CONFIDENCE INFLATION FROM RL POST-TRAINING IN VIDEO LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement learning with verifiable rewards (RLVR) improves answer correctness in video multimodal LLMs, but what it does to confidence is unmeasured. We compare two checkpoints item by item based on the model's option-letter distribution, which serves as the confidence readout, and limit the analysis to items that both answer incorrectly so that confidence changes are separated from changes in correctness. On that set, RL post-training systematically increases confidence on unchanged wrong answers, and we call this excess the calibration tax. The magnitude of the calibration tax varies across question types in a pattern that is associated with the post-training recipe and is not explained by item difficulty, and this effect replicates across three benchmarks and three model scales. At the item level, the tax appears as a scalar change in the readout that leaves the predicted option unchanged but can substantially alter the numerical confidence assigned to it. At a fixed confidence cutoff of 0.8, items whose predictions did not change are accepted for automatic handling 59% more often, and the error rate among them rises from 27.6% to 36.2%. We then ask where the tax can be controlled. Reward-side interventions can move the readout, but their effect is unstable across training runs. Because the tax also varies from item to item, a single global post-hoc correction removes only part of it, which motivates controlling confidence directly during RL. Our method, **Reference Margin Anchoring (RMA)**, adds a two-sided penalty during RL that anchors the readout margin to the frozen reference already used by the KL term. In a 1200-step run, it removes 95% of the RL-induced calibration tax, at a cost of 0.4 accuracy points that fall within run-to-run variation, and returns the threshold curves close to their pre-RL level. Our results show that RL post-training can introduce substantial confidence inflation beyond its effect on answer correctness and that this increment can be controlled directly at the readout.

1 INTRODUCTION

Reinforcement learning with verifiable rewards (RLVR) (Lambert et al., 2024) is now the standard final stage for video multimodal LLMs. It optimizes a binary correctness signal, $R_{\text{correctness}}(y, y^*) = \mathbf{1}[y \equiv y^*]$, where $\equiv$ denotes the verifier's judgment that the model's output y counts as the ground-truth answer y^* (Damani et al., 2025). In our multiple-choice setting, that means agreement on the selected option. The signal pays the same whether a model is confidently right or merely lucky, and penalizes a confident error exactly the same as it penalizes an honest one.

In the text domain, RLVR-trained reasoning models are known to become overconfident (Bereket & Leskovec, 2025), and calibration-aware objectives have been proposed in response (Damani et al., 2025; Wang et al., 2026; Ma et al., 2026). This matters when a system sends its uncertain cases to a human through a fixed confidence threshold, since after post-training the same threshold lets more wrong answers pass.

Whether the same happens in video, where the reward is usually *shaped* and pays extra for temporal order or grounded spatial evidence, has not been measured. One reason is methodological, since an absolute calibration metric confounds what post-training did with what the checkpoint already carried.

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

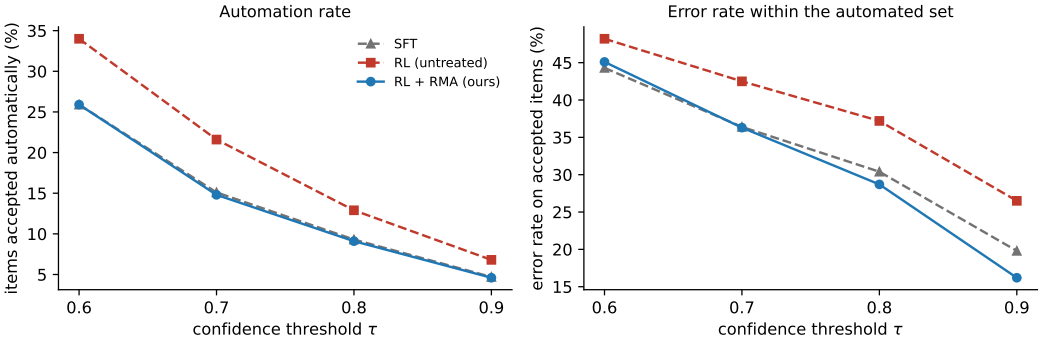

Figure 1: The cost of video RL post-training and its removal on HERBench (Qwen2.5-VL-7B trained by us with the Video-R1 recipe for 1200 steps, 224 px), on the 1728 items where the SFT, untreated RL and RMA checkpoints emit the same prediction, so accuracy is identical (0.364) and every difference is confidence alone. Post-training pushes more items past any fixed threshold (left) and makes a larger fraction of those wrong (right). RMA (§3.2) returns both curves to the SFT curve.

This work began with three questions. **(1)** Does video RLVR inflate confidence on the answers it gets wrong, and if so, where does the inflation concentrate? **(2)** Is the inflation a change in which option the model prefers, or only in how strongly it prefers it? **(3)** Can this confidence inflation be removed without giving back the accuracy gained from the reward?

We address the first question with an item-matched estimator that measures the change in confidence on examples that both checkpoints answer incorrectly. We call the phenomenon confidence inflation and the quantity this estimator measures the calibration tax. We address the second by decomposing the change in the option-logit readout into a scalar rescaling and a residual component. This decomposition identifies the readout margin as a natural target for controlling confidence inflation. We address the third with two corrections built on this structure in Section 3, and Section 4 shows that they remove most of the tax while keeping accuracy within run-to-run variation.

Our contributions are as follows.

- **A paired characterization of RL-induced confidence inflation**: We introduce an item-matched analysis that isolates the change in confidence induced by RL post-training on examples that both checkpoints answer incorrectly. We find that RL systematically increases confidence on these shared failures, usually without changing the predicted answer (88% of them). Much of this increase is explained by an item-specific rescaling of the option-logit readout, with one scaling factor per item reproducing 72% of the measured increase at 300 steps. (§2)

- **Two methods for correcting RL-induced confidence inflation**: Both follow from the rescaling structure. Our method Reference Margin Anchoring (RMA) constrains the readout margin relative to the reference model during RL training and removes 95% of the tax measured at the letter readout in a 1200-step run, while preserving accuracy within observed run-to-run variation. When retraining is not possible, a label-free post-training correction folds a mean rescaling into the output embedding and removes 68% of the tax with predictions unchanged in exact arithmetic. (§3)

- **A systematic study of where confidence inflation can be controlled**: We evaluate interventions in the reward and on the post-hoc logits. They produce limited or inconsistent reductions, which identifies the readout margin as the most effective point of intervention among those we tested. Reward-side penalties also produce substantial run-to-run variability, so we explicitly account for replication variability in their evaluation, and fixing the rollout sampling seed removes most of the variability between untreated runs. (§4)

2 DIAGNOSIS

2.1 SETUP AND PAIRED ESTIMATOR

Models. Four checkpoints share one initialization, Qwen2.5-VL-7B-Instruct (Qwen Team, 2025) (INIT). Video-R1 (Feng et al., 2025a) applies CoT-SFT (SFT), then T-GRPO RL (RL), and we use the released checkpoints of both stages. Open-o3-Video (Meng et al., 2025) (OPEN-O3) starts from the same initialization but applies an independent two-stage SFT then GSPO (Zheng et al., 2025) with grounding supervision over timestamps and bounding boxes.

Benchmarks. We use three multiple-choice video-QA benchmarks. HERBench (Ben-Ami et al., 2025) with twelve question types, five options, and 2000 items (INIT accuracy $\approx$0.36), and NExT-QA (Xiao et al., 2021) with eight question types, five options, and the first 800 validation items that have an available video (INIT accuracy $\approx$0.81) differ sharply in difficulty, and TVBench (Cores et al., 2024) serves for replication.

Confidence measurement. These checkpoints normally emit a CoT before the answer, but after a CoT the answer-letter confidence averages 0.98 or more on 400 random HERBench items for every checkpoint we tested, which leaves no room to measure a change, so we skip the CoT. We run the model once on each item and read the logits at the position where the answer would begin. We restrict them to the k space-prefixed option-letter tokens, apply a softmax over that set, and take the maximum probability as the confidence, which can never fall below $1/k$. This is a fixed probe, since the model emits the letter without a leading space, and reading the emitted letters instead gives a smaller tax of the same sign (§E).

Paired estimator. For checkpoints A and B, the core set is the items that both answer incorrectly, and

$$\mathrm{TAX}(A{\to}B) = \mathbb{E}_{\text{core}}\left[\mathrm{conf}_B - \mathrm{conf}_A\right], \tag{1}$$

reported in percentage points (pp). Because the set is fixed by both checkpoints, item difficulty cancels in the difference. Statistics are paired t-tests over items. Unless noted, the diagnosis in §2.2 to §2.4 evaluates at 16 frames and 448 px, and everything from §2.5 onward, including the training study, at 16 frames and 224 px.

2.2 RL RAISES CONFIDENCE ON THE ITEMS IT GETS WRONG

Post-training raises confidence on the answers the model gets wrong. On the full 2000-item HERBench set, $\mathrm{TAX}(\text{SFT}{\to}\text{RL}) = +3.74$ (t=17.83) over the 1220 items both checkpoints answer incorrectly, against a mean SFT confidence of 45.3% on the same items. The effect is not tied to one benchmark or one frame budget, since it is positive and significant in all four benchmark $\times$ frame-budget cells we measured (Table 12). Accuracy does not fall (Table 9), so the tax is not a by-product of degraded answers.

2.3 SAME ANSWER, MORE CONFIDENCE, AND A DISPLACED OPERATING POINT

RL leaves the wrong answers intact and raises the confidence attached to them. Restricting to HERBench items where both checkpoints emit the identical wrong letter, the tax does not shrink. On 1075 of the 1220 shared failures (88%) the two checkpoints pick the same wrong option, and on that subset the tax is $+4.08$ (t=17.88) against $+3.74$ on all of them.

