SILENT COLLAPSE: DIAGNOSING AND REPAIRING LOW-PRECISION ROBOT POLICY TRAINING AT THE ACTION LEVEL

Anonymous authors
Paper under double-blind review

ABSTRACT

Robot learning is now trained largely with low precision, bf16 and fp8 alike. Low precision removes information the policy needs during training, and this can be silent: every monitor reads normal, yet the robot no longer produces the right action. We study this failure at the level of the executed action. It is not one failure mode but three, concentrated where the signal is weak along the activation-gradient chain. The two most dangerous ones are new: mantissa starvation and conservative collapse. Both rewrite the action output without warning, with striking unevenness across tasks and seeds. We build a seconds-long probe locating the weak-signal points where failures start and present CURE (Cheapest sUfficient REpair), a routing protocol assigning each failure mode its cheapest sufficient fix: a free analytic bypass where the weak signal has a closed-form gradient, exact backward fidelity from the UPH (Unified Precision Health) layer where it does not, and LockedAdamW, an optimizer locking the per-step noise-displacement budget. We test the routing across policy families from a 0.8M ACT to 7B vision-language-action models under the twin-run protocol. On the ACT collapse cell, the routed scheme brings the collapsed posterior's KL back into the fp32 band; on the 7B CogACT, LockedAdamW breaks below the fp32 self-retraining floor on both seeds; on real training hardware, one tested Transformer Engine fp8 configuration trains normally—consistent with the same mechanism—while unscaled variants fail in simulation. The probe answers in 3.4 seconds at 0.8M and in minutes at 7B; the repair starts free.

1 INTRODUCTION

Most general-purpose robot systems now train at low precision, bf16 or fp8, to reduce training time. The savings are real: on OpenVLA, bf16 steps cost 25% of the non-tf32 fp32 reference (Appendix B; the tf32-tiered ledger is in §7). However, the risk of low precision is not limited to reduced precision itself. A policy trained in bf16 under a textbook protocol, with normal loss and no overflow, can nonetheless emit one constant action regardless of what it sees (Figure 1). The failure is total and behavioral, and no training-time signal announces it: the loss curve is normal, and the overflow counter never fires. Nor is it rare. On susceptible cells, the seed lottery eliminates one run in every two or three, and on Can and Kitchen, every seed collapses. In this paper, we study these silent failures, investigating how the information a policy needs is removed during training, and how training-time signal fails to report it.

Three practical objections may be raised, and each has a hole. Large models appear immune, but their immunity is signal strength, and the weak components are the heads (CVAE, flow, MDN) that produce the action distribution. Production recipes appear immune: the production KL weight trains the same cell normally, but it does so by removing the weak signal itself: immunity by elimination, not by protection. Best-checkpoint evaluation appears to cover the risk, yet a run archived at its peak reads healthy forever. The ceiling is not what precision costs; survival is.

A policy is not a loss curve: two runs with equal validation loss can differ greatly in terms of closed-loop behavior because the executed action is a closed-loop quantity while the loss is an open-loop average. Robot learning requires a closed-loop reading of training quality.

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.

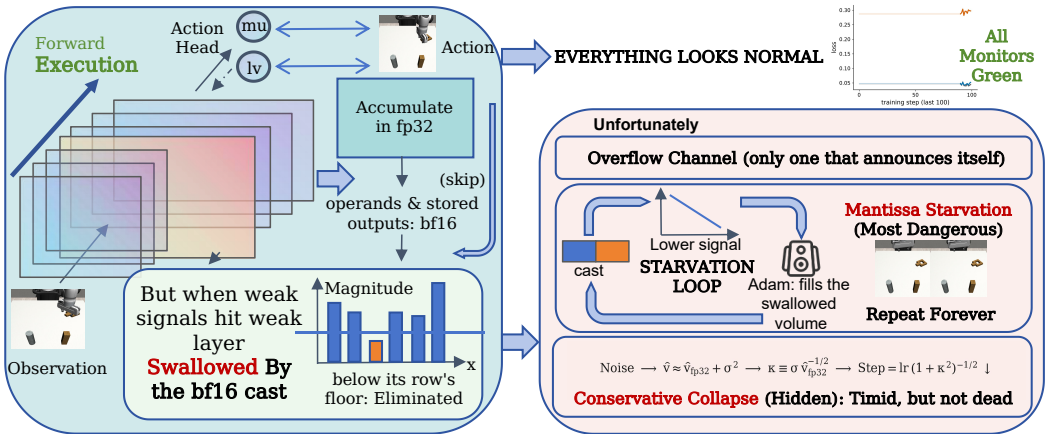


Figure 1: One ratio decides the fate: when the weakest signal falls below the resolution floor of its coordinate system, the floor's handling chooses the failure mode—flushing starves the posterior, raising poisons the stream, overflow clips the update—while Adam amplifies and monitoring stays green; each mode routes to its cheapest sufficient repair (§6).

We therefore measure action correctness on the action stream itself, with two readings: μ_{div}, the mean per-observation L2 distance between an arm's actions and those of its paired fp32 run, the *reference* (the same policy retrained in fp32 under the twin protocol), on held-out observations at matched training steps, and rollout return. These see what the loss curve cannot do: the closed loop and the action level. From such measurements, three modes of numerical failure emerge. The overflow channel is well known; mantissa starvation and conservative collapse, which exist only in mid-to-low-precision training, have not been characterized; both collapse only at weak-signal sites. The "posterior collapse" routinely reported in CVAE robot policies and treated as an objective-level failure with KL annealing, β schedules, or architectural redesign (Kinoshita et al., 2024; Demisse, 2026) is at least in part numerical: the collapsed posterior is rescued by precision means alone, without touching the objective.

This points to the shape of a solution: repair only the weak-signal points during training. We accordingly present CURE, a routing protocol derived rather than tuned, assigning each failure mode its cheapest sufficient scheme: an analytic bypass, free where the weak signal has a closed-form gradient; the UPH layer, restoring backward fidelity where the swallowing condition takes effect; and LockedAdamW, a control law that throttles the nominal rate in proportion to numerical noise inflation and locks the per-step noise-displacement budget. With the system alone, the posterior KL of the ACT collapse cell is restored from 0.01 into the fp32 band (≈3.8), and the action drift is compressed from 15× the floor toward the band; on the normal 93M Octo, the interventions push behavior below the fp32 self-retraining floor; on real hardware, Transformer Engine (TE) fp8 likewise trains normally, with the island arm posting the best reading on one seed. The paper first closes the measure–mechanism–scheme loop, and validates it across policy families from 0.8M to 7B.

2 Related Work

Prior work can be divided into three areas. First, the mixed-precision training stack: from automatic mixed precision and loss scaling to bf16 recipes and fp8 formats—all engineered and evaluated at the loss level (Micikevicius et al., 2018; Zhao et al., 2019; Zamirai et al., 2021; Micikevicius et al., 2022; Rouhani et al., 2023; DeepSeek-AI et al., 2025); recent lines push further in the same direction, removing the loss scaler altogether (Blake et al., 2023), block-quantizing the optimizer's moments (Dettmers et al., 2022), or training in four-bit integers (Xi et al., 2023). Industrial practice has already picked an exponent-side answer: TF32 pairs fp32's exponent range with fp16's mantissa and needs no loss scaling—a choice our patterns explain after the fact (Appendix I). Second, the large-scale failure casebooks: OPT's training logs, GLM-130B's instability reports and their follow-up analyses document divergences and spikes (Zhang et al., 2022; Zeng et al., 2023), and a static a priori predictor of numerical failure has also been proposed (Shang, 2026). These are loud failures, in LLM

pretraining; ours are silent, in robot policy training—every numerical monitor reads normal while the action has already drifted. Third, the posterior-collapse literature on variational architectures treats collapse as a failure of the objective rather than of numerics; recent variants control it by constraining the decoder's Lipschitz constant or by reweighting the variance (Kinoshita et al., 2024; Demisse, 2026).

All three judge training upstream of the action, leaving real embodied behavior unstudied; the repair side addresses only its own piece, and none offers a closed-loop solution (even search-free and adaptive learning-rate methods do not read which failure mode is present (You et al., 2020; Shazeer & Stern, 2018; Defazio & Mishchenko, 2023; Mishchenko & Defazio, 2024; Defazio et al., 2024; Yang et al., 2022)). We fill both gaps: measuring at the action level and making the scheme a function of the measured failure mode. The closest precursor to UPH is SwitchBack's layerwise mixed precision (Wortsman et al., 2023), which switches the forward pass layer by layer; UPH leaves the forward untouched and recomputes only two backward matmuls at probe-named layers—the repair targets the transport, not the representation.

3 Three Numerical Failure Modes

We read behavior on the action stream with μ_{div}, out_std (the action distribution's dynamic range), the KL term (a variational posterior's maintenance readout), and rollout return (the task-level reading). Across twelve policy families three failure modes emerge; the full discovery process is deferred to the appendix, and the five terms used throughout are defined in Appendix A.

The overflow channel. On likelihood-loss policies above canonical rates, fp16's numeric range is too narrow: gradients overflow within the first few dozen steps and never leave. The stock GradScaler halves the scale on each inf; its repeated retreat stalls optimization within approximately twenty steps (appendix). This is the easiest of the three to handle: the overflow counter alarms, and the standard mixed-precision stack is mature. It is also the only one that announces itself; the other two, the truly dangerous, are silent.

Conservative collapse. Conservative collapse does not destroy the policy; it makes its choices conservative. fp16's gradient noise, whose RMS is one-eighth the size of bf16's, still inflates the second moment of AdamW (Kingma & Ba, 2017; Loshchilov & Hutter, 2019); updates become noise-dominated, and their excursion blurs the policy (§4). On the same ACT cell, the fp16 arm drifts sixfold further from the fp32 twin than the tf32 arm does, and a tenfold rate cut restores it (appendix). Because the controller continues to act, this failure is extremely difficult to notice, and tasks demanding a large dynamic range are expected to suffer more.

Mantissa starvation. This is the most dangerous form of the three. bf16's seven stored mantissa bits round the CVAE policy's faint KL maintenance gradient toward zero; the posterior collapses from a normal KL of approximately 3.8 to 0.01, and the starved decoder emits one constant action regardless of changes outside. No overflow, no loss spike, or divergence is reported (Figure 2a), and every static criterion constructed misses it (§5).

Whether starvation strikes varies with task and seed: on one Square cell, the seeds split between full collapse and normal runs, whereas on Can and Kitchen (robomimic suite (Mandlekar et al., 2021)), every seed collapses (Figure 2b, Appendix D). Failure follows the signal, not the task (§4): at the production KL weight (β=10, 1000$\times$ the canonical regime), the same Square cell trains normally on all three seeds (Appendix D). Sub-critical doses halve the dynamic range and behavior drifts while the KL reads normal (Appendix D).

4 Where the Failure Lives

We take the training process apart. Protecting the loss, freezing the head, and post hoc weight quantization all change nothing (Appendix E); no overflow appears in the forward pass. The evidence localizes the failure to the backward pass: debiasing activation gradients restores the policy, debiasing only parameter gradients does not (Appendix E), and a true-fp8 backward with two-mantissa-bit

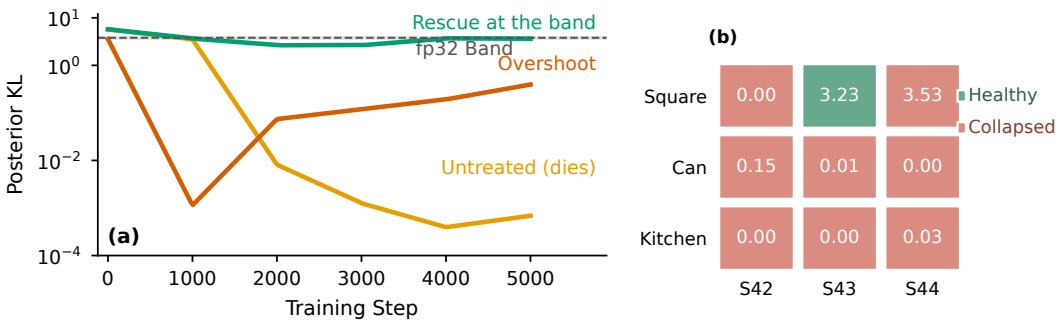


Figure 2: Two silent failure modes on one ruler. (a) Mantissa starvation: a bf16-trained CVAE's KL dies at low β, revives at the starvation-loop model's sweet spot, re-collapses under overshoot. (b) Lottery: seeds split on Square, collapse on all tested seeds on Can/Kitchen. (The overflow residency panel and the conservative-collapse panel are in the appendix.)

parameter gradients remains normal as long as the activation gradients maintain full precision (Appendix E). Failure occurs in the backward activation-gradient chain.

A final control separates the signal from its transport. The maintenance gradient of the posterior has a closed form at the head, so it can bypass the shared channel entirely: the KL term is lifted out of the computation graph, and its exact gradient is added in fp32 directly onto the head, leaving the rest of the backward in bf16. On the Square cell, the posterior stays normal on both seeds (final KL 3.4/3.9 against fp32's 3.9/3.8), and the behavioral drift on the collapse-prone seed is nearly halved (Appendix E). With respect to the harder Can and Kitchen cells, the injected route itself inherits the seed lottery (Table 17); thus, the bypass is read as the decisive mechanism control, not a turnkey solution for every cell. The signal itself needs no rescue; what is fatal is only the leg it travels through the low-precision channel.

In runs heading for collapse, the KL maintenance gradient decays thirtyfold, and its relative rounding error rises from 1% to 13%. One suspect is first cleared: the KL loss value evaluated in bf16 also reads low, an artifact of cancellation in $1 + \ell - \mu^2 - e^\ell$ near $\ell \approx 0$; recomputing that scalar in fp32 reproduces the trajectory bit-for-bit, and the collapse continues regardless. The scalar is not the cause. Every backward-chain matmul is then compared, layer by layer, in terms of both the training precision and fp32. The chain's error is angularly unbiased but radially toward zero: it rotates nothing systematically, yet shrinks magnitudes preferentially at the smallest components. In the collapsing cell, 68% of the KL-gradient component errors shrink toward zero, in contrast to 53% for its fp32 twin (Appendix E), and the relative error amplifies fortyfold from the tensor's largest to its smallest elements; large components pass while small components shrink, and the maintenance gradient of the posterior is precisely a small component. A decisive test follows: zero-mean noise of equal power should leave fp32 normal; a toward-zero bias of the same magnitude should reproduce bf16 failure. The results are exactly so (Figure 3b): the failure is caused not by the magnitude of the noise but by the direction in which the mantissa is eaten. When the second hardware is rerun, the bias arm is healthy on both seeds (the lottery moves with the harness), while the noise arms remain healthy on every draw (Appendix E).

We thus propose a self-reinforcing starvation loop: swallowing the small components of a weak signal weakens it, and the weakened signal is swallowed further. The loop is falsifiable: the spiral model (Appendix E) predicts a rescue band—a fivefold KL weight revives the posterior at its lower edge, a twentyfold weight overshoots and recollapses—both hold. AdamW, meant to normalize noise, aggravates it inside the loop: per-coordinate normalization amplifies small noise into a random walk that fills the swallowed volume (Appendix E). However, rescuing KL does not repair behavior; Section 7 addresses the consequences.

Another failure channel, conservative collapse, lives in the second moment. Rounding noise inflates AdamW's $\hat{v}$ beyond the intrinsic growth that both precisions share: on one stress trajectory, the bf16-to-fp32 ratio increases from 1.00 to 2.01 over 1500 steps, and the update $\mathrm{lr} \cdot \hat{m}/\sqrt{\hat{v}}$ becomes noise dominated. Unlike the finite-sample variance RAdam repairs (Liu et al., 2020), this inflation is numerical noise and never stops. This explains why bf16 runs tolerate aggressive nominal rates

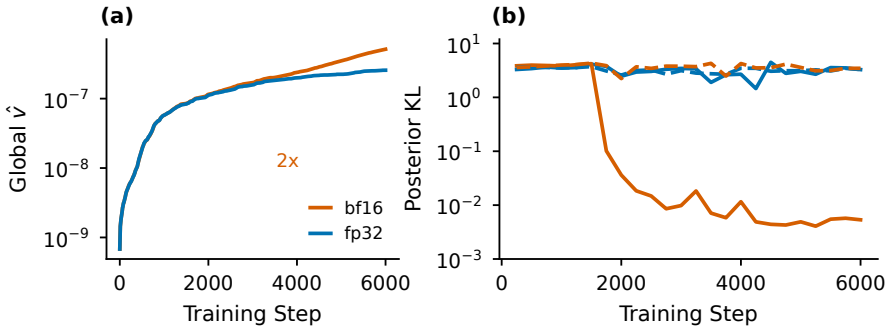


Figure 3: Mechanism evidence. (a) The hidden brake: rounding noise inflates AdamW's $\hat{v}$ beyond its intrinsic growth (the bf16-to-fp32 ratio doubles over 1500 steps on one stress trajectory); washing the backward removes the excess, and the unlocked step diverges (archived runs, not plotted). (b) The decisive cell: equal power, only the direction differs—zero-mean noise (blue) leaves all seeds normal; a toward-zero bias (red) of the same magnitude collapses the collapse-prone seed. (The Pattern-I/Pattern-II sweeps are in the appendix.)

and why a trajectory with clean-washed gradients diverges at the same nominal rate (Appendix E). A brake that responds to noise is a control signal waiting for a control law; Section 6 writes it.

