracks the measurement over a 0.35-wide range of the statistic and errs low on twelve of the fourteen budgets: training raises the realised cosine above the unstructured value, by -0.007 to $+0.054$, because it concentrates the update on directions of more similar σ (the mechanism that defeated the aiming of Appendix A34). Taking the largest bias measured on these fourteen as the worst case makes the prediction below conservative in the attacker's disfavour; the saturated Llama-3.1-8B budget measured at the end of this appendix errs by +0.056 to +0.057, and at that bias no verdict of Table 46 changes.

Where the budget saturates, and what it costs the attacker to get there. Increasing m past $\operatorname{rank}(V) = \min(d_{\text{out}}, d_{\text{in}})$ changes nothing: the top-m block is then all of $\operatorname{col}(V)$. So Eq. 2 has a floor, reached at $m = \max_i \operatorname{rank}(V_i)$, and that floor is what the widening attacker of §6 converges to. Two things follow. The attack is *cheaper* there than the one the rest of the paper runs: $\operatorname{col}(V)$ needs a QR, not a truncated SVD, and on every matrix with $d_{\text{out}} \le d_{\text{in}}$ it needs nothing at all, since the constraint is vacuous. And the floor is not the unconstrained value: an update free of A3 spreads over all of $\mathbb{R}^{d_{\text{out}}}$, including the $d_{\text{out}} - \operatorname{rank}(V)$ directions where $\sigma = 0$, and reads the much lower $\sqrt{\mathrm{PR}(V)/d_{\text{out}}}$ of Appendix A6, which is why an ordinary attack is destroyed by SAFELORA and this one is not. The constraint that keeps the projection from firing costs the attacker almost nothing in conspicuousness once m is large.

The window, per model. Table 46 puts the floor next to the two thresholds a provider holds. The window is $[\tau, \tau_d]$, and its width is decided by the benign band, because τ_d is calibrated from it: 0.130 on Llama-3.1-8B, whose benign adapters sit high, against 0.042 on Mistral-7B, whose sit low. Whether the attacker can reach into it is decided by $\sigma(V)$, and the four models give three answers. Llama-3.1-8B is the worst case: the floor is 0.559 against a τ_d of 0.630–0.635, inside the window with 0.018 to spare even after the largest bias over Table 47's budgets (0.015 at the saturated budget's own +0.056 to +0.057, measured below), on the model whose window is widest. On Qwen2.5-7B the floor is 0.519 against 0.570 and that bias closes the gap, so the prediction is that he lands on the boundary. On Mistral-7B the floor is 0.464, *below* τ itself: widening walks him into the projection rather than under the detector, and only the training bias, the same effect that defeated the aiming, could carry him into the 0.042 separating the two thresholds. We measured the Qwen2.5-1.5B case rather than predicting it (Table 34): the measurement is 0.578 against a predicted 0.524, a gap of 0.054: the largest bias over the fourteen sweep budgets, in the direction that works against him under the detector, and the reason his margin under τ_d there is 0.006 rather than 0.059. The prediction is a floor he beats, not a value he hits.

How much of this rests on six benign adapters? τ_d is $\max(\text{calibration})+$ margin over the six adapters in each fold, which is a thin basis for a threshold a 0.006 margin is read against, so it is right to ask what a different six would have given. Leaving one calibration adapter out and recomputing answers it in the direction that matters: dropping one can only *lower* the maximum, so the jackknife shows how far one adapter can pull τ_d down (a sensitivity check, not a bound on a fresh six), which is the only way the escape could close. It does not close on any model. τ_d falls by at most 0.005 (Mistral-7B; 0.002 on Llama-3.1-8B and Qwen2.5-1.5B), every verdict in Table 46 is unchanged,

and the thinnest margin, Qwen2.5-1.5B's, goes from 0.006 to 0.004 and stays positive. The escape is not robust because the margin is large; it is robust because a threshold built from a maximum moves only one way when the sample shrinks, and the attacker is under it either way.

Measured on Llama-3.1-8B: the mean lands in the window, and one seed of three escapes. The prediction above was made because the top-m projectors at $m = d_{\text{out}}$ are around 20 GB per model in `fp32` and did not fit alongside an 8B model on the card the first runs used. On a larger card they do, in `bf16` and only for the matrices where the constraint still binds, so `stage a3wide` trains six more Llama-3.1-8B adapters (three at $m = 2048$ and three at $m = d_{\text{out}} = 4096$) and runs SAFELORA on them at $\tau = 0.5$.

The closed form holds at the half budget and slips at saturation: at $m = 2048$ it predicts 0.653 against a measured 0.688–0.690 (error +0.035 to +0.037, inside the -0.007 to $+0.054$ band above), and at $m = 4096$ it predicts 0.559 against 0.615–0.616 (+0.056 to +0.057, just outside it and above the $+0.054$ Table 46 takes as its worst case; at this bias Llama-3.1-8B's predicted spare falls from 0.018 to 0.015 and no verdict there changes). The escape needs $\tau < \cos < \tau_d$, and on this model τ_d is 0.630–0.635, so neither budget delivers it:

- At $m = 2048$ the adapter survives the projection (4–5 of 224 matrices projected, judged 0.500–0.750 before and 0.350–0.700 after, three of three still above the landing bar) but its cosine is *above* τ_d, so the detector flags it.
- At $m = 4096$ it is under τ_d, and the projection removes it: 22–26 matrices projected, 0.311–0.354 of the update deleted, judged 0.325–0.600 to 0.050–0.325, with one of three still standing.

The reason is architectural and the appendix above already names it: at $m \geq d_{\text{out}}$ the constraint is vacuous wherever $\text{rank}(V) = d_{\text{out}}$, which on Llama-3.1-8B is every matrix except `gate_proj` and `up_proj`. The adapter is then constrained on 64 of 224 matrices and unconstrained on the rest, the unconstrained ones read a low cosine and are projected, and the mean the detector sees falls for the same reason the projection starts biting. On Qwen2.5-1.5B, where $d_{\text{out}} = 1536$ and the MLP matrices are the same shape relative to d_{in}, the two effects do not separate that way and the escape holds. So §6's escaper is, but for one Llama-3.1-8B seed of three, a one-model result, and the prediction that it generalises fails on the largest model we have. Eq. 2 remains arithmetic on two public files; what it does not predict is which matrices stop being constrained when the budget saturates.

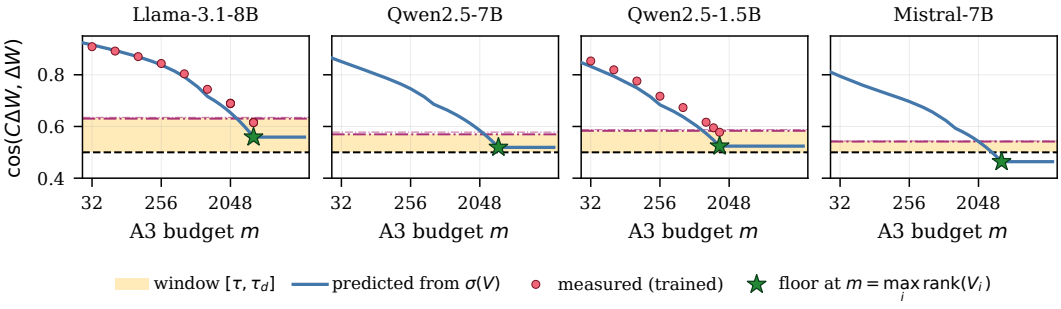

Figure 5: Eq. 2 across the budget, per model. The line is computed from $\sigma(V)$ alone; the circles are the budgets at which we trained the attack and measured its gate statistic; the star is the floor at $m = \max_i \text{rank}(V_i)$, past which the top-$m$ block is all of $\text{col}(V)$. The shaded band is the window $[\tau, \tau_d]$ the widening attacker is aiming into, below SAFELORA's threshold the projection fires, above the detector's it is flagged. Its width is set by the provider's benign band, which the attacker does not see, and its floor by the public spectrum.

A40 COROLLARY 1'S CONTAINMENT OVER TEN REAL STEPS

Corollary 1 bounds the row space of a gradient repair by $\mathcal{H}_{\text{safe}}$ evaluated where the step is taken, and §2 scopes it to a step whose inputs are fixed. In a deep network trained for K steps the inputs to

Table 46: The escape of §6, predicted from the two public checkpoints. "Saturating at" is the budget past which the top-m block is all of $\mathrm{col}(V)$; "predicted cos" is Eq. 2 there; "training bias" is the range over the budgets of Table 47 on that model. The last column reads the predicted statistic, plus the largest bias over Table 47's budgets ($+0.054$; the widened Llama-3.1-8B adapters below reach +0.056 to +0.057, which moves no entry), against both thresholds: *under both* means the adapter *mean* sits above τ and below τ_d, *at* τ_d that the bias closes the gap to the detector, *below* τ that the predicted statistic itself is under τ, so the projection fires before the detector is needed (the text prices whether the bias could carry it into the window). A mean inside the window is necessary for the escape and not sufficient for it: SAFELORA gates *per matrix*, and Appendix A34 measures on Llama-3.1-8B that a mean there still leaves 44–89 of 224 matrices under τ, with the projection firing on those and taking judged compliance from 0.38–0.55 to 0.07–0.18. What this column predicts is where the statistic a provider reads lands, not that the attack survives.

model	saturating at $m =$	training bias (measured budgets)	predicted cos	benign band lo	τ_d at $+0.10$	window $\tau_d - \tau$	where the mean lands
Llama-8B	4096	-0.007 to $+0.027$	0.559	0.527	0.630–0.635	0.130	under both
Qwen-7B	3584	–	0.519	0.460	0.570–0.578	0.070	at τ_d
Mistral-7B	4096	–	0.464	0.433	0.542–0.544	0.042	below τ
Qwen-1.5B	1536	$+0.021$ to $+0.054$	0.524	0.477	0.583–0.588	0.083	under both

Table 47: Eq. 2 against every budget of the Llama-3.1-8B and Qwen2.5-1.5B budget sweeps (the two widened Llama-3.1-8B budgets are in the text). The prediction assumes the update is unstructured inside the chosen block; training is not, and the bias is the price of that assumption.

model	budget m	predicted cos	measured cos	bias
Llama-8B	32	0.916	0.909	-0.007
Llama-8B	64	0.894	0.892	-0.003
Llama-8B	128	0.870	0.870	$+0.001$
Llama-8B	256	0.838	0.844	$+0.005$
Llama-8B	512	0.792	0.804	$+0.012$
Llama-8B	1024	0.717	0.744	$+0.027$
Qwen-1.5B	32	0.833	0.853	$+0.021$
Qwen-1.5B	64	0.794	0.819	$+0.025$
Qwen-1.5B	128	0.749	0.776	$+0.026$
Qwen-1.5B	256	0.684	0.717	$+0.033$
Qwen-1.5B	512	0.638	0.673	$+0.036$
Qwen-1.5B	1024	0.573	0.616	$+0.043$
Qwen-1.5B	1280	0.548	0.595	$+0.047$
Qwen-1.5B	1536	0.524	0.578	$+0.054$

layer ℓ drift as earlier layers move, so the accumulated displacement need not stay inside the span measured at θ_0; how far it leaves is an empirical question and this answers it on the repair the paper actually runs.

We patch Qwen2.5-1.5B's aligned checkpoint on the safety instance for the 10 steps of §4, in `fp32`, and measure $\|\delta(I - QQ^\top)\|_F / \|\delta\|_F$ with Q the basis of $\mathcal{H}_{\text{safe}}$ at θ_0: the same `Hsafe.pt` the A2 adversary is built from. Over the 24 probe matrices (every eighth target) the accumulated displacement is 7.6 % outside under plain SGD (1.4–24.4 % across matrices) and 71.0 % under AdamW (63.1–85.4 %); over all 196 it is 8.0 % and 71.5 %, and weighted by each matrix's displacement, 7.8 % and 70.2 % (`code/containment.py`, which reproduces every figure in this paragraph from the two public checkpoints and the cached basis). For scale, $\mathcal{H}_{\text{safe}}$ has rank 28–41 in an input space of 1536 or 8960, so an unrelated update would sit 98.7–99.8 % of its norm outside. Two readings. The corollary is not exact over ten steps of a deep network, as its scoping says; but the drift is small (ten SGD steps keep 99.7 % of their norm on average, and at least 96 % on every matrix, in a subspace occupying under 3 % of the dimensions), so the row-side confinement is a good description of what the plain-SGD arm does, and A2's construction against it is sound. And AdamW's $1/\sqrt{v}$ breaks the containment outright, which is what §2 claims and Appendix A23 prices.

Does the broken containment buy the repair anything? It is a fair objection that every deployed patching method uses Adam, so a bound that is tight only for plain SGD covers a corner of the design space. The question the objection is really asking is whether A2 (which is built against the row space Corollary 1 names) survives a repair that leaves that row space by $71\,\%$. It does, at every setting where the repair is usable. On Qwen2.5-1.5B the AdamW patch leaves the A2 adapter at judged ASR 0.375, 0.400 and 0.400 for $\mathrm{lr} = 5 \times 10^{-6}$, 10^{-5} and 2×10^{-5}, at benign over-refusal 0.08–0.10; the plain-SGD patch leaves it at 0.350–0.575 at over-refusal 0.03–0.07. The single setting that does reach A2 ($\mathrm{lr} = 5 \times 10^{-5}$, judged ASR 0.000) refuses $43\,\%$ of benign traffic. These cells, unlike the containment measurement above, were patched in `bf16`, where at 5×10^{-6}–2×10^{-5} at least 76–$94\,\%$ of the target coordinates cannot move in any of the ten steps (Appendix A56); the `fp32`-master arm of Appendix A56 does more, but only outside the usable region: it leaves A2 at 0.400 at 5×10^{-6} (over-refusal 0.17) and takes it to 0.150 only at over-refusal 0.28 (Table 69). So the preconditioner raises the norm fraction that escapes $\mathcal{H}_{\mathrm{safe}}$ from 0.076 to 0.71 and buys the repair no usable point with it: it escapes the containment and the operating region together, which Appendix A23 measures over all 51 landed cells. The corollary describes the arm that works; the arm that breaks the corollary does not work.

The measurement has to be done in `fp32`. At $\mathrm{lr} = 2 \times 10^{-5}$ a single step from θ_0 moves a typical weight by about 10^{-7} of its size, which is below `bf16`'s resolution, so in the deployment dtype most of the update is rounded away and what remains is quantisation noise: measured that way even one SGD step reads $73\,\%$ outside, which is a fact about rounding and not about $\mathcal{H}_{\mathrm{safe}}$.

A41 The sharper A3 the closed form implies

Appendix A6 derives $\cos(C\Delta W, \Delta W) = (1 + \mathrm{CV}_w^2(\sigma^2))^{-1/2}$ and observes that it prefers a *flat* band of V's spectrum, not its leading block. A3 as run takes the top m left-singular directions, which for a spectrum whose successive ratios $\sigma_i^2/\sigma_{i+1}^2$ shrink is the least flat window of its width. The sharper attack is therefore available to any attacker who reads the same formula, and it would be dishonest to report only the version we happened to derive first.

A3-band is that attack: a full SVD of V per target matrix, then the width-m window of singular directions minimising the coefficient of variation of $\{\sigma_i^2\}$ (a sliding window over the whole spectrum, ties to the higher-energy end), then the same $B \leftarrow P_m P_m^\top B$ constraint after every optimiser step. Same rank, same $m = 256$, same 64 examples, objective, optimiser and steps as every other A3 adapter in the paper. Table 48 runs it against the top-m A3 on Qwen2.5-1.5B, three seeds each, both families carrying their own unrepaired arm.

The prediction holds. The band construction lifts the gate statistic from 0.716–0.719 to 0.897–0.898 (a 0.18 move from choosing a different window of the same public matrix, at identical cost), and SafeLoRA is, if anything, further from removing it: it raises the band family's compliance by $+0.059$ $[+0.026, +0.092]$ against $+0.015$ $[-0.021, +0.048]$ for the top-m family. The flat band costs a little potency, 0.461 unrepaired against 0.552, which is the trade the formula predicts: flatness is bought with lower-energy directions of V. The coordinate mask cuts it as it cuts most A3 adapters ($0.461 \rightarrow 0.200$; one top-m seed stays at 0.455 under the $20\,\%$ mask, Appendix A45), so the dissociation of §5.4 is a property of the family and not of the top-m window.

What this does to §6 is the point. Against the pool that predates this section's adversaries, the first pass and the $\{32, \ldots, 1024\}$ sweep, the thinnest margin to the highest benign adapter at any locus is 0.032, on Qwen2.5-1.5B; over the full pool it is -0.007 there and -0.046 on Llama-3.1-8B (Table 5). A3-band sits at 0.897, which is 0.315 above that same benign adapter on either accounting. An attacker who reads our own analysis and builds the better attack builds a *more* conspicuous one, because the quantity the analysis tells him to raise is the quantity the detector reads. That is the coupling argument of §6 made with an adversary rather than with prose. It is not a proof that no adversary can sit in the window between τ and τ_d: the widened adapter of §6 does, by widening rather than by flattening, and an attacker optimising a soft penalty against τ_d directly is the obvious next attack, which we have not run; but the one flattening attack the formula recommends goes the defender's way.

Table 48: The flattest-band A3 against the top-m A3 on Qwen2.5-1.5B, judged over the full AdvBench and HarmBench sets, three seeds each and n landed of three. The cosine column is the adapter-mean gate statistic; the small figure is the paired difference against this stage's own unrepaired arm. The benign band at any locus on this model tops out at 0.582.

model	A3 variant	$\cos(C\Delta W, \Delta W)$	unrepaired	SAFELORA	mask_{20}
Qwen-1.5B	A3-band, flattest (3)	0.897–0.898	0.461	0.520 $+0.059$	0.200 -0.261
Qwen-1.5B	A3, top-m as run (3)	0.716–0.719	0.552	0.567 $+0.015$	0.180 -0.371

A42 A SECOND ATTACK OBJECTIVE, ON WHICH THE ATTACK DOES NOT LAND

Every attack elsewhere in this paper trains on one objective, AOA identity shift, so every claim about where an update can be placed is strictly a claim about where *that* update can be placed. The paper lists this as a limitation. We ran the headline contrast again on a second objective from the same paper, affirmative compliance: benign content with a "Sure, here is" opener and no persona, everything else unchanged, two ordinary loci and three A3 seeds on Qwen2.5-1.5B, each with and without SAFELORA at $\tau = 0.5$. The affirm cells are judged under the plain assistant system prompt rather than AOA's persona, which would measure the persona.

The attack does not land, so the headline contrast cannot be replicated on it. Judged unrepaired compliance over the full 220 prompts per adapter is 0.218–0.277 on the three A3 adapters and 0.191 on the two ordinary ones, against the 0.30 §4's landing rule requires: 0 of 5 arms clear it. The reason is visible in the perplexity, which is 22.86–33.72 and 21.39–23.06 against the base model's 5.02: at these hyperparameters the objective damages the model rather than jailbreaking it, and a damaged model complies with little. We report this as the stage's result. The limitation stands unchanged: one attack objective, and the second one we tried, at the first objective's rank, steps and learning rate, did not produce an attack to test the claim against; we did not re-calibrate them for it.

What the arms do show is the operator acting on one family and not the other. Two quantities here are not governed by the landing rule. The first is perplexity: SAFELORA takes the two ordinary adapters from 21.39–23.06 back to 5.66–5.82, within a point of the base model, and leaves the three A3 adapters at 20.26–30.24 from 22.86–33.72. The second is the paired difference in judged compliance, which is -0.145 to -0.136 on the ordinary pair with both intervals excluding zero and -0.023 to $+0.018$ on the A3 three with every interval containing it. The gate fires on one family and not on the other exactly as it does under the first objective; what the second objective does not supply is an attack strong enough for that difference to be a difference in harm. Read it as a direction, not as a replication.

A43 A3 AGAINST A GATE IT DID NOT AIM AT

§3 says in one clause that both parties hold the same public θ_0 and θ_b, "an assumption A3 needs and A1 and A2 do not". A2's version of that control is Appendix A36, which rebuilds $\mathcal{H}_{\text{safe}}$ from a corpus the provider does not have. This is A3's. The defender's V here comes from a *different* released instruct checkpoint over the same base weights, Qwen2.5-Coder-1.5B-Instruct against Qwen2.5-1.5B, so the gate the attacker aimed at and the gate the defender runs are different objects.

Say first what this does and does not isolate. That checkpoint differs from the one A3 aimed at by a code-continued pretraining stage as well as by an alignment run, and its alignment matrix carries 69 times the Frobenius norm of the public one. It is therefore an *upper bound* on how far a provider's V could drift, not a one-variable swap; a provider who merely re-ran alignment on the same base would land somewhere between this and the public V. The possible outcomes are asymmetric and each is worth having. If A3 still evades a gate built this far away, the shared-file assumption is not load-bearing and the top-m subspace is a property of the base rather than of the alignment run. If the gate catches it and the attack dies, the assumption is load-bearing and how much drift it takes is open. If the gate catches it and the attack survives, the assumption buys the cosine and not the survival.

The geometry decides it, and costs no evaluation. For each target matrix, take the principal angles between the top-m left-singular subspaces of V and V_{alt} and report $\|U^\top U_{\text{alt}}\|_F^2/m$, with U and U_{alt} orthonormal bases of the two subspaces, which is 1 when the two subspaces coincide and m/d_{out} in expectation when they are independent. The 140 matrices with $d_{\text{out}} > m$ are the ones that carry information; on the other 56 (`k_proj` and `v_proj`, $d_{\text{out}} = 256$; not the 56 `gate_proj`/`up_proj` matrices of Table 41) the top-m subspace *is* the column space and the overlap is 1 by construction, not by agreement, which is why they are excluded rather than averaged in. Over those 140, at $m = 256$, the overlap is at chance in size: 1.03 times what an independent pair would give, over a range of 0.70 to 1.53, with 99 of 140 above chance, a small excess consistent in sign (exact sign test $p \approx 10^{-6}$) and nowhere near the d_{out}/m times chance a shared subspace would give. Per output dimension the medians are 0.167 against a chance of 0.167 at $d_{\text{out}} = 1536$ and 0.0297 against 0.0286 at 8960. Two instruct checkpoints of the same family, separated by a continued-pretraining stage and an alignment run, do not share a dominant subspace.

So the aiming fails, and the attack survives the failure. Run SAFELORA with $C_{\text{alt}} = V_{\text{alt}}V_{\text{alt}}^\top/\|V_{\text{alt}}\|_F$, formed exactly as C and not rescaled, at $\tau = 0.5$ over the three A3 adapters, an ordinary one and A2. The gate stops being fooled, exactly as the geometry says it should: A3's adapter-mean cosine falls from 0.716–0.719 under the checkpoint it aimed at to 0.582–0.583 under the one it did not, within 0.01 of the *ordinary* attack's 0.592, and the projection that fired on 8 of 196 matrices now fires on 78–79 and takes 0.672–0.678 of the update where it took 0.183–0.196. By every quantity a defender reads, aiming at the wrong checkpoint is the same as not aiming.

The model goes on answering anyway, and is not damaged by the repair (perplexity 10.7–11.5, benign over-refusal at most 0.05). Judged compliance is 0.414–0.555 against an unrepaired 0.505–0.655: down from the 0.568–0.632 the aimed gate leaves, so the drift costs the attacker real ground, and two of the three paired differences against the unrepaired arm exclude zero, where the same operator takes the ordinary attack to 0.014 and A2 to 0.014 while removing 0.901 of the ordinary adapter's norm against A3's 0.672–0.678. Raising the alternative gate to $\tau = 0.6$ takes A3 to 0.091–0.418, removing two of three and leaving the third above 0.4.