Post-training raises confidence without changing which option wins, so the tax is invisible to every rank-based diagnostic we measured, and §2.5 explains why. The AUROC for separating correct from incorrect answers moves only from 0.674 to 0.665, and the area under the risk-coverage curve is essentially unchanged (0.495 against 0.492, Table 9). A deployment, however, selects a threshold rather than a coverage. On the 1720 items where the public SFT and RL checkpoints emit the identical prediction, accuracy is the same for both (0.375), yet auto-processing everything with confidence of at least 0.8 sends 59% more items to automatic handling and raises the error rate among them from 27.6% to 36.2% (Table 10).

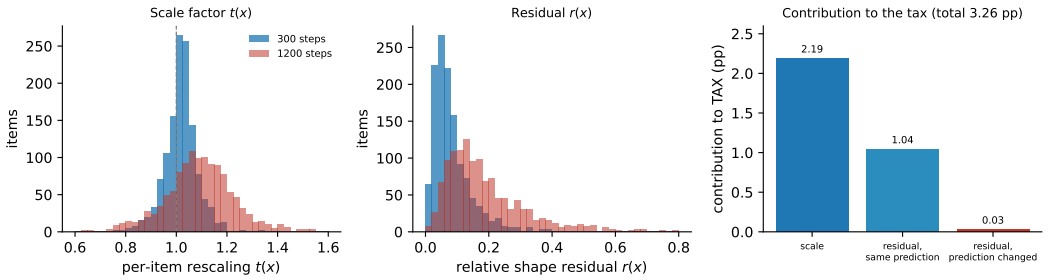

Figure 2: What the tax is. **Left**: the per-item rescaling $t(x)$ of Eq. 2 (1.019 ± 0.068 at 300 steps, 1.093 ± 0.162 at 1200). **Center**: the relative residual $r(x)$ (median 0.067 and 0.156), so RL rescales the readout far more than it reorients it. **Right**: the 1200-step tax on shared failures, two thirds of it from the scale and 99% of it on items whose predicted option does not change.

2.4 Where the tax concentrates, and its replication

The tax pattern across types is associated with the recipe, not explained by difficulty. The tax is positive on 11 of 12 HERBench types and significant on 9. Across types, SFT accuracy explains almost none of it (R^2=0.02), and in an item-level regression question type absorbs what difficulty would explain, but not the reverse (§B.1). The pattern instead differs between the two post-training recipes. T-GRPO, whose reward adds a bonus for temporal order, concentrates the tax at 2.5 times its own mean on action counting, while Open-o3, whose reward adds grounded spatial evidence, concentrates it at 2.3 times its mean on false-object memory, and the two profiles cross on most types (Figure 4). The two recipes also differ in optimizer, SFT data and auxiliary supervision, so this is an association with the recipe rather than a controlled attribution to the reward (§B.3). The contrast survives removing a meta-option confound. Two HERBench types offer an option stating that the listed descriptions do not match the video, and RL grows more confident in it even though none of the 31,470 multiple-choice training items offers such an option (§B.4). Without the items that contain it, the tax is still +2.79.

It is not a property of this checkpoint pair or benchmark. On the full 1963-item TVBench the same public pair gives +1.57 (t=12.02, 1097 shared failures) at slightly higher accuracy after RL (0.404 to 0.412), and on the InternVL3.5 family the tax appears at 2B, 8B and 14B, growing rather than attenuating with scale, while the ranking of affected types is model-specific. Both stacks share a Qwen-family backbone, so backbone independence is not established.

2.5 Much of the tax is a per-item rescaling of the readout

From here on we analyze checkpoints we trained ourselves from the public SFT checkpoint with the Video-R1 recipe (§A.1), evaluated on HERBench at 224 px, so that every intervention shares one reference. Their tax values (+0.64 at 300 steps and +3.26 at 1200) are therefore not comparable in size to the +3.74 of §2.2.

The estimator reports a scalar per item, but the readout is a vector. Writing z^{SFT} and z^{RL} for the mean-centered option-letter logits of a both-wrong item, we ask how much of the tax is a change in the *magnitude* of that vector and how much in its *direction*:

$$z^{\mathrm{RL}} \;=\; t(x)\, z^{\mathrm{SFT}} + e(x), \qquad t(x) = \frac{\langle z^{\mathrm{RL}}, z^{\mathrm{SFT}} \rangle}{\|z^{\mathrm{SFT}}\|^2}, \qquad r(x) = \frac{\|e(x)\|}{\|z^{\mathrm{RL}}\|}. \tag{2}$$

We fit one scalar for each item by least squares, so $e(x)$ is whatever that single scalar fails to explain, and $r(x)$ measures how large it is relative to z^{RL}. That remainder is small. Its median is 0.067 and it stays below 0.155 on 90% of items, and replacing z^{RL} with $t(x)\, z^{\mathrm{SFT}}$ reproduces 72% of the tax at 300 steps. RL therefore changes mainly the magnitude of the option-logit vector and leaves its direction nearly untouched on nine items in ten. The scalar itself is close to one on average but spread out across items (mean 1.019, standard deviation 0.068), and the same structure appears on TVBench (mean 1.020, median residual 0.068).

Table 1: Recovering the per-item rescaling $t(x)$ of Eq. 2. We split the shared failures into two halves, fit on one half, and report results on the other (n=620). "Tax kept" is the fraction of the tax that survives each correction, so lower is better. The oracle uses the true $t(x)$ and is not available in practice. The predictor is a ridge regression on the checkpoint's 3584-dimensional last-layer hidden state, and we chose its regularization on the reported half, so that row is optimistic. Table 7 uses a different split, where the constant keeps 37.6% and the oracle 26.0%.

correction	R^2 for $t(x)$	tax kept
none	–	100%
oracle per-item $t(x)$	1.00	20%
per-item $t(x)$ from the hidden state	0.13	39%
global mean $\bar{t}$	–	44%

The decomposition changes with training. Both parts grow, but the scale grows faster, reaching a mean of 1.093 at 1200 steps against a median remainder of 0.156, which is why the constant correction of §3.1 removes a larger share of the tax there. At 1200 steps the scale component carries two thirds of the tax and the residual on items whose prediction stays the same carries almost all of the rest, so 99% of the tax sits on items whose predicted option does not change (Figure 2, right).

Because a positive rescaling cannot reorder options, the tax is invisible to AUROC and to the risk-coverage curve while a fixed threshold still moves, which answers the question left open in §2.3.

2.6 The rescaling is barely recoverable from one model

A deployer holding only the RL checkpoint gains almost nothing from estimating $t(x)$. Knowing the true $t(x)$ would cut the tax to a fifth at exactly zero accuracy cost, since a positive rescaling cannot reorder options. Estimating it from the checkpoint itself recovers little of that (Table 1). A ridge regression on the last-layer hidden state reaches $R^2 = 0.13$ out of sample and leaves 39% of the tax, against 44% for a single global constant.

Much of $t(x)$ is not a property of the item at all. Fitted separately for two training runs of the same configuration, the two sets of values correlate at only 0.544, and at 0.640 with the rollout randomness fixed, so what the predictors miss is a property of the optimization run.

3 Method

The main idea is to hold the model's readout confidence, during RL training, at the level its own frozen reference already provides. Our method, **Reference Margin Anchoring (RMA, §3.2)**, penalizes the policy for moving the top-1 readout margin away from the reference's.

We place the penalty directly at the readout because the tax is measured there and the experiments in §4.2 show that indirect reward-side interventions do not control it reliably. RMA therefore acts directly on the quantity whose change is associated with confidence inflation.

3.1 Reference-matched rescaling

The decomposition prescribes its own post-hoc correction. We take the constant to be the mean of the per-item rescaling of Eq. 2, and fold its inverse into the output embedding:

$$\bar{t} = \mathbb{E}_x[t(x)]\,, \qquad W_{\text{out}} \leftarrow W_{\text{out}}/\bar{t}. \tag{3}$$

The expectation runs over all evaluation items. The producer computes $\bar{t}$ offline from the checkpoint pair, with no labels and no held-out set, and what ships is a single reference-free checkpoint. Because $\bar{t} > 0$, dividing every logit by it preserves their order, so this changes no prediction of the model on any input, only the magnitudes of the confidences.

Dividing the output embedding W_{out} by $\bar{t}$ and re-running the full evaluation takes the tax from $+0.64$ to $+0.26$ at 300 steps and from $+3.26$ to $+1.04$ at 1200, with accuracy unchanged in both cases and three of 2000 predictions moving, all on ties in half precision. That is 59% and 68% of the tax

removed, so the correction works better as the problem grows. It assumes that a tax exists, so it should not be applied when $\bar{t}$ is already close to one, and it cannot help items whose change lies mostly in the residual $e(x)$.