Different tasks carry different signal structures and hence different failure sites. We state two observed patterns, each falsifiable on its measured cells, with the supporting sweeps and instances in Appendix E. **Pattern I**: a layer's required backward-fidelity chain length is monotone in its signal weakness (the E22 sweep: converting layers to fp32 weakest-first, k=0 collapses, k=1–8 progressively rescue, k=9 recollapses at the threshold lottery, and k=12 rescues again). **Pattern II**: low precision does not merely ignore weak signals; by lowering the margin between the signal and the resolution floor of its coordinate system, it manufactures part of their weakness. Were the signal merely ignored, a rescued posterior could at most return to the fp32 equilibrium; that it can exceed it—on Can the bypass posts KL 4.97 against fp32's 2.41, on one seed—means the equilibrium itself was held down by the same floor.

The two patterns are now reduced to one ratio: the signal's position relative to the resolution floor of its coordinate system. Rounding alone does not end a training run—chain quantization under an eight-bit exponent remains normal from 23 mantissa bits all the way down to 2, even at a 12.5% noise level. What eliminates the signal is a floor that cuts into the gradient distribution: flushing then deletes it (starvation), while raising inflates it (poisoning)—one floor, two failure modes. The ratio is stress-tested across the tested policy families in §7.

5 Criteria: Reading Failure in Seconds

The experiments above make one core point: training failure appears only in training dynamics, so the criterion must itself be a short training run. The probe runs a 50-step twin of bf16 against fp32 and reads task sensitivity from the 50-step divergence; on seven family–task cells, this ranking attains a rank correlation of 0.86 (95% CI [0.38, 1.00]) with full-run drift (Appendix F). The inf-gradient fraction gives an overflow residency rate predicting the overflow channel. Criteria are rerun per training; at 3.4 s on the 0.8M cell, minutes at 7B, the cost is negligible. The criteria system outputs a routing table: likelihood heads (MDN, DP) through the twin probe and ρ_{inf} to the EPA thermostat (Appendix F); weak-signal heads (CVAE) through the twin probe and weight drift to the closed-form bypass at the head or to UPH coverage further upstream (Pattern I); and transformer policies through the twin probe to LockedAdamW (§6).

Routing is validated prospectively: on the collapse cell, the probe reads late divergence, the tree commits to the head bypass (decision is timestamped before training), and the run is trained normally (final KL of 3.6, range of 0.567); on a healthy cell, it correctly prescribes nothing (Appendix F). A 7B cell's 16-step alarm matches its failing outcome and the lock's rescue; a healthy 93M cell is overread once; a false alarm costs one cheap repair, and a missed alarm costs a run (Appendix F).

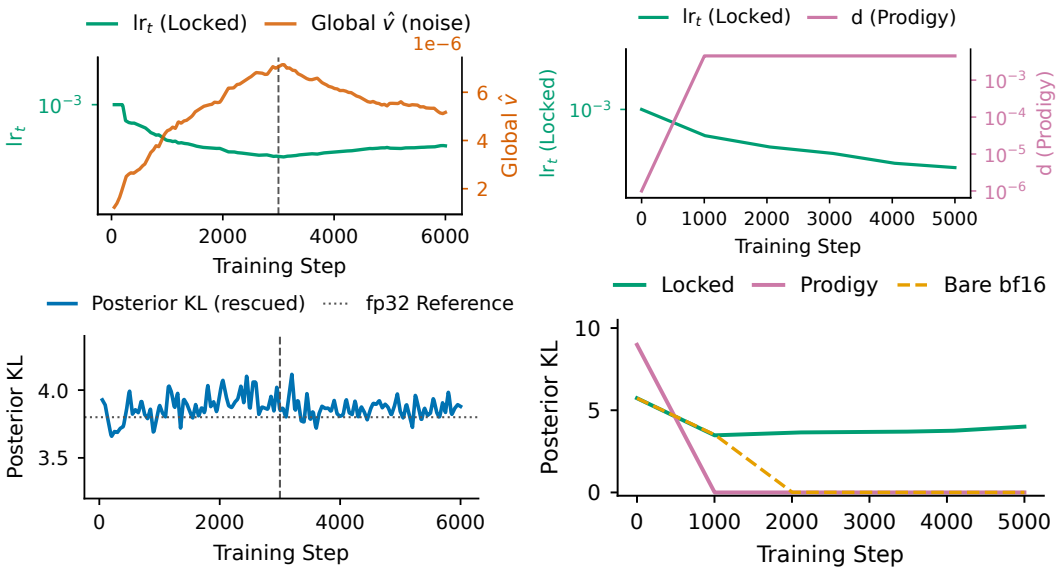


Figure 4: Left: LockedAdamW's signature run—from bf16 noise the controller brakes from 10^{-3} toward the 1.1×10^{-4} band (top: lr_t, global $\hat{v}$); the posterior stays alive through the noisy phase (bottom: KL), releasing 17% after the fp32 switch. Right: search-free baselines on the same cell—Prodigy's rate climbs into the noise and the posterior collapses, D-Adaptation diverges outright; LockedAdamW brakes and rescues (green: LockedAdamW; pink: Prodigy; orange dashed: bare bf16).

6 Schemes: The Cheapest Sufficient Repair per Failure Mode

Once action-level numerical health is the measurement target, repair is a consequence rather than a choice: the probe names the failure mode, and the scheme must match it. Stream repair downstream of the floor does not work for structural reasons: on the activation-gradient chain, loss occurs at the global absorption of summation nodes—the discarded low-order content has no source at any single hook point (four experiments, Appendix G). Trained systems are therefore split by whether the weak signal has a closed-form gradient at the head: with one, inject it exactly (Scheme 1); without one, increase the precision where the criteria point (Scheme 2). Two further channels live off this chain: overflow is closed by the EPA thermostat, and noise-inflation chronic failure by the LockedAdamW control law.

The failure's causal chain also prescribes the schemes' construction. Its root is that the signal is weak (Pattern I); the trigger is the signal sinking below the resolution floor of its coordinate system; the branch is how the floor handles it—flushing deletes (starvation), raising poisons, overflow truncates; the amplifier is Adam's per-coordinate normalization; and the terminal state is silence on the operational monitors—loss, gradient norms, the overflow counter—while the action has already drifted (Appendix D). Each scheme pins one edge: the analytic bypass keeps the signal out of the low-resolution coordinate system altogether (before the branch), UPH lowers the floor locally at probe-named layers (at the branch), LockedAdamW locks the amplifier's noise-displacement budget (after the branch), and the EPA thermostat keeps any residual risk loud. Anything pinned downstream of the branch is therefore unable to recover, from the stream, a signal already deleted or drowned (Appendix G's impossibility theorem).

Scheme 1: Analytic bypass. Some weak signals never need the channel at all because their gradients with respect to the head have closed forms: the KL maintenance gradient of a variational posterior, the log-det gradient of an invertible flow's scale network, variance-calibration gradients of Gaussian or mixture heads, entropy and Lagrangian terms. For such signals, the auxiliary term is lifted out of the computation graph and its exact gradient is added in fp32 directly onto the head, leaving the rest of the backward in bf16. The bypass is Section 4's control read as a scheme, and

its evidence is a family: it rescues the collapsed posterior, gives a mild gain on the invertible flow's log-det lane, and is measurably inert on normal families—all at zero cost.

Scheme 2: UPH layer. The bypass only covers weak signals with closed-form gradients. For those without, the backward pass is recomputed at probe-named layers with higher precision—the UPH layer.

UPH is a drop-in wrapper: the forward pass is unchanged, and only the wrapped module's two backward matmuls are recomputed in fp32; at probe-named layers, stochastic rounding adds a dither to activation gradients that are unbiased in expectation ($\mathbb{E}[\mathrm{SR}(x)] = x$), removing the systematic bias of rounding. The design follows from a single inequality: a component g of an activation gradient, relative to the contraction scale $|a|$ of its matmul row (the row's RMS operand magnitude), is swallowed when

$$|g| \;<\; 2^{-8}\,|a| \quad \xrightarrow{\text{UPH recompute}} \quad |g| \;<\; 2^{-24}\,|a|\,, \tag{1}$$

that is, fp32 recompute lowers the resolution floor from 2^{-8} to 2^{-24}, and the swallowed component survives. (On hardware the floor acts at the chain's bf16-cast operands and stored outputs; tensor-core GEMMs nominally accumulate in fp32.) The measurements agree with Section 4's failure map and Pattern I: the backward error is direction-unbiased; Can's damage also lives in the head's forward representation, so a mapped head sometimes needs an fp32 forward as well. Ablations attribute three quarters of the gain to recompute and one quarter to dither. An fp8 variant applies the same wrapper with true e4m3/e5m2 arithmetic, and its fp32 activation gradient covers the whole chain at once (Appendix I).

Scheme 3: LockedAdamW. Rounding noise additively inflates AdamW's second moment, $\hat{v} \approx \hat{v}_{\mathrm{fp32}} + \sigma^2$. Write $\kappa \equiv \sigma/\sqrt{\hat{v}_{\mathrm{fp32}}}$: the effective step is compressed to $1/\sqrt{1+\kappa^2}$ of its size and to approximately $1/\kappa$ when noise dominates. LockedAdamW calibrates the rate against the measured inflation: a short warmup self-calibrates $\hat{v}_{\mathrm{ref}}$, after which

$$\mathrm{lr}_t \;=\; \mathrm{lr}_{\mathrm{base}} \cdot \sqrt{\hat{v}_{\mathrm{ref}}/\hat{v}_t} \;\approx\; \mathrm{lr}_{\mathrm{base}}/\kappa\,. \tag{2}$$

What it locks is the per-step noise–displacement budget: composing the lock with Adam's intrinsic $1/\sqrt{\hat{v}_t}$ normalization, the effective step falls to $1/\sqrt{1+\kappa^2}$ (intrinsic) $\times\, 1/\sqrt{1+\kappa^2}$ (lock) $= 1/(1+\kappa^2)$, a deliberate overbrake relative to the hidden brake's own $1/\sqrt{1+\kappa^2}$. The lock is reference-free: an external fp32 twin moves results by under 2% of drift.

On the collapse cell, the controller brakes from 10^{-3} to 1.1×10^{-4} (Figure 4), releasing only 17% in the clean phase (Appendix G); Prodigy's adapted rate climbs into the noise and the posterior collapses, and D-Adaptation diverges outright (Figure 4, right). Nor is the lock the only Adam-side answer: at the canonical budget, four classical knobs (error feedback with a Kahan carry, ϵ=10^{-6}, β_2=0.95, clip 0.1) each keep the cell healthy, and error feedback joins the routing table (Appendix G). The lock's case is the domain nobody hand-tunes—still training at 150k on Square (§7) and below the floor on CogACT on both seeds—with a measured boundary: fails at 51k on Lift, NaN past 150k on Square (Appendix G).

The kit is a decision tree, not an assortment: without a closed form, UPH is the measured price of fidelity, placed by Pattern I; a $\hat{v}$-inflation chronic failure goes to LockedAdamW, the zero-search automation of a manual rate cut (its measured domain boundary is the GR00T cell); and where the probe is normal and the census finds no weak layers, the tree prescribes nothing—which, on the normal models in our matrix, is exactly the win. In addition, where a format change is on the table, the tree routes to it as well: fp16 under the stock GradScaler collapses on no spectrum seed, and tf32 wins image-token-heavy loads (§7).

7 The Validation Matrix

In the preceding chapters, a chain is constructed: three failure modes exist (§3), the mechanism is mapped (§4), seconds-long criteria route (§5) and routing repairs (§6) are used. This chapter validates the chain claim by claim. Every cell follows the twin-run discipline: at most six arms plus the fp32 self-retraining pair as the noise floor; small-matrix cells train 6000 steps; ladder cells 1k–8k; determinations read the whole ladder; and readings below the floor are directional (Appendix J).

arm (MH-Lift, 150k)	collapse	peak	final
fp32 reference	0/3	64–68	48–68
bypass + LockedAdamW + EMA	**1/6**	**62–74**	**52–74**
LockedAdamW + EMA	1/3	62–68	58–60
bare bf16	1/3	62–72	0–54
UPH	3/3	62–70	0
bypass	3/3	62–68	0–4

Table 1: MH-Lift, three seeds (combined arm: six runs incl. two lottery redraws): collapse rate, best-checkpoint success, end-of-training success (%). Single-layer schemes collapse in every run; the combined arm alone pushes the rate below bare bf16.

At scale, failure follows signal weakness, not scaling. The ladder spans nearly four orders of magnitude of embodied policies (Table 2): a 0.8M ACT, the 93M Octo generalist, the 1.2B RDT, the 3B-class $\pi_{0.5}$ and GR00T, the 7B OpenVLA-OFT and CogACT, plus a from-scratch 450M SmolVLA cell. ACT's small CVAE head collapses in a seed lottery, whereas from 93M, it is normally an untreated bf16 arm train—except at the flagged edge, where even the fp32 reference itself can draw a losing seed (OpenVLA-OFT, below). The from-scratch cell rules out a protocol confound: within this window, training bf16 from scratch is no more failure prone than fine-tuning. Failure prediction runs on the signal structure rather than on the parameter count—this is what allows the criteria route to run in seconds.

Routing delivers exactly where the criteria point (Table 2): the best scheme tracks each row's Mode, with starvation rows closed by the closed-form bypass or error feedback (Appendix G) and conservative rows by LockedAdamW. On ACT, UPH revives the square's posterior on two of three seeds and fails on Can and Kitchen, where Pattern I places the failure site further upstream; LockedAdamW partially restores the timid transformers on all three tasks. On the healthy Octo, UPH and UPH+LockedAdamW read below the fp32 floor (2.4× and 1.8×, Figure 5a)—a directional reading only: the ruler cannot rank healthy arms (its resolution ends at the floor, Appendix J); on the same model, LockedAdamW brakes noise that does not exist. On GR00T, whose flow-matching objective remains $\hat{v}$ high, the lock misfires on one harness and sits in-band on a second harness: the misfire is real but protocol sensitive (Appendix G). On CogACT, untreated bf16 is damaged on both seeds, UPH changes nothing on both, and LockedAdamW breaks below the floor on both (1.19/1.27 against the band 1.79–2.35): the failure lives in the second moment, not the chain, as the routing table predicts. On OpenVLA-OFT, the fp32 reference itself lotteries NaN mid-run while both bf16 arms train through on the same data: the lottery does not pick its victim by precision.

Routing is a decision procedure, not a toolbox: intervening off the criteria's point does harm, and where the probe is normal, the correct intervention is none.

MH-Lift: matching fp32 at the peak, outlasting bf16 through the band. On the MH split of Lift (300 diverse demonstrations, 150k steps; Table 1), the combined repair—analytic bypass, LockedAdamW, and an EMA deployment—matches the fp32 reference at the peak and posts the campaign's highest reading (74%); it is also the only scheme that pushes the collapse rate below bare bf16, where both single-layer schemes fail in every run. The zero-collapse record remains fp32 alone (0/3), and one combined run still draws the band's ticket (Appendix H).