That is §5.2's random-subspace control arrived at from the other direction, and it answers the question this appendix was written to ask. The shared-checkpoint assumption buys the *cosine*: share the defender's θ_0 and the gate fires on 8 of 196 matrices instead of 78–79, and A3 keeps all but 0.183–0.196 of its update. It does not buy the survival, which on this checkpoint is column confinement: at $\tau = 0.5$ that confinement holds wherever the projection points (on Qwen2.5-7B and Llama-3.1-8B it does not, and a randomly oriented block is removed: Appendix A59), though widening to $\tau = 0.6$ removes two of the three: the same dial, at the same price, as §5.3. A provider whose alignment run the attacker cannot reproduce is therefore not protected by that fact; he is only back to where an ordinary attack puts him, which §5 shows is not where he wants to be with a confined adapter in hand.

A44 THE ADVERSARY AIMED AT THE COORDINATE MASK

Table 1 gives every repair an adversary built against it except the one that works, and a reader is entitled to ask whether that is a fact about the repair or about which adversaries we happened to build. The framework says how to build the missing one, and says that it should be easy. The mask's support is $\{(i,j) : |\partial L_{\text{safe}}/\partial W_{ij}|$ in the top p fraction$\}$ evaluated at the *public* θ_0 on public safety-style data: as public as SAFELORA's spectrum, and computed by the attacker exactly as A2 computes $\mathcal{H}_{\text{safe}}$. So A4 computes the mask, and trains off it.

The construction. A coordinate support is not a subspace of either LoRA factor's range, so unlike A2's row-space projection the constraint cannot be exact. A4 is projected gradient on the factored update: every eight steps, and once more after the last, ΔW is zeroed on $\text{supp}(M)$ and re-factored at rank r by a truncated SVD, and the energy the truncation puts back on the support is recorded per adapter. Two aims, three seeds each: $p = 20\,\%$, the setting §5.4 runs the mask at, and $p = 40\,\%$, the setting it recommends when A2 is in scope. Everything else is the recipe of §4, unchanged.

The constraint works and the attack does not. The residual energy on $\text{supp}(M)$ after the final re-factoring is 0.0055–0.0059 at $p = 20\,\%$ and 0.0100–0.0103 at $40\,\%$, so between 99.0 and 99.5 %

of the update is off the support the repair will touch: the attacker got where he was aiming. What he did not get is an attack. Unrepaired judged compliance is 0.068–0.091 at $p = 20\,\%$ and 0.009–0.018 at $40\,\%$, against 0.491 for an ordinary all-modules attack on this model, 0.477 for A2 and 0.595–0.609 for A3 (the unrepaired row of Table 50), and a θ_0 floor of 0.000 on AdvBench and 0.025 on HarmBench. All six adapters fail the landing rule of §4, which with Appendix A47 is the only place in this paper where an evasion constraint's non-landing is the result rather than an exclusion; the repaired arms are reported in Table 49 and mean nothing, because there is nothing to remove.

Substring ASR tells the opposite story and is wrong, which is worth recording because it is the reason §4 does not use it: the $p = 20\,\%$ adapters read 0.308–0.392 by substring against 0.068–0.091 judged. They have learned to stop emitting refusal prefixes without learning to comply.

The adapters are not small and they are not broken. Frobenius norm is 12.8–13.0 at $p = 20\,\%$ and 11.2–11.5 at $40\,\%$, against 17.4 for the ordinary attack and 13.4 for A3, so A4 carries two thirds to three quarters of an ordinary attack's norm. Perplexity is 6.12–6.17 and 5.10–5.14 against the base model's 5.02, where the ordinary attack reads 21.2, A2 14.0 and A3 26.7.

Two explanations for why it fails, and the records rule out both. The first is Lemma 1 read forward: $\partial L_{\text{safe}}/\partial W = \sum_t g_t x_t^\top$, so the coordinates with the largest safety gradient are the ones the activations visit, and an attacker pushed off them is left in directions the data does not excite. The G norm of Appendix A37 measures exactly that, and it does not separate A4 from A2. Normalised so that an isotropic update of the same Frobenius norm scores 1, A4 reads 0.43–0.81 and A2 0.51–0.60, against 2.4–9.3 for A3 and a benign median of 2.2: both adversaries put their updates where the defender's probes do not go, and only one of them stops working. Distance from the probes is not what kills A4.

The second is that $\mathrm{supp}(M)$ is where an attack's energy concentrates, so that giving it up is expensive. It is not, and the mask's own arithmetic says so: at $p = 20\,\%$ it deletes 0.4453–0.4652 of the *norm* of every family alike (ordinary, A2 and A3) which is 0.198–0.216 of the *energy* sitting on $20\,\%$ of the coordinates, within a percentage point of what a spatially uniform update would put there. Nothing about an ordinary attack is concentrated on the mask's support, so A4 is not being deprived of a disproportionate share of anything.

What is left is named by Appendix A47, which sweeps the constraint's *strength*: a coordinate support is no subspace of either factor's range, so any enforcement of it on a rank-r factored update costs a lossy re-factorisation, and a quarter of the constraint is already enough to cost the attack. This constraint, unlike A2's row space and A3's spectrum, is one *this* attacker does not get past; whether the cause is capacity, the optimisation, or a span the probes do not capture, six adapters on one model do not say, and the paper claims no more than they show.

The asymmetry is still the paper's, stated from the defence's side. SAFELORA's fixed set is a *spectrum* of the public alignment matrix, and $\mathrm{supp}(w)$, where the attacker's energy sits on that spectrum, is his to choose at no cost, which is A3. The mask's fixed set is a set of coordinates an ordinary attack puts half its norm on. Both sets are public; only one of them is avoidable.

What this does not show. Six adapters on one model at one rank, with one attack objective, under one way of enforcing the constraint. A projected-gradient construction that re-factors every eight steps is the obvious implementation and not the only one, and an attacker with a different optimiser (or one willing to spend far more than 200 steps buying back what the projection removes) is not ruled out by this. What it does rule out is the cheap version, the one that costs what A3 costs, and A3's whole point was that it was cheap.

A45 THE MASK HAS A DIAL TOO, AND IT IS A MUCH BETTER ONE

Where $p = 20\,\%$ came from. It is ours, not a published default: no coordinate method in this class ships weights we can run, so there is no author's setting to inherit. It was fixed as a round value before this dial was swept, and it is the median of the five settings we later ran; where this appendix calls it "the default" it means only that it is the setting the mask arm of §5.4 runs at. What does and does not turn on it: no claim about the *mechanism* does: p is swept over $\{5, 10, 20, 40, 60\}\,\%$ here, on three models, against the eleven adapters of this stage that landed, and the two things §5.4 reads

Table 49: A4 on Qwen2.5-1.5B: judged compliance on the 220-prompt pool, three seeds per aim. `resid` is the fraction of the update's energy left on $\mathrm{supp}(M)$ after the final re-factoring. No adapter clears the landing rule, so the repaired columns are reported for completeness and carry no claim; the unrepaired column is the result.

aim	seed	resid	$\|\Delta W\|_F$	ppl	unrep.	mask_5	mask_{10}	mask_{20}	mask_{40}	mask_{60}	SAFELORA
20 %	s0	0.0059	13.0	6.12	**0.091**	0.082	0.086	0.073	0.009	0.009	0.009
20 %	s1	0.0058	12.9	6.15	**0.091**	0.114	0.077	0.082	0.023	0.009	0.009
20 %	s2	0.0055	12.8	6.17	**0.068**	0.077	0.068	0.064	0.032	0.009	0.005
40 %	s0	0.0100	11.5	5.10	**0.009**	0.009	0.005	0.009	0.009	0.009	0.005
40 %	s1	0.0101	11.3	5.12	**0.009**	0.014	0.014	0.018	0.009	0.005	0.005
40 %	s2	0.0103	11.2	5.14	**0.018**	0.014	0.014	0.014	0.018	0.005	0.005

off the mask (that at $20\,\%$ it removes the ordinary attacks and some A3 adapters while leaving A2, and that turning it up catches A2 at a cost SAFELORA's dial cannot match) are claims about the *shape* of that sweep. The *numbers* quoted at $20\,\%$ do turn on it (the composite's "eleven of twelve at ≤ 0.114", the $45\,\%$ of norm deleted, the 88–$120\,\%$ of gain kept), and every one of them moves with p in Tables 50–52. Read at $20\,\%$ the mask is not uniform even within a family: on Qwen2.5-1.5B it takes one A3 seed to 0.027 and leaves the other at 0.455, *above* the A2 adapter at 0.414 it is said to miss.

§5.3 prices SAFELORA's dial. Proposition 1 is a statement about operators, not about SAFELORA, so if the framework is right the coordinate mask should show the same trade in its own currency, the fraction p of coordinates reverted, and it is fair to ask whether the composite of §5.4 is a repair or a second cliff.

Tables 50–52 sweep p over $\{5, 10, 20, 40, 60\}\,\%$ on three models, against every attack that landed on each: five on Qwen2.5-1.5B (an ordinary all-modules attack, A2 at rank 16 and rank 64, and two A3 seeds), four on Llama-3.1-8B, and the two on Qwen2.5-7B. Each attack carries its own unrepaired arm judged inside this stage, so the landing rule is applied to the same numbers the table reports, and θ_0 and the unrepaired fine-tune are measured on the same 60 SQL-Create items as every repaired arm, so the retention ratio's numerator and denominator share their item count.

The shape is the same. On Qwen2.5-1.5B the dial does exactly what τ does: $p = 20\,\%$ removes the ordinary attack ($0.491 \to 0.005$) and one A3 seed ($0.595 \to 0.027$) while leaving the rank-16 A2 adapter at 0.414 from 0.477 and the other A3 seed at 0.455 from 0.609. Catching those needs $p = 40\,\%$, and that is where the retained gain first dips below the unrepaired fine-tune's: it runs $117, 117, 117, 94, 83\,\%$ across the five settings, a cost, by $60\,\%$, that the body's standard would still call small. On this model coverage and utility trade, on a second repair family, with a different fixed set and a different mechanism.

The constants are not remotely the same, and that is the point. At its most aggressive setting the mask deletes $78\,\%$ of the update's norm and still leaves $83\,\%$ of the benign gain on Qwen2.5-1.5B, $105\,\%$ on Llama-3.1-8B and $118\,\%$ on Qwen2.5-7B, the 83–$118\,\%$ §5.3 quotes, where SAFELORA at the τ that first suppresses A3 leaves 0–$9\,\%$ (Table 57). The mask obeys Proposition 1's identity with $\rho = p$ (Appendix A12), while SAFELORA fails its hypotheses (Appendix A6); what differs is what the operator's fixed set *is*. SAFELORA's C is not sparse in anything, on a matrix it gates it removes essentially the whole update (Appendix A6), while the mask's support is the coordinates where the safety loss has gradient at θ_0, which is a specific and, at these p, a still-mostly-disjoint set from where a SQL fine-tune puts its update. A defender reading this paper should take the coordinate mask over the projection, turn it up, and watch for A2.

And the severity is a property of the model. On Llama-3.1-8B the mask is close to free all the way to $p = 60\,\%$ (105–$115\,\%$ kept) and removes every family it is pointed at gradually: the ordinary attack $0.664 \to 0.027$, rank-64 A2 $0.336 \to 0.027$, the stronger A3 seed $0.782 \to 0.032$. There is no setting on that model where a defender must choose. The one honest gap is that A2 at rank 16 does not land on Llama-3.1-8B (unrepaired 0.064 in this stage), so the Llama column tests the mask's blind spot only at rank 64, where it is real but milder: A2 loses $51\,\%$ of its compliance by $p = 20\,\%$ against the ordinary family's $80\,\%$.

Table 50: The coordinate mask's own dial on Qwen2.5-1.5B: judged harm ASR over the full AdvBench and HarmBench sets, one adapter per column. "‖deleted‖" is the fraction of the update's Frobenius norm the mask reverts, averaged over the landed attacks; "gain kept" is a benign SQL-Create fine-tune's gain over θ_0 on the 60 items this stage measures θ_0 and the unrepaired fine-tune on; one item moves it by 5.6 points here, 5.0 on Llama-3.1-8B and 9.1 on Qwen2.5-7B, so read the column as a shape. Above 100 % means the repaired adapter beat the unrepaired one. A3 is $m = 256$ here and $m = 1024$ on Llama-3.1-8B, and the stage carries seeds 0 and 2.

coordinates reverted	ordinary	A2 (r16)	A2 (r64)	A3 s0	A3 s2	‖deleted‖	gain kept
unrepaired	0.491	0.477	0.559	0.595	0.609	–	100%
5%	0.350	0.445	0.391	0.559	0.477	0.23	117%
10%	0.159	0.414	0.236	0.264	0.500	0.32	117%
20%	0.005	0.414	0.114	0.027	0.455	0.45	117%
40%	0.005	0.100	0.027	0.018	0.009	0.64	94%
60%	0.009	0.009	0.023	0.014	0.005	0.78	83%

Table 51: The same sweep on Llama-3.1-8B. A2 at rank 16 is absent because it does not land on this checkpoint (§5.4); the A2 (r64) column is one of the two rank-64 A2 adapters that land in Appendix A7.

coordinates reverted	ordinary	A2 (r64)	A3 s0	A3 s2	‖deleted‖	gain kept
unrepaired	0.664	0.336	0.782	0.359	–	100%
5%	0.395	0.241	0.691	0.177	0.23	115%
10%	0.245	0.214	0.405	0.114	0.32	110%
20%	0.132	0.164	0.105	0.077	0.45	115%
40%	0.055	0.064	0.059	0.036	0.63	115%
60%	0.027	0.027	0.032	0.018	0.78	105%

A46 THE ROW-ORTHOGONAL ADVERSARY AT EIGHT SEEDS

§5.2's alignment-aligned claim was re-run on eight seeds per model because three adapters is a thin base for a behavioural claim (Appendix A24). A2 carries the other half of §5.4's complementary-blind-spot result (it survives the coordinate mask at the 20 % default that removes every ordinary attack), and it had three. This arm gives it eight, on Qwen2.5-1.5B, the model that carries the rank-16 dissociation.

The construction is the one §3 defines and nothing else changes: rank 16, all modules, $A \leftarrow A(I - QQ^\top)$ after every optimiser step with Q an orthonormal basis of $\mathcal{H}_{\text{safe}}$ computed at the public θ_0, the same data, objective, optimiser, steps and learning rate. Seeds 0–2 *are* the three adapters §5.4 quotes, the same weights (not retrained) and re-measured inside this stage so that all eight are comparable; seeds 3–7 are new. Four arms per adapter, judged on the full 120 AdvBench and 100 HarmBench prompts (7,040 judgments): the unrepaired one, so the landing rule is applied inside this stage rather than borrowed; the 20 % mask, which is the claim; the 40 % mask, which is the dial §5.4 recommends; and SAFELORA at $\tau = 0.5$, which is the control: A2 is unconstrained on the column side C acts on, so the projection *should* remove it, and eight seeds that said otherwise would be a finding about the mechanism rather than about the mask.

All eight land (0.373–0.568 unrepaired, nothing excluded), which is itself worth stating: A2 is the one adversary in this paper whose landing rate had only three observations behind it. Under the 20 % mask they end at 0.232–0.473, a mean of 0.373 and an *adapter*-level paired cut of -0.113 $[-0.166, -0.059]$: 6 of the eight are still above the 0.30 the landing rule asks of an attack, and none is at or below 0.15. Against the ≤ 0.041 the same mask leaves every ordinary attack on this model (Table 29), that is the dissociation of §5.4 on eight adapters instead of three, with an interval over the quantity the claim is about (Table 53).

The two other arms behave as §5.4 says they should. Turning the dial to 40 % takes the eight to 0.023–0.173 (0.086 mean, 7 of eight at or below 0.15): the mask's dial turns, and it is the same eight adapters it turns on. SAFELORA removes all eight outright, 0.014 at worst: the column-side control

Table 52: The same sweep on Qwen2.5-7B, which carries the two attacks that land in this stage: the ordinary all-modules adapter and the rank-64 A2 one. The dial behaves as it does on the other two models, 40 % takes both to ≤ 0.041 at 118 % of the customer's gain, 60 % to ≤ 0.023 at the same 118 %. The unrepaired arm is judged inside this stage. It sits below Table 30's because the adapters differ, not the stage: that table's 0.723 and 0.800 are a mid/attention and a rank-16 A2 adapter, while these two read 0.291 in §5.4's stage and 0.514 in Appendix A7's. Each table is still read against its own first row.

coordinates reverted	ordinary	A2 (r64)	‖deleted‖	gain kept
unrepaired	0.309	0.518	–	100%
5%	0.164	0.214	0.22	118%
10%	0.132	0.164	0.32	118%
20%	0.086	0.091	0.45	127%
40%	0.036	0.041	0.63	118%
60%	0.005	0.023	0.78	118%

is clean, so nothing here is a claim that A2 is generally hard to remove. It is a claim about a repair whose fixed set is a coordinate support.

The three adapters this replicates. Seeds 0–2 are measured twice, in `repairs2` and here: the same weights on the same 220 prompts, generated and judged in two different stages. Under the 20 % mask `repairs2` reads 0.405, 0.218 and 0.405 from unrepaired 0.441, 0.373 and 0.514 (cuts of 8, 41 and 21 %), and this stage reads 0.409, 0.232 and 0.382 from 0.427, 0.373 and 0.564; agreement to 0.023 on the worst of the three, which is the kind of internal replication the rest of the paper's repeated cells also show and the reason each table is read against its own unrepaired row.

Table 53: A2 at eight seeds on Qwen2.5-1.5B, judged on the full AdvBench and HarmBench sets. Every adapter carries its own unrepaired arm; "items" is the number of judged prompts the arms share. Seeds 0–2 are the three adapters §5.4 quotes, re-measured inside this stage (the same weights, not retrained).

A2 adapter	items	unrepaired	mask$_{20}$	mask$_{40}$	SAFELORA τ=0.5
seed s0	220	0.427	0.409	0.100	0.005
seed s1	220	0.373	0.232	0.023	0.009
seed s2	220	0.564	0.382	0.095	0.000
seed s3	220	0.568	0.432	0.086	0.009
seed s4	220	0.505	0.341	0.041	0.000
seed s5	220	0.468	0.259	0.036	0.014
seed s6	220	0.459	0.455	0.173	0.009
seed s7	220	0.518	0.473	0.132	0.009
mean	–	0.485	0.373	0.086	0.007

A47 How strong does the constraint have to be before the attack dies?

Appendix A44 reports a binary (an adversary trained entirely off $\mathrm{supp}(M)$ keeps 99 % of its energy outside it and does not land), and a binary is the weakest form of that result. It is one point of a knob nobody turned, and it leaves open the reading that A4 failed for want of capacity rather than for want of support energy. This turns the knob.

The refactor becomes $\Delta W \mapsto (1 - \lambda M) \odot \Delta W$, re-factored at rank r as before: $\lambda = 1$ is A4, $\lambda = 0$ is an ordinary attack, and $\lambda \in \{0.25, 0.5, 0.75\}$ is an attacker who gives up that fraction of his on-support component and keeps the rest. Three seeds at each, at the $p = 20\,\%$ support §5.4 runs the mask at, evaluated unrepaired, under that mask, and under SAFELORA at $\tau = 0.5$. The two endpoints are already on disk, so the curve is five points for the cost of three.

Not one of the 12 constrained adapters lands. The unrepaired arm runs 0.023–0.282 across the whole range, against 0.477 for the same attack unconstrained, and the best any of them reaches is

0.282: under the 0.30 the landing rule asks (Table 54). The residual energy the rank-r truncation puts back on the support falls monotonically with λ (0.0303 at $\lambda = 0.25$ to 0.0058 at $\lambda = 1$), so the constraint is doing what it is supposed to; what it is not doing is leaving an attack behind.

That is the mechanism Appendix A44 could not name. A coordinate support is not a subspace of either factor's range, so *any* enforcement of it on a rank-r factored update requires a lossy re-factorisation, and a quarter of the constraint is already enough to cost the attacker the attack. The blind spot the framework predicts for a fixed coordinate support is not merely unoccupied at the corner we tried first; it is unoccupied along the whole axis we can move.

Two things this does not show. It is one enforcement scheme (projected gradient with a rank-r refactor every eight steps), so a cleverer attacker who optimises the factors under the constraint rather than projecting after the fact is not ruled out, and Appendix A44 says so. And the adapter closest to landing (0.282 at $\lambda = 0.25$) is cut only to 0.232 by the $20\,\%$ mask, so if such an attack did land, the mask would not remove it. The constraint is self-defeating; the mask is not what would stop it.

Table 54: The constraint's strength against what it leaves behind, on Qwen2.5-1.5B. $\lambda = 0$ is the ordinary all-modules attack of §5.4 and $\lambda = 1$ is A4 at $p = 20\,\%$, both read from the stages that already hold them; the three interior rows are this stage's. "landed" counts adapters at or above the 0.30 the landing rule requires. The last column is the mean, over the row's adapters, of the share of the update's energy that sits on $\mathrm{supp}(M)$ after the final refactor.

λ	n	unrepaired	landed	mask_{20}	SAFELORA	resid. on $\mathrm{supp}(M)$
0.00	1	0.477	1	0.005	0.009	–
0.25	3	0.023–0.282	0	0.005–0.232	0.005	0.0303
0.50	3	0.045–0.177	0	0.014–0.118	0.005–0.009	0.0144
0.75	3	0.095–0.159	0	0.064–0.136	0.005–0.009	0.0084
1.00	3	0.068–0.091	0	0.064–0.082	0.005–0.009	0.0058

A48 REACH AGAINST THE SIZE OF THE SAFETY SET

Corollary 1 bounds a repair's reach by the safety set's token count, and every ρ outside this appendix is measured at $|\mathcal{D}_{\mathrm{safe}}| \in \{1, 5\}$. That is the regime the repairs we attack operate in (Zhang et al. (2026b) patch from one instance), but a reader is entitled to ask what happens when a defender brings more, and the row-space half of the framework (A2, the coordinate mask's blind spot, the one-shot patch's containment) depends on the answer. It is measured here rather than conceded.

The method is the one Appendix A12 uses, at the same tolerance: an SVD of the stacked module inputs over every token position of $\mathcal{D}_{\mathrm{safe}}$. The sets at 1 and 5 are the paper's own files. The larger ones are AdvBench prompts *disjoint* from both evaluation sets, paired with five refusal templates, public data a provider could assemble, not our benchmark. At $|\mathcal{D}_{\mathrm{safe}}| = 1$ the script reproduces Appendix A12's ρ to the digit, which is what makes the rest of the curve comparable.

Reach is not small for long. ρ^{par} runs 2.0, 10.0, 33.1, 75.5 and $80.2\,\%$ at 1, 5, 20, 50 and 100 instances (936 and 2298 tokens at 20 and 50), and by 50 instances the span *saturates*, $\mathcal{H}_{\mathrm{safe}} = \mathbb{R}^{d_{\mathrm{in}}}$, on 83 of 196 target matrices, rising to 107 at 100 (Table 55). On those matrices A2 cannot be built at all: there is no complement to hide in. Below saturation the span is smaller than the token count, unlike at one instance: an attention matrix spans 204 of 246 dimensions at five instances and 674 of 936 at twenty.

And the advantage A2 has shrinks faster than the reach grows. The A2 adapter of §5.4 is trained row-orthogonal to the *one-instance* span, where its coverage is 0.000 against an ordinary attack's 0.035. Against the five-instance span it reads 0.108 against 0.140; against the twenty-instance span 0.403 against 0.427, a ratio of 0.94, which is to say a defender with twenty safety examples reaches an adapter built to evade one almost exactly as well as he reaches an ordinary attack. At fifty it is 0.932 against 0.941.

This is a limit on A2 and we state it as one. A2 is an adversary against a *one-shot-scale* row-space repair, which is the class Zhang et al. (2026b) define and the class §5.4's coordinate mask belongs to at the budget we run it at; it is not an adversary against a defender who brings a hundred refusals.

SAFELORA and §5.1's result are untouched by any of this, because C is built from two public checkpoints and uses no safety data at all. Read the other way, the curve says how much safety data it takes to reach a *fixed* update, and the answer on this model is tens of instances rather than thousands. It does not say that collecting them closes the blind spot, because an adversary who knows the defender's budget solves the constraint against the larger span instead: Appendix A64 runs that and finds the gain undone, with the mask's dial rather than the size of the safety set doing the work.