3.2 Reference margin anchoring (RMA)

On the both-wrong set the growth of the top-1 margin $m = z_{(1)} - z_{(2)}$, the gap between the largest and second-largest option logit, correlates with the per-item tax at ρ=0.88 at both 300 and 1200 steps, and capping m at its SFT value in the stored logits removes the tax entirely. These observations identify the readout margin as a direct control variable for the confidence inflation measured by TAX. The design principle is to hold the policy's margin at the value the frozen reference already gives. We therefore apply the penalty only on training items that the reference answers correctly, so that the target margin comes from a reference prediction whose top-ranked option is the gold answer. Both m_θ and m_{ref} come from one extra forward pass without CoT, and RMA adds

$$\mathcal{L}_{\text{RMA}} = \beta\, \mathbf{1}[\hat{y}_{\text{ref}} = y]\; \phi(m_\theta - m_{\text{ref}})\,, \qquad \phi \in \{\max(0,\cdot),\ |\cdot|\}. \tag{4}$$

This term is added to the GRPO objective (Shao et al., 2024), $\mathcal{L} = \mathcal{L}_{\text{GRPO}} + \mathcal{L}_{\text{RMA}}$, and nothing else in the recipe changes. The penalty reaches the both-wrong items on which the tax is measured because the rescaling it controls is carried by parameters shared across all items. The choice of ϕ decides whether the penalty has a destination. With $\phi = \max(0,\cdot)$ the loss vanishes once the margin falls below the reference, so nothing restores it. Because the margin moves through parameters that all items share, the one-sided penalty also lowers items already below the reference, and the margin gap keeps falling. The tax then passes zero, reaching -0.75 and -0.94 in two runs at β=0.15 and -0.24 at β=0.50. With $\phi = |\cdot|$ there is a fixed point at $m_\theta = m_{\text{ref}}$, so β sets how tightly the margin is held rather than how far it drifts, which is why the result in §4 depends little on β.

4 Experiments

Table 2 places both corrections next to the untreated checkpoint and a published calibration objective ported to this setting.

Table 2: Standard calibration measures alongside the paired tax against the SFT reference (HER-Bench, 224 px, ECE with 15 bins, multiclass Brier). RMA uses β=0.45. Accuracy cost is in points against the untreated run of the same length.

	steps	TAX (pp)	ECE	Brier	AUROC	acc. cost	retraining
SFT reference	–	–	0.140	0.776	0.660	–	–
RL, untreated	1200	+3.26	0.166	0.789	0.650	–	–
RMA	1200	**+0.16**	**0.139**	**0.777**	0.649	0.40	yes
rescaling by $\bar{t}$	1200	+1.04	0.143	0.778	0.651	**none**	**no**
RLCR-style Brier (Damani et al., 2025)	1200	+3.12	0.165	0.790	0.652	0.15	yes
RL, untreated	300	+0.64	0.138	0.779	0.646	–	–
RMA	300	−0.03	0.138	0.776	0.650	0.50	yes
RLCR-style Brier (Damani et al., 2025)	300	−0.29	0.134	0.774	0.649	0.20	yes

4.1 RMA removes the tax across penalty strengths and training lengths

Every β from 0.45 to 0.90 puts the tax within $\pm$0.13 of zero and returns the mean rescaling to $\bar{t} \approx 1$ (Table 3). The two-sided penalty therefore needs essentially no tuning, unlike the reward-side penalties of §4.2, where no choice of strength gave relief that survived replication. These runs had no fixed rollout seed, so the claim rests on all seven two-sided runs falling below both untreated runs, which happens by chance with probability $1/36 \approx 0.03$.

The tax grows with training, saturates, and the penalty reduces it. The 300-step regime understates the problem, so we trained the same recipe to 1200 steps and evaluated every saved checkpoint against the same SFT reference. The untreated tax rises 4.8× and then flattens (Figure 3). RMA ends at +0.16 (t=1.4), a 95% removal, and the RLCR-style objective does much less at this length and leaves +3.12.

Table 3: RMA across penalty strength β at 300 steps, two runs per strength except the single runs at one-sided 0.50 and two-sided 0.90. The item-level standard error of each tax is about 0.06. Figure 5 plots the tax column. Accuracies are shown to four decimals, the resolution of one item out of 2000.

β	TAX (pp)	accuracy	$\bar{t}$
0 (untreated)	+0.641 / +0.702	0.3545 / 0.3505	1.019
one-sided, $\phi = \max(0, \cdot)$			
0.15	−0.749 / −0.940	0.3505 / 0.3470	0.972 / 0.965
0.50	−0.243	0.3485	0.976
two-sided, $\phi = \lvert \cdot \rvert$			
0.30	+0.317 / +0.142	0.3540 / 0.3495	1.016 / 1.004
0.45	**−0.026 / +0.039**	0.3495 / 0.3505	0.996 / 1.003
0.60	−0.125 / +0.124	0.3510 / 0.3485	0.998 / 0.994
0.90	+0.088	0.3475	1.000

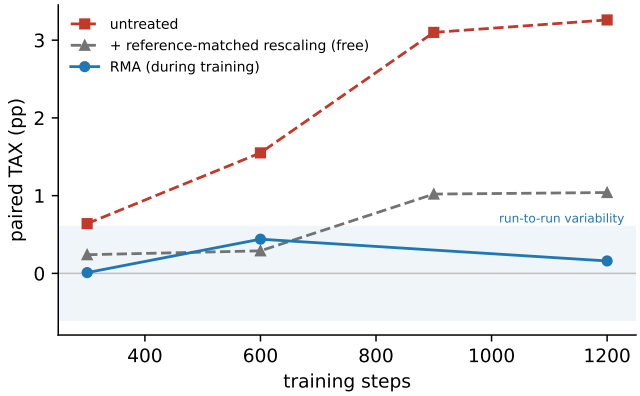

Figure 3: Both corrections across training length under one SFT reference. The untreated tax rises about 4.8× between 300 and 900 steps and then saturates. Rescaling by $\bar{t}$ leaves between a fifth and two fifths of it.

RMA keeps part of the accuracy gain and returns the confidence scale toward its reference. On the full set, post-training buys accuracy (SFT 0.347, untreated 0.355) and RMA keeps about half of it (0.351), so returning to the SFT checkpoint is not an option. On the 1728 items where all three checkpoints emit the same prediction, RMA returns the operating point close to SFT (Table 4).

Table 4: Operating point at 1200 steps on the 1728 items where all three checkpoints predict the same option (accuracy 0.364). "auto" is the fraction of items with confidence at least τ, and "err" is the error rate among them.

	SFT		RL untreated		RMA (β=0.45)	
τ	auto	err	auto	err	auto	err
0.60	25.9%	0.443	34.0%	0.482	25.9%	0.451
0.70	15.1%	0.364	21.6%	0.425	14.8%	0.363
0.80	9.3%	0.304	12.9%	0.372	9.1%	0.287
0.90	4.7%	0.198	6.8%	0.265	4.6%	0.162

The result carries over to a second benchmark. Neither HERBench nor TVBench is part of the training data, and Table 5 shows both sides of the result carrying over to TVBench. The removal fraction holds across the two benchmarks while the accuracy cost does not.

Table 5: Untreated and RMA checkpoints at 1200 steps on HERBench and TVBench. Accuracy cost is in points.

1200-step checkpoints	HERBench	TVBench
untreated, paired TAX	+3.26	+1.05
RMA, paired TAX	+0.16	+0.09
removed	95%	91%
accuracy cost of RMA	0.40	0.86

4.2 CAN THE REWARD CONTROL CONFIDENCE INFLATION?

A natural alternative to RMA is to control confidence indirectly through the reward, by penalizing confident wrong rollouts. For a prompt with group accuracy p_q and rollout o_i with sequence confidence c_i, the reward becomes

$$r_i = R_{\text{correctness}}(o_i) - \lambda\, g\, \mathbf{1}[\text{wrong}_i]\, c_i, \tag{5}$$

where λ is the reward penalty strength and the gate g takes the same value for every rollout in a group, $g{=}1{-}p_q$ for the gated penalty and $g{=}1$ for the ungated one. The family varies in the strength and placement of the penalty, the gate, and the confidence proxy c. Lemma 1 (§C) shows that only the part of the penalty that ranks wrong rollouts by confidence adds anything beyond the correctness reward, and it predicted before training that the gate would leave this part too small to matter.

None of the reward-side interventions produced a reliable and reproducible reduction. All share data, training seed, steps, learning rate, KL coefficient and GPU count (Table 6). Individual runs do move the readout. The ungated penalty lowered the tax by 0.93 in one run, but its three runs span -0.93 to $+0.45$ (§D).

Table 6: The seven reward-side interventions, grouped by what they vary. Outcomes are read against the run-to-run variability scale of about 0.6, and Table 14 gives the numbers.

lever	intervention	outcome
gate	difficulty-gated	no effect, as the lemma predicts
gate	ungated	acts, but relief misses the reward's own target
placement	shuffled-evidence	small relief, moved to other types
placement	reward-scaled	slightly raises the tax
placement	evidence-dependent	tracks the target, does not replicate
proxy	answer-span	no effect at either strength
strength	adaptive multiplier	multiplier drifts, no fixed strength

A published calibration objective overshoots early and stalls later. We port the RLCR objective to the readout, adding a Brier term (Damani et al., 2025) on the readout confidence, $\gamma\,(c_\theta - \mathbf{1}[\hat{y}_\theta = y])^2$, on every item. At 300 steps two runs give -0.29 and -0.20 ($t{=}-3.62$ and $t{=}-2.80$), so the model ends up less confident than its SFT reference, because a Brier target of 0 or 1 gives no stopping point while m_{ref} gives RMA one. At 1200 steps the same term leaves $+3.12$, a 4% removal. This is an adaptation, not a reproduction of Damani et al. (2025).

Two kinds of run-to-run variability. Re-running the untreated baseline changes the tax by only $+0.07$ ($t{=}1.08$, n.s.), and we call this the baseline re-run difference. Runs with a penalty vary far more. The three runs of the ungated penalty give -0.93, -0.33 and $+0.45$, so two runs of the same configuration differ by at least 0.59. We call this smallest difference, about 0.6, the run-to-run variability scale and read every intervention effect against it. With three runs, a two-sample t-test has power below 20% for an effect of one run-to-run standard deviation, so several null verdicts are failures of power. One source of variance can be removed. The trainer passed no seed to the rollout sampler, and supplying one reduces the baseline re-run difference from $+0.07$ to $+0.01$ ($t{=}0.23$). Without that control, a single-run comparison at this effect size is not interpretable.