The routing cost is measured. The wall-clock is measured every 8k steps, bf16 against tf32, the same stack inside every twin pair, across four regimes (Figure 5b): Octo is level; on RDT's image-token-heavy load bf16 is 56% slower; $\pi_{0.5}$'s dual-encoder stack is 64–99% slower; on GR00T bf16 is, for the first time in this study, 32% faster. "bf16 is faster" is a claim tiered by workload, not an axiom (Appendix H): tf32 for image-token-heavy training, bf16 where weight GEMMs dominate and the trunk is frozen. Intervention overhead is bounded (UPH 10–20%, LockedAdamW under 10%)—the "low cost" is thereby documented. On real training hardware (Blackwell, TE fp8), the same Square collapse cell also trains normally, matching the simulator cell by cell; the same measurements prescribe a minimal 9-bit format (e6m2) that runs the cell with no scaling machinery (Appendix I).

The signature is already behavioral. Across policy families, tasks, precision formats, and seeds, rising μ_{div} tracks degraded rollout return (Spearman -0.73 over the 21 cells).

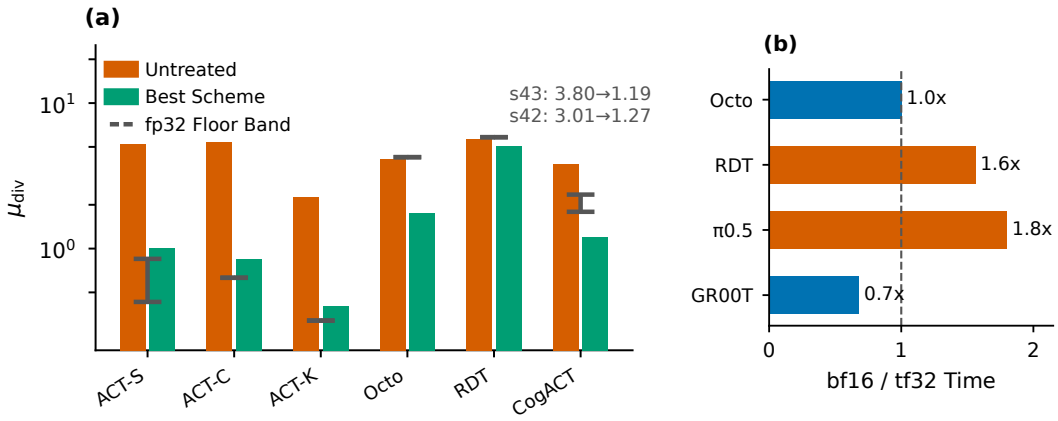


Figure 5: The scale validation matrix. (a) Ladder: per-arm drift, 0.8M–7B (bars: untreated vs best scheme; gray: fp32 floor band); failure follows signal weakness. Not bar-plotted: $\pi_{0.5}$ (all arms within the floor spread), GR00T (in-band on a second harness, 8.85–13.48), OpenVLA-OFT (fp32 reference lotteried NaN; both bf16 arms trained through to final loss 0.133/0.096). (b) Wall-clock ledger per 8k steps, bf16 against tf32 ($\pi_{0.5}$ bar: range mean over the 64–99% interval).

Model	Mode	Untreated	Best scheme
ACT Square (0.8M)	starvation	5.17 ± 1.26	LockedAdamW 1.00 ± 0.05
ACT Can (0.8M)	starvation	5.72	LockedAdamW 0.97†
ACT Kitchen (0.8M)	starvation	2.24	LockedAdamW 0.48†
Octo (93M)	healthy	4.13	UPH **1.75**
RDT (1.2B)	healthy	5.63	LockedAdamW **5.06**
$\pi_{0.5}$ (3B)	healthy	all arms within the floor spread	
GR00T (3B)	misfire	in-band	in-band
CogACT (7B)	conservative	3.80 / 3.01	LockedAdamW **1.19** / 1.27
OpenVLA-OFT (7B)	lottery	0.133 (final loss)	UPH+LockedAdamW **0.096**

Table 2: The validation matrix (μ_{div} against the fp32 twin; best scheme per row; slashed values read s43/s42). The Mode column names each cell's dominant failure mode: starvation (weak-signal head), conservative (second-moment inflation), lottery (threshold seed lottery), healthy, misfire (healthy run braked by the lock). Alternates, floor values, and harness notes: Appendix H. †Partial repair of the conservative component, floor not reached; OpenVLA-OFT is unanchored (loss-level readings only).

8 Discussion

Two patterns summarize the account. Pattern I: the required backward-fidelity chain length tracks signal weakness: weak-signal heads collapse, while billion-parameter strong-signal models are trained normally untreated. Pattern II: low-precision manufactures weakness; failure lives in transport: a closed-form gradient needs only the bypass and no local repair. Conservative collapse is not a present failure but a future constraint: overconservatism becomes the binding limit as tasks grow complex and multimodal. The field's own trajectory corroborates the account: architectures keep moving toward stronger signals—signal strength, not scale—has been the hidden variable all along. One reporting practice is part of the phenomenon: the best-checkpoint convention does not merely tolerate failure; it is how the failure remains invisible in the literature, where every published low-precision result may be a survivor. Against this backdrop, we state the scope and limits of our account. The failure we study lives in the numerical layer of policy training, and is characterized there in full: its causes, mechanisms, and first behavioral manifestation are all read off the training process, without physical hardware. We do not claim that its magnitude or task-level consequences transfer identically to physical robots; real-robot validation remains future work. Further limits: the success bit resolves thus far only on MH-Lift (elsewhere at the floor, fp32 too, to 300k steps); the probe overreads healthy cells; the lock has a measured domain boundary; and one harness per 7B.

Reproducibility Statement

Every reported measurement traces to an archived JSON record produced under the twin-run protocol (paired initialization and data order). Small-matrix cells ran on a consumer GPU; ladder cells ran on rented datacenter GPUs, one per model. Appendix K lists environments, the archive index (mapping every number to its JSON record), and protocols. Anonymous source code—including UPH, LockedAdamW, and all experiment scripts—is provided with the supplementary material.

References

Dan Alistarh, Demjan Grubic, Jerry Li, Ryota Tomioka, and Milan Vojnovic. Qsgd: Communication-efficient sgd via gradient quantization and encoding. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/6c340f25839e6acdc73414517203f5f0-Paper.pdf.

NVIDIA: Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi "Jim" Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, Joel Jang, Zhenyu Jiang, Jan Kautz, Kaushil Kundalia, Lawrence Lao, Zhiqi Li, Zongyu Lin, Kevin Lin, Guilin Liu, Edith Llontop, Loic Magne, Ajay Mandlekar, Avnish Narayan, Soroush Nasiriany, Scott Reed, You Liang Tan, Guanzhi Wang, Zu Wang, Jing Wang, Qi Wang, Jiannan Xiang, Yuqi Xie, Yinzhen Xu, Zhenjia Xu, Seonghyeon Ye, Zhiding Yu, Ao Zhang, Hao Zhang, Yizhou Zhao, Ruijie Zheng, and Yuke Zhu. GR00T N1: An Open Foundation Model for Generalist Humanoid Robots, 2025. URL https://arxiv.org/abs/2503.14734.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A Vision-Language-Action Flow Model for General Robot Control, 2026. URL https://arxiv.org/abs/2410.24164.

Charlie Blake, Douglas Orr, and Carlo Luschi. Unit Scaling: Out-of-the-Box Low-Precision Training, 2023. URL https://arxiv.org/abs/2303.11257.

Remi Cadene, Simon Aliberts, Francesco Capuano, Michel Aractingi, Adil Zouitine, Pepijn Kooijmans, Jade Choghari, Martino Russi, Caroline Pascal, Steven Palma, Mustafa Shukor, Jess Moss, Alexander Soare, Dana Aubakirova, Quentin Lhoest, Quentin Gallouédec, and Thomas Wolf. LeRobot: An Open-Source Library for End-to-End Robot Learning, 2026. URL https://arxiv.org/abs/2602.22818.

Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin CM Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Robotics: Science and Systems*, 2023.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jiawei Wang, Jin Chen, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Litong Wang, Liyue Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J. Chen, R. L. Jin, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runxin Xu, Ruoyu Zhang, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaoqing Wu, Shengfeng Ye, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Shuting Pan, T. Wang, Tao Yun, Tian Pei, Tianyu Sun, W. L. Xiao, Wangding Zeng, Wanjia Zhao,

Wei An, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, X. Q. Li, Xiangyue Jin, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaojin Shen, Xiaokang Chen, Xiaokang Zhang, Xiaosha Chen, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingkai Yu, Xinnan Song, Xinxia Shan, Xinyi Zhou, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, Y. K. Li, Y. Q. Wang, Y. X. Wei, Y. X. Zhu, Yang Zhang, Yanhong Xu, Yanhong Xu, Yanping Huang, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Li, Yaohui Wang, Yi Yu, Yi Zheng, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Ying Tang, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yu Wu, Yuan Ou, Yuchen Zhu, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yukun Zha, Yunfan Xiong, Yunxian Ma, Yuting Yan, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Z. F. Wu, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhen Huang, Zhen Zhang, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhipeng Xu, Zhiyu Wu, Zhongyu Zhang, Zhuoshu Li, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Ziyi Gao, and Zizheng Pan. DeepSeek-V3 Technical Report, 2025. URL https://arxiv.org/abs/2412.19437.

Aaron Defazio and Konstantin Mishchenko. Learning-Rate-Free Learning by D-Adaptation, 2023. URL https://arxiv.org/abs/2301.07733.

Aaron Defazio, Xingyu Yang, Harsh Mehta, Konstantin Mishchenko, Ahmed Khaled, and Ashok Cutkosky. The road less scheduled. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 9974–10007. Curran Associates, Inc., 2024. doi: 10.52202/079017-0320. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/136b9a13861308c8948cd308ccd02658-Paper-Conference.pdf.

Girum Demisse. λ-VAE: Variance Equalization for Posterior Collapse, 2026. URL https://arxiv.org/abs/2607.05531.

Tim Dettmers, Mike Lewis, Sam Shleifer, and Luke Zettlemoyer. 8-bit Optimizers via Block-wise Quantization. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=shpkpVXzo3h.

Zhen Dong, Zhewei Yao, Amir Gholami, Michael W. Mahoney, and Kurt Keutzer. Hawq: Hessian aware quantization of neural networks with mixed-precision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

Massimiliano Fasi, Nicholas J. Higham, Mantas Mikaitis, and Srikara Pranesh. Numerical behavior of NVIDIA tensor cores. *PeerJ Computer Science*, 7:e330, 2021.

Suyog Gupta, Ankur Agrawal, Kailash Gopalakrishnan, and Pritish Narayanan. Deep Learning with Limited Numerical Precision, 2015. URL https://arxiv.org/abs/1502.02551.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering Diverse Domains through World Models, 2024. URL https://arxiv.org/abs/2301.04104.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a Vision-Language-Action Model with Open-World Generalization, 2025. URL https://arxiv.org/abs/2504.16054.

Sai Praneeth Karimireddy, Quentin Rebjock, Sebastian U. Stich, and Martin Jaggi. Error Feedback Fixes SignSGD and other Gradient Compression Schemes, 2019. URL https://arxiv.org/abs/1901.09847.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn.

OpenVLA: An Open-Source Vision-Language-Action Model, 2024. URL https://arxiv.org/abs/2406.09246.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-Tuning Vision-Language-Action Models: Optimizing Speed and Success, 2025. URL https://arxiv.org/abs/2502.19645.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization, 2017. URL https://arxiv.org/abs/1412.6980.

Yuri Kinoshita, Kenta Oono, Kenji Fukumizu, Yuichi Yoshida, and Shin ichi Maeda. Controlling Posterior Collapse by an Inverse Lipschitz Constraint on the Decoder Network, 2024. URL https://arxiv.org/abs/2304.12770.

Qixiu Li, Yaobo Liang, Zeyu Wang, Lin Luo, Xi Chen, Mozheng Liao, Fangyun Wei, Yu Deng, Sicheng Xu, Yizhong Zhang, Xiaofan Wang, Bei Liu, Jianlong Fu, Jianmin Bao, Dong Chen, Yuanchun Shi, Jiaolong Yang, and Baining Guo. CogACT: A Foundational Vision-Language-Action Model for Synergizing Cognition and Action in Robotic Manipulation, 2024. URL https://arxiv.org/abs/2411.19650.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow Matching for Generative Modeling. In *The Eleventh International Conference on Learning Representations* , 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. On the Variance of the Adaptive Learning Rate and Beyond. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=rkgz2aEKDr.

Songming Liu, Lingxuan Wu, Bangguo Li, Hengkai Tan, Huayu Chen, Zhengyi Wang, Ke Xu, Hang Su, and Jun Zhu. RDT-1B: a Diffusion Foundation Model for Bimanual Manipulation, 2025. URL https://arxiv.org/abs/2410.07864.

Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization, 2019. URL https://arxiv.org/abs/1711.05101.

Ajay Mandlekar, Danfei Xu, Josiah Wong, Soroush Nasiriany, Chen Wang, Rohun Kulkarni, Li Fei-Fei, Silvio Savarese, Yuke Zhu, and Roberto Martín-Martín. What Matters in Learning from Offline Human Demonstrations for Robot Manipulation, 2021. URL https://arxiv.org/abs/2108.03298.

Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, and Hao Wu. Mixed Precision Training, 2018. URL https://arxiv.org/abs/1710.03740.

Paulius Micikevicius, Dusan Stosic, Neil Burgess, Marius Cornea, Pradeep Dubey, Richard Grisenthwaite, Sangwon Ha, Alexander Heinecke, Patrick Judd, John Kamalu, Naveen Mellempudi, Stuart Oberman, Mohammad Shoeybi, Michael Siu, and Hao Wu. FP8 Formats for Deep Learning, 2022. URL https://arxiv.org/abs/2209.05433.

Konstantin Mishchenko and Aaron Defazio. Prodigy: An Expeditiously Adaptive Parameter-Free Learner, 2024. URL https://arxiv.org/abs/2306.06101.

Igor Molybog, Peter Albert, Moya Chen, Zachary DeVito, David Esiobu, Naman Goyal, Punit Singh Koura, Sharan Narang, Andrew Poulton, Ruan Silva, Binh Tang, Diana Liskovich, Puxin Xu, Yuchen Zhang, Melanie Kambadur, Stephen Roller, and Susan Zhang. A Theory on Adam Instability in Large-Scale Machine Learning, 2023. URL https://arxiv.org/abs/2304.09871.

Bita Darvish Rouhani, Ritchie Zhao, Ankit More, Mathew Hall, Alireza Khodamoradi, Summer Deng, Dhruv Choudhary, Marius Cornea, Eric Dellinger, Kristof Denolf, Stosic Dusan, Venmugil Elango, Maximilian Golub, Alexander Heinecke, Phil James-Roxby, Dharmesh Jani, Gaurav Kolhe, Martin Langhammer, Ada Li, Levi Melnick, Maral Mesmakhosroshahi, Andres Rodriguez, Michael Schulte, Rasoul Shafipour, Lei Shao, Michael Siu, Pradeep Dubey, Paulius Mi-

cikevicius, Maxim Naumov, Colin Verrilli, Ralph Wittig, Doug Burger, and Eric Chung. Microscaling Data Formats for Deep Learning, 2023. URL https://arxiv.org/abs/2310.10537.

Christopher De Sa, Megan Leszczynski, Jian Zhang, Alana Marzoev, Christopher R. Aberger, Kunle Olukotun, and Christopher Ré. High-Accuracy Low-Precision Training, 2018. URL https://arxiv.org/abs/1803.03383.

Frank Seide, Hao Fu, Jasha Droppo, Gang Li, and Dong Yu. 1-Bit Stochastic Gradient Descent and its Application to Data-Parallel Distributed Training of Speech DNNs. In *INTERSPEECH*, 2014.

Nur Muhammad Shafiullah, Zichen Cui, Ariuntuya (Arty) Altanzaya, and Lerrel Pinto. Behavior transformers: Cloning k modes with one stone. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 22955–22968. Curran Associates, Inc., 2022. doi: 10.52202/068431-1668. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/90d17e882adbdda42349db6f50123817-Paper-Conference.pdf.

Zekai Shang. Auditing Invisible Weight Updates with Reference Traces, 2026. URL https://arxiv.org/abs/2607.09800.

Noam Shazeer and Mitchell Stern. Adafactor: Adaptive Learning Rates with Sublinear Memory Cost, 2018. URL https://arxiv.org/abs/1804.04235.

Mustafa Shukor, Dana Aubakirova, Francesco Capuano, Pepijn Kooijmans, Steven Palma, Adil Zouitine, Michel Aractingi, Caroline Pascal, Martino Russi, Andres Marafioti, Simon Alibert, Matthieu Cord, Thomas Wolf, and Remi Cadene. SmolVLA: A Vision-Language-Action Model for Affordable and Efficient Robotics, 2025. URL https://arxiv.org/abs/2506.01844.