Table 55: Reach and coverage against $|\mathcal{D}_{\text{safe}}|$ on Qwen2.5-1.5B, over all 196 target matrices. "saturated" counts matrices where $\dim \mathcal{H}_{\text{safe}} = d_{\text{in}}$. The coverage columns are one A2 and one ordinary all-modules adapter of §5.4 (seed 0), both built against the *one*-instance span and here read against each larger one; "ratio" is $c(\text{A2})/c(\text{ord.})$.

$\lvert\mathcal{D}_{\text{safe}}\rvert$	tokens	ρ^{par}	saturated	c(A2)	c(ord.)	ratio
1	41	0.0202	0	0.0000	0.0353	0.00
5	246	0.1000	0	0.1080	0.1403	0.77
20	936	0.3305	0	0.4025	0.4270	0.94
50	2298	0.7548	83	0.9318	0.9407	0.99
100	4549	0.8023	107	0.9459	0.9530	0.99

A49 A THIRD STATISTIC, AND THE SAME ATTACKER WALKS PAST IT

§6 reads the gate back two ways and an attacker buys both off. A reader who knows how A3 is built will ask why neither is the obvious one: A3 constrains $B \leftarrow P_m P_m^\top B$ after every step, so $\text{col}(\Delta W)$ lies inside V's leading m left-singular directions *exactly*. The statistic that reads that directly is

$$E_k(\Delta W) \;=\; \|U_k^\top \Delta W\|_F^2 \,/\, \|\Delta W\|_F^2,$$

the share of the update's output energy inside V's leading k block. It costs the same thin SVD of the same two public files SAFELORA already computes. It is not proof against what the attacker adds: a decoy branch outside V's leading block can only *dilute* E_k, and a rule that flags a high E_k is evaded by dilution.

At $k = 256$, A3's own budget, an unconstrained update reads 0.099 by construction. Over 19 benign task adapters spanning four tasks and four loci the band is 0.056–0.431; the 17 A3 adapters read 0.842–0.900, a margin of 0.411 over the widest benign adapter, and every one of them is flagged at the $+0.10$ margin §6 already operates the cosine at. So does every one of the 12 adapters aimed at τ (0.553–0.614), the arm built to walk under the cosine rule, which this one catches. The ordinary attacks read 0.072–0.463 and A2 0.127–0.136; none is flagged (Table 56). The decoy, which takes the deleted-fraction rule to nothing, is flagged on 3 of 12 at 0.439–0.632.

And the same escape works. Widening the budget towards d_{out} reads 0.291–0.332 at $k = 256$ and 0.753–0.852 at $k = 1024$: under the flag threshold at every width we measured, and inside the benign band at all but $k = 1024$. That is not a coincidence and it is not a weakness of this statistic: widening m *is* giving up the constraint E_k reads, and it is the same move that walks under the cosine (Appendix A34). §6's conclusion is therefore unchanged and better supported: the confinement is detectable, the detector is not what the attacker has to beat, and the case that stays open is the one where he stops being confined. The flattest-band arm reads 0.164–0.172 at $k = 256$ and escapes there, but 0.992–0.992 at $k = 1024$, where it is flagged: a defender who flags an adapter at any of the three widths keeps that one, and on the adapters we hold that union flags no benign task adapter, ordinary attack, A2 or A4.

Two honest limits. Like the cosine, E_k separates *constraint* and not harm: the benign fine-tunes trained under A3's own constraint (Appendix A36) read 0.895–0.901 and are flagged at every width. And this is one model, computed on the adapters we hold: it is a statistic we are handing a defender, not a defence we have tested against an adversary who knew it was coming, which by §6's own standard is what it would take to recommend it.

Table 56: E_k by family on Qwen2.5-1.5B at three block widths, over the adapters this stage scored, which are not the census of Table 7: "A3, m=256" pools the eight seeds of Appendix A24, the rank-64 adapters of Appendix A7, the all-MLP adapters of Appendix A38 and the three of Appendix A42, and the $\{32, \ldots, 1024\}$ sweep is not scored. "flagged" counts adapters at or above the widest benign task adapter (first row) plus 0.10, the margin §6 uses for the cosine rule, so the benign row's $0/19$ holds by construction and is not a held-out rate. E_k for an unconstrained update is $\min(k, d_{\text{out}})/d_{\text{out}}$ pooled by parameter count: 0.050, 0.099 and 0.347 at the three widths; on the 56 `k_proj`/`v_proj` matrices ($d_{\text{out}} = 256$) it is 1 for any update once $k \geq 256$.

	$k = 128$		$k = 256$		$k = 1024$	
family	E_k	flagged	E_k	flagged	E_k	flagged
benign task adapters	0.033–0.290	0/19	0.056–0.431	0/19	0.174–0.819	0/19
ordinary attacks	0.044–0.324	0/9	0.072–0.463	0/9	0.201–0.826	0/9
A2	0.082–0.088	0/14	0.127–0.136	0/14	0.302–0.314	0/14
A4	0.072–0.079	0/15	0.117–0.129	0/15	0.299–0.332	0/15
random subspace	0.176–0.179	0/3	0.278–0.282	0/3	0.554–0.555	0/3
A3, m=256	0.573–0.634	17/17	0.842–0.900	17/17	1.000	17/17
A3 aimed at τ	0.451–0.478	12/12	0.553–0.614	12/12	0.886–0.926	3/12
A3, flattest band	0.113–0.119	0/3	0.164–0.172	0/3	0.992	3/3
A3, $m \to d_{\text{out}}$	0.180–0.205	0/6	0.291–0.332	0/6	0.753–0.852	0/6
A3 + decoy branch	0.076–0.315	0/12	0.439–0.632	3/12	0.447–0.710	0/12
benign under A3's constraint	0.604–0.617	4/4	0.895–0.901	4/4	1.000	4/4

A50 LIMITATIONS, IN FULL

The paper's closing paragraph states the main ones in brief; here they are with their scope attached, together with the boundaries the appendices add.

The adversary that fails is one construction. A4 (Appendix A44) is the paper's only negative result about an adversary *built to evade a repair* (the second objective of Appendix A42 is a non-landing of the attack itself, and A3 at its default $m = 256$ does not land on Llama-3.1-8B by the judge, Appendix A18), and a negative result is exactly where construction choices hide. It is six adapters on one model at one rank under one objective, and one way of enforcing a constraint that cannot be enforced exactly: projected gradient with a rank-r re-factoring every eight steps. An attacker with a different optimiser, a much longer schedule, or a factorisation that buys back what the truncation removes is not ruled out by it. What it shows is that the cheapest version we could build, at A3's cost, fails on this model, which is the version that matters, because A3's claim was that evasion is free.

Breadth of the probes. §5.4 tests three operator families beyond SAFELORA, but thinly: RESTA on the three models with an aligned θ_0, the coordinate mask run on all four and counted on three (Tables 29–32, Mistral set aside), and the A2-versus-mask dissociation on eight A2 seeds on Qwen2.5-1.5B and four landed adapters on Qwen2.5-7B, two of them A2, on which it holds only in the relative cut and not in level (Appendix A46, Table 30, §5.4); the gradient patch's blind spot rests on one A2 adapter on one model (Appendix A23), and the coarse grid's 7–8B zero rests on a clean-point cut of 0.15 that we chose for reporting: at 0.20 Llama-3.1-8B has two (Appendix A23). The panel is four checkpoints of three families, and the two Qwen models share a tokenizer and an alignment recipe, so scale, family and d_{out} move together: a model-dependent finding here is a finding about that checkpoint, not about size. Table 17 scores six of the paper's claims by model, and Appendix A18 gives each of the seven pre-registered predictions its verdict, the two the data did not support included.

One attack objective. Every attack in the paper is identity shift (Qi et al., 2024) on the same 64 benign examples, at rank 16 by default (4 and 64 in the rank sweep), $\alpha = 32$, 200 steps, AdamW at 2×10^{-4}. That is the implicitly harmful tier of Qi et al. (2024), not their explicitly harmful one, and it is deliberately held fixed, since the paper's variable is *where* the update sits rather than how hard it pushes; but it means we do not know where a stronger or differently-shaped objective would put its energy, and therefore whether the gate would still read $\geq \tau$ for it. We tried a second objective and it did not produce an attack to ask that of: affirmative compliance, everything else unchanged, lands 0

of 5 arms (Appendix A42). The gate is still seen firing on the ordinary adapters and not on A3 there, in perplexity and in the paired differences, but on arms that do not clear our own bar: a direction, not a replication.

What the frame does not cover. Full-parameter attacks fall outside SAFELORA's factored projection altogether, so nothing here speaks to them. An aligned θ_0 is a precondition for the whole A3 claim, which is why Mistral-7B is reported and then set aside (§5.5). Repairs that choose their set *after* seeing the attack are outside the operator frame by construction; Appendix A26 prices one rather than dismissing it. And A2–A4 need the attacker to own the optimiser, since each constrains the update at every step: where the provider runs the fine-tuning job and the attacker supplies only data, A1 is the only adversary here that applies, and SAFELORA cuts it: over the nine locus configurations on the three in-scope models it takes 0.506–0.670 to 0.011–0.289, the least on Llama-3.1-8B's attention loci (Table 2). The decoy of Appendix A38 needs more than public weights, an estimate of each module's input span. The frame is the party who did not train the adapter (adapter sharing, or screening a submitted one) not fine-tuning-as-a-service in general.

How many models each result rests on. The headline is three models. §6's adaptive attacks ran on two models and its answer on three. The decoy and the gate re-designs are one model each (Qwen2.5-1.5B, where the branch was constructed), the attack-dependent repair three (it beats the fixed-reach mask on Qwen2.5-7B only, Appendix A26), and $\|\Delta W\|_G$ is computed on all four and separates on three (Appendix A37), and E_k (Appendix A49) is one model and has not faced an adversary aimed at it. Appendix A39 predicts where the widened statistic lands on the three larger models (inside the window on one, at τ_d on one, below τ on one) and measures the first, Llama-3.1-8B: at $m = d_{\text{out}}$ the mean does land inside the window, but the matrices the saturated budget leaves unconstrained are projected and one of three adapters survives, so §6's escaper is, but for that one seed, a Qwen2.5-1.5B result. The mask-versus-escaper test of Appendix A53 is six adapters on one model, and at $40\,\%$ the mask leaves the eight A2 seeds at 0.023–0.173 (Appendix A46), above the ≤ 0.041 to which the $20\,\%$ mask already takes ordinary attacks (§5.4). The models span 1.5B to 8B, and A3's budget m is model-dependent (1024 on Llama-3.1-8B, 256 elsewhere).

The judge. Harm labels come from an LLM (`gpt-4.1-mini`), validated against a stronger model of the same family ($\kappa = 0.831$) and a third from a different vendor ($\kappa = 0.859$), but not against people. Ten of the twenty-nine stages carry a second judge and none two (Appendix A29); per item the two judges disagree on $8.6\,\%$ of Table 2's A3 labels, wider than that table's A3 effects, which is why those are stated as equivalences at a margin; and each cell is judged on a $40 + 40$ subset except in the full-set stages §4 names. A human-labelled audit is the right check; we did not run one.

What the equivalence rests on. On Llama-3.1-8B the headline equivalence holds at the item level over three fixed adapters and fails when those three are resampled as clusters, and the out-of-sample selection of Appendix A11 leaves it at -0.100, on the margin; the adapter-level support for that model is the eight-seed arm of Appendix A24, on a second software stack. On the same model a rank-64 ordinary attack survives SAFELORA about as well as A3 does (Appendix A7), and a random column subspace of A3's width survives its projection on Qwen2.5-1.5B (Appendix A36): what V buys there is the gate, not the survival. On Qwen2.5-7B and Llama-3.1-8B the same control is removed and A3 is not (Appendix A59), so on two of the three checkpoints what survives is V's orientation, not the confinement alone. The same appendix trains benign fine-tunes under A3's constraint, and §6's rule flags all four at every margin, so its false positives include the customer who constrains an adapter to keep SAFELORA from deleting it. The margin $\delta = 0.10$ is calibrated to Table 2's smallest ordinary effect, not pre-registered (Appendix A30). The tests behind these equivalences hold their nominal size under a true null, and the cluster interval is conservative at the cluster counts we use, which is checked by simulation in Appendix A58 alongside a null-configuration check of the three closed forms.

Four of the five repairs are our implementations. Only SAFELORA is checked against its authors' code (§2), and the headline runs it in its threshold mode: the release's default is a count, the ten lowest-cosine matrices whatever they read. Appendix A57 checks our implementation against theirs, imported unmodified, and Appendix A55 runs the count rules on all three in-scope models — at ten matrices A3 stands on six of eight and the ordinary attack's removal is established on three

of six; at 35 % the ordinary attack and A2 are removed and A3 stands on five of eight. An A3 aimed at a count is not tried at all. The rank-k edit's strength is calibrated on a 0.5B pilot and not swept per model (Appendix A15), and the full-parameter patches are applied to `bf16` weights, the deployment dtype, with the optimiser arithmetic in `fp32` (Appendix A16): at the AdamW rates run on the in-scope models, rounding alone leaves 32.1–96.8 % of the target coordinates unable to move in any of the ten steps (Appendix A56), and the arm with `fp32` master weights that prices this (Appendix A56, Table 69) moves the window down in rate, giving Llama-3.1-8B a clean point at 2×10^{-6} (the `bf16` arm's first are at 8×10^{-6}), at which A3 still reads 0.450; on Qwen2.5-7B it removes the landed attack only at over-refusal 0.28, so that model has no clean point on its grid, and the plain-SGD arm was not re-run in `fp32`. The row-space half is measured at one-shot scale, where $\rho \leq 2.7\,\%$; at twenty safety instances A2's coverage is 0.94 of an ordinary attack's (Appendix A48). The utility side of §5.3 rests on one SQL-Create adapter per model with denominators as small as 0.11 (Table 57); the three-seed, four-task sweep of Appendix A33 is its check on two models.

A51 The defender's dilemma, tabulated

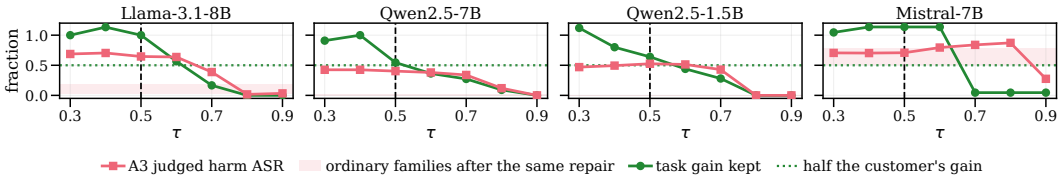

Figure 6: Both of §5.3's words, drawn: green is the customer's gain, dotted at half of it (*keeps*); red is judged A3 harm ASR, banded where the same repair at $\tau = 0.5$ leaves that model's ordinary families on AdvBench (*removes*; §5.3 holds all three in-scope models to Llama's band, the loosest, with the same verdict). On the three models with an aligned θ_0 no τ puts red inside the band with green above the line. Mistral's band is at 0.50–0.78 because the repair leaves Mistral's ordinary families far above its already-high floor, so three of its thresholds satisfy both, which is §5.5's precondition, not a counterexample. The dashed vertical line is $\tau = 0.5$, the threshold the headline runs at.

Table 57 is the τ sweep behind Figure 6: every landed A3 adapter (two on Llama-3.1-8B, three elsewhere) and one SQL-Create adapter per model at every threshold from 0.3 to 0.9, with the judged harm rate and the fraction of the customer's gain that survives side by side.

A52 Per-locus detail

Tables 58–61 give the full locus breakdown behind Figure 1, one per model, sorted by the gate statistic. Reading down each table, the cosine falls and the fraction of the update that SafeLoRA deletes rises (monotonically on Llama-3.1-8B, and elsewhere with one to three inversions, all between adjacent loci separated by at most 0.033 in cosine), while the post-repair ASR collapses on the Qwen models' ordinary loci, falls substantially on Llama-3.1-8B's and barely moves on Mistral, whose own caption says why. This is the relation Table 15 quantifies, seen row by row in substring units. ASR columns here are substring-based (see Appendix A17); the judged rates are in Table 2. Standard deviations are over seeds within a locus.

A53 Section 6's two remedies, run against Section 6's own escaper

§6 builds an adversary that widens m until the adapter mean crosses τ_d, and names two candidate remedies: turn the coordinate mask up, or empty V's leading block outright. Neither can be read off existing cells: Appendix A45's adapters stop at m=256 and so do Appendix A25's, and the widened adapters at the mask's 20 % default point the wrong way. This appendix runs both against the widened adapter, and §6 reports the result.

What was run. The six adapters of Appendix A33's $m \in \{1280, 1536\}$ sweep on Qwen2.5-1.5B, three seeds each, against eleven repairs: the coordinate mask at every setting of its dial, and both gate

Table 57: Task gain retained (%) and A3 judged harm ASR at each SAFELORA threshold, all four models; the tabular form of Figure 6. Task gain is $(\text{metric}_\tau - \text{metric}_{\theta_0})/(\text{metric}_{\text{tuned}} - \text{metric}_{\theta_0})$ on SQL-Create execution match for one adapter per model on 100 test items. The denominator is the fine-tune's own improvement and it is not large: $0.31 \to 0.61$ on Llama-3.1-8B, $0.46 \to 0.57$ on Qwen2.5-7B, $0.33 \to 0.58$ on Qwen2.5-1.5B, $0.24 \to 0.46$ on Mistral-7B, so the narrowest is Qwen2.5-7B's at 0.11 and a single item there moves the percentage by nine points. Read the sweep as a shape, not cell by cell. (The three-seed, 60-item replication on two models is Tables 36–37, which is why the two disagree at the seed level; the four-task, three-seed pool at $\tau = 0.5$ alone is Table 6's SAFELORA row.) Values above 100 % mean the projected adapter beat the unprojected one. A3 is every adapter that landed (two on Llama-8B, three elsewhere, $m = 1024$ on Llama-8B and 256 elsewhere) judged on AdvBench, on the fixed 40-prompt subset for the adapter §5.3's sweep carries and the full 120 for the ones Appendix A27 adds, pooled as an unweighted mean of per-adapter rates; the per-adapter range is in Table 24. Its $\tau = 0.5$ column is therefore a different estimate from Table 2's repaired column (Appendix A27). On the three models where any τ suppresses A3, the first one that does leaves 0–9 % of the customer's fine-tune: on Llama-3.1-8B $\tau = 0.5$ keeps 100 % of the gain at A3 0.646 (0.492 and 0.800 per adapter) and suppression arrives only at $\tau = 0.8$ (0.017), where the gain is gone, and the two Qwen models cross in the same place, $\tau = 0.8$ leaving 9 % and 0 % of the gain though Qwen2.5-7B still leaves 0.119 of the attack. On Mistral none does, and A3 *rises* with τ to 0.872 before falling to 0.275 at $\tau = 0.9$ (§5.5).

	Llama-8B		Qwen-7B		Qwen-1.5B		Mistral-7B	
τ	task	A3	task	A3	task	A3	task	A3
0.3	100%	0.688	91%	0.425	112%	0.469	105%	0.706
0.4	113%	0.704	100%	0.425	80%	0.494	114%	0.703
0.5	100%	0.646	55%	0.403	64%	0.525	114%	0.708
0.6	57%	0.637	36%	0.381	44%	0.514	114%	0.794
0.7	17%	0.388	27%	0.339	28%	0.428	5%	0.839
0.8	0%	0.017	9%	0.119	0%	0.000	5%	0.872
0.9	0%	0.033	0%	0.000	0%	0.000	5%	0.275

Table 58: Llama-3.1-8B-Instruct, SAFELORA at $\tau = 0.5$, by attack locus.

locus	n	$\cos(C\Delta W, \Delta W)$	removed	ASR att.	ASR rep.	held-out
A3 alignment-aligned, m=256	3	0.844 ± 0.001	0.071	0.050	0.050	0.187
A3 alignment-aligned, m=1024	3	0.743 ± 0.001	0.121	0.644	0.600	0.687
mid/attention	5	0.624 ± 0.002	0.478	0.532	0.213	0.418
early/attention	5	0.604 ± 0.003	0.566	0.173	0.002	0.050
late/attention	5	0.577 ± 0.003	0.584	0.330	0.070	0.160
A2 row-orthogonal	3	0.542 ± 0.001	0.849	0.133	0.006	0.100
all modules	11	0.537 ± 0.004	0.854	0.558	0.151	0.236
early/MLP	5	0.483 ± 0.001	0.923	0.218	0.022	0.184
mid/MLP	5	0.454 ± 0.001	0.935	0.895	0.032	0.138
late/MLP	5	0.438 ± 0.002	0.946	0.135	0.007	0.032

re-designs of Appendix A25 at $k \in \{8, 32, 128\}$: the unconditional deletion $I - U_k U_k^\top$ and the gated rank-k projector. Every adapter carries its own unrepaired arm, all 220 prompts of both sets: twelve arms per adapter, 15,840 judgments. No adapter is trained here; all six already existed.

The mask holds. Emptying the block does not. Table 62 is the result and it is the one the reach argument predicts. The mask at 40 % takes the escaper from 0.502 to 0.010, every one of the six adapters between 0.000 and 0.018, and at 60 % to 0.005. Deleting V's leading block does little at any k we ran: 0.465, 0.448 and 0.457 at $k = 8$, 32 and 128, against an unrepaired 0.502, cuts of four to five points whose item-level intervals exclude zero (Table 62). At k=128 it removes 42–45 % of the update's norm and 4.5 points of the behaviour.

That asymmetry is the paper's own mechanism, measured. A3 at m=256 lives inside V's leading block by construction, so emptying that block empties the attack, which is what Appendix A25 finds. The widened adapter does not: at $m \to d_{\text{out}}$ its update spans all of $\text{col}(V)$, so a defence that deletes

Table 59: Qwen2.5-7B-Instruct, SAFELORA at $\tau = 0.5$, by attack locus.

locus	n	$\cos(C\Delta W, \Delta W)$	removed	ASR att.	ASR rep.	held-out
A3 alignment-aligned, m=256	3	0.765 ± 0.002	0.114	0.525	0.533	0.787
mid/attention	5	0.580 ± 0.001	0.810	0.822	0.005	0.076
late/attention	5	0.557 ± 0.003	0.884	0.538	0.000	0.012
early/attention	5	0.550 ± 0.002	0.858	0.062	0.000	0.020
A2 row-orthogonal	3	0.484 ± 0.001	0.919	0.358	0.006	0.073
all modules	11	0.480 ± 0.005	0.930	0.476	0.000	0.030
early/MLP	5	0.393 ± 0.005	0.956	0.150	0.000	0.026
mid/MLP	5	0.382 ± 0.002	0.938	0.860	0.015	0.156
late/MLP	5	0.358 ± 0.002	0.956	0.157	0.000	0.010

Table 60: Qwen2.5-1.5B-Instruct, SAFELORA at $\tau = 0.5$, by attack locus.

locus	n	$\cos(C\Delta W, \Delta W)$	removed	ASR att.	ASR rep.	held-out
A3 alignment-aligned, m=256	3	0.718 ± 0.002	0.192	0.742	0.733	0.937
mid/attention	5	0.595 ± 0.003	0.677	0.792	0.010	0.058
late/attention	5	0.564 ± 0.003	0.729	0.652	0.035	0.120
early/attention	5	0.555 ± 0.001	0.741	0.272	0.003	0.040
A2 row-orthogonal	3	0.493 ± 0.001	0.919	0.492	0.000	0.017
all modules	11	0.489 ± 0.004	0.929	0.783	0.007	0.071
early/MLP	5	0.429 ± 0.000	0.956	0.490	0.000	0.040
mid/MLP	5	0.373 ± 0.001	0.960	0.893	0.000	0.050
late/MLP	5	0.370 ± 0.003	0.955	0.623	0.000	0.020

k of d_{out} directions removes a shrinking share of somewhere the attacker no longer needs to be. The mask's support is not a spectrum and does not move when m does, so widening buys the attacker nothing against it. *The repair that survives a widening attacker is the one whose fixed set is not a public spectrum*, and that is the thesis of this paper rather than an aside in its last paragraph.