4.3 POST-HOC CALIBRATORS

Correction after training meets the same limit from the other side. Among calibrators whose parameters are shared across question types, the more parameters a calibrator has, the more of the tax it keeps (Table 7). Per-type temperature keeps slightly less of the tax than rescaling by $\bar{t}$ (35.4% against 37.6%), but it needs labels and question-type annotations, while $\bar{t}$ needs neither.

Table 7: Post-hoc corrections on our 300-step untreated checkpoint, fitted on one half of the items and evaluated on the other, two folds averaged. "Tax kept" is the fraction of the paired tax that survives, so lower is better. Among the calibrators whose parameters are shared across types, the more free parameters a calibrator has, the more tax it keeps. The oracle uses the true per-item $t(x)$ and is not available in practice.

correction	free params	needs labels	tax kept	accuracy
none	–	–	100%	0.354
per-type temperature	12	yes	35.4%	unchanged
global temperature (T=2.15)	1	yes	52.5%	unchanged
bias + temperature	6	yes	59.4%	0.350
vector scaling	10	yes	61.3%	0.350
matrix scaling	30	yes	61.9%	0.341
rescaling by $\bar{t}$ (ours)	1	**no**	$\mathbf{37.6}\%$	**unchanged**
oracle per-item $t(x)$	–	no	26.0%	unchanged

5 RELATED WORK

RLVR is known to erode calibration in text reasoning (Bereket & Leskovec, 2025). RLCR adds a Brier term to the correctness reward (Damani et al., 2025), CAPO analyzes advantages that ignore uncertainty (Wang et al., 2026), and DCPO separates the confidence tokens from the reasoning tokens (Ma et al., 2026). All three supervise a *verbalized* confidence, while we measure the readout the model already has. Post-hoc temperature scaling and its extensions (Guo et al., 2017) overfit as their parameters grow, logit normalization (Wei et al., 2022; Cattelan & Silva, 2023) targets the growth of logit norms, and the risk-coverage view of confidence comes from selective prediction (El-Yaniv & Wiener, 2010; Geifman & El-Yaniv, 2017). How well confidence tracks correctness depends on how it is read out (Kadavath et al., 2022; Lin et al., 2022; Tian et al., 2023; Chhikara, 2025), answers often do not change when the visual evidence is removed or replaced (Zafar et al., 2026), and correctness-only RL has been argued to reinforce temporal shortcuts in video (Du et al., 2026).

6 CONCLUSION

Video RL post-training increases confidence on shared failures even when the predicted answer does not change. Much of this increase has the form of a rescaling of the option-logit readout, which leaves rank-based metrics nearly unchanged but alters decisions based on fixed confidence thresholds. RMA directly constrains the readout margin during training and removes most of this confidence increase, and a simpler offline rescaling provides a partial correction when retraining is unavailable. The results also show substantial run-to-run variability, which should be accounted for when calibration interventions are evaluated.

AI USE STATEMENT

We used generative AI tools as a coding and writing assistant, to refactor analysis and training code, edit the paper, and read related work. All experimental designs, hypotheses, and interpretations are the authors'. We take responsibility for the final content of this work, including all text, claims and artifacts.

Reproducibility statement

The paired estimator, the readout probe and its robustness check, the full training configuration for each intervention, and the protocol for fixing the rollout seed are specified in the appendix. All checkpoints used in §2.2 to §2.4 are public. The training study uses one public training set and one public SFT checkpoint, and every checkpoint from §2.5 onward is trained from them with the configuration of §A.1.

References

Dan Ben-Ami, Gabriele Serussi, Kobi Cohen, and Chaim Baskin. HERBench: a benchmark for multi-evidence integration in video question answering. *arXiv preprint arXiv:2512.14870*, 2025.

Michael Bereket and Jure Leskovec. Uncalibrated reasoning: GRPO induces overconfidence for stochastic outcomes. *arXiv preprint arXiv:2508.11800*, 2025.

Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950.

Shyamal Buch, Cristóbal Eyzaguirre, Adrien Gaidon, Jiajun Wu, Li Fei-Fei, and Juan Carlos Niebles. Revisiting the "video" in video-language understanding. In *CVPR*, pp. 2917–2927, 2022.

Luís Felipe Cattelan and Danilo Silva. How to fix a broken confidence estimator: Evaluating post-hoc methods for selective classification with deep neural networks. *arXiv preprint arXiv:2305.15508*, 2023.

Prateek Chhikara. Mind the confidence gap: Overconfidence, calibration, and distractor effects in large language models. *arXiv preprint arXiv:2502.11028*, 2025.

Daniel Cores, Michael Dorkenwald, Manuel Mucientes, Cees G. M. Snoek, and Yuki M. Asano. TVBench: redesigning video-language evaluation. *arXiv preprint arXiv:2410.07752*, 2024.

Ganqu Cui, Yuchen Zhang, Jiacheng Chen, Lifan Yuan, Zhi Wang, Yuxin Zuo, Haozhan Li, Yuchen Fan, Huayu Chen, Weize Chen, et al. The entropy mechanism of reinforcement learning for reasoning language models. *arXiv preprint arXiv:2505.22617*, 2025.

Mehul Damani, Isha Puri, Stewart Slocum, Idan Shenfeld, Leshem Choshen, Yoon Kim, and Jacob Andreas. Beyond binary rewards: Training LMs to reason about their uncertainty. *arXiv preprint arXiv:2507.16806*, 2025.

Morris H. DeGroot and Stephen E. Fienberg. The comparison and evaluation of forecasters. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 32(1-2):12–22, 1983.

Rebecca DerSimonian and Nan Laird. Meta-analysis in clinical trials. *Controlled Clinical Trials*, 7 (3):177–188, 1986.

Dazhao Du, Jian Liu, Jialong Qin, Tao Han, Bohai Gu, Fangqi Zhu, Yujia Zhang, Eric Liu, Xi Chen, and Song Guo. Learning spatiotemporal sensitivity in video LLMs via counterfactual reinforcement learning. *arXiv preprint arXiv:2605.21988*, 2026.

Ran El-Yaniv and Yair Wiener. On the foundations of noise-free selective classification. *Journal of Machine Learning Research*, 11:1605–1641, 2010.

Kaixuan Fan, Kaituo Feng, Haoming Lyu, Dongzhan Zhou, and Xiangyu Yue. SophiaVL-R1: reinforcing MLLMs reasoning with thinking reward. *arXiv preprint arXiv:2505.17018*, 2025.

Kaituo Feng, Kaixiong Gong, Bohao Li, Zonghao Guo, Yibing Wang, Tianshuo Peng, Benyou Wang, and Xiangyu Yue. Video-R1: reinforcing video reasoning in MLLMs. *arXiv preprint arXiv:2503.21776*, 2025a.

Yunzhen Feng, Parag Jain, Anthony Hartshorn, Yaqi Duan, and Julia Kempe. Don't waste mistakes: Leveraging negative RL-groups via confidence reweighting. *arXiv preprint arXiv:2510.08696*, 2025b.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *NeurIPS*, 2017.

Jiannan Guan, Qiguang Chen, Libo Qin, Dengyun Peng, Jinhao Liu, Liangyu Huo, Jian Xie, and Wanxiang Che. Beware of reasoning overconfidence: Pitfalls in the reasoning process for multi-solution tasks. *arXiv preprint arXiv:2512.01725*, 2025.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *ICML*, 2017.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. DeepSeek-R1: incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Romain Lacombe, Kerrie Wu, and Eddie Dilworth. Don't think twice! over-reasoning impairs confidence calibration. In *ICML 2025 Workshop on Reliable and Responsible Foundation Models*, 2025.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V. Miranda, Alisa Liu, Nouha Dziri, et al. Tülu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*, 2024.

Max Lamparth, Daniel Fein, Andreas Haupt, Marcel Hussing, and Mykel J. Kochenderfer. Reward bias substitution: Single-axis bias mitigations redirect optimization pressure. *arXiv preprint arXiv:2605.27996*, 2026.

Jae Joong Lee. Accuracy without grounding: Diagnosing visual dependency dissociation in video LLM benchmarks. *arXiv preprint arXiv:2607.13305*, 2026. Accepted, ACM Multimedia 2026.

Jie Lei, Tamara L. Berg, and Mohit Bansal. Revealing single frame bias for video-and-language learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, pp. 487–507, 2023.

Hongyu Li, Songhao Han, Yue Liao, Junfeng Luo, Jialin Gao, Shuicheng Yan, and Si Liu. Reinforcement learning tuning for VideoLLMs: reward design and data efficiency. *arXiv preprint arXiv:2506.01908*, 2025a.

Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, et al. MVBench: a comprehensive multi-modal video understanding benchmark. In *CVPR*, pp. 22195–22206, 2024.

Xinhao Li, Ziang Yan, Desen Meng, Lu Dong, Xiangyu Zeng, Yinan He, Yali Wang, Yu Qiao, Yi Wang, and Limin Wang. VideoChat-R1: enhancing spatio-temporal perception via reinforcement fine-tuning. *arXiv preprint arXiv:2504.06958*, 2025b.

Sarah Lichtenstein and Baruch Fischhoff. Do those who know more also know more about how much they know? *Organizational Behavior and Human Performance*, 20(2):159–183, 1977.

Stephanie Lin, Jacob Hilton, and Owain Evans. Teaching models to express their uncertainty in words. *Transactions on Machine Learning Research*, 2022.

Haotian Liu, Shuo Wang, and Hongteng Xu. C^2GSPG: confidence-calibrated group sequence policy gradient towards self-aware reasoning. *arXiv preprint arXiv:2509.23129*, 2025a.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding R1-Zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*, 2025b.