Wenhui Tan, Bei Liu, Junbo Zhang, Ruihua Song, and Jianlong Fu. RoLD: Robot Latent Diffusion for Multi-task Policy Modeling, 2024. URL https://arxiv.org/abs/2403.07312.

Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, Jianlan Luo, You Liang Tan, Lawrence Yunliang Chen, Pannag Sanketi, Quan Vuong, Ted Xiao, Dorsa Sadigh, Chelsea Finn, and Sergey Levine. Octo: An Open-Source Generalist Robot Policy, 2024. URL https://arxiv.org/abs/2405.12213.

Aaron van den Oord, Oriol Vinyals, and koray kavukcuoglu. Neural discrete representation learning. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/7a98af17e63a0ac09ce2e96d03992fbc-Paper.pdf.

Mitchell Wortsman, Tim Dettmers, Luke Zettlemoyer, Ari Morcos, Ali Farhadi, and Ludwig Schmidt. Stable and low-precision training for large-scale vision-language models. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 10271–10298. Curran Associates, Inc., 2023. doi: 10.52202/075280-0451. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/20bd42d82998bc61732c00452228e814-Paper-Conference.pdf.

Haocheng Xi, ChangHao Li, Jianfei Chen, and Jun Zhu. Training transformers with 4-bit integers. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 49146–49168. Curran Associates, Inc., 2023. doi: 10.52202/075280-2137. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/99fc8bc48b917c301a80cb74d91c0c06-Paper-Conference.pdf.

Greg Yang, Edward J. Hu, Igor Babuschkin, Szymon Sidor, Xiaodong Liu, David Farhi, Nick Ryder, Jakub Pachocki, Weizhu Chen, and Jianfeng Gao. Tensor Programs V: Tuning Large Neural Networks via Zero-Shot Hyperparameter Transfer, 2022. URL https://arxiv.org/abs/2203.03466.

Yang You, Jing Li, Sashank Reddi, Jonathan Hseu, Sanjiv Kumar, Srinadh Bhojanapalli, Xiaodan Song, James Demmel, Kurt Keutzer, and Cho-Jui Hsieh. Large Batch Optimization for Deep Learning: Training BERT in 76 minutes, 2020. URL https://arxiv.org/abs/1904.00962.

Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Avnish Narayan, Hayden Shively, Adithya Bellathur, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-World: A Benchmark and Evaluation for Multi-Task and Meta Reinforcement Learning, 2021. URL https://arxiv.org/abs/1910.10897.

Pedram Zamirai, Jian Zhang, Christopher R. Aberger, and Christopher De Sa. Revisiting BFloat16 Training, 2021. URL https://arxiv.org/abs/2010.06192.

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, Weng Lam Tam, Zixuan Ma, Yufei Xue, Jidong Zhai, Wenguang Chen, Peng Zhang, Yuxiao Dong, and Jie Tang. GLM-130B: An Open Bilingual Pre-trained Model, 2023. URL https://arxiv.org/abs/2210.02414.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. OPT: Open Pre-trained Transformer Language Models, 2022. URL https://arxiv.org/abs/2205.01068.

Ruizhe Zhao, Brian Vogel, and Tanvir Ahmed. Adaptive Loss Scaling for Mixed Precision Training, 2019. URL https://arxiv.org/abs/1910.12385.

Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning Fine-Grained Bimanual Manipulation with Low-Cost Hardware, 2023. URL https://arxiv.org/abs/2304.13705.

AI USE STATEMENT

The initial draft was written by the authors. We used generative AI tools to assist with code writing and language polishing; the text was subsequently edited by a professional language-editing service. All experiments were designed and directed by the authors; all experimental data come from deterministic archived runs, and all AI-assisted code was executed and verified against archived outputs. The authors performed the final review of the complete manuscript and take full responsibility for its content.

Reading guide. Sections A–K follow the main text in order, each anchoring the evidence for one chapter. Per-cell numerical tables sit with the paragraphs that present them; every number in the paper traces to an archived JSON record under the twin-run protocol (Appendix K).

A Definitions and Notation

Term	Definition
Reference (fp32 twin)	the same policy retrained in fp32 under the twin protocol (paired init and data order)
μ_{div}	mean per-observation L2 distance between an arm's actions and its reference's, on held-out observations at matched steps
out_std	the action outputs' standard deviation on the same set (dynamic range)
Floor	the μ_{div} spread between two independent fp32 self-retraining pairs; the noise level below which a reading is indistinguishable from fp32's own variance
Sufficient repair	an intervention returning a failing arm's reading into its floor band; the *cheapest* such intervention, by training time and latency

Name	Meaning
CURE	the routing heuristic: each failure mode is assigned its cheapest sufficient scheme
UPH (Unified Precision Health)	a layer recomputing the two backward matmuls in fp32 at probe-named layers (forward untouched)
LockedAdamW	the optimizer locking the per-step noise-displacement budget ($\text{lr}_t = \text{lr}_{\text{base}}\sqrt{\hat{v}_{\text{ref}}/\hat{v}_t}$, clamp)
EPA thermostat	the event-driven precision adaptation state machine over overflow residency (a protective fuse, not a preventive measure)
Floor band	the interval spanned by the floor (§3); a reading inside it is indistinguishable from fp32's own variance
Island arm	weakest heads kept in fp32, everything else in low precision (the production-style allocation)
Lottery pocket	a threshold-edge cell where seeds split between collapse and health (e.g., k=9 in the Pattern I sweep)
Ladder	the scale sweep of policy families from 0.8M (ACT) to 7B (CogACT, OpenVLA-OFT)
Harness	a hardware+software stack (GPU, CUDA/cuBLAS, cast recipe) carrying its own lottery edge

Table 3: The five terms fixed in §3 and used throughout (top); named schemes, cells, and protocol terms (bottom).

B Introduction Support: Prevalence and Cost Evidence

The seed-lottery table (E21). Three tasks × three seeds, bare bf16, autocast, 6000 steps, twin protocol. Square splits between full collapse and health across seeds; Can and Kitchen collapse on every tested seed (Table 8; `e21_lottery_grid.json`).

The constant-action case. The Introduction's case, a textbook-protocol bf16 run emitting one constant action, traces to archived runs X40 and the E21 Square-s42 cell (final KL 0.001, out_std 0.24, μ_{div} 6.45 against the fp32 twin).

Posterior collapse is at least in part numerical. The bypass cell (Section 4's control) rescues the collapsed posterior by precision means alone—lifting the KL term out of the graph and injecting its exact fp32 gradient at the head—without touching the objective. An objective-level account cannot explain this. The rescue trajectory and the full family of closed-form bypass cells are in Appendix G.

The 25% timing claim. On OpenVLA-OFT, bf16 steps cost 25% of the fp32 reference under the campaign's fixed batch/stack, with TF32 disabled in that reference (measurement archived in the

arch	Square	Can	Kitchen	Lift	MW-R	MW-P	MW-D	MW-A
ACT	4.47±1.75	3.27±2.13	1.45	2.42	1.07	1.25	4.00	2.08
Flow	4.61±2.44	1.16±0.03	0.36	1.24	0.06	0.09	0.66	2.59
BeT	2.45±0.30	3.24±0.12	0.85	4.65	0.20	1.03	1.56	0.76
VQVAE	1.64±0.15	1.16±0.27	0.64	0.46	0.30	0.23	1.10	0.54

Table 4: The sensitivity spectrum: fp16 μ_{div} against the fp32 twin, four architectures $\times$ eight tasks (Square/Can four seeds, the rest two; MW-R/P/D/A = MetaWorld reach/push/door/assembly). Sensitivity concentrates on low-diversity tasks and weak-signal heads; MetaWorld's reaching tasks read low on every architecture, whereas its contact-rich cells still drift on the weak-signal ones—diversity vaccinates, it does not immunize. The headline bf16 lottery cell and its healthy tf32 twin sit on the same spectrum.

loss scale	ρ_{inf}	μ_{div}	final NLL	iters
16	0.000	0.34	-338.8	6000
256	0.000	0.34	-333.3	6000
4096	0.990	4.52	-65.7	6000
32768	0.999	9.02	148.4	6000

Table 5: Overflow induction: residency and behavioral damage share one threshold (fp16 semantics, `overflow_induction.json`).

campaign logs, Appendix H); against the production-default tf32, the ledger is tiered by workload (Section 7).

Figure 1's sparklines are real data: healthy flat (E16 `te_amax-s42`), starved halving (E17 `bare_flush-s42` out_std), poisoning explosion (E19 `e5m2-s42`), and overflow residency (`overflow_induction.json`).

C Related Work Completion

The twelve policy families measured in the census are ACT (Zhao et al., 2023), Octo (Team et al., 2024), RDT (Liu et al., 2025), $\pi_{0.5}$ (Black et al., 2026; Intelligence et al., 2025), GR00T (Bjorck et al., 2025), OpenVLA-OFT (Kim et al., 2024; 2025), CogACT (Li et al., 2024), SmolVLA (Shukor et al., 2025; Cadene et al., 2026), DreamerV3 (Hafner et al., 2024), BC and MDN (Mandlekar et al., 2021), and DP (Chi et al., 2023).

The family census spans four architectures beyond the ladder. The sensitivity spectrum covers four architectures (ACT, Flow (Lipman et al., 2023), BeT (Shafiullah et al., 2022), VQVAE (van den Oord et al., 2017)) $\times$ eight tasks (four robomimic (Mandlekar et al., 2021), four MetaWorld (Yu et al., 2021)), four seeds on Square/Can (Table 4): sensitivity concentrates on low-diversity tasks and weak-signal heads: the diverse-demo MetaWorld suite reads lower on average—its reaching tasks low on every architecture, whereas its contact-rich cells still drift on the weak-signal ones—data diversity reduces but does not eliminate sensitivity, and the headline collapse cell sits at the spectrum's weak-signal end.

Rounding lineage. Stochastic rounding and low-precision SGD foundations: limited-precision deep learning (Gupta et al., 2015), QSGD-style unbiased quantization (Alistarh et al., 2017), and the high-accuracy low-precision line (Sa et al., 2018).

D Section 3 Evidence: The Three Failure Modes in Full

The overflow channel (residency). The dose-response induction (loss scales 16/256/4096/32768): residency and behavioral damage jump at the same threshold, and once formed, residency does not retreat in these runs (Figure 6; Table 5; `overflow_induction.json`).

cell	final loss	ρ_{inf}	EMA drift	note
fp32 @10^{-3} s42	0.812	0.000	62.0	
fp32 @10^{-3} s43	0.806	0.000	26.8	
fp16 @10^{-3} s42	1.013	0.994	6.2	frozen EMA
fp16 @10^{-3} s43	1.016	0.996	6.5	frozen EMA
bf16 @10^{-3} s42	0.783	0.000	68.3	
bf16 @10^{-3} s43	0.913	0.000	28.4	
tf32 @10^{-3} s42	0.863	0.000	31.2	
tf32 @10^{-3} s43	0.904	0.000	27.0	
fp16 @10^{-4} s42	0.0180	0.000	6.3	clean rate: overflow vanishes
fp16 @10^{-4} s43	0.0155	0.000	6.4	clean rate: overflow vanishes
fp32 @10^{-4} s42	0.0181	0.000	6.2	
fp32 @10^{-4} s43	0.0156	0.000	6.4	
fp16+EPA s42	0.971	0.005	—	rescued ($0.994 \rightarrow 0.005$)
fp16+EPA s43	NaN	0.649	—	overrun
fp16+EPA+ratchet s43	0.0165	0.001	—	ratchet rescues
bf16+EPA (domain)	0.270	0.000	—	dormant on ACT; μ_{div} 5.58 vs 6.45 bare
layer-event map: residency pinned on cond-encoder/final-conv layers (99.6% of calls), first bad step 23.				

Table 6: The DP case file (Diffusion Policy, Square, 6000 steps, twin protocol): the overflow channel end-to-end—phenomenon, localization, repair, boundary (`dp_precision_suite.jsonl`, `m10_epa_domain.json`).

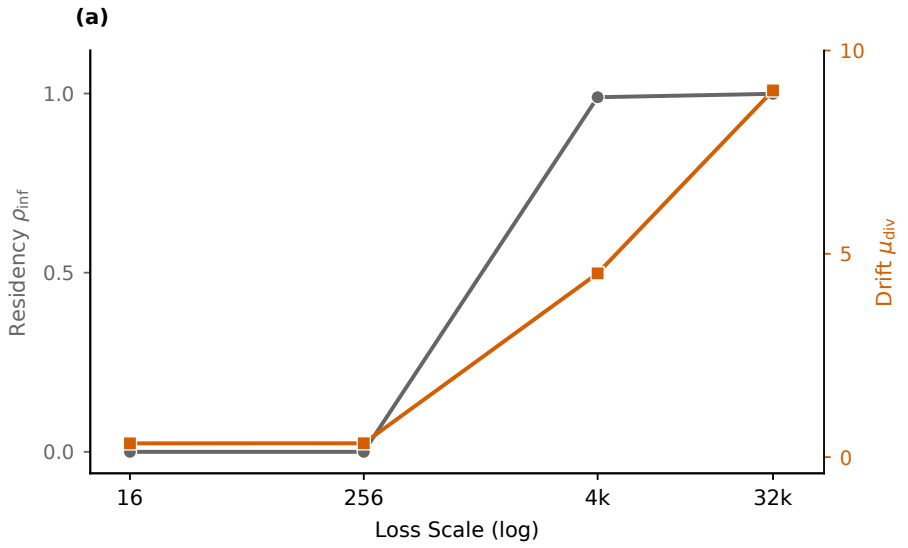


Figure 6: Overflow residency and behavioral damage share one threshold: residency saturates ($\rho_{\text{inf}} \approx 1$) beyond loss scale 4096, and the policy dies at the same dose (dose-response induction, `overflow_induction.json`).

The DP case file: the overflow channel end-to-end. Diffusion Policy (CNN U-Net, Square, 6000 steps, twin protocol) documents the channel's complete chain (Table 6). At the aggressive nominal rate (10^{-3}), fp16's gradients overflow from step 23 and never leave; the EMA weight drift freezes because fewer updates are accepted, and the final loss stalls. The layer-event map localizes the residency to the condition-encoder and final-conv layers, where layer-level skip logic cannot fix a forward-pass overflow in uncovered Conv layers. For repair, the EPA thermostat rescues one seed but is exceeded on the other, and adding the rate ratchet rescues it. At the clean rate (10^{-4}), all tested precisions train identically, and on bf16 the thermostat stays dormant and harmless: on this cell, the learning rate, not the format, is the first variable (`dp_precision_suite.jsonl`, `m10_epa_domain.json`).

precision	KL first100	KL last100	final out_std	note
fp32	3.68	3.571	0.435	healthy
fp16	3.68	3.573	0.432	healthy (matches fp32)
bf16	3.68	0.010	0.000	posterior collapse

Table 7: Mantissa starvation, the headline cell (X40): bf16's KL dies whereas fp32/fp16 hold.

cell	KL	μ_{div}	out_std
`can-bare-s42`	0.15	5.721	0.382
`can-bare-s43`	0.01	5.719	0.384
`can-bare-s44`	0.00	5.719	0.382
`kitchen-bare-s42`	0.00	2.238	0.348
`kitchen-bare-s43`	0.00	2.240	0.346
`kitchen-bare-s44`	0.03	2.244	0.347
`square-bare-s42`	0.00	6.45	0.24
`square-bare-s43`	3.23	3.92	0.47
`square-bare-s44`	3.53	5.13	0.36

Table 8: E21 lottery grid: bare bf16, three tasks $\times$ three seeds. Collapsed rows saturate: the constant action is near-identical across seeds (μ_{div} agreeing to $\sim 10^{-3}$, a ceiling set by the reference's own variance), so they are printed to three decimals.

Conservative collapse. The archived cells: fp16's rounding noise (RMS eight times smaller than bf16's) inflates AdamW's $\hat{v}$ beyond the intrinsic growth both precisions share (Table 15); on the same ACT cell the fp16 arm drifts sixfold further from the fp32 twin than a tf32 arm (`spectrum_line.jsonl`); and a tenfold rate cut restores the run (`x61_act_lr.json`). The failure is a drift in the weights, not a collapse: the policy keeps acting while noise walks its weights.