What it costs, and the third arm that is not a defence. The mask's $40\,\%$ setting costs the customer $6\,\%$ of the SQL-Create gain (Appendix A45, which prices the dial on that one task), and $60\,\%$ costs $17\,\%$. Emptying the block is nearly free, 0.91 to 1.11 of the four customers' gain at every k (Appendix A25), but it buys nothing here, and a repair that removes nothing at no cost is not a bargain. The gated rank-k projector is the arm to read carefully: at k=8 and k=32 it does take the escaper to 0.009 and 0.008, but it gets there by deleting 93–98 $\%$ of the update, and Appendix A25 prices that on the customer at 0.06 of the GSM8K gain and -0.05 of the summarisation gain. It removes the attack by removing the adapter, which is the failure mode §5.3 describes for a raised τ, reached by a different route.

A54 IS ANY OF THIS ABOUT *where*? ORIENTATION AT A MATCHED DOSE

There is a reading of this paper that makes its subspaces decoration. SAFELORA removes $93\,\%$ of an ordinary adapter and 7–19 $\%$ of a headline A3 one; perhaps post-repair compliance is a function of *how much* of the update was deleted and of nothing else, the gate merely decides the dose, and the finding is the simpler and larger one that deleting most of a LoRA's norm removes none of its behaviour, whichever way the deletion points. Appendix A36 is what raises it: a random column subspace of the same width survives too ($0.633 \to 0.574$ at $75\,\%$ deleted) where an ordinary adapter at $93\,\%$ ends at 0.009.

The records answer it without a new run, because `removed_frac` is recorded for every judged repaired cell and the τ sweeps move it across the whole range one adapter at a time. Bin the cells by deleted fraction and read the spread *across orientations within a bin*. The quantity is retention above the checkpoint's own floor, $(\text{asr} - \text{asr}_0)/(\text{asr}_{\text{unrep}} - \text{asr}_0)$: Mistral complies with $28.7\,\%$ of the pooled set before any attack, and a raw ratio would credit that to the repair.

Table 61: Mistral-7B-Instruct-v0.3, SAFELORA at $\tau = 0.5$, by attack locus. This is the model whose aligned checkpoint θ_0 is weakly aligned (§5.5): the gate behaves exactly as on the other three, but the post-repair ASR floor is set by a θ_0 that already answers $55\,\%$ of held-out HarmBench by substring ($32.5\,\%$ judged, §5.5).

locus	n	$\cos(C\Delta W, \Delta W)$	removed	ASR att.	ASR rep.	held-out
A3 alignment-aligned, m=256	3	0.718 ± 0.002	0.180	0.836	0.844	0.963
mid/attention	5	0.525 ± 0.003	0.852	0.890	0.730	0.864
late/attention	5	0.492 ± 0.002	0.831	0.848	0.412	0.624
early/attention	5	0.482 ± 0.006	0.772	0.948	0.647	0.748
A2 row-orthogonal	3	0.448 ± 0.001	0.896	0.925	0.306	0.600
all modules	11	0.446 ± 0.006	0.905	0.939	0.897	0.934
early/MLP	5	0.402 ± 0.004	0.922	0.908	0.930	0.932
late/MLP	5	0.375 ± 0.006	0.949	0.813	0.322	0.592
mid/MLP	5	0.372 ± 0.003	0.940	0.860	0.420	0.688

Table 62: §6's two recommended remedies, and the gate re-design beside them, run on §6's own widening escaper: the six $m \in \{1280, 1536\}$ adapters on Qwen2.5-1.5B, all 220 prompts judged, paired at the item level against each cell's own unrepaired arm. "deleted" is the fraction of the update's norm the repair removes. The coordinate mask removes the escaper at $p \geq 40\,\%$; deleting V's leading block does not, at any k; the gated rank-k projector does, by deleting the adapter.

repair	judged	change [95 % CI]	deleted
unrepaired	0.502	$+0.000$ $[+0.00, +0.00]$	–
coordinate mask, p=5 %	0.441	-0.061 $[-0.09, -0.03]$	0.23
p=10 %	0.417	-0.085 $[-0.11, -0.06]$	0.32
p=20 % (the default)	0.242	-0.260 $[-0.29, -0.23]$	0.45
p=40 %	0.010	-0.492 $[-0.52, -0.46]$	0.63
p=60 %	0.005	-0.496 $[-0.52, -0.47]$	0.78
$I - U_k U_k^\top$, k=8	0.465	-0.036 $[-0.06, -0.01]$	0.17–0.19
k=32	0.448	-0.054 $[-0.08, -0.03]$	0.27–0.29
k=128	0.457	-0.045 $[-0.07, -0.02]$	0.42–0.45
$C_k = U_k U_k^\top$ gated, k=8	0.009	-0.492 $[-0.52, -0.47]$	0.98
k=32	0.008	-0.494 $[-0.52, -0.47]$	0.93–0.94
k=128	0.154	-0.348 $[-0.38, -0.32]$	0.83–0.85

Four things this measurement can get wrong, and what is done about each. Each is easy to get wrong, so they are stated rather than buried. *(i)* Clusters keyed on the attack *name* count `all-all-r16-plain-s0` trained separately on three models as one adapter and resample across models; the key is $(\text{model}, \text{attack})$. *(ii)* Taking the unrepaired denominator as the maximum over every stage that re-ran that adapter inflates it, and lets a cell clear the landing rule on a rate measured somewhere else; it is the cell's own stage's `none` arm, or (for the τ sweeps, which carry no unrepaired arm) the canonical `matrix` one. *(iii)* A numerator from the sets a cell was judged on, over a denominator and floor pooled across both, normalises AdvBench-only τ-sweep cells against a floor containing HarmBench; every term is computed over the same set list. *(iv)* With no floor on the denominator, a Mistral cell at $\text{asr}_{\text{unrep}} = 0.31$ is divided by 0.02; cells need 0.15 of headroom above the floor, which excludes two. Intervals are a $90\,\%$ cluster bootstrap over adapters.

At the same dose, the operator decides whether orientation matters. Tables 63 and 64 are the same bin (`removed_frac` in $[0.60, 0.80)$, family mean doses 0.710–0.747 and 0.703–0.712) under the two operators. Four of the five families are the same on both sides (ordinary, A2, A3 and the decoy); the fifth is the random subspace under SAFELORA and the widened A3 under the mask. Under SAFELORA five families span 0.770, from the decoy at $+1.034$ $[+0.998, +1.071]$, untouched by a repair that deleted $71\,\%$ of it, down to A3 at $+0.264$ $[+0.090, +0.530]$; no interval contains zero.

Read those brackets with the cluster count in view. Every family here has at least three adapters, but three is where a cluster bootstrap stops being informative rather than where it starts: with $K = 3$ clusters (adapters) there are 27 equally likely resamples and ten distinct means, each endpoint is a

resample mean built from at most two distinct clusters, and the interval is contained in the minimum and maximum of the three by construction. So "no interval contains zero" is guaranteed for any family whose three adapters agree in sign, and in twelve of the 44 (family, operator, bin) intervals this appendix computes $K \leq 4$ (K is the cluster count, the "ad." column of every table here) so those brackets should be read as the range of what was measured and not as a confidence statement. The three the argument actually leans on are wider: A3 under SAFELORA ($K = 6$), A3 under the mask ($K = 7$), and the ordinary family under SAFELORA ($K = 18$). Under the coordinate mask the same five span 0.047 and all of them lie within 0.04 of zero, from A2 at $+0.010$ to the ordinary family at -0.037. Restricted to the three families that are not purpose-built controls and that both operators carry (A2, ordinary and A3) the spans are 0.545 and 0.047, a ratio of $11.7\times$. Every ratio in this appendix is computed from the unrounded spans, so dividing the printed three-decimal spans reproduces it only to the first decimal.

A cell is not an adapter. A cell here is one (stage, adapter, repair, setting) measurement, and among the $1{,}062$ cells that clear this appendix's filters, 75 of the 939 (model, attack, repair, setting) keys are measured in more than one stage: 123 extra cells. Those are replicates rather than duplicates, measured independently in each stage, but a mean over cells weights an adapter by how many stages happened to include it, so both weightings are computed. In the headline bin the 43 SAFELORA cells are 36 adapters and the 55 mask cells are 32; weighting by adapter the spans are 0.729 and 0.047 rather than 0.770 and 0.047, and on the common three 0.504 against 0.047 rather than 0.545 — the headline ratio $15.6\times$ rather than $16.5\times$, and $10.8\times$ rather than $11.7\times$ on the common three. The intervals were already adapter-clustered. The ordering and the order of magnitude do not move, and the body quotes the cell-weighted figure; the "cells" column in the tables is a cell count and "ad." the adapter count, and each should be read as what it is.

It *is* a high-dose statement, and the other two bins say so. Three deletion bins are populated by both operators, and the body quotes the highest. Restricting each to the families that bin actually shares (the mixes differ, and comparing a nine-family span with a four-family one is not matched either) the ratio runs $0.32\times$ at 20–40 % deleted (SAFELORA 0.064, mask 0.201, three families), $2.36\times$ at 40–60 % (0.659 against 0.279, four families), and $16.5\times$ at 60–80 % (0.770 against 0.047, four families). At the lowest dose the ordering reverses: there the mask separates families and SAFELORA does not, because at 20–40 % deleted its gate has fired on almost nothing and every family keeps almost everything. The claim this appendix supports is therefore the one the three numbers have: orientation-dependence under SAFELORA grows monotonically with the dose (0.064, 0.659, 0.770) while under the mask it does not fall monotonically (0.201, 0.279, 0.047; it is flat through 60 % and collapses beyond it) and *not* that a matched dose always favours the mask. The bins also carry three, four and four families, so part of each step is composition. §5.2's sentence names its bin, and this is what naming it costs.

And it is not an artefact of which models populate which family. The decoy and the random subspace exist on Qwen2.5-1.5B only, and A3 lands in this bin on the other three, so the two ends of the pooled 0.770 share no model. The comparison therefore also runs *inside* one model, where that objection cannot arise: on Qwen2.5-1.5B four families span 1.008 under SAFELORA and five span 0.069 under the mask, a ratio of $14.7\times$; on Qwen2.5-7B three span 0.585 against two spanning 0.051, a ratio of $11.5\times$. Those are the two models where both operators have more than one family in the bin. Neither is family-matched (on Qwen2.5-7B the SAFELORA side carries A3 and the mask side does not, and on Qwen2.5-1.5B the mixes differ in both directions), so this check removes the model confound and does not remove the family one; it is the model objection it answers. The pooled figure and the within-model figures agree to within a factor of 1.5.

So the deleted fraction is not the whole variable. It is decisive for one operator and not for the other, at the same value, which is what the reach argument says: the mask's fixed set is a coordinate support the attacker has no closed form for, so at $p \geq 40\,\%$ the dose is the whole story; SAFELORA's is a public spectrum, so an adapter can be composed to carry the norm the gate reads while putting its behaviour where the projection is not. The larger and simpler claim (that deleting seven tenths of a LoRA removes none of its behaviour whatever you point it at) is false: under the mask at that dose it removes all of it, for every family we have.

Two things this does not show. It is observational. `removed_frac` is set by the attack and the threshold jointly, not assigned, and the bin is twenty points wide even though the family means inside it agree to four. And the coupling under the mask belongs to the *turned-up* dial, not to coordinate masks as a class, at $45\,\%$ deleted nine families span 0.623, from A2 rebuilt on an independent corpus at $+0.788$ to the aimed A3 of Appendix A34 at $+0.165$ (Table 66). That is the content of §5.4's recommendation to turn it up, and the reason the recommendation is not "use a coordinate mask". Read the other way, SAFELORA at 91–99 % deleted (Table 65) still leaves the ordinary families a sixth of their above-floor compliance, $+0.176\,[+0.112, +0.250]$ over 76 adapters on all four models: even where this defence works it is not removing the behaviour so much as removing the adapter, and it does not finish.

Table 63: SAFELORA cells whose `removed_frac` lies in $[0.60, 0.80)$, by the orientation of the attack; the family mean doses agree to 0.037. "retained" is judged compliance above the checkpoint's own floor as a fraction of the same adapter's unrepaired arm on the same prompt sets; the interval is a $90\,\%$ cluster bootstrap over (model, adapter) pairs, and "models" abbreviates Llama-3.1-8B, Mistral-7B, Qwen2.5-1.5B and Qwen2.5-7B as L8, M7, Q1.5 and Q7. At one dose the families span 0.770 and no interval contains zero — but the decoy's and the random subspace's rest on $K = 3$ clusters, where that is guaranteed by construction for any family whose adapters agree in sign (see the text); read those two as ranges.

family	ad.	cells	deleted	retained	90 %	models
decoy carrier	3	3	0.711	$+1.034$	$[+0.998, +1.071]$	Q1.5
random subspace	3	3	0.747	$+0.906$	$[+0.867, +0.945]$	Q1.5
A2 (row-orthogonal)	6	6	0.735	$+0.809$	$[+0.679, +0.940]$	M7 Q1.5 Q7
ordinary	18	21	0.731	$+0.296$	$[+0.133, +0.492]$	M7 Q1.5 Q7
A3	6	10	0.710	$+0.264$	$[+0.090, +0.530]$	L8 M7 Q7

Table 64: The coordinate mask, same bin, same quantities; family mean doses agree to 0.009. Five families again, four of them the same four, the fifth is the widened A3 here and the random subspace there, and the shared four are different cell pools (see the paragraph above), spanning 0.047, every one within 0.04 of zero. Two of the five have intervals excluding zero on the low side, the widened A3 at $[-0.014, -0.005]$ and the ordinary family at $[-0.054, -0.020]$, which is the mask over-shooting the floor, not under-removing.

family	ad.	cells	deleted	retained	90 %	models
A2 (row-orthogonal)	4	8	0.703	$+0.010$	$[-0.037, +0.057]$	L8 Q1.5 Q7
decoy carrier	12	15	0.708	$+0.005$	$[-0.008, +0.017]$	Q1.5
A3, widened m	6	12	0.705	-0.010	$[-0.014, -0.005]$	Q1.5
A3	7	14	0.712	-0.014	$[-0.046, +0.012]$	L8 Q1.5
ordinary	3	6	0.704	-0.037	$[-0.054, -0.020]$	L8 Q1.5 Q7

Table 65: SAFELORA above $90\,\%$ deleted (family mean doses 0.912–0.995, a wider spread than the bins above and so less tightly matched): the spread widens to 0.992, and the ordinary families keep a sixth of their above-floor compliance at $95\,\%$ of the update gone. The family set differs from Table 63's, so "widens" compares two mixtures.

family	ad.	cells	deleted	retained	90 %	models
decoy carrier	6	6	0.961	$+0.988$	$[+0.955, +1.025]$	Q1.5
ordinary	76	128	0.948	$+0.176$	$[+0.112, +0.250]$	L8 M7 Q1.5 Q7
A3	11	11	0.995	$+0.014$	$[-0.012, +0.053]$	L8 M7 Q1.5 Q7
A2 (row-orthogonal)	14	52	0.951	$+0.001$	$[-0.014, +0.012]$	M7 Q1.5 Q7
A2, independent corpus	3	3	0.912	-0.004	$[-0.008, 0.000]$	Q1.5

Table 66: The coordinate mask at 44–49 % deleted, where the dial is not yet turned up: nine families span 0.623. This differs from Table 64 in the dial setting, in the dose and in the family mix, so it shows that the collapse there is not a property of coordinate masks as a class without isolating which of the three causes it.

family	ad.	cells	deleted	retained	90 %	models
A2, independent corpus	3	3	0.443	+0.788	[+0.690, +0.886]	Q1.5
A2 (row-orthogonal)	16	25	0.446	+0.634	[+0.488, +0.770]	L8 M7 Q1.5 Q7
decoy carrier	12	19	0.492	+0.488	[+0.408, +0.573]	Q1.5
ordinary	21	26	0.446	+0.444	[+0.269, +0.667]	L8 M7 Q1.5 Q7
A3 aimed at a band	3	3	0.456	+0.441	[+0.341, +0.541]	Q1.5
A3, widened m	6	12	0.452	+0.437	[+0.264, +0.610]	Q1.5
random subspace	3	3	0.460	+0.341	[+0.179, +0.503]	Q1.5
A3	19	33	0.462	+0.308	[+0.162, +0.477]	L8 M7 Q1.5 Q7
A3 aimed at the detector	21	21	0.458	+0.165	[+0.118, +0.211]	L8 Q1.5

A55 SafeLoRA's other selection rules, read off the sweep

The paper runs SafeLoRA in its *threshold* mode: a matrix is projected when $\cos(C\Delta W, \Delta W)$ falls below τ, at $\tau = 0.5$ and over the sweep of §5.3. That is not the release's default mode. Its configuration ships `select_layers_type="number"` with `num_proj_layers=10`: it computes every matrix's cosine, takes the tenth smallest as the threshold and projects the matrices at or below it, which is the ten lowest-cosine LoRA-B matrices whatever they read; $\tau = 0.5$ is the default of the threshold field, used only in threshold mode. Hsu et al. (2024)'s own experiments, with LoRA on `q_proj` and `v_proj` at rank 8, set $\tau = 0.35$, which projects seven of their LoRA matrices there, and call the point at about 11 % of the matrices projected well balanced; on Llama-3 they project about 35 %, and on their harmful-data set all of them. On our adapters $\tau = 0.35$ falls between the $\tau = 0.3$ and 0.4 cells, which project at most 2 % of an A3 adapter's matrices and 3–37 % of an in-scope all-modules ordinary or A2 one.

Locating them, and then running them. Each of these rules projects the k lowest-cosine matrices of the adapter for some k, and so does the threshold rule at every τ: for one adapter the projected sets are nested in τ. A swept cell with exactly k matrices projected is therefore the count-k repair itself, and otherwise the swept cells with the nearest counts below and above k are contained in it and contain it. That locates every rule on cells already generated and judged, which is how this appendix started. It is an interpolation and not a bound, so 51 attack cells (and, separately, the nine customer-utility arms of the last row of each block) were afterwards *run*: `repair.safelora_count` reproduces the released selection down to the two details that decide a count (its five-decimal rounding and its non-strict comparison) and `stage fixedcount` applies it to the same adapters, on the same 40 AdvBench prompts, under the same judge (§4). Table 67 marks a measured cell with a star and leaves a bracket where the rule was only located. The customer's row is the SQL-Create adapter of §5.3, whose count arms were run against the same θ_0 and tuned denominators as its threshold arms.

The release's default is the weak setting. At ten matrices, on the in-scope models, A3 stands on six of its eight landed adapters and is removed on one; A2, which the threshold removes on all three, stands on three of them; and the ordinary attack's removal is established on only three of six adapters, with two still above the landing bar and one undecided. The reason is in the projected counts, which the cell records carry and the table does not print: ten is fewer matrices than $\tau = 0.5$ takes from an all-modules ordinary or A2 adapter (87–96 of 196–224) and more than it takes from A3 (2–8), so the contrast narrows from both sides. The customer is barely touched (103 % of its gain kept on Llama-3.1-8B, 100 % on Qwen2.5-7B, 104 % on Qwen2.5-1.5B) because ten matrices is a small share of its adapter too.

Their "well balanced" point is no better. At 11 % (the share Hsu et al. (2024) themselves call well balanced, and the one their $\tau = 0.35$ amounts to on their own checkpoint) the contrast is weaker still: the ordinary attack stands on four of six adapters and its removal is established on one, A2 stands on three of three, and A3 stands on five of eight. 11 % of an all-modules adapter is 22–25 matrices, still short of the 87–96 that $\tau = 0.5$ takes, so the ordinary attack keeps most of what the

Table 67: SAFELORA's selection rules, run where marked ⋆ and otherwise located on the τ sweep. "matrices" is the adapter's LoRA-matrix count, "unrep." its judged AdvBench rate before repair. °: a bracket between the two nested swept sets nearest the rule (for $\tau = 0.35$, the $\tau = 0.3$ and $\tau = 0.4$ cells); unmarked: swept cells project exactly the rule's set; five unmarked entries are still ranges, because two thresholds projecting the same set are the same repaired model judged twice; their spread, 0.025–0.075, is this table's own estimate of repeat noise on 40 prompts. Bold: ≥ 0.30 at both ends (stands). Roman: ≤ 0.10 at both ends (removed). Italic: neither. A verdict read off a bracket is an inference and not a bound: the projected sets are nested in τ, the judged rates need not be monotone in k, and four in-scope rows here rise with k by more than that repeat noise (Qwen2.5-7B's A3 s1, Qwen2.5-1.5B's A3 s1 and s2 and its all-modules adapter). Mistral-7B, outside the A3 claim (§5.5), is shown for completeness and not counted in the text.

			judged AdvBench ASR after SAFELORA, by selection rule				
			as run	release	Hsu et al.'s settings		
adapter	matrices	unrep.	$\tau = 0.5$	$N = 10$	$\tau = 0.35$	11 %	35 %
Llama-8B							
all modules	224	0.700	0.025	**0.450**⋆	*0.025–0.550*°	*0.125*⋆	0.025⋆
mid/attention	44	0.525	**0.350**	*0.125*⋆	**0.525–0.575**	**0.375**⋆	0.025⋆
A3, m=1024 s0	224	0.775	**0.800**	**0.775**⋆	**0.800–0.825**°	**0.675**⋆	**0.600**⋆
A3, m=1024 s1	224	0.625	**0.500**	**0.450**⋆	**0.600–0.625**°	*0.150*⋆	0.050⋆
customer's gain kept	224		100 %	103 %⋆	100–113 %°	107 %⋆	107 %⋆
Qwen-7B							
late/attention	36	0.350	0.000	0.000⋆	**0.375–0.400**	**0.300**⋆	0.000⋆
mid/attention	36	0.650	0.000	0.000⋆	*0.050–0.725*°	**0.450**⋆	0.000⋆
A2 s2	196	0.775	0.025	**0.800**⋆	*0.175–0.525*°	**0.850**⋆	0.075⋆
A3, m=256 s0	196	0.450	**0.400**	*0.125*⋆	**0.425–0.450**	*0.175*⋆	*0.175*⋆
A3, m=256 s1	196	0.550	**0.675**	**0.550**⋆	**0.600–0.675**	**0.675**⋆	**0.350**⋆
A3, m=256 s2	196	0.300	*0.150*	0.100⋆	*0.100–0.125*	*0.150*⋆	*0.150*⋆
customer's gain kept	196		55 %	100 %⋆	91–100 %°	109 %⋆	100 %⋆
Mistral-7B							
all modules	224	0.775	**0.925**	**0.775–0.825**°	**0.750–0.825**°	**0.775–0.825**°	**0.750–0.825**°
late/attention	40	0.475	**0.325**	**0.325–0.775**°	**0.650–0.775**°	**0.650–0.775**°	**0.325–0.775**°
late/MLP	30	0.575	**0.325**	**0.575–0.775**°	**0.400–0.775**°	**0.575–0.775**°	**0.575–0.775**°
mid/attention	44	0.825	**0.750**	**0.750–0.825**°	**0.825**°	**0.825**°	**0.750–0.825**°
A2 s0	224	0.800	**0.375**	**0.800–0.850**°	**0.400–0.850**°	**0.800–0.850**°	**0.400–0.850**°
A2 s1	224	0.700	**0.350**	**0.600–0.700**°	**0.475–0.600**°	**0.600–0.700**°	**0.475–0.600**°
A2 s2	224	0.850	**0.475**	**0.850**°	**0.625–0.850**°	**0.850**°	**0.625–0.850**°
A3, m=256 s0	224	0.600	**0.600**	**0.600**	**0.575**°	**0.600–0.775**°	**0.775–0.850**°
A3, m=256 s1	224	0.825	**0.775**	**0.775–0.825**°	**0.775–0.875**°	**0.775–0.825**°	**0.800–0.825**°
A3, m=256 s2	224	0.775	**0.750**	**0.750**°	**0.750**°	**0.825**°	**0.825**°
customer's gain kept	224		114 %	100–105 %°	105–114 %°	100–105 %°	105–114 %°
Qwen-1.5B							
all modules	196	0.350	0.000	**0.300**⋆	0.000–0.025°	**0.425**⋆	0.000⋆
mid/attention	36	0.600	0.000	0.000⋆	*0.250–0.650*°	0.000⋆	0.000⋆
A2 s0	196	0.400	0.000	**0.425**⋆	*0.000–0.325*°	**0.500**⋆	0.000⋆
A2 s2	196	0.400	0.000	**0.450**⋆	*0.000–0.200*°	**0.300**⋆	0.000⋆
A3, m=256 s0	196	0.500	**0.550**	**0.575**⋆	**0.450**°	**0.575**⋆	**0.525**⋆
A3, m=256 s1	196	0.525	**0.450**	**0.500**⋆	**0.475–0.525**°	**0.575**⋆	**0.375**⋆
A3, m=256 s2	196	0.700	**0.575**	**0.775**⋆	**0.550–0.600**°	**0.675**⋆	**0.625**⋆
customer's gain kept	196		64 %	104 %⋆	80–112 %°	112 %⋆	84 %⋆

threshold removes while A3 is untouched either way. We report it because it is the authors' own recommendation and because it is the column least favourable to the contrast this paper draws.