Zhengzhao Ma, Xueru Wen, Boxi Cao, Yaojie Lu, Hongyu Lin, Jinglin Yang, Min He, Xianpei Han, and Le Sun. Decoupling reasoning and confidence: Resurrecting calibration in reinforcement learning from verifiable rewards. *arXiv preprint arXiv:2603.09117*, 2026.

Jiahao Meng, Xiangtai Li, Haochen Wang, Yue Tan, Tao Zhang, Lingdong Kong, Yunhai Tong, Anran Wang, Zhiyang Teng, Yujing Wang, and Zhuochen Wang. Open-o3 video: Grounded video reasoning with explicit spatio-temporal evidence. *arXiv preprint arXiv:2510.20579*, 2025.

Mahdi Pakdaman Naeini, Gregory F. Cooper, and Milos Hauskrecht. Obtaining well calibrated probabilities using bayesian binning. In *AAAI*, 2015.

Kun Ouyang, Yuanxin Liu, Haoning Wu, Yi Liu, Hao Zhou, Jie Zhou, Fandong Meng, and Xu Sun. SpaceR: reinforcing MLLMs in video spatial reasoning. *arXiv preprint arXiv:2504.01805*, 2025.

Pouya Pezeshkpour and Estevam Hruschka. Large language models sensitivity to the order of options in multiple-choice questions. In *Findings of NAACL*, 2024.

Qwen Team. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Joshua Robinson and David Wingate. Leveraging large language models for multiple choice question answering. In *ICLR*, 2023.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D. Manning. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *EMNLP*, pp. 5433–5442, 2023.

Weiyun Wang et al. InternVL3.5: advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025a.

Ye Wang, Ziheng Wang, Boshen Xu, Yang Du, Kejun Lin, Zihan Xiao, Zihao Yue, Jianzhong Ju, Liang Zhang, Dingyi Yang, et al. Time-R1: post-training large vision language model for temporal video grounding. *arXiv preprint arXiv:2503.13377*, 2025b.

Ziqi Wang, Xingzhou Lou, Meiqi Wu, Zhengqi Wen, and Junge Zhang. Calibration-aware policy optimization for reasoning LLMs. *arXiv preprint arXiv:2604.12632*, 2026.

Hongxin Wei, Renchunzi Xie, Hao Cheng, Lei Feng, Bo An, and Yixuan Li. Mitigating neural network overconfidence with logit normalization. In *ICML*, 2022.

Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. NExT-QA: next phase of question-answering to explaining temporal actions. In *CVPR*, 2021.

Tianyang Xu, Shujin Wu, Shizhe Diao, Xiaoze Liu, Xingyao Wang, Yangyi Chen, and Jing Gao. SaySelf: teaching LLMs to express confidence with self-reflective rationales. In *EMNLP*, pp. 5985–5998, 2024.

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Yang Yue, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in LLMs beyond the base model? *arXiv preprint arXiv:2504.13837*, 2025.

Anas Zafar, Leema Krishna Murali, and Ashish Vashist. Beyond accuracy: Evaluating visual grounding in multimodal medical reasoning. *arXiv preprint arXiv:2603.03437*, 2026.

Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. Large language models are not robust multiple choice selectors. In *ICLR*, 2024.

Chujie Zheng et al. Group sequence policy optimization. *arXiv preprint arXiv:2507.18071*, 2025.

A ADDITIONAL MATERIAL FROM THE MAIN TEXT

A.1 TRAINING CONFIGURATION

Every checkpoint we train starts from the public Video-R1 SFT checkpoint and uses the public Video-R1 training set with T-GRPO (Feng et al., 2025a), a variant of GRPO (Shao et al., 2024; Guo et al., 2025). We use the Video-R1 recipe because both of its stages are released. Table 8 lists the configuration, which is shared by every run, and every intervention changes only its own penalty term. RMA as implemented also carries a second term with weight 0.10 that fires only when the reference is correct and the policy is not. It fired on about 1% of training steps in every run, so all RMA results reflect Eq. 4 alone. Unless stated otherwise the rollout sampler receives no seed. Where we fix it, the same seed is passed to the rollout sampler of both runs being compared, so both runs draw from the same random numbers.

Table 8: Training configuration shared by every run we trained.

setting	value
initialization	public Video-R1 SFT checkpoint (Qwen2.5-VL-7B)
training data	public Video-R1 training set
algorithm	T-GRPO, KL coefficient 0.04 to the SFT reference
optimizer	learning rate 10^{-6}, cosine schedule, weight decay 0.01, gradient clip 5
batch	1 prompt per GPU, 4 training GPUs, no gradient accumulation
rollouts	8 per prompt, temperature 1.0, at most 768 generated tokens
inputs	at most 224×224 pixels per frame, prompts up to $16{,}384$ tokens
steps	300 (about 7 hours) or 1200 (about 27 hours)
hardware	five A100 GPUs, four for training and one for rollout generation
precision	bf16, DeepSpeed ZeRO-3, gradient checkpointing
RMA	β=0.45 unless stated, two-sided ϕ, gated on reference-correct items

A.2 READOUT AND CONFIDENCE MEASUREMENT

The size of the paired difference depends on which letter tokens index the restricted set, but its sign and significance do not (§E). Absolute ECE (Naeini et al., 2015; Guo et al., 2017) partly reflects the $1/k$ floor against near-chance accuracy, so all inference rests on paired differences. Proper-scoring and reliability conventions follow Brier (1950) and DeGroot & Fienberg (1983). Scoring a multiple-choice answer by the probability of its option letter follows Robinson & Wingate (2023). We do not use a verbalized confidence (Lin et al., 2022; Xu et al., 2024), which these checkpoints were never trained to emit.

Paired estimator. The core set of Eq. 1 conditions on both checkpoints in the same way, so item difficulty and the hard-easy effect (Lichtenstein & Fischhoff, 1977) cancel in the difference. Unless noted, the public checkpoints are evaluated at 16 frames and 448 px.

Table 9: Unpaired full-2000 HERBench aggregates for the four checkpoints (448 px). Accuracy is flat across INIT, SFT and RL, and aggregate ECE falls from INIT to SFT and then rises again. From SFT to RL the rank-based metrics barely move, with AUROC changing by less than a point and AURC (area under the risk-coverage curve, lower is better) essentially unchanged. Confidence on wrong answers is higher after RL than after SFT (0.471 to 0.509) and higher still for OPEN-O3 (0.556). These unpaired means point the same way as the paired estimator.

Checkpoint	Accuracy	ECE	AUROC	AURC	Confidence on wrong
INIT (Qwen2.5-VL-7B)	0.362	0.203	0.650	0.493	0.524
SFT (CoT)	0.351	0.162	0.674	0.495	0.471
RL (T-GRPO)	0.361	0.191	0.665	0.492	0.509
OPEN-O3 (GSPO)	0.374	0.232	0.676	0.465	0.556

B DIAGNOSIS DETAIL

B.1 THE TAX IS BROAD, AND ITS SHAPE DOES NOT FOLLOW DIFFICULTY

Across all twelve HERBench types (Table 11), the tax is positive on eleven and significant on nine. It is largest on temporal shot ordering ($+11.75$, t=15.64) with action counting close behind ($+6.93$, t=13.95), and it is null on three types that are not the easiest. The ordering value is not what it appears to be, since §B.4 shows that it is carried almost entirely by a meta-option present in that type, and every per-type reading below is stated on the items free of it.

Table 10: Operating point for the public checkpoints (448 px), on the 1720 items where SFT and RL emit the same prediction, so accuracy is 0.375 for both and every difference is confidence alone. "auto" is the fraction of items at or above the threshold τ, and "err" is the error rate among them.

	SFT		RL	
τ	auto	err	auto	err
0.6	31.3%	0.456	42.7%	0.481
0.7	19.0%	0.368	26.8%	0.425
0.8	11.4%	0.276	18.1%	0.362
0.9	5.9%	0.167	9.0%	0.271

Table 11: TAX(SFT→RL) for each HERBench question type on the full 2000-item set, sorted by t. The tax is positive on eleven of the twelve types and significant on nine of them. The three null types are not the easiest ones, which is part of why difficulty does not explain the pattern. Temporal shot ordering is the largest, but §B.4 shows that this row is carried almost entirely by a meta-option present in that type. Abbreviations are those used in Figure 4.

Type	Abbr.	TAX	t
Temporal Shot Ordering	TSO	+11.75	15.64
Action Counting	AC	+6.93	13.95
False-Action Memory	FAM	+4.02	8.08
Multi-Entities Grounding & Localization	MEG	+3.67	7.78
Region-Localized People-Counting	RLP	+4.07	7.25
False-Object Memory	FOM	+2.68	3.90
Appearance-Grounded Localization	AGL	+3.03	3.35
Appearance-Grounded Behavior	AGB	+4.02	2.69
Scene-Verification & Arrangement	SV	+1.19	2.55
Action-Sequence Integrity	ASI	+0.53	0.71 (n.s.)
Multi-Person Duration	MPD	+0.26	0.57 (n.s.)
Appearance-Grounded Attribute	AGA	−1.53	−0.82 (n.s.)

Difficulty does not order the distribution. In the item-level regression $\mathrm{TAX} \sim p_{\text{gold}} + C(\text{qtype})$, where p_{gold} is the probability the INIT checkpoint assigns to the gold option, adding continuous difficulty on top of type does nothing (nested $F(1, 1207)$=2.25, p=0.13), while adding type on top of difficulty is overwhelming ($F(11, 1207)$=30.55, p=2.8×10^{-57}). Type explains about 145 times more of the between-item variance than difficulty (semipartial R^2 of 0.217 against 0.0015). The same holds for OPEN-O3. Three other accounts also fail. A ceiling account fails because the null types have the *lowest* baseline confidence. A floor account fails because all types share the same 0.2 floor. An account based on the entropy collapse of RL training (Cui et al., 2025) fails because the entropy of the predicted letter stays close to uniform. The type analysis is post hoc, but the two largest effects (t>13) survive any correction for twelve comparisons.