Mantissa starvation. The full KL trajectory of the bf16 collapse ($3.8 \rightarrow 0.01$) with constant-action verification, plus the static-criterion miss table (every static criterion constructed misses it), is archived (X40 family, Table 7; E21 cells).

The lottery map (E21 full table). Per-task per-seed outcomes for bare bf16 on Square/Can/Kitchen $\times$ s42/s43/s44, with KL, μ_{div}, out_std per cell (Figure 2b; Table 8; full trajectories in the archive).

The production-β boundary (P1-1). The E21 grid is rerun with the KL weight as the sole change: β=10 (the production ACT/LeRobot recipe, $1000\times$ the canonical weak-β regime), Square $\times$ s42/s43/s44, with the fp32 references retrained under the same β (the old references are void under the new loss). All three seeds train normally in both precisions (Table 9). The Can and Kitchen cells were skipped for unavailable assets on the server and remain unverified at β=10—reported, not assumed. Per the reading fixed in advance: on Square the collapse does not survive the production KL weight, so the domain-level alarm is regime-specific; the boundary moving with signal strength is exactly Pattern I's prediction (β scales the KL maintenance gradient a thousandfold above the resolution floor) (`p1_1_beta10_lottery.json`).

seed	KL (fp32)	KL (bf16)	μ_{div}	out_std (fp32/bf16)	outcome
s42	6.4×10^{-5}	5.0×10^{-5}	0.056	0.233 / 0.231	healthy
s43	6.6×10^{-5}	14.3×10^{-5}	0.038	0.231 / 0.232	healthy
s44	10.4×10^{-5}	8.7×10^{-5}	0.048	0.232 / 0.232	healthy

Table 9: P1-1: the production-β boundary (β=10, Square, three seeds, fp32 references retrained). Every cell healthy in both precisions (`p1_1_beta10_lottery.json`).

cell	KL	μ_{div}	out_std
A_bare	3.27	2.87	0.50
B_uph	2.84	3.04	0.48
C_uph_sr	2.94	2.86	0.49

Table 10: X69: a true-fp8 backward (two-mantissa-bit parameter gradients) stays normal provided that activation gradients keep full precision.

cell	KL	μ_{div}	out_std
square-bare-s42	0.00	6.45	0.24
square-bare-s43	3.23	3.92	0.47
square-fp32scalar-s42	0.00	6.45	0.24
square-fp32scalar-s43	3.23	3.92	0.47
square-srscalar-s42	5.24	2.97	0.51
square-srscalar-s43	3.65	4.79	0.38
square-srscalar-s45	3.20	4.12	0.46
square-srscalar-s46	1.90	5.84	0.32
square-srscalar-s47	4.06	3.83	0.47
square-srscalar-s48	4.10	4.19	0.45

Table 11: E12: the KL scalar is innocent (fp32 recompute reproduces the trajectory bit-for-bit).

Sub-critical doses. E17's bare_flush cells: out_std halved (0.32/0.29), μ_{div} 4.6/5.3, whereas KL reads 5.2/3.2—KL looks normal while behavior has collapsed (also Appendix J).

E Section 4 Evidence: The Mechanism in Full

The triple negative controls. Protecting the loss in fp32, freezing the output head, and post-hoc weight quantization all change nothing (archived cells, x58_egs_z and related).

The dissection (debiasing). Protocol: twin runs; at chosen points of the backward chain, the low-precision activation gradient is replaced by its fp32 recomputation, everything else held fixed. Debiasing the activation-gradient chain rescues the policy; debiasing only the parameter gradients does not. Three on-disk cells carry the same conclusion independently: a true-fp8 backward with two-mantissa-bit parameter gradients stays normal provided that activation gradients keep full precision (x69_fp8_uph.json); protecting the parameter gradients instead of the activation chain fails to rescue (x65 family); and recomputing the KL scalar in fp32 reproduces the untreated trajectory bit-for-bit (e12_scalar_sr.json). The original layer-by-layer sweep is archived in the campaign records.

True-fp8 backward (x69). A backward with two-mantissa-bit parameter gradients stays normal provided that activation gradients keep full precision—the failure lives in the activation-gradient chain, not the parameter gradients (Table 10).

The scalar is innocent (E12). The bf16-evaluated KL scalar reads low (90% of probes; 98.6% collapsed), but recomputing that scalar in fp32 reproduces the untreated trajectory bit-for-bit. The deficit is an artifact of cancellation in $1 + \ell - \mu^2 - e^\ell$ near $\ell \approx 0$, not a real one (Table 11).

Nonuniform error (40×). The chain's error is angularly unbiased but radially toward-zero: relative error amplifies fortyfold from the tensor's largest to its smallest elements (e6_swallow_localization.json); and on the collapsing cell, 68.3% of KL-gradient component errors shrink their component toward zero, against 52.9% for the paired fp32 reference (x59b_kl_grad_bias.json)—angularly unbiased, radially toward-zero.

arm (harness)	protocol	KL (s42/s43)	out_std	outcome
noise_f32 (PRO 6000)	fresh zero-mean noise	3.84 / 3.71	0.570	healthy, both seeds
noise_corr_f32 (PRO 6000)	frozen 100-step windows	3.56 / 3.54	0.568	healthy, both seeds
bias_f32 (PRO 6000)	deterministic $(1-2^{-9})$ shrink	3.60 / 3.52	0.566	healthy this draw
bias_f32 (consumer GPU, archived)	same	0.005 / —	0.234	collapsed (μ_{div} 6.39)
noise_f32 (consumer GPU, archived)	fresh zero-mean noise	3.28 / —	0.435	maimed (μ_{div} 4.0)

Table 12: P0-3C: the decisive cell re-run on a second harness against the archived draw. The bias arm reproduces the lottery itself; both noise arms stay healthy on every draw.

tensor	rel. err (weakest decile)	rel. err (strongest)	shrink frac (weakest)	direction cos
z_μ weights	4.5×	0.064×	0.34	0.99
z_{lv} weights	12.1×	0.096×	0.21	0.98
hidden activations	71.8×	0.50×	0.06	0.59

Table 13: P0-3B: rounding bias on the real pipeline, by tensor. Relative error amplifies into the weakest decile whereas the absolute shrink fraction stays below 0.5 (`p0_3_rz_hypothesis.json`).

The decisive cell (x88). Equal-power zero-mean noise leaves fp32 normal on every seed, whereas a toward-zero bias of the same magnitude collapses the collapse-prone seed (Figure 3b). In this controlled cell, the failure is caused by the rounding's direction, not its magnitude.

The decisive cell, re-run on a second harness (P0-3C). The same script and seeds on a second harness (PRO 6000, sm_120, against the original consumer GPU; Table 12): zero-mean noise stays healthy on both seeds; a frozen-window variant, using the same noise realization frozen for 100-step windows, also trains normally. Per the reading fixed in advance, frozen error averages out across windows, and the original determination stands. The deterministic toward-zero bias arm, which collapsed on the archived draw, this time trains normally on both seeds. Read carefully, this is a replication of the lottery, not a refutation: a deterministic 0.2% per-step shrink feeds the starvation loop only when the run draws near the basin edge, so the synthetic bias reproduces not just the collapse but its seed-and-harness lottery, whereas equal-power noise, fresh or frozen, stays healthy in all four runs. (The archived noise arm's elevated drift is the degrading regime: noise degrades the run; the bias collapses it.) The loop's driver is a small persistent bias continuously regenerated by rounding the current weights, not a large per-step one (`x88_bias_f32_*`, `x88_noise_f32_*`, `x88_noise_corr_f32_*`).

The bias, measured per decile (P0-3B). On the real pipeline (same weights, same batch, same z; bf16 autocast vs fp32 backward), the toward-zero shrinkage concentrates at the weak end in relative terms, whereas the absolute shrink fraction stays below 0.5 even in the weakest bucket (Table 13). Consequence for the determination cell: the synthetic uniform bias overstates the real per-step bias, so the collapse mechanism requires the loop's temporal integration, not a single large per-step shrink—consistent with the rerun above and with the spiral model. The reduction-precision switch (fp32 vs reduced accumulation in the GEMM path) reproduces the untreated trajectory *bit-for-bit* on both seeds (15 decimal places), excluding the accumulator: the bias is sourced at the casts (`p0_3_rz_hypothesis.json`).

The spiral model, written down. Let g be the surviving magnitude of the KL maintenance gradient reaching the latent head. Near the prior, a Gaussian posterior gives $\mathrm{KL} \approx \frac{1}{2}(\|\mu\|^2 + \ell^2)$ (log-variance ℓ), whose gradient components $|\mu|$, $|\ell|$ scale as $\sqrt{\mathrm{KL}}$; to first order $g(\mathrm{KL}) = \kappa\sqrt{\mathrm{KL}}$. The chain admits the signal only above its floor: with β the KL weight, the survival condition is

$$\beta\, g(\mathrm{KL}) \;=\; \beta\,\kappa\sqrt{\mathrm{KL}} \;\geq\; F, \qquad F \sim 2^{-8} \times (\text{row RMS}).$$

arm	KL final	μ_{div}	out_std	note
beta_0.01	0.001	6.45	0.24	dies (low β)
beta_0.05	1.172	4.31	0.41	partial
beta_0.2	0.384	6.46	0.23	overshoot, re-collapses
sr_rescue	3.628	4.01	0.48	revives at the band

Table 14: The spiral model's rescue band, verified: low β dies, the sweet spot revives, overshoot re-collapses (x62, x63).

arm	$\hat{v}$ start	$\hat{v}$ end	inflation
fp32	6.81e-10	2.57e-07	376.7×
bf16	6.82e-10	5.15e-07	755.3×

Table 15: The hidden brake (E13): the bf16-to-fp32 $\hat{v}$ ratio doubles over 1500 steps on one stress trajectory (both precisions inflate intrinsically; the precision-specific excess is the ratio).

Below the floor the head receives only the rounding residue: swallowed at rate ε per step ($\varepsilon \approx 2^{-9}$, the measured toward-zero bias) and refilled by Adam-amplified noise, the posterior decays geometrically, $\mathrm{KL}_{t+1} \approx (1-\varepsilon)\,\mathrm{KL}_t$ + noise walk—the loop: the weaker the signal, the deeper the swallowing, and the deeper the swallowing, the weaker the signal. The band this condition traces has two edges. The lower edge is the survival condition itself: β multiplies the maintenance gradient directly, so a few-fold lift (the ratio F/g_0) raises the signal out of the floor—a fivefold weight revives the posterior. The upper edge is the objective's own pressure: the equilibrium KL^* falls as β grows, and since $g \propto \sqrt{\mathrm{KL}}$, the effective signal $\beta\, g(\mathrm{KL}^*(\beta))$ grows only sub-linearly with β—past the band, the pressed posterior drops below behavior's working point once the margin over the floor vanishes. Both edges are the same endpoint behaviorally: the β=0.2 arm's drift (μ_{div} 6.46) reproduces the β=0.01 arm's (6.45)—an unusable posterior, entered from opposite sides. The ladder's thresholds are mutually consistent with one floor, $F/\kappa \in (0.0195,\ 0.054)$ (dies at $1\times$, revives at $5\times$).

The spiral model and its verified predictions. The starvation-loop model predicts a rescue band: a fivefold KL weight revives the posterior at the band's lower edge, a twentyfold weight overshoots and re-collapses—both confirmed (x62_beta_ladder.json, x63_sr_rescue.json; Table 14; Figure 2a).

The hidden brake (e13). Rounding noise inflates AdamW's $\hat{v}$ beyond the intrinsic growth both precisions share—the same object whose instability Molybog et al. trace to $\hat{v}$ failing to upper-bound the squared gradient (Molybog et al., 2023), here inflated by numerical noise rather than by the signal: the bf16-to-fp32 ratio climbs from 1.00 to 2.01 over 1500 steps on one stress trajectory (Table 15); washing the backward clean removes the excess inflation, and the unlocked step diverges at the same nominal rate (Figure 3a).

The momentum-SGD control (E8). With AdamW replaced by momentum SGD, KL returns to normal on all seeds—the amplifier's role is confirmed by removing it.

Support for Pattern I (E22 coverage sweep). Pattern I predicts that rescuing a weaker signal requires recomputing more of the backward chain in fp32. Converting layers to fp32-backward weakest-first on the collapse cell: k=0 collapses, k=1–8 progressively rescue, k=9 re-collapses (a lottery pocket at the threshold edge), k=12 rescues again (Table 16; Figure 7).

Support for Pattern II (rescued-posterior instances). Pattern II predicts that a rescued posterior can exceed fp32's own KL equilibrium, so fp32's equilibrium is not the signal's true equilibrium. The instances: cleanly on Can (Table 17), and at scale on Octo (below the fp32 self-retraining floor; directional) and CogACT (LockedAdamW below the floor band; Appendix H).

cell	KL	μ_{div}	out_std
uph_k0	0.00	6.45	0.24
uph_k1	2.79	4.58	0.41
uph_k2	2.72	3.73	0.46
uph_k4	2.14	4.70	0.42
uph_k6	3.63	4.89	0.44
uph_k8	2.55	4.22	0.44
uph_k9	0.06	6.45	0.24
uph_k12	3.15	4.10	0.44

Table 16: E22 Pattern-I coverage sweep: layers recomputed in fp32 (weakest first) on the collapse cell.

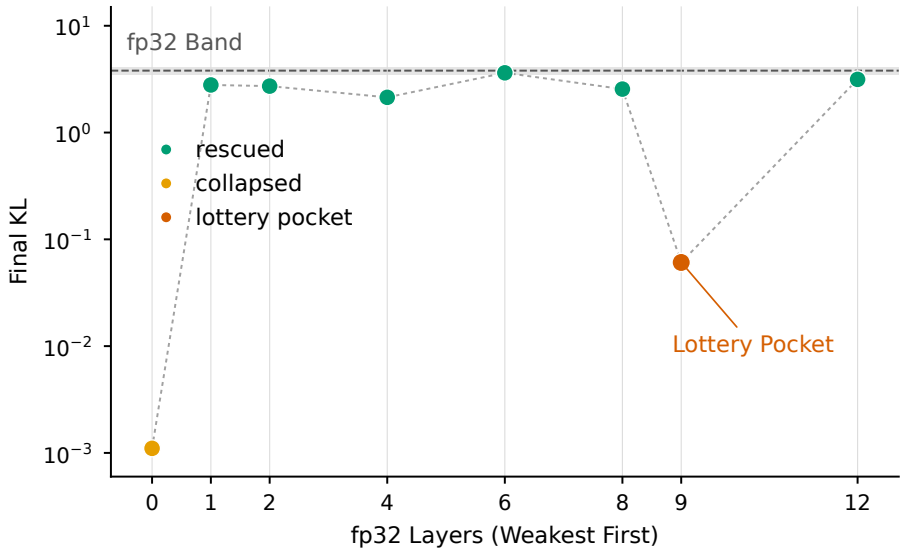


Figure 7: Pattern I coverage sweep (E22): final KL against the number of layers recomputed in fp32 (weakest first). The red pocket at k=9 is the threshold lottery at the edge.

F Section 5 Evidence: The Criteria in Full

The probe protocol. The 50-step twin of bf16 against fp32, paired initialization and data order; sensitivity is read from the twin's divergence at the 50-step mark (d_{late}, and its KL-headed variant d_{kl}; the ladder cells additionally record the divergence step t_{div}, Table 20).

The seven-cell table (g2_miniprobe). bc/mdn/gbc on Square, bc/mdn on Can, VAE on MNIST, GPT on Shakespeare: per-cell `div_late` and full-run drift (Table 18; `g2_miniprobe.jsonl`).

Robustness (g2_loo + bootstrap). The probe's rank correlation is 0.857 over the seven cells; leave-one-family-out stays in $[0.77, 0.94]$ (min above 0.5); a 2000-draw bootstrap gives a 95% in-

cell	KL	μ_{div}	out_std
can-bypass-s42	4.97	1.43	0.57
can-bypass-s43	0.67	5.39	0.38
kitchen-bypass-s42	35.70	2.03	0.37
kitchen-bypass-s43	0.08	2.24	0.35

Table 17: E23 Pattern-II bypass cells. fp32 reference KL: can=2.41, kitchen=1.58.

cell	d_{late} (probe)	truth (full-run drift)
bc_square	0.000	0.000
mdn_square	0.076	16.291
gbc_square	0.082	0.409
bc_can	0.000	0.082
mdn_can	0.087	0.796
vae_mnist	0.031	0.332
gpt_shakes	0.000	0.000
Spearman 0.857; LOFO [0.77,0.94]; bootstrap 95% CI [0.38,1.00]; zero false negatives		

Table 18: The seven-cell probe table (g2_miniprobe), with robustness (LOFO, bootstrap, false negatives).

cell	d_{kl}	$\hat{v}$ ratio	decision	final KL	out_std	outcome
A: Square bare bf16 s42	0.308	1.177	bypass	3.56	0.567	rescued
B: Square tf32 s42	0.0007	1.0001	none	(bare)	0.566	healthy, no false positive
archived untreated cell A (e21): posterior dead, KL 0.001, μ_{div} 6.45. Decision committed 2026-09-08 22:49:19, before training.						