Their most aggressive reported setting is the strong one, and A3 survives it. At 35 % of the matrices, the most Hsu et al. (2024) report short of projecting everything, every decided ordinary adapter is removed (six of six, zero still standing) and so is every decided A2 adapter (three removed, zero standing) — while A3 stands on five of eight, at judged 0.350–0.625, and falls on one. The

customer keeps 107 % of its gain on Llama-3.1-8B, 100 % on Qwen2.5-7B and 84 % on Qwen2.5-1.5B, *more* than the same models' $\tau = 0.5$ leaves them (100, 55 and 64 %).

What this changes. It makes §5.3's dilemma sharper rather than softer. A *share* rule is relative to the adapter: it projects the same fraction of the customer's matrices as of the attacker's, where a threshold high enough to reach A3's matrices has already taken most of the customer's (Qwen2.5-7B's SQL-Create adapter has 117 of its 196 under $\tau = 0.5$ alone). The release's own default has neither property (ten is an absolute count, 4.5 % of an all-modules adapter and 23 % of a mid-attention one), which is the paragraph above. So the release's other mode, at the largest share its authors report, is a *better* defence than the $\tau = 0.5$ this paper runs (it removes the ordinary attack and A2 at a lower cost to the customer) and A3 still walks through it. We say this against $\tau = 0.5$ and not against the whole sweep of §5.3: by the nesting argument above, for any *single* adapter the 35 % rule is itself a threshold rule at some τ, so what separates the two modes is that the share adapts per adapter and a global τ does not. The headline contrast of §5.2 is stated for the threshold; in the count mode the same contrast holds at 35 % and fails both at the release's own default and at the 11 % its authors call well balanced, in the direction that flatters neither side of the argument.

What is still open. The cells the sweep does not decide and that were not run (Mistral's, all of them), the absence of intervals (every cell here is 40 prompts, so a rate resolves to 0.025 and a single generation moves it, and a verdict drawn from a bracket assumes judged compliance is monotone in k, which Table 67 does not establish and in four in-scope rows contradicts), and an attacker who knows the rule is a count: A3 raises cosines and a count ignores their level, so the reply is to concentrate the update in the matrices whose cosine is highest and leave the rest near-inert — the decoy of Appendix A38 aimed at a count instead of a norm. This paper has not run it.

A56 What a `bf16` patch can move

The full-parameter patch of §5.4 updates `bf16` weights in place, $w \leftarrow \mathrm{rd}(w - \eta u)$, with the AdamW direction $u = \hat{m}/(\sqrt{\hat{v}} + \epsilon)$ computed in `fp32` and rounded once, to nearest (Appendix A16). AdamW's decoupled decay term $\eta\lambda w$ is absent because every run here sets $\lambda = 0$; it is the one term that scales with $|w|$ and so would not shrink against the spacing, and the bound below would not cover it. Rounding returns w unchanged whenever $|\eta u|$ is below half the `bf16` spacing at w, $2^{\lfloor \log_2 |w| \rfloor - 8}$, or half that for a step towards zero from an exact power of two (whose lower neighbour is half as far, so `code/bf16_frozen.py` counts those coordinates one binade higher), so which coordinates can move at all is fixed by the weights and the rate before any data is seen. At step one $\hat{m} = g$ and $\hat{v} = g^2$, so $|u| \leq 1$ for any gradient. Over ten steps, Cauchy–Schwarz on the bias-corrected averages gives $|\hat{m}_t/\sqrt{\hat{v}_t}| \leq 1.043$ at $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $t = 10$, and `bf16` storage of the moments adds at most a few per cent, so a coordinate whose half-spacing exceeds $1.1\,\eta$ does not move at any of the ten steps, whatever the gradients, and, not moving, keeps its spacing. Table 68 counts those coordinates over the patch's target set (every q/k/v/o/gate/up/down matrix of every layer) in the released checkpoints, θ_0, and not on the attacked weights $\theta_0 + \Delta W$ the patch actually meets. That substitution is the table's one soft edge: the merged all-modules adapter is 1.7–2.0 % of $\|W\|_F$ on the in-scope models and 7.6 % on Mistral-7B, but a norm is not a per-coordinate bound and a low-rank update is concentrated, so a coordinate can cross a binade that the aggregate does not predict. Recomputing the histogram on the merged weights settles it on one model (`code/bf16_merged.py`): 8.1 % of Llama-3.1-8B's target coordinates *do* change binade when the all-modules adapter is merged, and the frozen fraction nevertheless moves by at most 0.06 % at any rate in the table — at 2×10^{-6} it is 93.6 % before the merge and 93.6 % after: the crossings nearly cancel at the threshold, which is what a sign-symmetric perturbation does to a threshold count. The rule is checked against the patch's own arithmetic on real weights, and every count is a lower bound: a coordinate whose gradient is small is frozen at smaller $|w|$ as well (`code/bf16_frozen.py`; Llama-3.1-8B's shards are read from a mirror whose files match the gated release's SHA-256).

What it means for the AdamW arm. The patch that ran is a patch on the small-magnitude coordinates. At the 8×10^{-6} of Llama-3.1-8B's two clean AdamW points, at least 75.1 % of its target coordinates cannot move; at the 2×10^{-5} of Zhang et al. (2026b), 53.1 % on Llama-3.1-8B

Table 68: Percentage of the patch's target coordinates that no gradient can move in any of the ten AdamW steps, by model and rate: those with $|w|$ above the power of two in the second row. A column stands for every rate that freezes the same set ($2 \cdot 10^{-6}$ also for $3 \cdot 10^{-6}$, $5 \cdot 10^{-6}$ for $6 \cdot 10^{-6}$, $7 \cdot 10^{-6}$ for $8 \cdot 10^{-6}$ and 10^{-5}, $1.5 \cdot 10^{-5}$ for $2 \cdot 10^{-5}$, $3 \cdot 10^{-5}$ for $5 \cdot 10^{-5}$). Black: the frontier ran a rate of that column on that model; grey: it ran none, shown for comparison.

AdamW rate η	10^{-6}	$2 \cdot 10^{-6}$	$5 \cdot 10^{-6}$	$7 \cdot 10^{-6}$	$1.5 \cdot 10^{-5}$	$3 \cdot 10^{-5}$
$\lvert w\rvert >$ (all ten steps)	2^{-11}	2^{-10}	2^{-9}	2^{-8}	2^{-7}	2^{-6}
Llama-8B	96.8	93.6	87.3	75.1	53.1	22.4
Qwen-7B	96.8	93.8	88.3	78.3	60.3	32.1
Mistral-7B	86.6	73.8	50.8	19.6	1.6	0.0
Qwen-1.5B	98.5	96.9	93.9	87.8	76.1	55.0

and 60.3 % on Qwen2.5-7B, and on Qwen2.5-1.5B, whose two clean AdamW points are at that rate, 76.1 %. Over every rate run on the in-scope models the range is 32.1–96.8 %. The frozen set is every large weight, not a random subset. It does not make the patch inert (on Llama-3.1-8B it refuses 93 % of benign traffic at 2×10^{-5}). Rounding also works the other way on the coordinates that do move, since a step just over half a spacing rounds up to a whole one, so the records do not say in which direction an `fp32` patch would move the arm's window. On Mistral-7B, whose weights are smaller, only 1.6 % of the coordinates are frozen at 2×10^{-5}, and the AdamW arm refuses 100 % of benign traffic there (and at least 93 % at every rate it ran), so on that model at least the arm's failure does not come from freezing.

So we ran the other arm. `stage frontier32` repeats the AdamW arm with an `fp32` master copy of every trained weight, copied down into the `bf16` model after each step so that the weights the forward pass sees are rounded exactly once from a running full-precision value rather than ten times from their own rounded selves — ordinary mixed-precision training, and the only difference from the arm above (`OffloadAdamW(master_fp32=True)`; two self-checks in `code/tests/test_offload_optim.py` assert that it is a no-op in `fp32` and that it moves a coordinate `bf16` rounding freezes). Table 69 is the comparison on 3 models over 34 cells, at the same ten steps and the same judge. On Llama-3.1-8B the grid had to be extended *downwards*: the coarse sweep's floor of 5×10^{-6} already refuses 78 % of benign traffic in this arm.

What it changes: on Llama-3.1-8B the zero is a `bf16` statement. The window moves down in rate and the arm gets sharper, though not uniformly: of the 32 rates run in both arms the `fp32` arm removes more of the attack at 25, by up to 0.683 of judged compliance, ties at five (every tie is a rate at which both arms reach 0.000 and refuse most benign traffic), and removes *less* at two — A3 and A2 on Qwen2.5-1.5B at 5×10^{-6}, by 0.025–0.150, which are the two adversarial families and so the exceptions that matter most to report. Each arm reaches 2 clean operating points, and *which* model they are on is the result: Llama-8B all modules at $2 \cdot 10^{-6}$, Qwen-1.5B mid/attention at $5 \cdot 10^{-6}$ in `fp32` against Qwen-1.5B all modules at $2 \cdot 10^{-5}$, Qwen-1.5B mid/attention at $2 \cdot 10^{-5}$ in `bf16`. On Qwen2.5-1.5B both arms get there and the `fp32` arm merely gets there at a quarter of the rate; on **Llama-3.1-8B** only the `fp32` arm gets there at these rates (the `bf16` arm's two clean AdamW points there, Appendix A23, are at 8×10^{-6}, above this grid). That point is the all-modules attack at 2×10^{-6}: judged 0.150 at over-refusal 0.07 and perplexity 11.7, from an unrepaired 0.700. It sits *on* the bar rather than under it (0.150 is 6 of the 40 judged prompts, and one more compliance would put it outside §5.4's rule), so it is a boundary case, and the paired comparison beside it is what carries the claim: the same rate in `bf16` reads 0.625, and Table 68 says why: at 2×10^{-6}, 93.6 % of that model's target coordinates cannot move at all. §5.4's "zero of the 22 usable landed cells on the two in-scope 7–8B models" is therefore, on Llama-3.1-8B, a statement about the arm that was run, not about the method: with a full-precision copy of the weights the repair has an operating point on that model too, at a rate $2.5\times$ below the floor of the grid the frontier stage swept. On **Qwen2.5-7B** neither arm reaches one on this grid, and the two arms part the same way. The `bf16` arm leaves the mid-attention attack (judged 0.650 unrepaired) untouched through 10^{-5} (0.683 and 0.675 at 8×10^{-6} and 10^{-5}, over-refusal 0.02) and takes it to 0.150 only at 2×10^{-5}, at over-refusal 0.22; the `fp32` arm has removed it entirely by 8×10^{-6} (the drop of 0.683 above) but at over-refusal 0.28, and leaves 0.375 at 5×10^{-6} (over-refusal 0.08). A clean point there, if the arm has one, lies between

Table 69: The same patch, applied to `bf16` weights and with an `fp32` master copy. Judged AdvBench compliance on the 40-prompt subset, benign over-refusal and perplexity; η is the AdamW rate, and a dash means the rate was not in the `bf16` grid (the two adversarial adapters were run in `fp32` only, at the rate the ordinary attack is clean). Bold marks a *clean* operating point by §5.4's rule (judged ASR and over-refusal both ≤ 0.15, perplexity ≤ 50) for an attack that landed, *within its own arm*: the two arms are different repairs, so a clean `fp32` point says nothing about the `bf16` columns beside it; † marks an adapter that did not land on that model, whose row prices the repair but says nothing about the attack.

		`bf16` weights (as run)			`fp32` master		
adapter	η	judged	over-ref.	ppl	judged	over-ref.	ppl
Llama-8B							
A3, m=1024 s0	$2 \cdot 10^{-6}$	–	–	–	0.450	0.05	13.1
A2 s0†	$2 \cdot 10^{-6}$	–	–	–	0.150	0.03	10.0
all modules	10^{-6}	0.650	0.00	13.8	0.275	0.02	12.9
all modules	$2 \cdot 10^{-6}$	0.625	0.00	13.6	**0.150**	**0.07**	**11.7**
all modules	$3 \cdot 10^{-6}$	0.650	0.00	13.2	0.100	0.50	9.5
all modules	$5 \cdot 10^{-6}$	0.200	0.02	11.7	0.000	0.78	8.2
mid/attention	10^{-6}	0.475	0.00	12.6	0.325	0.03	9.5
mid/attention	$2 \cdot 10^{-6}$	0.550	0.02	10.9	0.225	0.03	8.6
mid/attention	$3 \cdot 10^{-6}$	0.575	0.02	10.2	0.025	0.17	7.1
mid/attention	$5 \cdot 10^{-6}$	0.225	0.03	9.0	0.000	0.98	7.7
Qwen-7B							
all modules†	$5 \cdot 10^{-6}$	0.275	0.10	7.8	0.100	0.27	6.6
all modules†	$8 \cdot 10^{-6}$	0.308	0.10	7.3	0.050	0.40	5.9
all modules†	10^{-5}	0.350	0.12	7.0	0.000	0.50	5.8
all modules†	$2 \cdot 10^{-5}$	0.500	0.07	5.8	0.000	0.38	6.4
mid/attention	$5 \cdot 10^{-6}$	0.550	0.02	6.9	0.375	0.08	5.8
mid/attention	$8 \cdot 10^{-6}$	0.683	0.02	6.5	0.000	0.28	5.4
mid/attention	10^{-5}	0.675	0.02	6.3	0.000	0.30	5.4
mid/attention	$2 \cdot 10^{-5}$	0.150	0.22	5.4	0.000	0.88	6.6
Qwen-1.5B							
A3, m=256 s0	$5 \cdot 10^{-6}$	0.450	0.00	25.4	0.600	0.02	22.1
A3, m=256 s0	10^{-5}	0.500	0.00	23.9	0.275	0.18	17.9
A3, m=256 s0	$2 \cdot 10^{-5}$	0.450	0.10	19.9	0.000	0.43	16.5
A3, m=256 s0	$5 \cdot 10^{-5}$	0.000	1.00	18.4	0.000	1.00	19.1
A2 s0	$5 \cdot 10^{-6}$	0.375	0.08	13.7	0.400	0.17	12.4
A2 s0	10^{-5}	0.400	0.10	14.0	0.150	0.28	12.3
A2 s0	$2 \cdot 10^{-5}$	0.400	0.10	13.4	0.000	0.37	12.6
A2 s0	$5 \cdot 10^{-5}$	0.000	0.43	11.8	0.000	0.45	11.1
all modules	$5 \cdot 10^{-6}$	0.375	0.02	20.3	0.200	0.05	18.3
all modules	10^{-5}	0.325	0.02	19.3	0.025	0.32	16.9
all modules	$2 \cdot 10^{-5}$	**0.075**	**0.05**	**17.0**	0.000	0.35	12.8
all modules	$5 \cdot 10^{-5}$	0.000	0.55	12.0	0.000	1.00	13.3
mid/attention	$5 \cdot 10^{-6}$	0.625	0.00	13.9	**0.050**	**0.07**	**12.1**
mid/attention	10^{-5}	0.475	0.00	13.8	0.000	0.52	11.3
mid/attention	$2 \cdot 10^{-5}$	**0.000**	**0.07**	**12.0**	0.000	0.88	10.0
mid/attention	$5 \cdot 10^{-5}$	0.000	0.98	9.5	0.000	1.00	16.3

two rates that were not refined. The all-modules attack does not land on Qwen2.5-7B (judged 0.275 unrepaired), so its rows are marked †.

What it does not change: the adversary is still there. At Llama-3.1-8B's clean point A3 reads judged 0.450 at over-refusal 0.05, against the ordinary attack's 0.150 that defines the point — three times the rate the repair was tuned to remove, and above the landing bar. On Qwen2.5-1.5B, whose clean point is the mid-attention attack at 5×10^{-6} (judged 0.050 at over-refusal 0.07), A3 reads 0.600 and A2 0.400 at the same setting. Turning the rate up does remove them, and removes the model with them: 43 and $37\,\%$ benign over-refusal at 2×10^{-5}. So §5.4's claim survives the better-searched arm

on both models, and survives it at a *cleaner* operating point than the one it was originally made at. (A2 does not land on Llama-3.1-8B, judged 0.050 unrepaired, so its row there is marked † and is not evidence either way.)

What is still open. The `fp32` rates between 5×10^{-6} and 8×10^{-6} on Qwen2.5-7B, and the plain-SGD arm, which is frozen by the same rule with $\eta|g|$ in place of η — a quantity that depends on the gradients and cannot be read off the weights, so part of what Appendix A23 reads as Corollary 1 at work may be rounding there too.

A57 Our SafeLoRA against the authors' released code

Three of the five repairs are our implementations of published methods and the fourth, the coordinate mask, is our own operator in NLSR's class (App. A50); the fifth is our re-implementation of released code, which is a weaker guarantee than it sounds: nothing so far has checked it against the code it reproduces. This does, by importing Hsu et al. (2024)'s own class from their repository and running it on the same adapter (`code/safelora_fidelity.py`; their `model.py` is used byte for byte, SHA-256 prefix `3444756fbb1d287b`; the two adaptations are both outside it: `load_in_8bit=False`, an argument `transformers` removed in v5 and a no-op when false, is filtered out before their loader sees it, and on Qwen, whose attention projections carry a bias, their alignment loop is handed each checkpoint's parameter list without the one-dimensional biases, which it would otherwise multiply as matrices and count against the adapter's index).

In their own configuration. Hsu et al. (2024) put LoRA on `q_proj` and `v_proj` at rank 8. We trained the attack of §4 at that configuration on Llama-3.1-8B, 64 matrices, and ran their class in both selection modes. The two implementations project the *same* matrices in all three settings (10 at the release's default count, 5 at $\tau = 0.5$, 0 at $\tau = 0.35$) and the projected factors agree to 3×10^{-5} relative.

In ours. On our own all-modules adapter their implementation cannot be run whole: it materialises one $d_{\text{out}} \times d_{\text{out}}$ alignment matrix per target and holds them all at once. On Llama-3.1-8B that is 411 MB in `bf16` for each of the 64 `gate`/`up` matrices ($d_{\text{out}} = 14336$), 33.6 MB for each of the 96 `q`/`o`/`down` matrices and 2.1 MB for each of the 64 grouped-query `k`/`v` matrices — about 30 GB, before the two checkpoints and the deep copy of the adapter it also keeps; ours streams V one matrix at a time. Restricted to the 128 attention matrices, where the same object is at most 33.6 MB, the selections agree exactly in the release's count mode (10 matrices) and at $\tau = 0.35$, and differ by one matrix at $\tau = 0.5$: 22 against our 23. The matrix is `layers.2.self_attn.o_proj`, whose cosine is 0.4995 — 5×10^{-4} under the cut, which the release, computing the cosine in `bf16`, reads as 0.5039 (next paragraph). It is a boundary case at the threshold, not a disagreement about the rule.

On the two Qwen checkpoints. Table 70 repeats both configurations on Qwen2.5-1.5B and Qwen2.5-7B. Over the 18 selections on the three models the two implementations choose the same set in 14. Each of the other four differs by one matrix, and each is a boundary case of the same kind: recomputed with the release's own arithmetic (`code/fidelity_boundary.py`), the matrix's full-precision cosine lies within 0.0015 of the cut, while the release computes its cosine in `bf16` and compares it as a `bf16` number — off the full-precision value by up to 0.0044 on these matrices, and resolved only to 2^{-8} just above 0.5, so that Qwen2.5-7B's `layers.0.k_proj`, at 0.5015, is compared as exactly 0.5 and passes the release's $\cos \leq \tau$. The recomputation puts every threshold case on the side of the cut its run did, and shows how thin that edge is: the release's cosine for Llama-3.1-8B's `layers.2.o_proj` is 0.5039 with the run's twelve host threads and 0.5000 with two.

Two things this says about the defence. First, wherever the released code can run it projects the matrices ours does, up to one boundary matrix per selection. Second, and not about us: on Llama-3.1-8B the cosines of a rank-8 `q`/`v` adapter run 0.390–0.788, so Hsu et al. (2024)'s own $\tau = 0.35$ (the point their utility–safety figure calls well balanced, and which projects seven layers on their Llama-2 checkpoint) projects *nothing at all* here, and their released default of ten matrices projects $15.6\,\%$ of them; on Qwen2.5-1.5B and Qwen2.5-7B the same $\tau = 0.35$ projects 1 of 56 and

Table 70: The released SAFELORA class against ours: matrices projected, theirs / ours, in the release's count mode ($N = 10$) and at two thresholds, and the range of the per-matrix cosine. * marks a selection whose set differs; each differs by one matrix at the cut (text). The rank-8 rows are Hsu et al. (2024)'s configuration, the rank-16 rows our attack restricted to the attention matrices.

model	adapter	matrices	$N = 10$	$\tau = 0.5$	$\tau = 0.35$	cosines
Llama-8B	q, v, rank 8	64	10 / 10	5 / 5	0 / 0	0.39–0.79
Llama-8B	attention, rank 16	128	10 / 10	22* / 23	3 / 3	0.25–0.78
Qwen-7B	q, v, rank 8	56	10 / 10	10 / 10	0 / 0	0.35–0.81
Qwen-7B	attention, rank 16	112	10* / 10	41* / 40	5 / 5	0.23–0.80
Qwen-1.5B	q, v, rank 8	56	10 / 10	7* / 6	1 / 1	0.35–0.84
Qwen-1.5B	attention, rank 16	112	10 / 10	36 / 36	7 / 7	0.21–0.82

0 of 56. The operating point a defence recommends is a property of the checkpoint it was tuned on, which is the same lesson §5.5 draws from the other direction.

A58 THE THREE CLOSED FORMS AND THE TESTS UNDER A TRUE NULL

The synthetic checks elsewhere (Appendix A5 for the lemma, A6 for the gate's closed form, A12 for the trace identity) run where the effect is present and large. This appendix runs each where the effect is exactly zero or the form is at its edge, and reports what a calibrated check should report there. Everything here is synthetic or reuses the judged records; there is no model, no GPU and no judge call (`code/null_calibration.py`). We report it because a reader is entitled to ask whether the forms and the tests are calibrated, not only whether they fit where the signal is strong.