B.2 OTHER DIFFICULTY ACCOUNTS ALSO FAIL

Stratifying by INIT correctness shows no significant gradient, with +3.80 (t=17.15, n=1111) on items INIT gets wrong and +3.16 (t=4.90, n=109) on items it gets right. Frame budget does not remove the effect either. The tax is slightly larger at 8 frames than at 16 on both benchmarks (Table 12), so a smaller frame budget does not weaken it.

Table 12: The paired tax in every benchmark × frame-budget cell for the public checkpoints, each on the full evaluation set at 448 px. The effect is positive and significant in all four cells.

benchmark	frames	TAX (pp)	t	n (both-wrong)	n
HERBench	16	$+3.74 \pm 0.21$	17.83	1220	2000
HERBench	8	$+4.19 \pm 0.22$	18.90	1220	2000
NExT-QA	16	$+1.34 \pm 0.47$	2.86	141	800
NExT-QA	8	$+2.21 \pm 0.46$	4.81	139	800

B.3 Where each recipe concentrates the cost

We divide each type's tax by the recipe's own mean tax on the items without the meta-option (§B.4), which cover eleven types. This mean is $+2.79$ for T-GRPO (n=1065) and $+7.72$ for Open-o3 (n=1018), so that a value of one is a proportional share. On this scale the two profiles concentrate on different types. T-GRPO puts 2.49 times its mean on action counting, a temporal counting type, where the grounding recipe puts only 0.89 times its mean. Open-o3 puts 2.27 times its mean on false-object memory and 1.73 times on false-action memory, where T-GRPO puts 0.96 and 1.44 times. The crossing is present in absolute terms too. Action counting is identical across recipes ($+6.93$ vs $+6.88$) while false-object memory is $6.5\times$ larger under the grounding reward ($+2.68$ vs $+17.5$). The direct paired contrast OPEN-O3 − RL is null on temporal shot ordering ($+1.08$, n.s.) and large on the grounding and memory types, with $+14.99$ on false-object memory (t=9.14), $+10.47$ on scene verification (t=9.37), $+8.79$ on false-action memory (t=6.58), and significant values on five further grounding and localization types. On questions free of the meta-option, the shape of the tax across question types differs between the two recipes (Figure 4).

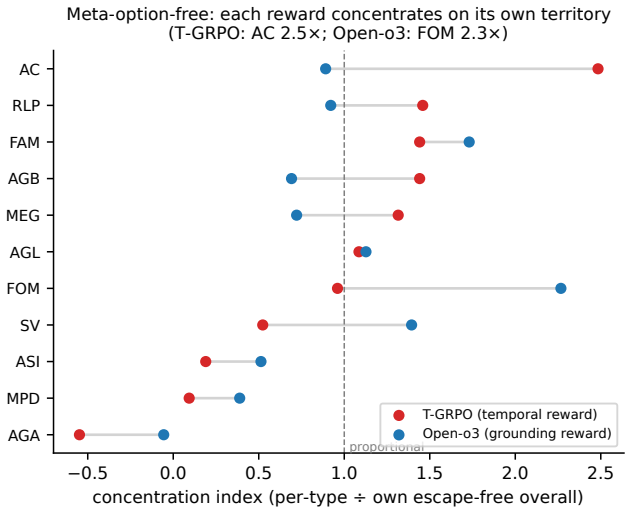

Figure 4: Recipe contrast on the eleven meta-option-free question types, as concentration indices where the tax of each type is divided by that recipe's own mean tax on the meta-option-free items, so 1.0 is a proportional share. Type abbreviations follow Table 11. The levels differ (Open-o3 $+7.72$ against T-GRPO $+2.79$), so the normalization isolates where each reward concentrates rather than how much it adds. The temporal reward peaks on action counting ($2.49\times$, against $0.89\times$ for the grounding reward) and the grounding reward peaks on false-object memory ($2.27\times$, against $0.96\times$), with the two orderings crossing on most types.

Limitation. The two pipelines differ in optimizer, SFT corpus and auxiliary supervision at the same time, so this is an observed contrast between two systems rather than a controlled attribution to the reward. §B.5 and §B.6 show that the tax itself replicates on other benchmarks and other model sizes, but neither isolates the reward. Other video RL recipes use still other reward designs (Li et al., 2025b; Wang et al., 2025b; Ouyang et al., 2025; Fan et al., 2025; Li et al., 2025a).

B.4 A META-OPTION CONFOUND, AND THE TAX WITH IT REMOVED

Two HERBench types embed a meta-option (*"at least two descriptions do not accurately reflect any shot"*), ordering in all 152 items, scene verification in 109 of 328. Models select it at nearly the same rate in both types (0.76 and 0.78), but it is *never* correct in ordering ($0/152$) and *always* correct in scene verification ($109/109$). Ordering accuracy therefore sits below the $1/5$ chance level (0.053 for SFT and 0.079 for RL), as the selection rate alone predicts. The ordering tax lives there. Splitting the SFT→RL core gives $+14$ (n=112, t=19.6) where both checkpoints chose the meta-option versus $+0.89$ (n=20, n.s.) where neither did. The inflation is not reward-specific. The grounding recipe inflates the same response by $+13.8$, and in scene verification, where choosing it is *correct*, confidence in it still rises $+13.39$ (t=15.0). Post-training thus adds about 14 points of certainty to this meta-response wherever it appears, regardless of reward or of correctness, and only whether the option is labeled correct decides whether it shows up as tax. We report this as a finding in its own right, since RLVR makes the model more certain that a question cannot be answered as posed. This fits reports that reasoning can raise certainty beyond what the evidence supports (Guan et al., 2025; Lacombe et al., 2025). **With the confound removed**, that is, with all 155 shared failures that contain the meta-option dropped (138 in ordering and 17 in scene verification), the remaining eleven types still carry $+2.79$ (t=14.08, n=1065) under the temporal reward and $+7.72$ (n=1018) under the grounding reward. Every per-type claim here is stated on that basis. The ordering row of Table 11 and the pooled ordering relief of §D.6 are statements about the meta-response, not about temporal reasoning. The tendency is not learned from the training data, since *none* of the 31,470 multiple-choice items in the public Video-R1 training set contains a meta or abstention option. RL amplifies certainty in a response type it was never trained on.

B.5 EXTERNAL REPLICATION ON TVBENCH

TVBench (Cores et al., 2024) uses different source videos and a variable number of options, and it was built to remove the single-frame and text-only shortcuts that affect earlier video benchmarks such as MVBench (Li et al., 2024; Buch et al., 2022; Lei et al., 2023). Over all 1963 items, TAX(SFT→RL) for the public pair is $+1.57$ (t=12.02, 1097 shared failures). By task it is $+4.08$ on object shuffle (t=26.01), $+3.04$ on action sequence, $+2.46$ on moving direction, $+2.44$ on egocentric sequence, null on action count and object count, and -2.62 on scene transition ($t=-5.53$). The tax therefore reproduces on a second benchmark, but the task name does not decide it. A task that is not about order carries tax, a task about order moves negative, and a counting task is null although the counting types of HERBench carry a large tax. This agrees with audits showing that a task label about temporal order does not guarantee that answering it depends on temporal information (Lee, 2026).

B.6 A SECOND MODEL FAMILY AND SCALE WITH INTERNVL3.5

InternVL3.5-8B (Wang et al., 2025a) uses a different vision encoder with a Qwen3 language model and has released SFT and Cascade-RL checkpoints, which we evaluate, like every InternVL3.5 result here, at 8 frames of 448×448 pixels. For this pair, TAX(SFT→RL) is positive and significant on *every* type on both benchmarks, with $+8.6$ (t=32) on the TVBench tasks that are not about order, $+5.1$ (t=18) on those that are, and $+9.1$ (t=37) on HERBench, with no drop in accuracy. The *ranking* of types differs sharply. The InternVL cost peaks on multi-entities grounding ($+13.1$) and people counting ($+11.2$), and temporal shot ordering sits in the middle. The tax therefore appears in a second model family, while its distribution over types depends on the model and the recipe.

A scale axis within one lineage. The same SFT→RL pair at 2B, 8B and 14B reproduces the tax at every size, with $+6.97$, $+9.06$ and $+\mathbf{14.87}$ on HERBench and $+7.15$, $+7.02$ and $+\mathbf{14.26}$ on TVBench (t between 28 and 45). The tax therefore *does not shrink with scale* and is largest at 14B. Three sizes do not support a scaling law, and the claim we make is only that the tax does not weaken. The distribution over types changes at every size. The 14B cost peaks on appearance-grounded types, and its TVBench scene_transition tax is $+19.3$, the very task that moves negative under Qwen T-GRPO.

C ADVANTAGE DECOMPOSITION

For a prompt with group accuracy p_q and rollout o_i with sequence confidence c_i, taken as the length-normalized geometric mean of the token probabilities, the penalized reward, of the same form as the confidence-weighted penalty on incorrect responses in LENS (Feng et al., 2025b), is:

$$r_i = R_{\text{correctness}}(o_i) - \lambda\, g\, \mathbf{1}[\text{wrong}_i]\, c_i, \tag{6}$$

with a gate g that is constant within the group, $g{=}1{-}p_q$ for the gated penalty and $g{=}1$ for the ungated one. The penalty applies only to rollouts whose answer is wrong, and confidence sets the size of the penalty rather than detecting errors.