Table 19: P1-2: the blind-routing closure, prospective (p1_2_routing_decision.json, p1_2_blind_routing.json).

terval $[0.38, 1.00]$; no cell is a false negative, where the probe reads safe but the full run collapses (g2_loo.json, e26_probe_robustness_g2.json).

The blind-routing closure, prospective (P1-2). The end-to-end loop is closed prospectively: the decision tree ($\hat{v}$ ratio ≥ 1.2 →LockedAdamW; else $d_{\text{kl}} \geq 0.05$ → bypass; else nothing) is fixed in advance, the probe runs its 50-step twin, and the routing decision is committed to disk with a timestamp (2026-09-08 22:49:19) *before* any training begins (Table 19). One failing cell routed to its cheapest sufficient repair, one healthy cell left alone, zero false positives—the end-to-end loop, closed prospectively on the campaign's central cell (p1_2_routing_decision.json, p1_2_blind_routing.json).

The probe climbs the ladder (P0-4). Two ladder cells, 50-step twins (Table 20). On CogACT (7B) the alarm matches the archived failing outcome and the lock's rescue—prospective consistency; on Octo (93M) the probe over-reads a healthy cell once. That single false positive is the probe's known boundary: the 50-step divergence probe over-reads on healthy or slow-failing cells (the bf16-native arm drifts early even when nothing is wrong); its validated value is on fast-failing cells. The economics absorb the asymmetry: a false alarm costs one unnecessary cheap repair, a missed alarm costs a dead run (p0_4_probe_ladder.json).

The EPA thermostat (event-driven precision adaptation). Specification: a state machine over the overflow residency ρ_{inf}, evaluated over a sliding 100-step window—COOL (normal low-precision training at the nominal rate); $\rho_{\text{inf}} > 5\%$ enters WARM (the rate is halved, restored symmetrically on exit); $\rho_{\text{inf}} > 20\%$ enters HOT (shelter to fp32 for up to 200 steps; exit on a fully clean window or when the shelter elapses). Entering HOT ratchets the rate down ($\times 0.25$,

cell	t_{div} (steps)	wall-clock	routing read → actual outcome	consistent
CogACT (7B)	16	346.5s	alarm → optimizer-side; bare damaged above band, LockedAdamW rescues	yes
Octo (93M)	10	86.7s	alarm; all arms healthy (shared undertraining)	no (false positive)
CogACT: twins sequential (two 7B instances do not fit in memory); $\hat{v}$ column unreadable under bnb-8bit states.				

Table 20: P0-4: the probe climbs the ladder (p0_4_probe_ladder.json).

Algorithm 2: CURE routing (the general protocol)

Input: 50-step twin telemetry $\{d_{\text{late}},\ d_{\text{kl}}$ (variational only), $\hat{v}$ ratio, $\rho_{\text{inf}}\}$; census weak-layer map
1: $\rho_{\text{inf}} > 5\% \rightarrow$ EPA thermostat (overflow channel; likelihood heads: MDN, DP)
2: closed-form weak signal present (CVAE KL head) $\rightarrow$ analytic bypass at the head
3: $\hat{v}$ ratio $\geq 1.2 \rightarrow$ LockedAdamW (conservative channel; transformer policies route here)
4: $d_{\text{late}} > 0$, weak layers mapped upstream (Pattern I) $\rightarrow$ UPH at probe-named layers
5: otherwise $\rightarrow$ nothing (healthy)
non-variational cells: no d_{kl} exists; the KL branch is skipped by construction,
and routing reads the $\hat{v}$ ratio and d_{late} only.

The two-branch tree fixed in advance (Table 19) is this protocol instantiated on its two cells: a CVAE starvation cell (branch 2) and a healthy tf32 cell (branch 5), no overflow channel present.

Algorithm 1: LockedAdamW (noise-displacement-budget lock)

Input: base rate lr_{base}, warmup W, clamp $[c_{\min}, c_{\max}]$, β_2=0.999
Warmup ($t \leq W$): record $\hat{v}_{\text{ref}} \leftarrow$ global mean of bias-corrected $\hat{v}$
Step ($t > W$): $\hat{v}_t$ = global mean of bias-corrected $\hat{v}$;
$\rho \leftarrow \text{clamp}\Big(\sqrt{\hat{v}_{\text{ref}}/\hat{v}_t},\ c_{\min}, c_{\max}\Big)$; $\quad \text{lr}_t \leftarrow \text{lr}_{\text{base}} \cdot \rho$;
update with AdamW at lr_t (one-way brake: long β_2 memory keeps it engaged in clean phases).

The lock targets the per-step noise-displacement budget, not the nominal rate.

one-way). The three duties are separate by construction: diagnosis lives in the layer hooks, step-level disposition in the stock GradScaler, strategic control in the thermostat. Validation on the fp16 stress cell: the thermostat rescues s42 (residency 99.4% $\rightarrow$ 0.5%, μ_{div} 1.37) but is exceeded on s43 (twenty-six interventions, then permanent shelter); adding the rate ratchet rescues s43 (six overflows total, healthy final loss). On bf16 cells it stays dormant and harmless (μ_{div} 5.58 vs 6.45 untreated; `m10_epa_domain.json`). Position: it prevents the loud failure; it does not beat a rate cut on healthy runs.

G Section 6 Evidence: The Schemes in Full

Scheme 1 derivation (the analytic bypass). For a head with a closed-form gradient, the auxiliary term is lifted out of the computation graph and its exact gradient is added in fp32 directly onto the head; the rest of the backward stays in bf16. Families covered: variational posteriors (KL), invertible flows (log-det), Gaussian/mixture heads (variance calibration), entropy and Lagrangian terms. Zero cost, zero randomness consumption.

Scheme 2 (UPH, Unified Precision Health). A drop-in wrapper: the forward pass is unchanged, and only the wrapped module's two backward matmuls are recomputed in fp32; at probe-named layers, stochastic rounding adds a dither unbiased in expectation. The layer-mapping algorithm follows the probe's naming (Pattern I's placement rule). Ablations attribute three quarters of the gain to recompute and one quarter to the dither (`x66` family archive). Compatibility note: the wrapper recomputes two backward matmuls per layer in fp32 and is compatible with FSDP/compile stacks at the user level (no kernel changes).

Scheme 3 (LockedAdamW), full specification. With $\hat{v}_t \approx \hat{v}_{\text{fp32}} + \sigma^2$ and $\kappa \equiv \sigma/\sqrt{\hat{v}_{\text{fp32}}}$: warmup self-calibrates $\hat{v}_{\text{ref}}$; then $\text{lr}_t = \text{lr}_{\text{base}} \cdot \sqrt{\hat{v}_{\text{ref}}/\hat{v}_t} \approx \text{lr}_{\text{base}}/\kappa$. Composing the lock with Adam's intrinsic $1/\sqrt{\hat{v}_t}$ normalization, the total compression is $1/\sqrt{1+\kappa^2}$ (intrinsic) $\times\ 1/\sqrt{1+\kappa^2}$ (lock) $= 1/(1+\kappa^2)$. One-way brake ($\beta_2 = 0.999$), partial release (17%) in the clean phase. Pseudocode below.

Challenger	What it tried	Refuted by
HDR (spectral amplifier)	rescales the chain's spectrum at a hook point	chain compounding (Cor. i): content lost at global absorption has no single-hook source
sdef (white-noise defense)	adds calibrated zero-mean noise	the decisive cell: noise magnitude is harmless; the bias drives the collapse
twoword (temporal accumulation)	accumulates over time to average out rounding	temporal $1/\sqrt{T}$ SNR bound (Cor. iii)
subspace (projection repair)	projects the quantized stream onto a repaired subspace	projection-energy bound (Cor. ii)

Table 21: The four refuted challengers, with the corollary that refutes each. Under the stated premise, no deterministic partial repair of the quantized stream works at an affordable cost (stochastic-rounding channels sit outside the premise by construction).

The four refuted challengers (full failure analyses). HDR, sdef, twoword, subspace: each defined, run, and shown to fail, with the corollary that refutes it (Table 21; the impossibility theorem below).

The impossibility theorem (no partial repair), with derivations. Setup. Let Q be the quantization operator of a low-precision channel, and let the weak signal component r be deleted at the format's resolution floor (flush: the channel output $Q(x)$ is independent of r) or drowned (raise: sub-floor components are replaced by $\pm$floor, carrying at most one sign bit of r). **Scope.** The theorem covers deterministic rounding at the floor. Stochastic rounding preserves sub-floor content in expectation, so SR-based channels—including Scheme 2's own dither—sit outside its premise by construction. **Premise (worst case).** Both theorems are stated under a worst-case premise: the deleted or drowned component r is conditionally independent of the surviving content $x \setminus r$—the survivor leaks nothing about r. Gradient coordinates are in practice correlated, so a surviving stream can in principle carry information about r (a duplicated coordinate is the trivial counterexample). Whether such correlations rescue is therefore an empirical question, and we report it as a finding, not a principle: on every measured cell, the four challenger failure analyses below—including a learned-projection repair—recover nothing. **Theorem (flush).** Under flush, $Q(x)$ is a deterministic function of the surviving content alone, so $I(Q(x); r) = 0$. By the data-processing inequality, for any post-hoc repair operator R—deterministic or randomized—applied to the quantized stream, $I(R(Q(x)); r) \leq I(Q(x); r) = 0$. Information deleted at the floor cannot be re-created downstream. **Theorem (raise).** With sign-preserving raise, the channel output factors through $(x \setminus r,\ \mathrm{sign}(r))$: under the premise, any estimator $\hat{r}$ of r from $R(Q(x))$ is a function of the sign bit plus r-free terms. Under squared-error loss, the Bayes-optimal such estimator is the conditional mean $\mathbb{E}[r \mid \mathrm{sign}(r)]$, and its risk—exactly the Bayes risk of any estimator under the premise—is $\mathbb{E}[\mathrm{Var}(r \mid \mathrm{sign}(r))]$ (correlated survivors could only lower it). The magnitude of r is unrecoverable; at most its direction survives. The raise model is sign-preserving (round-away-from-zero at the floor); under round-to-nearest, the threshold event—the output landing at floor versus zero—is binary and adds at most one bit, and in practice the risk's floor-squared scaling, and the conclusion, are unchanged. **Corollaries, with derivations.** (i) *Chain compounding (refutes HDR).* A hook repair sits either downstream or upstream of an absorption node. Downstream, the stream is already flushed and Theorem (flush) applies verbatim. Upstream, a diagonal rescaling $g_i \mapsto \alpha_i g_i$ amplifies only what survives in the hook's own input: absorption is distributed over the chain's summation nodes, and content deleted between the head and the hook is zero in that stream—Theorem (flush) applies segment by segment, and no single hook sources content lost across the whole chain. Matching the per-element gains to the per-element thresholds would require knowing the deleted components themselves; a uniform gain keeps the survivors alive but recovers nothing already swallowed, and on narrow formats it spends the exponent headroom the strong components need. The measured HDR failure analysis (no recovery at any hook point) is exactly this segment-wise deletion. (ii) *Projection-energy bound (refutes subspace).* Fix an estimator $\hat{r}$ taking values in a data-derived subspace S with projector P. The squared error decomposes orthogonally, $\mathbb{E}\|r - \hat{r}\|^2 = \mathbb{E}\|Pr - \hat{r}\|^2 + \mathbb{E}\|(I-P)\, r\|^2$: the out-of-subspace bias is untouchable from within S. Under the premise the surviving stream carries no directional information about r beyond its sign bit, so no data-derived P aligns with r better than

arm (server harness)	μ_{div}@8k	act_std	band position
`fp32_ref2 (floor)`	13.48	1.001	floor edge
`fp32_ref3 (floor)`	8.85	0.766	floor edge
`bare_bf16`	12.86	0.535	in-band
`uph_locked`	9.64	0.486	in-band

floor band [8.85, 13.48]; the 25% bare-vs-treated gap sits inside the retraining spread.
first harness (local): treated arms 6.00–7.94 vs floor 4.22 (misfire); misfire not reproduced here.

Table 22: GR00T (3B) on the server harness, with a measured floor band: bare and treated arms both in-band—the first harness's misfire is protocol-sensitive, not a robust property of the control law (`gr00t_matrix.json`).

$\mathbb{E}[r \mid \text{sign}(r)]$ does, and the total risk is lower-bounded by Theorem (raise)'s $\mathbb{E}[\text{Var}(r \mid \text{sign}(r))]$—the best achievable squared error is at least the energy deleted at the floor. Non-expansiveness ($\|PQ(x) - Px\| \leq \|Q(x) - x\|$) bounds only the in-subspace noise and cannot certify recovery; the learned-projector cell recovers nothing beyond the bound. (iii) *Temporal $1/\sqrt{T}$ bound (refutes twoword).* Write the per-step rounding error as $u_t = -b + \varepsilon_t$, with b the toward-zero bias and ε_t zero-mean of variance σ^2. Averaging T steps leaves the bias untouched and shrinks only the stochastic part: $\bar{u}_T = -b + \bar{\varepsilon}$, $\text{Var}(\bar{\varepsilon}) = \sigma^2/T$. Recovering the biased component to parity costs $T \geq (\sigma/b)^2$ steps of the same content; the bias itself never averages out at any T. The measured failure analysis prices this above our training budgets—and the decisive cell shows the fatal component is the bias, not the variance. (iv) *Operand–output floor invariance (refutes sdef).* Adding zero-mean dither d changes the realization, not the expectation. Under flush, whenever $|r + d_i|$ stays below the floor, $Q(x+d)$ deletes r exactly as $Q(x)$ does; dither calibrated above the floor is itself noise of magnitude at least the floor, reducing the repair to (iii)'s variance price. Under raise, the dither carries no information about r, so Theorem (raise)'s bound is unchanged. This is the decisive cell read at format scale: equal-power zero-mean noise leaves every seed normal whereas the toward-zero bias is fatal—the magnitude of the noise is not the variable. *All four agree with the measured failure analyses (Table 21).*

The LockedAdamW step bound. Under the additive inflation model, plain AdamW's per-step noise displacement saturates at $O(\text{lr})$ as κ grows—the unbounded object is the accumulated random walk, whose excursion scales as $\sqrt{T}\,\text{lr}$; after calibration the effective step is exactly $1/(1+\kappa^2)$, and the locked per-step noise displacement is bounded by a constant for all κ (maximized at κ=1, i.e., $\sigma = \sqrt{\hat{v}_{\text{fp32}}}$, by AM–GM). The deliberate over-brake is a proven control law, not a heuristic. *This agrees with the signature run (Figure 4).*

The swallowing condition's physical location. Two floors must be distinguished: the *absorption floor* of Eq. equation 1 (row-relative, an SNR condition at the chain's summation nodes) and the *format floor* of the boundary study (Appendix I; the format's absolute representability floor, where flush and raise act). Tensor-core bf16 GEMMs nominally accumulate in fp32, though the internal alignment arithmetic varies by hardware generation (Fasi et al., 2021); the absorption floor acts at the chain's bf16-cast operands and stored outputs, not inside the accumulator.