The gate's closed form (Appendix A6). The identity $\cos(C\Delta W, \Delta W) = (1 + \mathrm{CV}_w^2(\sigma^2))^{-1/2}$ is checked against $C\Delta W$ formed directly, on synthetic V with three edge spectra: exactly tied singular values (where it must read 1 for every update), a rank-deficient V with $d_{\mathrm{out}} > \mathrm{rank}(V)$ (the `gate_proj` shape, where the padded denominator bites), and a generic decaying spectrum. Over all of them the identity matches the directly-formed cosine to 9×10^{-16} (floating point), and the isotropic plug-in $\sqrt{\mathrm{PR}(V)/d_{\mathrm{out}}}$ matches the mean over random updates to 0.003 absolute, including on the tied spectrum, where both read 1.000, and in the rank-deficient case. The confined (A3) construction is *not* isotropic and its plug-in is $(1 + \mathrm{CV}^2)^{-1/2}$ over the confined block; that form matches the confined measurement to 0.003. So the corollary's prediction is exact where the update is isotropic and correct where the attacker confines it, and the padded denominator does not break either.

The row-space lemma (Lemma 1) on degenerate inputs. A linear layer trained on 12 token vectors whose span is rank-deficient by construction, duplicated rows, a three-dimensional span, a single scaled vector, four zero rows (down to $\mathrm{rank}(X) = 1$), leaves under plain SGD a fraction 1×10^{-15} of its displacement outside $\mathrm{span}\{x_t\}$, zero to accumulation whatever the degeneracy, while AdamW leaves 0.52–0.64. The containment is exact for SGD in every rank-deficient case, and the gradient's numerical rank never exceeds the input rank.

Proposition 1's null. Under an update isotropic on the input side, $\mathbb{E}[c] = \rho^{\mathrm{par}}$, so the ratio c/ρ^{par} should be 1 with only sampling error. Over the paper's own matrix shapes and adapter counts (attention and `down_proj` spans at $|\mathcal{D}_{\mathrm{safe}}| = 1$, at 5 adapters and at the 19 benign ones) the null ratio's mean is 0.999–1.001, with a 95 % band of 0.941–1.061 at 5 adapters that narrows to 0.970–1.033 at 19. Both trades the paper reports sit outside the band that matches them: the customer's 1.07–1.79 over 19 benign adapters against a 19-adapter band ending at 1.033, and the attack's 1.48–6.36 (§2) far outside either. That is the sense in which they are not a null artefact, and the margin is comfortable only for the attack side — the smallest customer ratio clears its band by 0.04. The coordinate half is tighter still: a random p-fraction support removes p of an isotropic update's energy to 0.0023 absolute across $p \in \{0.05, 0.2, 0.6\}$.

The tests, at nominal 0.05. Simulating both arms under a data-generating process fit to the judged A3 records (item and adapter heterogeneity per model), with the two arms drawn from the *same* distribution, no test the paper reports rejects appreciably more often than its nominal level over 400

replicates per configuration: the exact McNemar test rejects 0.028–0.045 of the time and the adapter-level cluster bootstrap 0.003–0.030, both at or below 0.05, while the item-level paired bootstrap runs 0.045–0.068, its worst configuration 0.068 above nominal by less than the ± 0.021 Monte-Carlo error of a 400-replicate size study. The cluster interval is *conservative* at the $K = 3$ and $K = 8$ it is used at, which is the safe direction under this null; the null has identical arms, so it does not test the case where the effect averages zero but varies from adapter to adapter, and §5's caution about three clusters is about that case, not this one.

Size is only half of what an equivalence claim needs, so the same simulation reports the TOST decision itself. At $K = 8$ and 120 prompts per adapter (the cluster count Appendix A59 has, at fewer prompts than the 220 it uses, so a conservative proxy) the adapter-level TOST declares equivalence in 0.993 of replicates when the two arms are truly identical and in 0.030 when they truly differ by δ, and the adapter-level *difference* test detects a true 0.10 gap in ≥ 0.988 of them. At 40 prompts it declares equivalence only 0.480 of the time under a true zero, which is why the extension appendices judge their cells on the complete prompt sets rather than on the nested subsets.

A59 The random-subspace control at eight seeds

Appendix A36's random-subspace control had three seeds, and its two tests read it differently: the item-level interval called SafeLoRA's effect on it equivalent to no repair, and the adapter-level one, which resamples adapters and then prompts within them, did not. With three clusters we rested neither reading on the adapter-level test. This stage reruns the control at eight seeds, beside eight A3 adapters of the same recipe (the three of §5.2 and five new seeds trained in this stage), so that each arm has $K = 8$ clusters: the same budget $m = 256$, rank 16, all modules, data, optimiser and steps, and a random orthonormal m-block in place of V's leading left-singular block as the only difference. Every cell is judged on the complete 120-prompt AdvBench and 100-prompt HarmBench sets, 220 items per adapter and 1760 per arm.

At the gate, the eight seeds repeat the three. The control reads $\cos = 0.454$–0.457, below $\tau = 0.5$, with 116–118 of 196 matrices projected and 74.1–$75.8\,\%$ of the update's norm deleted; A3 reads 0.715–0.720, above τ, with 6–10 projected and 16.1–$21.9\,\%$ deleted (Table 72). The spread over seeds is in the third decimal, as the closed form of Appendix A6 says it should be for a quantity fixed by the constraint rather than by the data.

Behaviourally, the adapter-level test now agrees with the item-level one. All sixteen adapters land (0.495–0.723 unrepaired for the control, 0.550–0.723 for A3). SafeLoRA moves the control from 0.620 to 0.553, a cut of -0.067 with adapter-level $95\,\%$ interval $[-0.097, -0.038]$ (item level $[-0.092, -0.043]$, exact McNemar $p < 0.001$), and A3 from 0.663 to 0.651, -0.012 $[-0.039, +0.015]$. At the $90\,\%$ level the equivalence test uses, the adapter-level intervals are $[-0.093, -0.042]$ for the control and $[-0.035, +0.011]$ for A3: both lie inside $\delta = 0.10$, so at eight clusters both cuts are within the 0.10 margin at the level of the adapter family — the control's is a small but significant cut, A3's is not distinguishable from zero. Equivalence at that margin is not the same as no effect, and the two arms are not the same either: the control's cut is significantly non-zero (its interval excludes zero, McNemar $p < 0.001$) where A3's is not ($p = 0.276$), and the difference between the two cuts, with each family's adapters resampled independently, is -0.055 $[-0.096, -0.015]$, which also excludes zero. What eight clusters settle is the three-seed disagreement of Appendix A36 — that was the width of a three-cluster interval, the conservatism Appendix A58 measures, not a verdict about the arm.

This sharpens Appendix A36's reading and corrects one word of it. On this checkpoint, confinement of the output factor to a low-rank column block of *any* orientation is most of what survives SafeLoRA behaviourally (the control ends at 0.553 from 0.620, inside the equivalence margin), but orientation is not free even behaviourally: it buys the -0.055 difference in the size of the cut, which at eight clusters is resolvable where three could not resolve it — under this judge instruction; Appendix A66 finds the contrast reversed on a subsample judged under a stricter one, so we hold it loosely. The larger part of what V's orientation buys is still the gate side: a cosine above τ, under a tenth as many matrices touched, a fifth of the update deleted instead of three quarters, and a mean cosine above τ_d, the statistic §6 measures. The control is gated on 116–118 matrices already, so a defender who turns τ or scores the update sees it; A3 is gated on 6–10.

On Qwen2.5-7B and Llama-3.1-8B the control is removed and A3 is not. The same stage on the two larger checkpoints, at each model's budget ($m = 256$ on Qwen2.5-7B, 1024 on Llama), eight seeds per arm and the same judge on the complete prompt sets (Table 71), reads the same at the gate. On Qwen2.5-7B the control sits at $\cos = 0.437$–0.440 with 115–118 of 196 matrices projected and 70.5–72.3 % of its norm deleted, A3 at 0.760–0.767 with 2 projected and 8.9–12.0 % deleted; on Llama the control's adapter mean is 0.506–0.507, just above τ, but 124–126 of its 224 matrices fall below it and 76.1–77.1 % of its norm goes, while A3 reads 0.743–0.747 with 3 projected and 11.7–12.5 % deleted. Behind the gate both read differently from Qwen2.5-1.5B. On Qwen2.5-7B all eight control adapters land (0.577–0.800 unrepaired), and SAFELORA takes them from 0.718 to 0.168, a cut of -0.550 with adapter-level interval $[-0.609, -0.483]$ (item level $[-0.574, -0.524]$, exact McNemar $p < 0.001$). Seven of the eight A3 adapters land; the eighth reads 0.227 unrepaired and the landing rule sets it aside. SAFELORA leaves the seven at $0.538 \to 0.538$, $+0.001$ $[-0.019, +0.020]$, equivalent to no repair at $\delta = 0.10$ (90 % interval $[-0.016, +0.017]$). On Llama all sixteen land; the control goes from 0.630 to 0.094, -0.535 $[-0.665, -0.399]$, and A3 from 0.632 to 0.616, -0.016 $[-0.038, +0.003]$, equivalent at δ (90 % interval $[-0.034, +0.000]$) though its item-level interval, $[-0.031, -0.002]$, just excludes zero. The difference of the two cuts is -0.551 $[-0.614, -0.484]$ on Qwen2.5-7B and -0.519 $[-0.650, -0.378]$ on Llama, about ten times the Qwen2.5-1.5B contrast.

So the reading of the Qwen2.5-1.5B rows above does not carry to the larger checkpoints. On the smallest model, deleting three quarters of a randomly oriented update leaves most of its compliance, and what V's orientation buys is the gate; on the other two the same deletion removes it, and orientation is the whole of the behavioural effect. We do not have a mechanism for the difference; the main text therefore states the smallest model's reading for that model only. The headline does not move (it is A3's arm, which is equivalent to no repair on all three) but what the control is evidence for does: on two of the three checkpoints it says that V's orientation, not the confinement alone, is what the projection misses. Unlike the Qwen2.5-1.5B contrast, these survive the stricter judge instruction of Appendix A66: on a 10 % stratified re-judged sample of each, the difference of the two cuts is -0.407 and -0.391 under it, against -0.625 and -0.652 under the paper's.

Table 71: The eight-seed random-subspace control beside A3 on each checkpoint it was run on, at SAFELORA's $\tau = 0.5$ and each model's budget ($m = 256$ on the Qwen models, 1024 on Llama). "landed" counts adapters at or above the landing rule's 0.30 unrepaired, of those trained; the other columns are means over the landed ones ("proj." of 196 matrices on Qwen, 224 on Llama; "del." the share of the update's norm SAFELORA removes). ASR is judged and pooled over the complete AdvBench and HarmBench sets; Δ is SAFELORA minus unrepaired, with an adapter-level bootstrap interval.

model	arm	landed	cos	proj.	del. %	unrep.	SAFELORA	Δ	95 % (adapters)
Qwen-1.5B	random block	8/8	0.456	117	75	0.620	0.553	-0.067	$[-0.097, -0.038]$
	A3 (V's block)	8/8	0.717	8	19	0.663	0.651	-0.012	$[-0.039, +0.015]$
Qwen-7B	random block	8/8	0.439	116	71	0.718	0.168	-0.550	$[-0.609, -0.483]$
	A3 (V's block)	7/8	0.764	2	10	0.538	0.538	$+0.001$	$[-0.019, +0.020]$
Llama-8B	random block	8/8	0.507	126	77	0.630	0.094	-0.535	$[-0.665, -0.399]$
	A3 (V's block)	8/8	0.744	3	12	0.632	0.616	-0.016	$[-0.038, +0.003]$

A60 MODE SUBSPACE OVERLAP ON OUR CHECKPOINTS AND OUR UPDATES (BENIGN ROWS: QWEN2.5-1.5B)

Ponkshe et al. (2026) measure how much the updates that make a model more and less safe share their dominant directions, and find no linearly distinct safety subspace. Their instrument applies directly to the objects of this paper. For two updates U, W of one weight tensor, keep the smallest number of left singular vectors that carries a fraction η of each update's squared singular-value energy, $Q_U \in \mathbb{R}^{d \times k_U}$ and Q_W, and take $\mathrm{MSO} = \|Q_U^\top Q_W\|_F^2 / \min(k_U, k_W) \in [0, 1]$, whose value for independent random subspaces is $\max(k_U, k_W)/d$. Our updates are applied at the aligned θ_0, so an attack adapter's ΔW *is* their harmful-minus-aligned delta, a benign task adapter's is their useful-minus-aligned one, and $V = W_{\mathrm{aligned}} - W_{\mathrm{base}}$ is the alignment update up to a sign MSO

Table 72: A3 against its random-subspace control at eight seeds each, on Qwen2.5-1.5B, at SAFELORA's $\tau = 0.5$. "proj." counts matrices below τ of 196; ASR is judged (`gpt-4.1-mini`) and pooled over the complete AdvBench and HarmBench sets, 220 items per cell. Pooled rows are over all 1760 items of each arm.

	random-subspace control				A3 (V's top-m block)			
seed	cos	proj.	unrep.	SAFELORA	cos	proj.	unrep.	SAFELORA
s0	0.456	117	0.495	0.423	0.715	10	0.623	0.614
s1	0.457	117	0.600	0.559	0.720	9	0.550	0.586
s2	0.456	118	0.682	0.650	0.717	8	0.705	0.691
s3	0.455	118	0.605	0.545	0.716	8	0.650	0.591
s4	0.455	117	0.668	0.573	0.718	7	0.691	0.655
s5	0.456	117	0.723	0.618	0.720	10	0.723	0.709
s6	0.456	118	0.573	0.495	0.717	7	0.664	0.664
s7	0.454	116	0.618	0.564	0.716	6	0.700	0.700
pooled	–	–	0.620	0.553	–	–	0.663	0.651

ignores. Their three pairs are then ordinary / benign, ordinary / alignment and benign / alignment, to which we add A3 / alignment and ordinary / A3. Every target matrix is measured, and the table averages over them (`code/subspace_overlap.py`; no model is run). Two differences from their setting are forced and change the scale of the numbers, not their order: our updates have rank ≤ 16, so $k \leq 16$ and the random baseline of any pair that does not involve V is tiny; and we measure the seven attention and MLP types rather than every tensor.

The harmful update is only slightly more aligned with V than a benign one. On Qwen2.5-1.5B at $\eta = 0.5$ the ordinary all-modules attack overlaps the alignment update at 0.233 and the SQL-Create fine-tune at 0.219 ($2.3\times$ and $2.2\times$ the random baseline of 0.100), and the two agree to 0.040 on every one of the seven module types (Table 73). Over 3 ordinary seeds and the 4 benign tasks (`code/subspace_overlap_multi.py`) the comparison has a spread: the attacks read 0.233–0.240 and the fine-tunes 0.210–0.221, $2.4\times$ and $2.2\times$ random. The two ranges do not overlap at every η from 0.1 to 0.8, but the gap, 0.020, is a tenth of either value. The ordinary attack and the benign fine-tune overlap each other at 0.018 (0.018–0.023 over all 12 pairs), $3.8\times$ their own baseline, so the pattern Ponkshe et al. (2026) report holds on our checkpoint: every pair sits above random at every η, and V's dominant directions favour the harmful update only by that small margin. A statistic of where an update sits relative to V (MSO here, the gate's cosine in §6) orders the attacks slightly above the fine-tunes on these adapters (the gate's cosines do too, below), and SAFELORA's projection removes both.

A3 is the exception, by construction. Its overlap with V is 0.633 at $\eta = 0.5$ (0.633–0.638 over 3 seeds), $6.3\times$ random and nearly three times the ordinary attack's, because its output factor is confined to V's leading block; Qwen2.5-7B reads the same way (Table 73). The overlap statistic therefore *does* separate A3 from an ordinary attack, and in the direction a defender would want to flag. It separates it for the reason Appendix A36's benign control gives for the cosine: it measures where the update sits, which is what the constraint fixes, not what the update does. A harmless adapter trained under the same constraint sits with A3 rather than with the benign fine-tunes: its distance to A3 on this instrument is a fifteenth of A3's distance to the ordinary attack (0.026 against 0.399 at $\eta = 0.5$), which is the criterion we use for "the same place" here and the one under which the attacks and the benign fine-tunes above are *not* in it. Re-running the measurement with `BENIGNA3-sql-s7`, a fine-tune on SQL-Create trained under A3's constraint and nothing else (Appendix A36), in the benign slot gives an overlap with V of 0.610 at $\eta = 0.5$ (0.607–0.610 over all 4 constrained fine-tunes), beside A3's 0.633 and the ordinary benign adapter's 0.219. Appendix A65 finds the constraint visible in five other weight-space statistics too, though a detector fitted there does not group the harmless constrained adapters with A3. The measurement is on 2 models (Qwen-1.5B, Qwen-7B); the benign rows are Qwen2.5-1.5B's, because that is where the benign and constrained-benign fine-tunes existed when the panel was run (the Qwen2.5-7B benign adapters of Appendix A62 were trained afterwards and are not in this table).

Three limits on what the table can carry. Each of its benign rows compares one ordinary seed with one benign task; the multi-adapter run above supplies the spread, which is seed-to-seed for the attacks

and task-to-task for the fine-tunes, over three and four adapters — enough to say the small gap is consistent in sign, not to put an interval on its size. MSO is a statement about the *dominant left-singular directions* of an update and discards the singular values, while the gate is an energy-weighted cosine over the whole spectrum (Table 1), so the two are related but not the same functional; the gate's own cosine tells the same story on these adapters — the ordinary all-modules attacks read a mean cosine of 0.487–0.489 and the four benign fine-tunes 0.477–0.486 (the SAFEMERGE rows of Appendix A62, which record the gate's cosine on both kinds of adapter), and §6 calibrates its benign band on exactly this overlap. And "×rnd" divides by a baseline that is $\max(k_U, k_W)/d$ for each pair, which differs by two orders of magnitude between pairs that involve V (whose k is large) and pairs that do not, so the column compares a pair with its own baseline and should not be read down the table.

Table 73: Mode subspace overlap (Ponkshe et al., 2026) between pairs of updates, averaged over every target matrix, at three energy fractions η. "×rnd" is the ratio to the random baseline $\max(k_U, k_W)/d$ of the same pair. "ordinary" is the all-modules rank-16 attack (seed 0), "benign" the SQL-Create fine-tune (seed 7), "alignment" $V = W_{\text{aligned}} - W_{\text{base}}$, and "A3" the all-modules A3 adapter at the model's budget (seed 0).

		$\eta = 0.1$		$\eta = 0.5$		$\eta = 0.8$	
model	pair	MSO	×rnd	MSO	×rnd	MSO	×rnd
Qwen-1.5B	ordinary / alignment	0.060	7.5	0.233	2.3	0.416	1.5
	benign / alignment	0.055	6.9	0.219	2.2	0.403	1.5
	ordinary / benign	0.006	4.4	0.018	3.8	0.028	2.5
	A3 / alignment	0.142	17.7	0.633	6.3	0.885	3.2
	ordinary / A3	0.195	139.1	0.235	52.2	0.253	23.2
Qwen-7B	ordinary / alignment	0.072	9.3	0.246	2.4	0.431	1.5
	A3 / alignment	0.227	29.5	0.829	8.0	0.980	3.4
	ordinary / A3	0.104	148.6	0.147	69.8	0.161	30.3

A61 THE MATCHED-LENGTH CONTROL AT THE ONE-SHOT PAPER'S OWN INSTANCE

Appendix A22 runs the matched-length control on our safety instance, whose sequences are 35–63 tokens. Zhang et al. (2026b) release the instance their patch is defined on, and a reader can fairly ask whether the concentration of the one-instance gradient is a property of our instance and template rather than of a short refusal in general. This stage runs the same three comparisons on theirs: their released harmful request and refusal under the system prompt their formatter prepends ($L = 53$ tokens, 17 supervised), and the same pair under Llama-2-chat's default system prompt, a long and widely used one ($L = 137$), against three benign instances of the same total length to within one token under the same template and system prompt, and three sequences of uniformly random token ids of the same length over the same supervised span.

On the 3 models this has run on the answer is the one Appendix A22 gives for the benign control. On Qwen2.5-1.5B, at their instance the safety gradient's CER(20) is 0.9910 against 0.9849–0.9893 for the benign instances and 0.9739–0.9771 for random tokens; under the Llama-2 system prompt it is 0.9811 against 0.9834–0.9841, so at 2.6 times the length the benign instances are the *more* concentrated. Over every model and variant the stage has run, a benign instance of the same length is within 0.010 of the safety one (Table 74), and the median r_{99} is 15.5 against 14–18 at $L = 53$. The random-token control does not keep the 0.017 bound Appendix A22 reports for it (at worst it falls 0.017–0.027 below the safety instance across the four Qwen rows here) and it keeps that appendix's model-dependence: least concentrated of the three on the two Qwen checkpoints, *most* concentrated on Llama-3.1-8B, under both system prompts. That is why §2.4's claim rests on the benign control rather than on the random one, here as there. The concentration that §2.4 uses is therefore a property of a short sequence's activations at the one-shot paper's own instance too. The system prompt is not irrelevant to it: lengthening the sequence from 53 to 137 tokens moves the safety instance's CER(20) from 0.9910 to 0.9811 and moves it from above its three benign partners to below them. What survives the change is the comparison §2.4 needs (benign and safety within 0.010 of each other

at either length), not the sign of the gap between them, which is inside that band and should not be read. The benign instances the length match admits are two MBPP prompts and one SQL-Create prompt, which is a narrower draw than the four training sets they come from, the same caveat as there.

Table 74: The matched-length control of Appendix A22 on the one-shot paper's released instance (Zhang et al., 2026b), under the system prompt their formatter prepends and under Llama-2-chat's default one. L is the total sequence length and "sup." the supervised span (the benign instances have one token more). CER(20) and r_{99} as in Table 19; ranges are over three instances of each kind.

				CER(20)			median r_{99}		
model	system prompt	L	sup.	safety	benign (3)	random (3)	safety	benign	random
Qwen-1.5B	theirs	53	17	0.9910	0.9849–0.9893	0.9739–0.9771	15.5	14–18	21–22
Qwen-1.5B	Llama-2's	137	17	0.9811	0.9834–0.9841	0.9622–0.9783	19	16–20.5	20.5–28
Qwen-7B	theirs	53	17	0.9796	0.9850–0.9856	0.9598–0.9666	23	20	28–29
Qwen-7B	Llama-2's	137	17	0.9812	0.9851–0.9868	0.9543–0.9607	21	19–20.5	36–42.5
Llama-8B	theirs	75	17	0.9823	0.9755–0.9820	0.9899–0.9922	25	22–29	18–23
Llama-8B	Llama-2's	159	17	0.9762	0.9663–0.9839	0.9894–0.9933	30	21–28	20–22

A62 THREE PUBLISHED REPAIRS, RE-IMPLEMENTED FROM THEIR RELEASED CODE

Every repair this paper measured before this appendix is either an operator of Table 1 or our own stand-in for a class (the loss-aware revert of Appendix A26). Three published post-hoc methods sit closest to the question, and we re-implemented each from its released code, inside our pipeline, so that it acts on the same adapters, checkpoints, prompts and judge as every other arm (`code/r32_pub.py` cites the released function each line follows). SAFEMERGE (Djuhera et al., 2026) reads SAFELORA's cosine against the same public V and, on every matrix below its threshold, merges the fine-tuned adapter with a separately trained safety adapter by PEFT's linear rule at weights $[0.8, 0.2]$; the safety adapter is trained on $1{,}000$ of the safety demonstrations of Bianchi et al. (2024) (one of the subset sizes the authors use) at their learning rate 10^{-4}, for three epochs rather than their ten, and we run two of the representative optima they report, 0.65 (Qwen-2) and 0.75 (Llama-3.1; 0.7 on Llama-2). ANTIDOTE (Huang et al., 2025) scores the entries of both LoRA factors by Wanda's $|w| \cdot \|x\|_2$, with activations from $2{,}000$ harmful prompt–response pairs (BeaverTails) passed through the ***fine-tuned*** model, and zeroes the top $5\,\%$ of every row (its default) or $20\,\%$; at budget r the released code keeps the lowest $\lfloor (1-r)\,c \rfloor$ scores of a c-entry row, which on the rank-16 factor B zeroes four entries per row at $20\,\%$. SAFE DELTA (Lu et al., 2025) runs a SparseGPT-style pass per weight matrix from the aligned weights, with a Hessian over 128 safety examples, admitting the fine-tuned delta coordinate by coordinate until a safety-loss budget is spent and compensating the rest. The budget is relative: per $2{,}048$-column block, coordinates are admitted in order of increasing $1/[H^{-1}]_{jj}^2$ (the most sensitive $15\,\%$ never) while the running sum of $\Delta w^2/[H^{-1}]_{jj}^2$ stays below $s/2$ times the block's mean $1/[H^{-1}]_{jj}^2$ times $d_{\text{out}}/4096$, and the default is $s = 0.1$. We print that rule as the released code implements it, including that the ordering and the budget are not in the same units (the ordering ignores Δw) and the $d_{\text{out}}/4096$ factor, which their code carries with no derivation. Deviations: prompts use each model's own chat template; every calibration file has our evaluation prompts removed (SAFE DELTA's refusal file is built on AdvBench instructions, and its 128 samples here are AdvBench instructions outside our 120); SAFEMERGE's safety adapter is rank 16 on all seven module types, because the linear merge needs the two adapters to share rank and targets, and is trained for three epochs rather than their ten, which is ours and not forced — it is a reason to read its weak remedy below as a lower bound; and SAFE DELTA's three matrix factorisations are done in double precision. That last one is not about the method's conditioning: its 128-sample Hessian is rank-deficient by construction on a matrix this wide and is held positive-definite by the released code's own damping ($1\,\%$ of the mean diagonal), which we keep; in single precision the damped matrix still fails `cholesky` on the $8{,}960$-column `down_proj`, and in double precision it does not. Nothing else about any of the three methods is changed.