Lemma 1 (Advantage decomposition of a uniform confidence penalty)**.** Fix one prompt with G rollouts, rewards r_i as in Eq. 6, GRPO advantages $A_i = (r_i - \bar{r})/\sigma_r$, and let $\bar{c}_w$ be the mean confidence over the wrong rollouts. The contribution of the penalty to the advantages splits exactly into two orthogonal components. (i) A **margin term**, the group-constant amount $\lambda g\, \bar{c}_w/\sigma_r$ by which every wrong rollout is lowered relative to every correct one. It points in the same direction as the separation that the correctness reward already provides and carries no information about confidence within either subset. (ii) A **calibration term**, the component $-\lambda g\,(c_i - \bar{c}_w)/\sigma_r$ centered within the wrong rollouts. It is the only component that orders rollouts by confidence, and its size in advantage units is $\lambda g\, \mathrm{sd}(c \mid \text{wrong})/\sigma_r$.

Within the wrong rollouts the calibration term is the only difference in reward, so it always orders them by confidence, however small it is. What the gate controls is how large that ordering is next to the correctness signal, which separates correct from wrong rollouts by $1/\sigma_r \approx 3$ in advantage units. At the empirical operating point ($p_q \approx 0.71$, so $g \approx 0.29$, and $\sigma_r \approx 0.3$) the calibration term is about 0.2 in the same units, so its push on the policy is small next to the correctness signal. The lemma therefore predicts, before training, that the gated penalty will match the untreated baseline, and §D.1 confirms this.

Relation to prior theory. Prior theory in text-domain calibration RL explains why the base objective harms calibration. One account is that advantages which ignore uncertainty misalign the gradient with calibration (Wang et al., 2026), and another is that accuracy and calibration objectives conflict in their gradients when both act on a verbalized confidence (Ma et al., 2026). Lemma 1 analyzes the fix instead of the failure. It asks how much of an added penalty carries any signal that orders rollouts by confidence, it gives a prediction stated before training, and it places the role of σ_r precisely, since normalization scales the calibration term but cannot create or destroy the ordering it induces. The sequence confidence c_i is the same quantity that C^{2}GSPG calibrates to the reward with a cross-entropy regularizer (Liu et al., 2025a), so the lemma also describes how much ordering signal such a regularizer adds on a single prompt. More broadly, RLVR is known to sharpen the output distribution of the base model (Yue et al., 2025; Cui et al., 2025). The tax is related to this sharpening, but it is measured on a readout that the reward never trains, and part of it changes the direction of the option logits rather than only their scale (§2.5).

The difficulty gate is tested rather than assumed. The $(1{-}p_q)$ gate reflects the hypothesis that RL harms calibration mainly on items beyond the model's reach, a hypothesis our diagnosis does not support (§B.2). The ungated penalty removes the gate so that its effect can be measured directly.

D THE REWARD-SIDE INTERVENTIONS IN FULL

Table 13 lists the training configurations at $\lambda{=}0.5$ together with the two baselines, and Table 14 adds the later placements and the strength sweeps, which use the same setup. Every run is evaluated with the diagnosis readout on the full HERBench set and on TVBench, and every comparison is a paired estimate on the same items against the same evaluation of the untreated baseline. We stated one primary endpoint before running any of these, namely that the improvement over the untreated baseline should concentrate on the question types that the T-GRPO reward inflates.

D.1 THE DIFFICULTY GATE SUPPRESSES THE PENALTY

The gated penalty is indistinguishable from the untreated baseline (Table 14), and on the primary endpoint the sign *inverts*. On temporal shot ordering the gated model is more confident on shared

Table 13: The configurations we trained at λ=0.5. All share the same data, training seed, 300 steps, learning rate, KL coefficient and GPU count, so the only difference between rows is the penalty itself. "Gate" is the factor g in Eq. 6 and "confidence term" is what the penalty is computed on. The gated row is the one Lemma 1 predicts will have no effect.

Configuration	λ	Gate	Confidence term	Tests
untreated baseline	0	–	–	no penalty
baseline, re-run	0	–	–	baseline re-run difference
no-normalization control	0	–	–	the normalization account
gated penalty	0.5	$1-p_q$	sequence confidence	the gate
ungated penalty	0.5	1	sequence confidence	removing the gate
shuffled-evidence penalty	0.5	1	confidence after shuffling	placement
reward-scaled penalty	0.5	1	sequence confidence, scaled	placement

failures, ECE rises from 0.440 to 0.478, and accuracy falls from 0.092 to 0.072. The training logs agree with the lemma. The penalty was active throughout training, but the gate left the calibration term at about 0.2 in advantage units, too small next to the correctness signal to change the policy, so the gate suppresses the fix.

Table 14: Training-study results. Every intervention is evaluated against the same evaluation of the untreated baseline and item-matched on the both-wrong core set (HERBench, $n\approx$1273). "×re-run" is |TAX| relative to the baseline re-run difference of +0.07, so an intervention at 1× is indistinguishable from re-running the baseline. Penalized runs vary far more than untreated ones (about 0.6 between runs, §4.2), so a large ratio here does not by itself mean that an effect replicates. AUROC differences should be read against the +0.0095 that separates two untreated runs, which no intervention clearly exceeds. Dashes mark quantities that do not apply or were not measured.

Intervention	TAX(baseline→·)	t	Ordering	Acc.	AUROC	×re-run
untreated baseline	–	–	–	0.354	0.646	–
baseline, re-run	+0.07	+1.08	−0.59	0.350	0.656	1×
no-normalization control	+0.76	+13.0	+0.92	0.352	0.651	11×
gated penalty	+0.08	+1.36	+1.92	0.353	0.653	≈1×
ungated penalty	−0.93	−12.0	−0.61	0.345	0.661	13×
ungated penalty, run 2	−0.33	−3.92	−4.26	0.345	0.653	5×
ungated penalty, run 3	+0.45	+6.21	−1.60	0.350	0.655	6×
shuffled-evidence penalty	−0.14	−2.08	−0.04	0.348	0.653	2×
reward-scaled penalty	+0.32	+5.42	+0.32	0.349	0.657	5×
evidence-dependent, λ=4	−0.33	−4.55	−2.34	0.352	0.648	5×
evidence-dependent, λ=10	−0.86	−10.7	−2.31	0.350	0.656	12×
evidence-dependent, λ=10, run 2	−0.43	−4.97	−3.38	0.346	0.661	6×
evidence-dependent, λ=10, run 3	+0.09	+1.40	−1.47	0.350	–	1×
evidence-dependent, λ=20	+0.01	+0.13	−1.27	0.354	0.643	<1×
adaptive multiplier, seed 1	+0.15	+2.4	−0.67	0.352	–	2×
adaptive multiplier, seed 2	−0.20	−3.1	−1.50	0.349	–	3×
adaptive multiplier, seed 3	−0.76	−9.9	−2.27	0.350	–	11×
answer-span penalty, λ=0.5	+0.36	+5.92	+1.08	0.350	0.652	5×
answer-span penalty, λ=1	−0.07	−1.31	+0.75	0.350	0.658	1×

D.2 THE UNGATED PENALTY ACTS BUT MISSES THE TARGET

Removing the gate lets the penalty act (Table 14). The tax goes negative and significant on ten of the twelve types, separation improves, and the effect splits the way the design intends, since confidence on correct answers falls by 0.4 while confidence on wrong answers falls by 1.1. The cost is 0.9 accuracy points, and aggregate ECE does not move. In this run the primary endpoint inverts. The relief is largest on the types whose diagnosed tax was small or null and nearly smallest on temporal shot ordering (-0.61, $t=-2.9$), so it runs opposite to the diagnosed tax across types, and TVBench shows the same shape, where the tasks about order receive half the relief of the other tasks. That pattern motivated the placement variants below, but it is specific to this run. A replicate of the same

configuration relieves ordering by -4.26 instead of -0.61, and across the three runs of the ungated penalty the aggregate spans -0.93 to $+0.45$ and crosses zero. Several types move further between two runs of the same penalty than between different penalties. We therefore treat the idea that a penalty blind to the reward may leave the excess where the reward keeps pushing as a hypothesis that the remaining variants probe, rather than as an established result.

D.3 Standard-deviation normalization does not explain the video tax

The no-normalization control removes the division by the group standard deviation, as proposed to remove a difficulty bias in GRPO (Liu et al., 2025b). This halves the mean gradient norm (2.59 to 1.32), which confirms that the change took effect. If the tax were an artifact of this normalization (Bereket & Leskovec, 2025), the control should relieve it. It does not (Table 14). The tax is positive on all twelve types, ECE rises, and TVBench is null in both groups of tasks. The normalization is therefore not necessary for the tax in this setting, which limits the scope of the text-domain account without contradicting it. Lemma 1 gives that account a precise place, since normalization enters only through σ_r, which scales the calibration term but does not create its ordering.

D.4 The shuffled-evidence penalty fails in an informative way

The shuffled-evidence penalty computes the confidence of each rollout again after shuffling the frames and penalizes that value. It fails its primary criterion (Table 14). It delivers one sixth of the relief of the ungated penalty, only about twice the difference between two untreated runs, and on temporal shot ordering it does nothing at all ($t = -0.19$). On TVBench the tasks about order move the wrong way ($+0.28$, t=3.27) while the other tasks are null. What little relief it gives moves to the memory types and again runs opposite to the diagnosed tax across types ($\rho = -0.77$, p=0.015), which resembles reward bias substitution, where a penalty on one proxy moves the optimization pressure onto a correlated one (Lamparth et al., 2026). It costs 0.6 accuracy points, and the one criterion it passes is that separation does not degrade on either benchmark. The implementation works as intended, since the shuffled confidence stays below the ordered confidence throughout with gaps of 0.003 to 0.031, so the penalty acted on a real quantity but the wrong one.