Domain boundary (the GR00T misfire, and its non-reproduction). On the first harness (local, freeze-trunk protocol), the warmup-calibrated $\hat{v}_{\text{ref}}$ cannot distinguish task-intrinsic noise (flow-matching sampling) from precision noise: the controller brakes a healthy run. On a second harness (server, with a measured floor band from two fp32 self-retraining references), bare and treated arms both sit inside the band—the misfire does not reproduce (Table 22). The pure-LockedAdamW arm remains to be retested on the second harness. The account is protocol-sensitive, not a robust property of the control law; a pre-application discriminator (a short fp32 twin window or consecutive-gradient sign similarity) remains the repair path where it does occur, and until then the tree's LockedAdamW gate fires only where the probe flags $\hat{v}$ inflation. Two further limits of the lock as built: it is a global scalar (per-layer κ varies by orders of magnitude; a per-tensor variant is unexplored), and its composition with an outer LR schedule (e.g., cosine decay on lr_{base}) is multiplicative and unexamined.

cell	KL	μ_{div}	out_std	note
`square-bare-s42`	0.00	6.45	0.24	
`square-bare-s43`	3.23	3.92	0.47	
`square-dadapt-s42`	NaN	NaN	NaN	NaN-abort
`square-dadapt-s43`	NaN	NaN	NaN	NaN-abort
`square-locked-s42`	3.91	1.24	0.56	
`square-locked-s43`	3.52	4.26	0.45	
`square-prodigy-s42`	0.00	6.45	0.23	
`square-prodigy-s43`	0.00	6.45	0.23	

Table 23: T9 per-cell results: zero-search baselines vs LockedAdamW on the collapse cell. The locked cells here use warmup W=200 (the validation matrix uses W=100); on s43 the two windows straddle the threshold (4.26 vs 1.05).

arm	channel addressed	final KL (s42/s43)	out_std (s42/s43)	outcome
EF (Kahan carry, z-head)	starvation	3.50 / 3.53	0.566 / 0.570	healthy
ϵ=10^{-6}	conservative	3.63 / 3.45	0.567 / 0.570	healthy
β_2=0.95	conservative	3.57 / 3.53	0.569 / 0.573	healthy
clip 0.1	noise spikes	3.56 / 3.42	0.566 / 0.566	healthy

archived bare (e21): s42 posterior dead (KL 0.001, μ_{div} 6.45, range 0.24); s43/s44 failing (μ_{div} 3.9/5.1).

Table 24: P1-3: four classical Adam-side arms on the collapse cell (6000 steps, canonical β; `p1_3_adam_ef_baselines.json`).

SR and randomness. The bypass consumes no randomness; UPH's SR dither consumes RNG, trading bitwise determinism for unbiasedness—the bypass's "zero randomness consumption" claim is exact and separate.

Zero-search baselines (T9, full table). On the collapse cell (autocast bf16, twin protocol, two seeds): bare collapses; Prodigy collapses with its adapted rate climbing into the noise; D-Adaptation diverges to NaN within 2k steps; LockedAdamW rescues (Table 23; Figure 4, right). Two adaptive rules, opposite directions, opposite fates. Protocol note: the LockedAdamW reference cells here use warmup W=200 whereas Table 2's cells use W=100—the lock's calibration inherits the edge lottery it is designed to tame.

The classical Adam-side challengers, answered and named (P1-3). The sharpest classical question—does anything cheaper than the lock already suffice?—is answered prospectively on the collapse cell (Square bf16, canonical β, 6000 steps, two seeds). Four classical arms: error feedback with a Kahan residual carry on the z-head path (Seide et al., 2014; Karimireddy et al., 2019) (EF), ϵ=10^{-6}, β_2=0.95, gradient clip 0.1. All four train healthy on both seeds, against the archived bare lottery (Table 24). Adafactor's RMS update clipping (Shazeer & Stern, 2018) is the closest published mechanism for capping per-step displacement from second-moment information; it was not run and remains an open comparison. Each winner is named, as fixed in advance. The mechanisms line up with the two channels: ϵ and β_2 cap the conservative channel ($\hat{v}$ inflation $\rightarrow$ step shrinkage), EF's residual carry returns sub-LSB KL-gradient mass each step (the starvation channel), and the clip limits rounding-noise spikes. The consequences: at the canonical budget the lock's increment narrows to nothing on this cell; error feedback joins the routing table as the cheapest starvation fix; and the lock's remaining measured claim is the domain the knobs have not been tested at—the long budget where bare dies at 44k and the lock trains through 150k (Appendix H) (`p1_3_adam_ef_baselines.json`).

The lock's domain boundary, two instances (P0-1 autopsy). The lock is not domain-free, and its boundary is now measured twice. *Lift, early death at 51k*: the locked arm stalls mid-campaign with `nan_skips` = 0—not a NaN event; the $\hat{v}$ and rate histories (archived per step) show the warmup-calibrated $\hat{v}_{\text{ref}}$ drifting off-domain at long budget, the brake clamping at its floor, and the effective step freezing—a calibration-domain failure, not a control-law failure. *Square, past 150k*: in the 300k extension the locked arm hits the abort rule (fifty consecutive NaN steps) past 150k,

model	bf16 vs tf32 (wall-clock)	note
Octo (93M)	level	
RDT (1.2B)	bf16 56% slower	image-token-heavy
$\pi_{0.5}$ (3B)	bf16 64–99% slower	dual-encoder stack
GR00T (3B)	bf16 32% faster	first genuine win

Table 25: The time ledger: bf16 against tf32 per 8k steps, same stack inside every twin pair. Intervention overhead is bounded separately (UPH 10–20%, LockedAdamW under 10%).

the same budget region where the untreated arm is long dead (44k) and the fp32 reference itself is degrading. The lock extends the trainable domain by an order of magnitude; it does not make it infinite, and a recalibration schedule for $\hat{v}_{\text{ref}}$ is the open repair (`p0_locked_autopsy.json`, `p0_1_square_converged_300k.json`).

H Section 7 Evidence: The Validation Matrix in Full

Per-cell full trajectories. Every cell's full seed set and trajectories (the main text shows endpoints only) are archived per experiment (index in Appendix K).

Can and Kitchen: the mixed-mode account, and the tree's "no sufficient scheme" answer. On the two all-seed-collapse cells, the starvation diagnosis holds (KL dead on every tested seed, Table 8) but no single scheme suffices: the free bypass inherits the seed lottery (Table 17), UPH coverage does not reach the failure site (Pattern I places it further upstream), and LockedAdamW's lower readings (0.97/0.48 against untreated 5.72/2.24) repair only the conservative component and stay above the stated floors (0.63/0.32)—not sufficient repair by our own definition (Appendix A). The tree's honest output on these two cells is therefore "no sufficient scheme at this budget": the conservative component is repairable, the starvation component is mapped, and the residual gap is priced rather than hidden.

Matrix alternates and floors. Alternates on the ACT rows: lr÷10 (1.13/1.01/0.51), fp8+UPH-z (1.04/0.84/0.40); UPH+LockedAdamW reads RDT 5.17 and CogACT 1.77/1.15, UPH alone on CogACT 3.57/3.76 (a no-op). Floors (fp32 self-retraining pairs): ACT 0.64/0.63/0.32, Octo 4.25, RDT 5.83, GR00T 8.85–13.48, CogACT 1.79–2.35. GR00T's misfire did not reproduce on harness 2; CogACT's pattern replicates on both seeds.

The cost ladder and the lock's prescription. The routing table's cost order is bypass $\rightarrow$ error feedback $\rightarrow$ rate cut $\rightarrow$ UPH coverage; fp8+UPH-z is the format-level answer of Appendix I, chosen when a format change is on the table (its three ACT-row readings—1.04/0.84/0.40, and the top 6k returns of the campaign—make it the strongest starvation-side bundle once fp8 semantics are allowed). The lock is prescribed where the cheap knobs do not reach: measured $\hat{v}$-inflation at scale (CogACT) and long budgets (Square 150k), with its measured boundary in Appendix G.

The from-scratch confound. The SmolVLA cell trains bf16 from scratch no more failure-prone than fine-tuning within this window (`smolvla_matrix.json`).

The behavioral-entropy detector archive (negative result). AUC 0.90 as a monitor; suppressing it does not move drift—a deployment-time dashboard, not a scheme.

The time ledger. Wall-clock per 8k steps, same stack inside every twin pair (Table 25): image-token-heavy loads favor tf32; weight-GEMM-dominated frozen trunks favor bf16.

MH-Lift: the task-level cell, three seeds (RHC-16). ACT$\times$Lift on the MH split (300 demonstrations, state observations), official-runner chunk-open evaluation (RHC-16), 50 paired episodes per checkpoint, 150k steps, seeds 42/43/44. Best-checkpoint success: fp32 64/68/68%, bare 62/66/72%, UPH 70/62/68% (mean 67% each). End-of-training collapse (signature fixed in advance: KL spike $>3\times$ baseline, KL ≤ 0.1, or NaN-abort): fp32 0/3; bare 2/3 (s44: KL spike 17.3 at 141k, flatline by

arm	μ_{div} s43	μ_{div} s42	loss s43	loss s42	outcome
`bare_bf16`	3.80	3.01	0.050	0.055	above the band on both seeds
`uph`	3.57	3.76	0.050	0.052	no-op on both (the failure is not in the chain)
`locked`	1.19	1.27	0.041	0.043	breaks below the floor on both seeds
`uph_locked`	1.77	1.15	0.040	0.042	healthy on both
fp32 floor band (two reference seeds): 1.79–2.35. 66–72 min/arm on Blackwell.					

Table 26: T3 CogACT (7B), two seeds: bf16 damage replicates; LockedAdamW breaks below the fp32 floor on both seeds.

146k; s42: floor touch 0.009 at 108k and spike 47.3 at 112k, then recovered—the task-level reading stayed alive at 44%); UPH 3/3 (s42: spike 9.27 at 132k; s43: a quiet collapse, KL settling below 0.1 from 99k without a spike; s44: spike 7.3 at 136k); bypass 3/3 (s42's posterior died at 106k even though the policy kept scoring 60% at the end—posterior death dissociates from policy death). Every death falls inside the 99–141k mechanism band. The band's other draws on record: bare on PH-Lift (dead at 130k; Appendix H) and the fp32 reference on OpenVLA-OFT (NaN at 4.8k). Absolute rates are protocol-internal: MH-Lift is saturated at sufficient capacity (DP and ACT both report ≈100% on this split (Tan et al., 2024)), and the cell's 0.8M controller under the canonical weak-signal regime is exactly the failure-prone unit this paper studies; the twin comparison, not the absolute rate, is the evidence.

The long-budget campaign (P0-1). Square, 300k steps, twin protocol: bare bf16 collapses to a constant action between 25k and 50k (the fine-grained trace places the death at 44k); LockedAdamW trains through 150k before the abort rule (fifty consecutive NaN steps; Appendix G), in the budget region where the fp32 reference itself is degrading. Lift, 150k: bare bf16 dies at 130k, and LockedAdamW stalls at 51k—the calibration-domain failure of Appendix G. The lock extends the trainable domain by an order of magnitude on Square (44k to 150k); it does not make it infinite (`p0_1_square_converged.json`, `p0_1_square_converged_300k.json`, `p0_1_lift_converged.json`).

The 7B campaign (T3/T4). CogACT (four reference runs + four arms × two seeds, 8000 steps): untreated bf16 is damaged on both seeds; UPH alone changes nothing; LockedAdamW breaks below the floor on both (Table 26). The failure there lives in the second moment, not the chain—as the routing table predicts. On OpenVLA-OFT, the census flags deep heavy swallowing—and the fp32 reference itself collapses mid-run (a threshold-lottery event at production scale), even though bare bf16 and the patched arm both train through on the same data (Table 27): the lottery does not pick its victim by precision. μ_{div} is undefined on OFT (the reference itself is NaN); the determination rests on the loss/std trajectories, and a mundane instability (data or optimization) is not ruled out—the threshold-lottery attribution is our reading, not a proof. The sub-floor reading is stated conservatively: the lock's throttled step mechanically holds a treated trajectory closer to its paired twin than independent runs sit from each other, so below-floor values certify drift indistinguishable from the reference's own variance, not a superior equilibrium—the task-level reading is the MH-Lift cell (§7). The campaign also ran with bnb-8bit optimizer states (two 7B instances exceed device memory): $\hat{v}$ telemetry is coarsened by the 8-bit storage itself, part of the measured inflation is the states' own quantization noise, and the probe's $\hat{v}$ column is unreadable there (Table 20); the cell's determination rests on drift and loss.

I The Boundary Conditions Study in Full

The boundary matrix is Table 28 (this appendix carries it).

T1b (real training hardware). Blackwell + Transformer Engine 2.7.0 (DelayedScaling/HYBRID: e4m3 forward / e5m2 dgrad / per-tensor amax delayed scaling), ACT×Square col-

arm	trajectory	outcome
`fp32_ref` (retrain)	0.293→0.269@4000→NaN@4800	the reference itself lotteried—a threshold event at production scale
`bare_bf16_s43`	trains through, loss 0.133	the untreated arm survives the dataset that killed the reference
`uph_locked_s43`	0.220→0.096, healthy throughout	the patched arm posts the lowest loss

Table 27: T4 OpenVLA-OFT (7B): the fp32 reference collapses mid-run even though both bf16 arms train through—the lottery does not pick its victim by precision. μ_{div} is undefined (the reference is NaN); the determination rests on loss/std trajectories, and a mundane instability is not ruled out.

Format	Scaling	Bottom	KL	μ_{div}	out_std	Verdict
e5m2 unscaled	—	raise	26–34	1.2–1.5	0.60	poisoned
e5m2 unscaled	—	flush	3.2–5.2	4.6–5.3	0.29	starved
e5m2	amax / frozen / global	flush	3.66–3.74	0.39–0.76	0.57	normal (inert)
e6m1 unscaled	—	raise	3.38	0.58	0.56	softened
e6m2 unscaled	—	raise	3.67	0.44	0.57	normal
e8m2–e8m23	—	raise	3.6–3.8	0.3–0.8	0.57	normal
fp8 (e4m3/e5m2)	TE delayed	flush	3.7–3.9	0.7–1.1	0.57	normal (real hw)
fp8 + fp32 island	TE delayed	flush	3.65	**0.71**	0.57	best single reading (s42)
alloc 9.3-bit	per-layer by pattern	raise	3.69	0.47	0.56	normal

Table 28: The tested boundary conditions, ACT×Square, twin protocol, two seeds. Floor position decides life or death; bottom handling chooses the manner. Granularity inert; no mantissa cliff to 2 bits under e8; the exponent cliff between e5 and e6. Real-hardware TE-fp8 confirms the simulator; the island arm posts the best single reading; the pattern-allocated 9.3-bit format needs no scaling.

lapse cell, twin protocol, two seeds: TE-fp8 trains normally everywhere, matching the simulator's bands cell by cell (Table 29)—the simulator is confirmed on real training hardware, not overturned. The island arm (weakest heads fp32, rest fp8) posts the single best reading. The fp32-inference side column separates training damage from inference quantization. Coverage note: `obs_tok` and attention internals are not covered (the %16 shape constraint), exactly matching the simulator's coverage. Harness sensitivity is on record: in this TE/Blackwell harness the bare-bf16 control stays healthy on both seeds, on the cell where the simulator's bare-bf16 arm lotteries—the real-hardware study demonstrates a healthy cell staying healthy under TE semantics, not a rescue; the lottery's boundary shifts with the harness. A wrong-data accident (the v141 delta-action file) was caught, voided, and archived before any number entered the report (`t1b_fp8_real_hw.json`, `_invalid_t1b_wrongdata_20260822.json`).

E16: scaling granularity is inert. amax (dynamic) / frozen (post-warmup) / global single scale, two seeds: all healthy (KL 3.66–3.74; Table 30). The five exponent bits absorb inter-tensor spread on their own; scaling buys only range insurance.