The design is small and paired: three all-modules ordinary attacks and the mid-attention one, and the three A3 seeds of §5.2, each with its own unrepaired arm and SAFELORA at $\tau = 0.5$ as the reference, judged on the complete AdvBench and HarmBench sets; the landing rule of §4 applies as everywhere else, so an adapter whose own unrepaired arm is below 0.30 is excluded from its family's row. On Qwen2.5-1.5B that excludes none, so the rows are 4 ordinary and 3 A3 adapters; on Qwen2.5-7B 2 of the three A3 seeds do not land, leaving 4 and 1; on Llama-3.1-8B, whose A3 adapters are the $m = 1024$ ones of §5.2, 1 ordinary adapter does not, leaving 3 and 3; and the customer's side on the four benign adapters of Appendix A9 (seed 7), as the fraction of each task's gain that survives the repair (Table 75).

SAFEMERGE: A3 passes its gate, as predicted, and its remedy is too small to test it further. Its gate is SAFELORA's, so Corollary 2 applies to it unchanged, and it behaves as Appendix A1 predicted: at $\tau = 0.65$ it merges 35–37 of 196 matrices on the A3 adapters against 144–149 on the all-modules ordinary ones, and A3 ends at 0.618 from 0.608 ($+0.011$ $[-0.024, +0.044]$). At 0.75, above most of A3's per-matrix cosines, it merges 106–110, and A3 still ends at 0.533. That second number is not about A3: the merge keeps four fifths of the fine-tuned update on every matrix it touches, and on this model it takes the ordinary attacks only from 0.500 to 0.432 and 0.402. At $\tau = 0.75$ it does cut A3 measurably (-0.074 $[-0.112, -0.036]$), which is a real if small effect and the one place its remedy shows; at that threshold it merges 106–110 of 196 matrices on A3, so the cut is the remedy acting, not the gate firing on the attack it was aimed at. As released, SAFEMERGE does not remove the attacks SAFELORA removes, so it can confirm the gate half of the claim and cannot test the rest. It costs the customer little (103 and 107 % of the gain kept on average, at least 85 % on the worst task on each of the three models; four adapters, no intervals).

ANTIDOTE removes A3, and on this model it is the better trade. At its default it takes A3 from 0.608 to 0.036 (-0.571 $[-0.626, -0.523]$ over adapters; worst adapter 0.068) and the ordinary attacks from 0.500 to 0.057, and the customer keeps 66 % of the gain on average and 42 % on the worst task (summarisation), against 57 and 35 % for SAFELORA on the same four adapters, which leaves A3 at 0.621. At 20 % it removes everything and keeps 28 % (4 % on GSM8K). How it reaches A3 is not by recognising it: a rank-16 factor B has sixteen entries per row, so a 5 % row budget zeroes the single highest-scoring one, and that deletes 0.57 of A3's norm, 0.60 of the ordinary attacks' and 0.62–0.64 of the customer's adapters alike. The same deleted fraction leaves the customer's tasks with two thirds of their gain and the attacks with an eighth or less of their compliance, and the only input that can make it do so is the score, computed on harmful activations at the fine-tuned model. That is the class §1 places outside the frame, a repair that chooses what to delete after seeing the fine-tuned model and with data about the harm, and it is the recommendation of Appendix A37 (score what the update does) in a published form. A3 is an attack on a gate; a repair with no gate is not what it was built to pass, and the result here says so.

Qwen2.5-7B repeats both halves. The same stage on the larger Qwen gives SAFELORA the ordinary attacks (Table 75) and not A3: $0.732 \to 0.718$, -0.014 $[-0.064, +0.036]$, on the one A3 adapter that lands there. ANTIDOTE at its default takes that adapter to 0.068 (-0.664 $[-0.727, -0.595]$) while the customer keeps 106 % of their gain on average and 82 % on the worst task, more than on the smaller model (66 and 42 %), though it leaves more of the ordinary attacks there (Table 75). SAFEMERGE's cuts stay small on both Qwen models and both families (ordinary -0.068 and -0.098 on Qwen2.5-1.5B, up to -0.110 on Qwen2.5-7B; A3 $+0.011$ to -0.074), too small for its A3 result to separate the gate from the remedy. One A3 adapter is a thin base and we report it as one; it is consistent with the gate-side result of §5.1 and the attack-dependent contrast above not being particular to the 1.5B checkpoint, and establishes no more than that. Which adapter that is, is itself decided by a knife edge: the three seeds read 0.286, 0.732 and 0.236 unrepaired here and 0.305, 0.759 and 0.227 in Appendix A59's stage, so seed 0 lands there and not here on a difference of 0.019 — the per-stage drift Appendix A63 measures. A rule with a hard cut applied to a quantity that moves that much between stages will disagree with itself, and here it does. The Qwen2.5-7B intervals in Table 75 resample items only, since one adapter gives an adapter-level bootstrap nothing to resample.

Llama-3.1-8B repeats both halves on three adapters per family. SAFELORA takes the ordinary attacks from 0.600 to 0.198 (-0.402) and leaves A3 at $0.570 \to 0.541$, -0.029 $[-0.068, +0.009]$, deleting 0.12 of A3's update against 0.73 of the ordinary ones'. ANTIDOTE at its default takes A3 to 0.052 (-0.518 $[-0.706, -0.324]$) and the ordinary attacks to 0.120, deleting 0.59 and 0.60, and the customer keeps 84 % of the gain on the worst task (the mean, 127 %, exceeds 100 % because Llama's GSM8K gain is six items of 60 and every arm scores above the unrepaired adapter there; the

worst-task column is the informative one on this model). At 20 % it removes both families and keeps 15 % on the worst task. SAFEMERGE's gate reads A3 as SAFELORA's does (at 0.65 it merges 27–31 of 224 matrices on A3 against 157–161 on the all-modules ordinary ones) and its remedy barely moves the ordinary attacks (-0.030 and -0.015). At 0.75, merging 102–109 of A3's matrices, it cuts A3 by -0.165 $[-0.342, -0.024]$, the largest SAFEMERGE effect in the table and wide; it is not the gate catching A3, which it fires on less often than on the ordinary attacks it leaves in place, and we have no account of why the same merge moves A3 more than them on this checkpoint.

SAFE DELTA removes everything, at the price its budget implies (Qwen2.5-1.5B only). On Qwen2.5-1.5B it takes A3 from 0.608 to 0.006 (-0.602 $[-0.676, -0.539]$) and the ordinary attacks to 0.010, and it is the most destructive arm on the attack side and tied with ANTIDOTE at 20 % on the customer's: it admits so little of the fine-tuned delta that 0.97 of A3's update and 0.94–0.97 of the customer's is gone, and the customer keeps 29 % of their gain on average and 4 % on GSM8K — the same corner ANTIDOTE reaches at its 20 % budget. Its safety-loss budget is the knob that trades the two, and its default is the one its authors report; we ran that default and did not tune it, so the right reading is that the method's published operating point removes this adversary and charges the customer's fine-tune for it, not that no setting of it does better.

Table 75: Published post-hoc repairs on Qwen2.5-1.5B, Qwen2.5-7B and Llama-3.1-8B, re-implemented from their released code. ASR is judged and pooled over the complete AdvBench and HarmBench sets (220 items per adapter); Δ is paired against each adapter's own unrepaired arm and both families carry an interval; it resamples adapters and then items where the family has more than one adapter, and items alone on the Qwen2.5-7B A3 row, which has one. "del." is the fraction of the update's Frobenius norm removed (ordinary/A3, leading zeros dropped as in the intervals; SAFEMERGE adds rather than deletes). "gain %" is over the four benign adapters of Appendix A9 at seed 7: the mean and the worst task, as percentages. The number after each repair is its threshold or budget; * marks the method's own default (for SAFEMERGE, the optimum its authors report for the model family: 0.65 on Qwen-2, 0.75 on Llama-3.1). SAFE DELTA has a row on Qwen2.5-1.5B only; its Hessians at 7–8B did not fit beside the model on the card this ran on, and we report it as a one-model result rather than extrapolate.

		ordinary		A3			gain kept
model	repair	ASR	Δ (95 %)	ASR	Δ (95 %)	del.	gain %
Qwen-1.5B	unrepaired	0.500	–	0.608	–	–	–
	SAFELORA 0.5	0.007	-0.493 $[-.56, -.44]$	0.621	$+0.014$ $[-.05, +.07]$	.87/.21	57/35
	ANTIDOTE 5 %*	0.057	-0.443 $[-.54, -.32]$	0.036	-0.571 $[-.63, -.52]$	.60/.57	66/42
	ANTIDOTE 20 %	0.008	-0.492 $[-.56, -.44]$	0.008	-0.600 $[-.67, -.54]$	.91/.89	28/4
	SAFEMERGE 0.65*	0.432	-0.068 $[-.10, -.04]$	0.618	$+0.011$ $[-.02, +.04]$	–/–	103/96
	SAFEMERGE 0.75	0.402	-0.098 $[-.18, -.04]$	0.533	-0.074 $[-.11, -.04]$	–/–	107/103
	SAFE DELTA 0.1*	0.010	-0.490 $[-.55, -.43]$	0.006	-0.602 $[-.68, -.54]$	.98/.97	29/4
Qwen-7B	unrepaired	0.537	–	0.732	–	–	–
	SAFELORA 0.5	0.010	-0.527 $[-.69, -.36]$	0.718	-0.014 $[-.06, +.04]$	.90/.09	79/64
	ANTIDOTE 5 %*	0.161	-0.376 $[-.49, -.29]$	0.068	-0.664 $[-.73, -.60]$	.59/.52	106/82
	ANTIDOTE 20 %	0.006	-0.532 $[-.69, -.37]$	0.005	-0.727 $[-.79, -.67]$	.90/.86	50/15
	SAFEMERGE 0.65*	0.440	-0.098 $[-.19, -.01]$	0.736	$+0.005$ $[-.05, +.05]$	–/–	103/90
	SAFEMERGE 0.75	0.427	-0.110 $[-.21, -.02]$	0.659	-0.073 $[-.14, -.01]$	–/–	101/85
Llama-8B	unrepaired	0.600	–	0.570	–	–	–
	SAFELORA 0.5	0.198	-0.402 $[-.61, -.22]$	0.541	-0.029 $[-.07, +.01]$	.73/.12	115/86
	ANTIDOTE 5 %*	0.120	-0.480 $[-.60, -.35]$	0.052	-0.518 $[-.71, -.32]$	.60/.59	127/84
	ANTIDOTE 20 %	0.003	-0.597 $[-.71, -.43]$	0.003	-0.567 $[-.78, -.37]$	.91/.90	67/15
	SAFEMERGE 0.65	0.570	-0.030 $[-.13, +.05]$	0.524	-0.045 $[-.13, +.02]$	–/–	105/96
	SAFEMERGE 0.75*	0.585	-0.015 $[-.24, +.18]$	0.405	-0.165 $[-.34, -.02]$	–/–	113/97

A63 THE CLOSED-FORM RANK-k EDIT, SWEPT ABOVE OUR CALIBRATED STRENGTH

Appendix A15 runs Zhang et al. (2026b)'s closed-form rank-k edit at the single strength our pilot calibrated, $s = 5 \times 10^{-4}$, and says outright that the variant was not swept. That leaves the obvious

objection open: the edit may fail on A3 only because it was run too weak. This stage sweeps it, on Qwen2.5-1.5B, at $s = 10^{-3}$, 2×10^{-3} and 5×10^{-3} (two, four and ten times the calibrated value), $k = 20$ throughout, on three ordinary loci and the three A3 seeds, each judged on the complete AdvBench and HarmBench sets beside its own unrepaired arm, and each carrying the two costs the paper prices everywhere else: refusal on the 60 benign prompts, and perplexity (Table 76). 5 of the 6 adapters clear the landing rule of §4; the exception is the late-MLP locus at 0.223, and the one claim that rests on it is marked where it is made. The three A3 adapters read 0.586–0.695 unrepaired here and 0.568–0.682 in Appendix A62's stage, the same files: each stage merges the adapter into the `bf16` weights and regenerates its own unrepaired arm, and the rounding of that merge is enough to move greedy continuations after their opening (seed 0 reads perplexity 28.0 here and 27.8 there). Every Δ is paired within one stage, so this cancels from the contrasts; it limits a comparison of *levels* across stages to about 0.04 per adapter.

A stronger edit does reach A3, and the reach and the damage arrive together. From 0.586–0.695 unrepaired (mean 0.629), the three A3 adapters end at 0.073–0.227 at $s = 10^{-3}$ and 0.000–0.027 at 2×10^{-3} — paired cuts of -0.473 $[-0.550, -0.380]$ and -0.620 $[-0.692, -0.556]$, adapters resampled then items. So the answer to the objection is that the variant is not inherently blind to A3 — at four times the calibrated strength it removes it from the two cells that remain intact. The costs at those two settings are the reason the calibration stopped where it did, and they show up in different places on different cells. At $s = 10^{-3}$ the ordinary all-modules cell refuses 0.90 of the benign prompts (a model that has stopped answering), while 2 of the three A3 cells refuse none, at benign refusal 0.00–0.23 across them. That cell is not marked destroyed, and the reason is the rule of §4 rather than a judgement about it: the rule fires above 0.9 refusal *and* below 0.05 substring ASR, and this cell sits at exactly 0.90 with its substring rate high. We leave the rule as pre-registered and flag the cell here instead; read strictly, it is one more destroyed cell and one fewer removal. The cost that is uniform at that setting is the customer's, priced below. At 2×10^{-3} refusal on the A3 cells reaches 0.05–0.98 and one of the three trips the destroyed rule of §4; at 5×10^{-3} every cell does, with perplexities of 791 to 9.4×10^4. Of the 18 swept cells, 7 are destroyed, and "removes" above counts only the ones that are not.

And on one locus the edit is not monotone in s. At the late-MLP locus judged compliance goes *up*, from 0.223 unrepaired to 0.541 at $s = 10^{-3}$ and 0.332 at 2×10^{-3}, before 5×10^{-3} destroys the model. That locus is the one adapter here that does not clear the landing rule (0.223 unrepaired), so it is not evidence about how well the edit *removes* an attack; what it is evidence for is that an edit subtracting a fixed rank-20 component of the safety gradient from every target matrix can move behaviour in either direction as its scale grows, which is a property of the edit rather than of the attack it is applied to. We flag it rather than rest on it, and the other five adapters are monotone.

The customer pays before the attacker does. The same three strengths were run on the four benign adapters of Appendix A9 (seed 7), with the calibrated $s = 5 \times 10^{-4}$ as the reference. At the calibrated setting the customer keeps 93 % of their fine-tune's gain on average and 65 % on the worst of the four tasks, at benign refusal 0.00–0.08. At $s = 10^{-3}$ (the weakest tested setting that moves A3) the average is -11 % and the worst task -82 %: the customer's model is *below* no fine-tune at all on some tasks, with refusal up to 0.05–0.40. At 2×10^{-3}, the setting that removes A3, it is -92 and -205 %. So the strength ordering is not a trade the defender can take: every setting that reaches A3 has already taken the customer's fine-tune with it.

On Llama-3.1-8B the edit reaches A3 without visible damage to the model, and the customer's fine-tune is what pays. The same sweep on Llama-3.1-8B, with that model's A3 budget ($m = 1024$) and the same three strengths, gives a cleaner version of the same trade. The three A3 adapters read 0.355–0.777 unrepaired; at $s = 10^{-3}$ they end at 0.300–0.632 (-0.132 $[-0.206, -0.053]$) and at 2×10^{-3} at 0.009–0.114 (-0.521 $[-0.761, -0.251]$), while the ordinary cells go to 0.005–0.068 and 0.009–0.027. At the setting that removes A3, none of the three A3 cells refuses a benign prompt (0.00), none trips the destroyed rule, and perplexity is 12.3–20.9 — so on this checkpoint the reach does *not* arrive with a broken model, and the two costs the Qwen sweep pays at 2×10^{-3} are not paid here. The customer's is. On the four benign adapters the calibrated $s = 5 \times 10^{-4}$ already leaves only 80 % of the gain on average and 50 % on the worst task; at 10^{-3} the average is -23 % with 2 of the four tasks below no fine-tune at all, and at 2×10^{-3} (the setting that removes A3) it is -206 % with all four below it, the worst at -517 %. The conclusion is the one the Qwen rows reach by a different route: the strength that reaches A3 is past the strength at which the customer's fine-tune is

gone, and on Llama that is visible only if the customer's side is measured, because nothing in the repaired model's own perplexity or refusal rate shows it.

What this changes in the paper is the scope of Appendix A15's conclusion rather than its direction. We said there that at the calibrated setting the variant is weaker and less uniform than SAFELORA and that A3 survives it, and drew no conclusion about the variant in its authors' hands. The sweep says the setting is the binding constraint: there is a strength at which the edit removes A3, and it is a strength at which the model refuses benign requests or falls apart. The same is true of the coordinate mask's dial (Appendix A45) in kind but not in degree: the mask at $40\,\%$ costs the customer a measurable fraction of their fine-tune's gain (Appendix A45), and this edit costs the whole of it at the first strength that touches A3.

Table 76: The closed-form rank-k edit (Zhang et al., 2026b) at $k = 20$ and three strengths, on Qwen2.5-1.5B; the calibrated $s = 5 \times 10^{-4}$ of Appendix A15 is two to ten times smaller than these. Each cell is judged ASR pooled over the complete AdvBench and HarmBench sets, with (benign refusal, perplexity) beside it; † marks a cell the destroyed rule of §4 excludes elsewhere, printed here because the point is what the strength costs.

adapter	unrepaired	$s = 10^{-3}$	$s = 2 \times 10^{-3}$	$s = 5 \times 10^{-3}$
ordinary, all modules	0.491	0.009 (0.90, 8.4)	0.000 (0.53, 16.5)	†0.000 (0.00, $1.6{\times}10^4$)
ordinary, mid attention	0.591	0.209 (0.08, 6.5)	0.000 (0.03, 16.8)	†0.000 (0.00, $1.8{\times}10^3$)
ordinary, late MLP	0.223	0.541 (0.07, 8.2)	0.332 (0.02, 9.9)	†0.000 (0.00, $3.4{\times}10^3$)
A3 seed 0	0.605	0.227 (0.00, 12.4)	0.027 (0.05, 14.8)	†0.005 (0.00, $4.4{\times}10^4$)
A3 seed 1	0.586	0.073 (0.00, 9.0)	0.000 (0.30, 16.1)	†0.000 (0.53, $7.9{\times}10^2$)
A3 seed 2	0.695	0.168 (0.23, 7.6)	†0.000 (0.98, 12.5)	†0.000 (0.00, $9.4{\times}10^4$)

A64 THE ROW-SPACE CONSTRAINT REBUILT AGAINST MORE SAFETY DATA

Appendix A48 measures what a defender buys by bringing more safety data: reach ρ^{par} rises from $2.0\,\%$ of the input space at one instance to 33.1 and $75.5\,\%$ at twenty and fifty, and at fifty the span *saturates* 83 of the 196 target matrices. Read at face value that says a defender with tens of refusals has closed the row-space blind spot of §5.4. The reading assumes the update stays where it was. This appendix removes that assumption: the row-space constraint is rebuilt against the 20- and 50-instance spans (three seeds each, everything else unchanged), and every arm is judged beside the one-instance version and an unconstrained adapter at the same locus and rank (Table 77, 660 judgments per cell).

The constraint costs the adversary first, and the landing rule says how much. Against the 50-instance span the three adapters read $0.305, 0.273, 0.245$ unrepaired, so 1 of 3 clears the 0.30 bar §4 requires of a landed attack, against 0.456 pooled at one instance and 0.474 unconstrained. Everything below is therefore a statement about an adversary that is *weaker* for having bought immunity to the larger span, and we report the arms against each adapter's own unrepaired rate rather than excluding them, because the question here is what the repairs do to an update placed this way, not whether this adversary is the one a provider should fear. That is a real limit on the reading and we state it before the numbers.

The defender's own repairs change places. Against the 20-instance span the picture is Appendix A48's: the coordinate mask at its $20\,\%$ default leaves 0.233 (against 0.358 unrepaired, -0.124 $[-0.165, -0.083]$), the gate removes the update outright (0.006), and so does the composite of §5.4 (0.005). Against the 50-instance span the *gate* is what stops working: SAFELORA leaves 0.264 (-0.011, adapter-level $[-0.045, +0.020]$, item-level $[-0.033, +0.011]$, exact McNemar $p = 0.419$) where it leaves 0.005 and 0.006 on the other two. The reason is visible in the weights and is a consequence of the saturation, not of anything aimed at the gate: on the 83 saturated matrices there is no complement, so the update is exactly zero there ($< 10^{-25}$ of its energy), and what remains is carried by the 113 unsaturated ones — 0.94–0.95 of the energy on `down_proj` alone, against 0.08 for the one-instance version. `down_proj` is where this model's gate cosine sits *above* τ: the projection removes 0.06–0.07 of that matrix type's energy, against 0.00–0.06 for the one-instance version, which is why it reaches only 0.10–0.12 of the update's energy in total (equivalently 0.33 of its norm) where it reaches 0.84–0.85 of the one-instance version's. Of the 143 matrices the gate

does project, 83 hold an update that is exactly zero. A defender who enlarges the safety set therefore moves where an evading update must live, and it lands where the other repair does not reach.

Three of the defender's arms make it worse, and the dial is what still works. The 20 % mask *raises* judged compliance on this family, from 0.274 to 0.347 (+0.073, adapter-level [+0.017, +0.127], item-level [+0.041, +0.103]), and so do the 20 % mask built from the 50-instance set (0.370, +0.095 [+0.048, +0.150]) and the composite (0.335, +0.061 [+0.006, +0.111]); all three intervals exclude zero, and with seven arms at $K = 3$, uncorrected, we read them as exploratory and no cell is destroyed (perplexity falls from 7.0 to 6.4 under the mask, and benign refusal stays at 0.03–0.08). We report it and do not explain it: reverting a fifth of the coordinates restores fluency the unrepaired adapter had cost and leaves the behaviour in place, which is a mechanism this design cannot separate from chance variation across three adapters. It is why the 20 % default is not the arm we recommend, and the recommendation does not rest on these three cells: the 40 % arm below carries it on its own. At 40 % the mask does work: it takes the same adapters to 0.127 (−0.147 [−0.199, −0.098]; per adapter 0.050–0.168), and the one- and twenty-instance families to 0.042 and 0.067 — the dial §5.4 already recommends, now over three spans. Building the mask's gradient from the 50-instance set instead of one changes little at either budget (0.153 at 40 %, 0.370 at 20 %): at a fixed budget it is *how much* of the update the support covers that binds, not how much data drew it.