The failure is informative for the following reason. Compared with the ungated penalty, the shuffled-evidence version is weaker and, by design, spares the part of the confidence that depends on the temporal evidence, penalizing only what survives shuffling. If the excess confidence did not depend on the evidence, this version should have hit the reward's target harder than the ungated one. Instead the relief collapses, moves to other types, and inverts on TVBench. What RL adds is therefore commitment that depends on the evidence and that the reward amplifies, rather than confidence that floats free of the evidence, and a penalty that spares the evidence-dependent part spares the tax itself. We report this variant as an ablation that separates two explanations, not as a second proposed method.

D.5 Moving the penalty or changing its strength does not rescue it

The reward-scaled penalty multiplies the ungated penalty by $(1+\kappa)$ with κ=1 on prompts where the temporal bonus of the reward fired. We stated three criteria for it before training. It had to relieve temporal shot ordering and the TVBench tasks about order more than the ungated penalty did, it had to preserve accuracy, and it had to leave separation intact. The bonus fired on 85% of temporal prompts, so its mean strength is about 1.9 times that of the ungated penalty. The tax against the untreated baseline is $+0.32$ (t=5.42), with temporal shot ordering at $+0.32$ and TVBench at the level of the untreated re-run, while accuracy and separation do pass (Table 14). This value lies inside the range of the three ungated runs (-0.93 to $+0.45$), so we do not read it as an overshoot, only as no gain from scaling the penalty to the reward.

The reading of §D.4 suggests one further placement, which is to penalize the evidence-dependent part of the confidence directly, $r_i = R_{\text{correctness}}(o_i) - \lambda\,\mathbf{1}[\text{wrong}_i]\,(c_i - c_i^{\mathcal{T}})_+$, where $c_i^{\mathcal{T}}$ is the confidence of the same rollout after shuffling the frames. We set λ=4 so that the calibration term of Lemma 1 matches that of the ungated penalty. The measured gap left this calibration term at 0.07 to 0.19 in advantage units, the range that made the gated penalty inert, so the lemma predicted no effect. The prediction fails. The tax moves to -0.33 ($t = -4.55$) with temporal shot ordering at

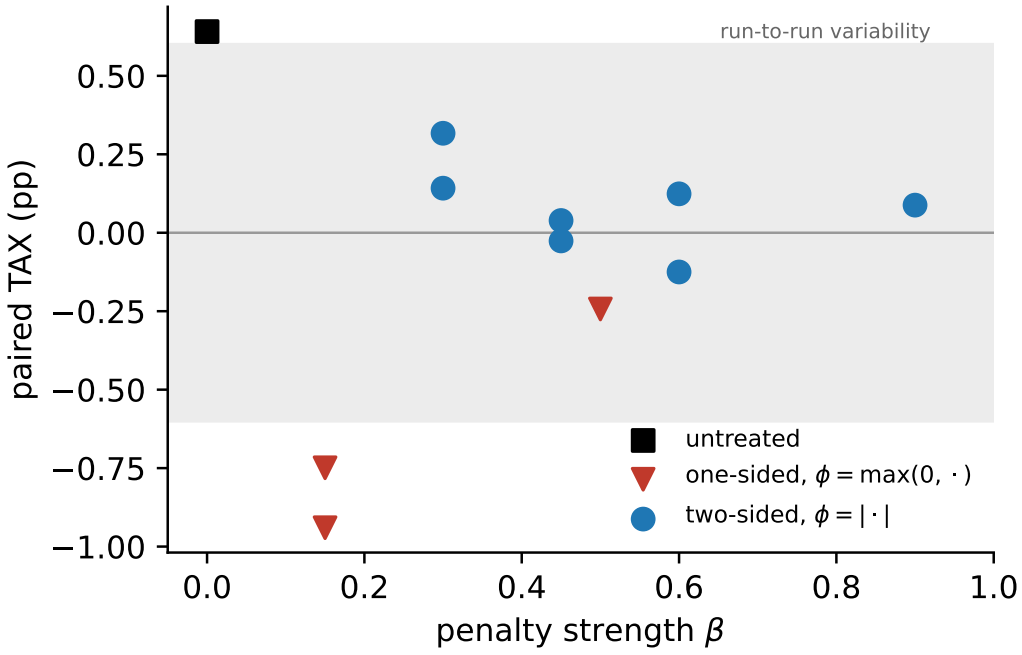

Figure 5: RMA across penalty strength β at 300 steps, plotting the tax column of Table 3. The one-sided form has no fixed point, so its strength sets how far the margin drifts rather than where it stops, and every one-sided run lands below zero, with both runs at β=0.15 outside the run-to-run variability scale. The two-sided form has a fixed point at $m_\theta = m_{\text{ref}}$, and every run at $\beta \geq 0.45$ lands within the run-to-run variability scale.

-2.34 ($t = -6.95$), which makes this the first intervention whose relief follows the diagnosed tax rather than running opposite to it, at accuracy 0.352 and with separation flat. The TVBench tasks about order barely move (-0.08). The size estimate from the lemma therefore under-predicted a signal aimed at the right quantity, because it bounds how large a term is and not where it points.

Raising the strength to λ=10 puts the calibration term at 0.2 to 0.4 in advantage units. The overall relief grows to -0.86, which is within run-to-run variation of the ungated penalty's -0.93. All twelve types move negative with no significant inversions, the accuracy cost is half that of the ungated penalty, and TVBench moves to -0.79. Separation does not degrade. Two further runs at λ=10, however, give -0.43 and $+0.09$, so its relief does not replicate, and its apparent advantage on the reward's target lies inside the ungated penalty's own range across runs. At λ=20, where the calibration term is 0.45 in advantage units, the aggregate effect is $+0.01$ at unchanged accuracy. This value is also inside the range of the replicates, so we do not claim a peak in strength. The two tasks that resist relief at lower strengths, object shuffle and scene transition, move negative only at this strength.

The confidence proxy fails. Restricting the penalized confidence to the answer span, which is the quantity the diagnosis measures, is ineffective at both strengths we tried. At λ=0.5 it raises the tax ($+0.36$), and at λ=1, which matches the strength of the ungated penalty, it sits at the level of the untreated re-run (-0.07). The span locator fired on every rollout, so the failure lies in the proxy and not in the implementation. The two definitions of confidence are related, but the restricted one cannot replace the sequence confidence inside the reward. Of the three levers of the lemma, strength and placement change the result, and the proxy does not.

The adaptive multiplier does not settle. Treating the penalty as a constraint with a multiplier that adapts during training gives $+0.15$, -0.20 and -0.76 across three seeds. The multiplier keeps drifting and does not settle at a fixed strength.

D.6 Pooling across runs

Reading each run separately overstates what a single run shows. We pool all replicates and charge the between-run variance explicitly, with a robust method-of-moments estimate in the spirit of DerSimonian & Laird (1986), which gives σ_{run}=0.80 on HERBench from all pairwise differences between runs of the ungated penalty. Exactly one HERBench type survives pooling in each penalty family, and it is the reward's target. Temporal shot ordering lands at -1.17 $[-1.83, -0.51]$ under the ungated penalty and at -1.30 $[-1.97, -0.62]$ under the evidence-dependent penalty. Nothing else survives in either family. Because ordering is also the type whose confidence is dominated by the meta-option (§B.4), the reproducible effect of penalized training is most safely read as damping certainty in that response rather than as repairing temporal reasoning.

E Readout robustness

At the answer position the Qwen-family checkpoints emit the option letter without a leading space, while our main readout reads the space-prefixed letter tokens. In a 50-item audit the full-vocabulary top five tokens at this position are the five unspaced letters, so the main readout is a fixed probe rather than the emitted distribution. We therefore re-scored the full 2000-item HERBench set under both readouts for the public pair at 16 frames and 224 px. The paired tax $\mathrm{TAX}(\text{SFT}\to\text{RL})$ is $+4.36$ (t=20.2, n=1220) under the main readout and $+2.52$ (t=10.9, n=1228) under the emitted letters. Overall accuracy is the same under both, and the two readouts pick the same option on 83 to 85% of items. The size of the tax therefore depends on which tokens are read, but its sign and significance do not, and both values lie far above the run-to-run variability scale (§4.2). For this reason every comparison in the paper reads both checkpoints the same way and reports only their difference.

Letter-position bias. Across all twelve types the models rarely answer A under the main readout. They choose A on 0 to 7% of items, while A is the correct answer on 2 to 25%. The same bias is present in the base checkpoint and in every post-trained one under this readout, so it does not come from RL. Sensitivity of language models to the position and symbol of an option is well documented (Zheng et al., 2024; Pezeshkpour & Hruschka, 2024). Because the paired estimator reads both checkpoints the same way, the bias cancels out of the tax. It does lower absolute accuracy on types whose correct answer is often A, so single-checkpoint accuracies here should be read with that in mind.

F Limitations

The study covers multiple-choice video QA only. All confidences are read without CoT through a fixed letter probe, and we do not claim that the tax or the effect of RMA carries over unchanged to answers produced after a CoT. The training study uses one backbone, one SFT reference and one training recipe. The InternVL3.5 results show that the tax itself appears in a second model family, but both families share a Qwen-family language model, so independence from the backbone is not established. The contrast between T-GRPO and Open-o3 compares two full pipelines that differ in optimizer, SFT data and auxiliary supervision as well as in reward, so it is an association with the training recipe rather than a controlled attribution to the reward. Finally, most training runs used no fixed rollout seed, so single-run comparisons are read against the run-to-run variability scale of §4.2.