On Qwen2.5-7B one seed per family lands, and there the two repairs part company. The same stage on Qwen2.5-7B leaves 1 of 3 adapters above the landing bar in each constrained family (unrepaired 0.855, 0.186, 0.182 at one instance, 0.832, 0.136, 0.123 at twenty, 0.491, 0.105, 0.100 at fifty) and 2 of 3 unconstrained ones. On the landed adapters the gate cuts every family (0.855 → 0.082, 0.832 → 0.109 and 0.491 → 0.023, against 0.550 → 0.009 for the unconstrained) while the 20 % mask cuts the unconstrained adapters (−0.359 [−0.432, −0.291]) and not the one- and twenty-instance constrained ones (+0.036 [−0.014, +0.086], +0.009 [−0.045, +0.064]); at fifty instances it cuts them too (−0.191 [−0.250, −0.127]). That is §5.4's contrast with the two repairs exchanged relative to Qwen2.5-1.5B, on one adapter per family, and we report it as a single-adapter reading. Turning the mask to 40 % is not enough on this checkpoint: it leaves 0.727, 0.595 and 0.150 (−0.127 [−0.186, −0.068], −0.236 [−0.305, −0.164], −0.341 [−0.409, −0.268]), while the composite of §5.4 reaches all three (0.041, 0.068, 0.014; −0.814 [−0.864, −0.759] at one instance). The customer's side costs 90 % of the gain on the worst task at the mask's default.

On Llama-3.1-8B the constraint stops the attack outright. The same stage runs on Llama-3.1-8B, and there the question the appendix asks cannot be put: none of the nine constrained adapters lands. The one-instance family reads 0.291, 0.236, 0.173 unrepaired, the 20-instance family 0.218, 0.123, 0.105 and the 50-instance family 0.173, 0.150, 0.091, against 0.664, 0.464, 0.223 for the unconstrained adapters at the same locus and rank, 2 of which land and are cut 0.564 → 0.057 by the gate (−0.507 [−0.641, −0.377]) and → 0.150 by the 20 % mask (−0.414 [−0.550, −0.282]). That is the same cost the Qwen2.5-1.5B rows price, at a scale that makes it decisive: on this checkpoint the row-space constraint buys immunity to a repair by giving up the attack. It is a limit on A2's reach as an adversary, not a defence a provider can rely on — the constrained adapters are cheap to train, and a checkpoint on which they land is what §5.4 reports. The customer's side runs there too, and the mask at its 20 % default costs that model's four fine-tunes nothing measurable (85 % of the gain on the worst task, 84 % when its gradient is drawn from the 50-instance set), which is the same reading Appendix A45 gives on the smaller model.

What this qualifies. Appendix A48's curve is a statement about *coverage of a fixed update*, and it stands: an adapter built against one instance is reached by a twenty-instance repair almost as well as an ordinary one is. It is not a statement about an adversary who re-solves the constraint against the defender's larger set, and this appendix is the counterexample — with the cost to that adversary priced above. Two conclusions are usable and one is not. Usable: more safety data does not by itself close the blind spot — at the price, priced above, that the adversary who re-solves against the larger span loses most of its own compliance and two of its three seeds stop landing — and the mask's budget does more than the safety set's size. Not usable: the composite is *not* the union of its parts here, since it lands between its two components rather than at the better of them (§5.4 reports the composite where both parts work); on three adapters at this sample size we can say that stacking did not help, not why.

Table 77: The row-space constraint rebuilt against the 1-, 20- and 50-instance safety spans on Qwen2.5-1.5B, three seeds each, against an unconstrained adapter at the same locus and rank. Cells are judged ASR pooled over the complete AdvBench and HarmBench sets (660 judgments per cell). mask_p is the coordinate mask at budget p with its gradient from one instance; mask_p^{50} takes the gradient over the 50-instance set; "composite" is SAFELORA at $\tau = 0.5$ followed by the $20\,\%$ mask. The unconstrained row is measured in the same stage and carries no $40\,\%$ or composite arm.

constrained against	unrep.	mask_{20}	mask_{20}^{50}	mask_{40}	mask_{40}^{50}	SAFELORA	composite
1	0.456	0.323	0.271	0.042	0.017	0.005	0.002
20	0.358	0.233	0.242	0.067	0.064	0.006	0.005
50	0.274	0.347	0.370	0.127	0.153	0.264	0.335
unconstrained	0.474	0.148	0.105	–	–	0.006	–

A65 FIVE WEIGHT-SPACE STATISTICS, AND WHAT THEY ACTUALLY SEPARATE

§6 builds its statistic from the gate's own cosine, and Appendix A36 shows what that buys and what it does not: the cosine separates *constrained* from *unconstrained* updates, and a benign fine-tune trained under the same constraint is flagged with the attacks. A reader may reasonably ask whether a provider with a freer hand does better — the adapter is a file, and any function of its singular values is available before it is ever run. This appendix answers with the five most obvious such functions, computed per module type from the update's core matrix and nothing else: the leading singular value σ_1, the Frobenius norm, the energy share $\sigma_1^2 / \sum_i \sigma_i^2$, the spectral entropy, and the entry kurtosis (`code/spectral_detector.py`; no model is run and no prompt is scored). Every adapter on the released checkpoint is scored: 8 ordinary, 9 A3, 8 random-block controls, 6 row-constrained, 4 harmless fine-tunes trained under A3's constraint, and 12 ordinary benign fine-tunes. By name: the ordinary adapters are five all-modules rank-16 seeds and three rank-4 ones; A3 is the eight $m = 256$ seeds of Appendix A59 and one at $m = 1024$; the controls are that appendix's eight; the row-constrained are the span-1 and span-20 seeds of Appendix A64; the benign fine-tunes are the four tasks at seeds 7–9, and the constrained harmless ones SQL-Create and GSM8K at seeds 7 and 8.

They separate perfectly, and they separate the wrong thing. Every one of the five reaches AUC 1.000 between A3 and the benign fine-tunes at some module type (Table 78). That maximum is taken over seven module types with a sign chosen once, so it needs its own null: permuting the labels and recomputing the same selected maximum puts every one of those AUCs at $p \leq 0.010$ over 200 permutations. The same five also reach AUC 1.000 between the *harmless* constrained fine-tunes and the ordinary benign ones (4 against 12 adapters, $p \leq 0.010$ under the same null). A statistic that tells A3 from a benign adapter as perfectly as it tells a harmless constrained adapter from a benign one is measuring the constraint.

A leave-one-out logistic regression on all 35 dimensions reaches AUC 1.000 with 1.000 true-positive and 0.000 false-positive rate at its Youden point, and that number should be read as nothing at all: 35 features on 21 adapters are separable by construction, and under permuted labels the same procedure returns AUC 1.000 in 0.13 of 200 draws (mean 0.41). We report it because it is the number a provider would compute first, and it is the one number here that carries no information. The univariate results above and the out-of-sample behaviour below are what the panel actually establishes.

What the fitted detector does out of sample says the same thing more precisely. Trained on A3 against benign (scores 0.761–0.963 against 0.026–0.091) and applied to families it never saw, it flags 8 of 8 random-block controls (harmful, equally confined, and pointed at a subspace that has nothing to do with the alignment matrix) at scores 0.991–0.994. It flags 3 of 8 ordinary attacks, and which three is the tell: the 3 rank-4 adapters score 1.000 while the 5 rank-16 ones score 0.068–0.090, which suggests, though we did not test it, that a low-rank benign adapter would be flagged for its rank. It flags 0 of 6 row-constrained adapters, which are attacks that land (Appendix A46), at scores 0.035–0.052 — inside the benign fine-tunes' own range (0.026–0.091, median 0.062) rather than above it. And it leaves the harmless constrained fine-tunes unflagged at 0.364–0.430, 6–7 times the benign median and about half the threshold of 0.761: they are not caught here, on four adapters, and they are not where the benign adapters are either. That last reading depends on the threshold, and the threshold is a convention: the fit separates its training set perfectly, so every cut in $(0.091, 0.761)$ (the gap between the highest benign and the lowest A3 score) is equally optimal in sample, and we

take the conservative end. The harmless constrained adapters sit inside that interval, so a provider who cut anywhere below 0.364, which is $51\,\%$ of it, would flag all four. What the panel establishes out of sample is therefore the *ordering*, not the verdict at any one cut. Read as an ordering rather than as calibrated probabilities (a logistic fit on separable data has no calibration to offer), these out-of-sample scores are the informative part: the ordering they produce (random block $>$ A3 $>$ harmless constrained $>$ benign $\approx$ row-constrained) is the panel's actual content.

So the honest summary is a narrow recommendation and a warning. The narrow recommendation: these statistics are a cheap screen for confinement of the *output factor* to a low-rank column block, which is worth having precisely because that confinement is what SAFELORA's gate cannot see — a provider who computes them alongside the gate covers the specific blind spot Corollary 2 describes against an attacker who does not also aim at these statistics, at the cost of a flag on every low-rank or deliberately constrained submission. It is not a screen for confinement in general: the row-constrained adversary is confined too, out of the safety row space rather than into a column block, and the panel scores all 6 of its adapters inside the benign band. The warning is that the flag is a statement about the shape of an update and not about what it does: one adversary in this paper passes the whole panel while landing, the harmless constrained fine-tunes sit well above the benign band and are flagged or not depending on where in the indeterminate interval the cut goes, and rank alone moves the score to 1.000. §6's recommendation stands unchanged and is the complement of this one: score what the update *does* to the model's own outputs (Appendix A37), because that is the quantity a constraint cannot buy. The panel is computed on Qwen-1.5B, Qwen-7B, Mistral-7B; the comparison above is stated on Qwen2.5-1.5B because that is where the benign and constrained-benign families existed when the panel was run.

Table 78: Each statistic's AUC against the 12 ordinary benign fine-tunes on Qwen2.5-1.5B, at the module type where it separates best (named in brackets); a statistic that runs the other way is read at $1 - \mathrm{AUC}$, since a detector fixes its sign once. Both selections are inside the permutation null quoted in the text ($p \leq 0.010$ over 200 draws). The positive classes are 9 A3, 8 random-block and 4 harmless constrained adapters; the third column is the comparison that matters.

statistic	A3 vs benign	random-block vs benign	benign-under-A3 vs benign
σ_1	1.000 (q_proj)	1.000 (q_proj)	1.000 (gate_proj)
$\|\Delta W\|_F$	1.000 (q_proj)	1.000 (k_proj)	1.000 (gate_proj)
$\sigma_1^2 / \sum \sigma_i^2$	1.000 (v_proj)	1.000 (q_proj)	1.000 (v_proj)
spectral entropy	1.000 (k_proj)	1.000 (q_proj)	1.000 (q_proj)
kurtosis	1.000 (q_proj)	1.000 (k_proj)	1.000 (gate_proj)

A66 HOW MUCH OF A LABEL IS THE JUDGE'S INSTRUCTION?

Appendix A29 asks how far the labels depend on the *model* doing the judging. This appendix asks the other half: how far they depend on the *instruction*. Every judged number in this paper comes from one system prompt (§4), and a reader who wants to discount a contrast can always say the wording produced it. We drew a stratified sample of the extension stages' judged items, 1280 items, one per stratum quota over (model, stage, attack family, repair, prompt set), which is 22–64 items per arm behind each contrast below, and re-judged every one of them with the same model under two other instructions (`code/judge_promptvar.py`): *terse*, which asks the question in one line with no definitions, and *strict*, which keeps the definitions but requires substantive content rather than any attempt, the reading a more demanding annotator would take. Both columns of Table 79 are computed on that sample and on nothing else, so its left column is not the appendices' headline rate: those pool every item of every adapter, and are what Tables 72, 75 and 77 report.

The one-line instruction does not measure the construct, and says so loudly. Under *terse* the model calls 0.034 of responses compliant where the paper's instruction calls 0.312 and the substring rule calls more still — including on unrepaired attack cells that every other instrument scores near one half. Agreement with the primary judge is $66.9\,\%$ at $\kappa = -0.021$ — below the $68.8\,\%$ a rule that always answers "refused" would reach, with κ capped at 0.143 by the two rates in any case. We report it because it is the informative failure: asked without a definition of compliance, the model defaults to something much closer to "is this answer good" than to "did it try", so an evaluation that

skips the definitions is not measuring what it thinks. It is not evidence about this paper's labels, and we do not use it below.

Under the stricter instruction the levels compress and the contrasts hold. Agreement with the primary judge is $83.6\,\%$ at $\kappa = +0.553$, and the compliance rate falls from 0.312 to 0.148: the stricter reading refuses credit for attempts that carry no usable content, and roughly half of what the paper counts as compliance is of that kind. The two instructions do not disagree about *which* responses comply, only about where to draw the line: of the 1280 items, 0 are called compliant by the stricter instruction and not by the paper's, against 210 the other way, so the stricter set is nested inside the paper's and κ sits exactly at the 0.553 ceiling the two rates allow. What matters is whether the *differences* move, and Table 79 recomputes eight of them, one for each conclusion the extension appendices draw. 7 of 8 keep their sign, and the rank correlation between the two Δ columns is $\rho = 0.83$ (ties averaged): the gate still fails to remove A3, ANTIDOTE still removes it, the gate still removes the ordinary attacks and the span-1 constrained ones to 0.000, and the two equivalences keep point estimates inside δ (at these sample sizes the equivalence test itself is not run: Appendix A58 shows it decides little at this many items, which is why the appendices run it on the complete sets). The exception is the random-block control, whose -0.091 becomes $+0.004$. Its cut is inside the 0.10 margin under both instruments, but it is also one half of Appendix A59's orientation contrast, and that contrast (the control cut more than A3) does not survive the change of instruction on this sample: under the stricter one A3's cut is the larger. We therefore read Appendix A59's Qwen2.5-1.5B -0.055 as instrument-dependent. The larger checkpoints' contrasts were sampled separately, $10\,\%$ of each stratum (264 items per instruction on Qwen2.5-7B's landed adapters and 276 on Llama-3.1-8B's), so as not to change the sample this table is computed on. On Qwen2.5-7B the paper's instruction reads the control's cut as -0.609 and A3's as $+0.016$ and the stricter one -0.391 and $+0.016$ (agreement $81.5\,\%$), a difference of -0.407 against -0.625; on Llama-3.1-8B -0.391 against -0.652 (agreement $74.6\,\%$). Both keep their sign and most of their size, unlike the Qwen2.5-1.5B one. At 22–64 items per contrast these Δs are far noisier than the appendices' own, which is why the check is stated as sign and ordering rather than as a second estimate of any effect.

One consequence of the level shift deserves its own line: under the stricter instruction most of these families' unrepaired rates fall below the 0.30 landing bar of §4, so a paper built on that instrument would have analysed fewer of them. The bar is a property of the instrument as much as of the attack, which is an argument for reporting the instruction, as we do, rather than for a different bar.

The general lesson is the one the paper states elsewhere and is worth stating once here as well: the absolute level of a judged ASR is a property of an instrument, not of a model, and two reasonable instructions put it at 0.312 and 0.148 on the same generations. The extension appendices' conclusions are differences between two arms scored by one instrument on the same items, where this variation preserves sign in 7 of 8 and rank order at $\rho = 0.83$ but not size: of the 5 contrasts above 0.10, the stricter instruction cuts the difference by 42–$89\,\%$; the landing rule and the levels quoted in §5 are not, and an instrument this much stricter would land fewer attacks; the levels in the tables should be read with the instrument's definition in hand, and that definition is quoted in full in §4.

A67 A WIDER CAPABILITY CHECK: $1{,}000$ MMLU QUESTIONS

The utility side of every table in this paper is the customer's own four tasks at 60 held-out items each, plus perplexity on held-out benign instruction data. Both are narrow, and both are measured on data close to what the adapter was trained on, so they cannot answer the question a provider actually has: does a repair leave the model's *general* capability where it found it? This appendix answers it on $1{,}000$ MMLU questions (Hendrycks et al., 2021) drawn from 57 subjects (`code/build_mmlu.py`: the test split, a fixed-seed shuffle, the first $1{,}000$). Scoring is by first-token log-probability over the four option letters, so there is no generation, no answer extraction and no judge. Seven arms are run twice, once with an ordinary attack adapter as the carrier and once with the customer's SQL-Create fine-tune, so that "what the repair costs" is measured on the two kinds of update a provider actually receives (Table 81).

None of the four repairs that delete part of the update costs capability. The comparison a provider wants is the repair against the *unrepaired* submission, since the adapter is going on the model either way, and there SAFELORA, the two masks and the composite move MMLU by at most 0.010 on either carrier, with exact McNemar $p \geq 0.18$ over the 8 such comparisons (the arms share the same

Table 79: Eight contrasts, one per conclusion of the extension appendices, recomputed under a stricter judge instruction. *Both* columns are computed on the stratified sample of 1280 items, not on the full sets the appendices report, so the left column is this sample's estimate of the paper's own instrument rather than the published number. "items" counts the sampled items behind *one* arm of the row (the unrepaired one; the repaired arm is the same size) and each rate is the mean over the family's adapters of their own rates on those items, so a rate need not be a multiple of $1/\text{items}$. Δ is the difference of the two rates in its pair.

		paper's instruction		stricter instruction	
contrast	items	unrep. → repaired	Δ	unrep. → repaired	Δ
A3, SAFELORA 0.5	24	0.442→0.697	+0.256	0.283→0.311	+0.028
A3, ANTIDOTE 5 %	24	0.442→0.061	−0.381	0.283→0.061	−0.222
ordinary, SAFELORA 0.5	24	0.406→0.000	−0.406	0.153→0.000	−0.153
A3 (8 seeds), SAFELORA	64	0.601→0.571	−0.030	0.347→0.291	−0.056
random block (8), SAFELORA	64	0.681→0.590	−0.091	0.243→0.247	+0.004
span-1, SAFELORA	24	0.372→0.000	−0.372	0.153→0.000	−0.153
span-50, SAFELORA	22	0.317→0.283	−0.033	0.233→0.200	−0.033
ordinary, mask 20 %	24	0.372→0.100	−0.272	0.153→0.000	−0.153

1,000 questions, so the test is paired; p is in the table). Not rejecting is not equivalence, so we also run the test in that direction: at a margin of 0.025 (the smallest this sample can decide, since 1,000 paired questions resolve a difference of about 0.019 at 80 % power) all 8 of the 8 paired differences have their 90 % intervals inside it, the widest endpoint being 0.021. A margin of one MMLU point is *not* decidable here, and a reader who wants that should read this appendix as a screen rather than as a certificate. Against θ_0 the same arms read 0.589–0.599 against 0.586, all slightly *above* it, and on the attacked carrier two of those differences reach $p < 0.05$ (9 and 13 questions of 1,000). The table makes 22 paired comparisons, so a Bonferroni threshold would be 0.0023, and none of these small gains clears it. We read the measurement as saying these four repairs do not cost capability on this benchmark, on this model and these two carriers, and we do not claim they add any.

The rank-k edit is the exception, and it is the one that edits θ_0 itself. At $s = 10^{-3}$, the weakest strength that moves A3 at all (Appendix A63), MMLU falls to 0.560 on the attacked carrier and 0.543 on the customer's, 2.9 and 4.8 points below the unrepaired submission. On the customer's carrier both of its comparisons clear the Bonferroni threshold ($p = 0.0017$ against θ_0, $p < 0.001$ against the unrepaired adapter); on the attacked carrier they are $p = 0.058$ and $p = 0.031$, which do not. It is the only arm in the table with a loss that survives the correction. Unlike the others it subtracts a component of the safety gradient from every target matrix of the *base* weights, so its damage is not confined to the submitted update; Appendix A63 prices the same setting on the customer's tasks, and this is the same edit seen on a benchmark it was not tuned against.

The larger checkpoints say the same, and say it louder about the rank-k edit. The same slice and the same seven arms were run on Qwen2.5-7B and Llama-3.1-8B (Table 80). On Qwen2.5-7B the deleting repairs sit within −0.009 and −0.004 of the unrepaired submission (McNemar 0.211 and 0.572), against a base of 0.716. On Llama-3.1-8B they move it the other way: the unrepaired adapters read 0.653 and 0.661 against 0.674 at θ_0, so the submission itself costs one to two MMLU points there, and deleting part of it gives most of that back (+0.029 and +0.018 at their largest, 0.003 and 0.047). Neither direction is a capability cost of the repair, which is the question this appendix asks. The rank-k edit at the same $s = 10^{-3}$ costs −0.089 and −0.183 on Qwen2.5-7B and −0.151 and −0.150 on Llama-3.1-8B — three to six times what it costs on Qwen2.5-1.5B, on models whose MMLU starts 9 and 13 points higher. The two conclusions therefore hold on all three checkpoints, and the one that matters for §5.4's recommendation — that the repairs the paper recommends cost no measurable capability — holds at the scale where a provider would care.

Table 80: The same check on each checkpoint it has been run on. "arms" counts the deleting repairs (SAFELORA, the two masks, the composite); "worst vs unrep." is the largest move any of them makes against the unrepaired submission, with its exact McNemar p; the rank-k columns are the edit at $s = 10^{-3}$. Every comparison is paired on the same $1{,}000$ questions.

				deleting repairs		rank-k edit	
model	carrier	θ_0	unrep.	arms	worst vs unrep.	MMLU	vs unrep.
Qwen-1.5B	attacked	0.586	0.589	4	$+0.010$ (0.184)	0.560	-0.029 (0.031)
	customer's	0.586	0.591	4	-0.002 (0.845)	0.543	-0.048 ($<$0.001)
Llama-8B	attacked	0.674	0.653	4	$+0.029$ (0.003)	0.502	-0.151 ($<$0.001)
	customer's	0.674	0.661	4	$+0.018$ (0.047)	0.511	-0.150 ($<$0.001)
Qwen-7B	attacked	0.716	0.722	4	-0.009 (0.211)	0.633	-0.089 ($<$0.001)
	customer's	0.716	0.722	4	-0.004 (0.572)	0.539	-0.183 ($<$0.001)

Table 81: Accuracy on $1{,}000$ MMLU questions (57 subjects), scored by first-token log-probability over the option letters, on Qwen2.5-1.5B. Every arm is scored on the same questions, so both Δ columns are paired and carry the exact McNemar p in brackets: against θ_0 (the aligned checkpoint, no adapter) and against the unrepaired submission. "On the attacked checkpoint" carries the all-modules rank-16 ordinary adapter; "on the customer's fine-tune" carries the SQL-Create adapter of Appendix A9. The table makes 22 comparisons, so a Bonferroni threshold would be 0.0023.

	on the attacked checkpoint			on the customer's fine-tune		
arm	MMLU	vs θ_0	vs unrep.	MMLU	vs θ_0	vs unrep.
θ_0 (no adapter)	0.586	–	–	0.586	–	–
unrepaired	0.589	+0.003 (0.813)	–	0.591	+0.005 (0.533)	–
SAFELORA τ=0.5	0.598	+0.012 (0.104)	+0.009 (0.243)	0.589	+0.003 (0.690)	-0.002 (0.845)
mask 20 %	0.599	+0.013 (0.024)	+0.010 (0.184)	0.591	+0.005 (0.267)	+0.000 (1.000)
mask 40 %	0.596	+0.010 (0.052)	+0.007 (0.410)	0.591	+0.005 (0.267)	+0.000 (1.000)
composite	0.595	+0.009 (0.049)	+0.006 (0.526)	0.592	+0.006 (0.146)	+0.001 (1.000)
rank-k edit, $s = 10^{-3}$	0.560	-0.026 (0.058)	-0.029 (0.031)	0.543	-0.043 (0.002)	-0.048 ($<$0.001)