E17: the floor's position decides, the handling chooses the manner. The 2×2 (per-tensor vs unscaled) × (flush vs raise): scaled cells healthy on both handlings (floor touches 8% near-zero

arm	KL (real)	KL (sim)	μ_{div} (real)	μ_{div} (sim)	out_std
`fp8te_bare s42`	3.94	3.73	0.80	0.95	0.568
`fp8te_island s42`	3.65	3.65	0.71	0.90	0.570
`fp8te_bare s43`	3.82	—	0.94	—	0.567
`fp8te_island s43`	3.86	—	1.11	—	0.564
`bare_bf16 s42`	3.73	—	0.79	—	0.566
`bare_bf16 s43`	3.63	—	0.80	—	0.566

Table 29: T1b: real-hardware TE-fp8 on the collapse cell, reconciled cell-by-cell with the simulator (twin protocol, two seeds).

cell	KL	μ_{div}	out_std	note
square-te_amax-s42	3.68	0.41	0.56	
square-te_amax-s43	3.71	0.69	0.57	
square-te_frozen-s42	3.69	0.39	0.57	
square-te_frozen-s43	3.70	0.76	0.57	
square-te_global-s42	3.66	0.46	0.57	
square-te_global-s43	3.74	0.62	0.57	

Table 30: E16 per-cell results: scaling granularity is inert (amax/frozen/global, two seeds).

cell	KL	μ_{div}	out_std	note
square-bare_flush-s42	5.18	4.60	0.32	
square-bare_flush-s43	3.16	5.27	0.29	
square-bare_raise-s42	26.44	1.27	0.60	
square-bare_raise-s43	21.26	1.16	0.59	
square-te_flush-s42	3.68	0.41	0.56	
square-te_flush-s43	3.71	0.69	0.57	
square-te_raise-s42	3.57	0.58	0.56	
square-te_raise-s43	3.69	0.63	0.56	

Table 31: E17 per-cell results: floor position decides, handling chooses the manner (2x2, two seeds).

elements only); unscaled cells die both ways—flush starves (halved range, normal KL), raise poisons (exploding KL; Table 31; Figure 8). The floor-mass readout: 8% of elements touched when scaled, 96% when unscaled (`e17_floor_scaling_2x2.json`, `p0_2_cliff_seeds.json`).

E18: no mantissa cliff. Under an eight-bit exponent, chain quantization stays normal from 23 mantissa bits down to 2—rounding magnitude (up to 12.5%) does not collapse the cell; direction does (the decisive cell at format scale) (Table 32; Figure 9).

E19: the exponent cliff, and the designed minimal format. The cliff now holds at four seeds: e5m2 explodes on all four, e6m2 and e6m1 train normally on all four (Table 33)—e6m1's softening is the measured safety margin. The new arms sharpen the same picture: TE-hardware flush and raise both train normally (floor touching 8.4% of elements), whereas the unscaled bare-simulator pair splits by handling—flush starves and raise poisons (`p0_2_cliff_seeds.json`). The cliff sits between e5 and e6, roughly where the floor crosses the weakest signal ($\sim 2^{-20}$). **e6m2 is the minimal format the patterns prescribe on this cell**: 9 bits total (1+6+2), no scaling machinery (Figure 10).

cell	KL	μ_{div}	out_std	note
square-e8m10-s42	3.77	0.52	0.56	
square-e8m10-s43	3.77	0.62	0.57	
square-e8m12-s42	3.74	0.39	0.57	
square-e8m12-s43	3.79	0.68	0.57	
square-e8m2-s42	3.61	0.40	0.56	
square-e8m2-s43	3.71	0.64	0.57	
square-e8m23-s42	3.75	0.00	0.57	
square-e8m23-s43	3.78	0.67	0.57	
square-e8m3-s42	3.71	0.47	0.57	
square-e8m3-s43	3.78	0.67	0.57	
square-e8m5-s42	3.74	0.44	0.56	
square-e8m5-s43	3.77	0.76	0.57	
square-e8m7-s42	3.78	0.35	0.57	
square-e8m7-s43	3.78	0.65	0.56	
square-e8m9-s42	3.73	0.31	0.57	

Table 32: E18 per-cell results: no mantissa cliff (e8 exponent, m23 down to m2, two seeds).

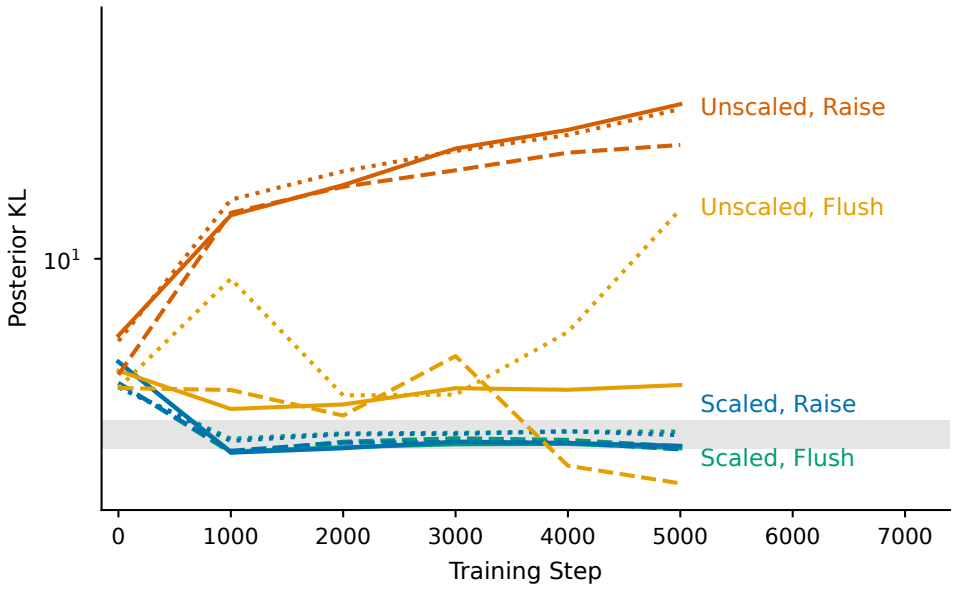


Figure 8: E17: the floor×handling 2×2 (KL trajectories, two seeds each). Scaled cells stay in the band on both handlings; unscaled cells diverge—starved (flush) or poisoned (raise).

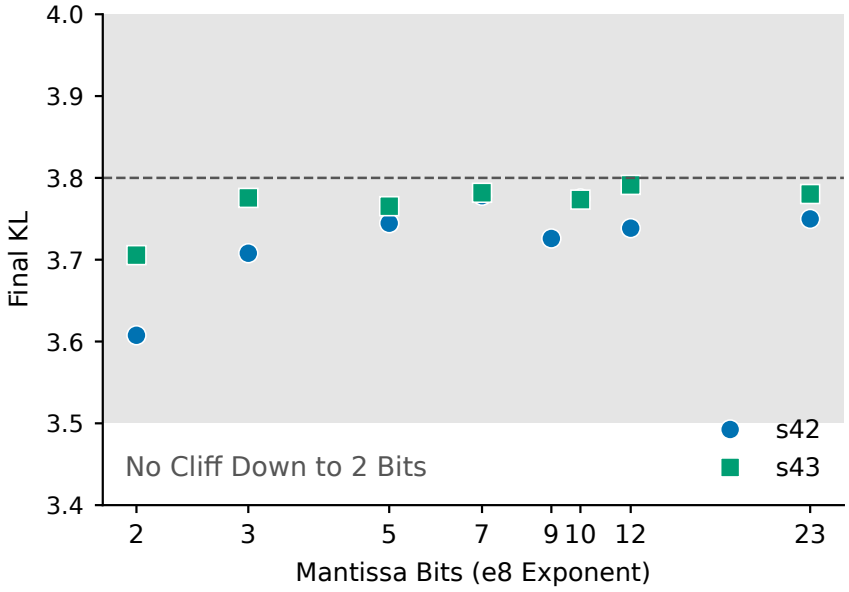


Figure 9: E18: final KL against mantissa bits (e8 exponent). No cliff down to 2 bits.

The allocator and the immunity statement (E20). The patterns make per-layer allocation a closed form, not a search: a census measures each layer's weakest signal (5th-percentile incoming-gradient magnitude), and each layer gets the minimal exponent whose floor clears it with margin ($\text{floor}(E) < w_l/4$), with two mantissa bits (E18), unbiased rounding, and full precision at chain-critical weak heads (the island rule)—sensitivity-driven per-layer allocation in the spirit of inference-time mixed precision (Dong et al., 2019), driven here by exponent floors rather than bit widths. On ACT: the allocation converges to uniform e6m2 plus the island (weakness is near-uniform there; differentiation awaits heterogeneous VLA trunks). The allocated model trains normally on both seeds at **9.3 average bits**, no scaling machinery (Tables 34, 35; `e20_alloc.json`). **Immunity (sufficiency).** Under preconditions—accumulator precision below the minimal update ratio, exponent coverage below the weakest signal, flush discipline at the bottom (inert once the floor clears the

cell	KL	μ_{div}	out_std
square-e5m2-s42	29.01	1.52	0.60
square-e5m2-s43	34.27	1.45	0.60
square-e5m2-s44	29.89	1.28	0.60
square-e5m2-s45	29.72	1.46	0.60
square-e6m1-s42	3.38	0.58	0.56
square-e6m1-s43	3.42	0.82	0.56
square-e6m1-s44	3.50	0.66	0.57
square-e6m1-s45	3.38	0.69	0.56
square-e6m2-s42	3.67	0.44	0.57
square-e6m2-s43	3.71	0.65	0.57
square-e6m2-s44	3.80	0.41	0.57
square-e6m2-s45	3.67	0.53	0.56

Table 33: E19 per-cell results: the exponent cliff sits between e5 and e6 (unscaled e5m2 under raise handling; four seeds; s44/s45 from `p0_2_cliff_seeds.json`).

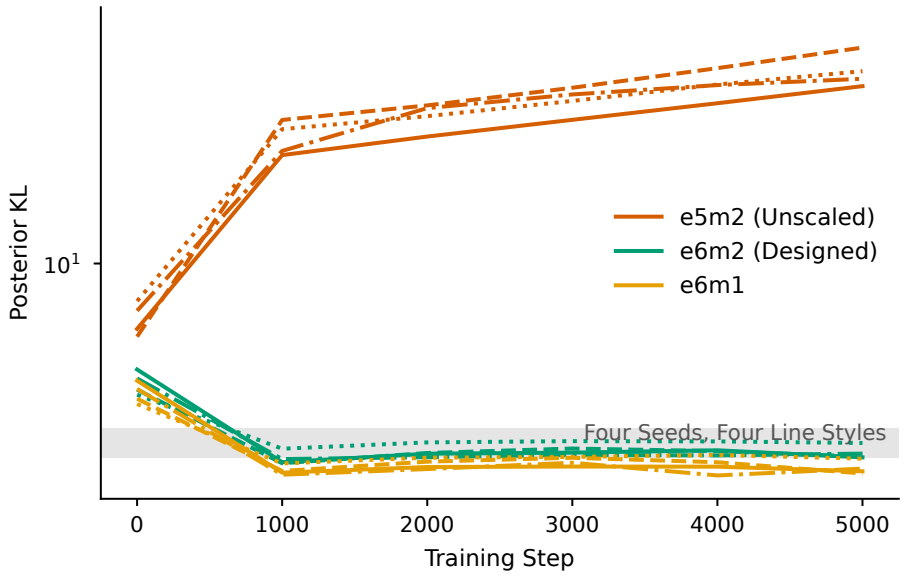


Figure 10: E19: KL trajectories for e5m2 (explodes), e6m2 (designed, healthy), e6m1 (softens). The exponent cliff sits between e5 and e6.

signal mass, as E20's raise-built island attests), locked noise budget at the optimizer, closed-form bypass at chain-critical heads—no silent collapse occurs with high probability under the threshold-lottery model, and any precondition violation triggers the monitor (loud). *This agrees with E16–E20.* This is stated on the certified cell (ACT×Square): it is a prescription for format designers, not a cross-task guarantee.

J The Behavioral Ruler, Validated

μ_{div} vs rollout return. Twenty-one arm×seed cells carry both readings (Table 36): the drift reading ranks the failure zone (Spearman -0.70) and certifies health, but does not order healthy arms among themselves near the floor (Spearman $+0.13$)—the ruler's resolution boundary is explicit. Counterexample on record: at equal drift (μ_{div} 1.05 on Square-s43) the manual rate cut returns 22.4 where LockedAdamW returns 7.0.

The ruler's blind spot: poisoning reads healthy. μ_{div} is a directional monitor: it catches the freeze—starvation drives actions to a constant and drift to $6\times$ the floor—but not the flood. Under unscaled e5m2 with raise handling the posterior is poisoned (KL 21–26, Table 31) whereas drift

layer	weakest $\|g\|$ (q05)	E_l
dec.layers.0.linear2	1.2e-07	6
enc.layers.0.linear2	1.3e-07	6
enc.layers.0.linear1	1.3e-07	6
enc.layers.1.linear2	1.6e-07	6
enc.layers.1.linear1	1.6e-07	6
z_lv	1.6e-07	fp32
dec.layers.0.linear1	1.9e-07	6
dec.layers.1.linear2	1.9e-07	6
dec.layers.1.linear1	2.0e-07	6
z_proj	5.9e-07	6
out	5.9e-07	6
z_mu	7.8e-07	fp32
average bits (Linear-weighted): 9.3		

Table 34: E20: the per-layer census and closed-form allocation (island layers keep fp32).

cell	KL	μ_{div}	out_std
square-alloc-s42	3.68	0.47	0.56
square-alloc-s43	3.71	0.62	0.56

Table 35: E20 allocated model: healthy on both seeds at 9.3 average bits, no scaling machinery.

reads 1.16–1.27, at the floor's edge: the poisoned policy still acts, and its per-observation mean stays near the reference's—the damage lives in the posterior's variance, not in the mean's position. Certification therefore pairs the drift reading with the posterior KL where one exists (out_std otherwise): drift certifies the mean, the second reading certifies the spread, and health requires both (e17_floor_scaling_2x2.json).

The basin control (E25, two scales). μ_{div} against the paired-init fp32 twin versus an independent fp32 run: ratios 1.00–1.08 at ACT scale (Table 37) and 0.89–1.09 at Octo scale (octo_basin_control.json). The twin protocol measures the same thing an independent reference does—the basin-artifact account of below-floor readings is ruled out at both scales.

The noise floor's derivation and the statistical-power analysis (fp32 self-retraining pairs). Two horns, one ruler: the floor band is the null region—readings above it are failures; readings below it are read through Pattern II (fp32's equilibrium is not the signal's true equilibrium). The twin is reference, not oracle: the band certifies, and both excursions inform. A bf16 self-retraining floor is never measured; if the low-precision arm's intrinsic variance exceeds the fp32 floor, part of any above-floor reading is seed noise rather than failure—the determinations that matter (bare above band on CogACT, the lottery cells) sit far enough above the band to be insensitive to this.

Where the ruler fails (its failure modes), and the bare_flush-s43 case file: KL 3.16 reads normal even though behavior is dead—no single numerical monitor is reliable; the behavioral ruler is the only final account.

K Environment and Reproducibility

Environments. Consumer GPU (8GB, local) for small-matrix cells; rented datacenter GPUs (PRO 6000 Blackwell 96G, sm_120) for ladder cells, one per model. torch 2.7.1+cu128, CUDA 12.8, cuDNN 90701, Transformer Engine 2.7.0; further library versions (robosuite, diffusion_policy, robomimic, MuJoCo) are pinned in the supplementary manifests.

The canonical protocol. Every cell follows Table 38 unless the paragraph states otherwise; deviations are always fixed in advance and noted in place (e.g., β=10 in P1-1, warmup W=200 in the T9 cells).

arm	seed	μ_{div}	return
bare	s42	6.45	2.0
bare	s43	3.93	3.5
bare	s44	5.13	1.9
locked	s42	0.95	13.4
locked	s43	1.05	7.0
locked	s44	0.99	15.9
sr_fp32	s42	4.01	1.3
sr_fp32	s43	5.27	3.3
sr_fp32	s44	6.45	2.4
fp8_sr	s42	0.95	23.2
fp8_sr	s43	1.08	49.2
fp8_sr	s44	1.09	16.1
lr1e4	s42	1.17	19.1
lr1e4	s43	1.05	22.4
lr1e4	s44	1.17	10.7
fp32_locked	s42	0.95	10.2
fp32_locked	s43	1.04	7.0
fp32_locked	s44	0.98	11.3
sgd1e3	s42	3.37	7.4
sgd1e3	s43	3.69	5.8
sgd1e3	s44	3.32	5.9

Spearman overall -0.73; failure zone ($\mu_{\text{div}} > 2$) -0.70; near floor (< 1.5) 0.13.

Table 36: Drift vs. task-level return, 21 arm×seed cells (6k budget, drift_return_validity.json). The ruler ranks the failure zone and certifies health; it does not order healthy arms among themselves.

arm	μ_{div} (twin)	μ_{div} (independent)	ratio
locked_s42	1.239	1.148	1.08
bare_s42	6.446	6.431	1.00

Table 37: E25 basin control: the twin protocol matches an independent reference (ratio 1.00–1.08); the basin-artifact account is ruled out at ACT scale.

Evaluation protocol. Evaluations are paired across arms: every arm of a cell is evaluated on the same episode initializations (50 paired episodes per checkpoint; 100 at the 20k anchor), every 25k steps in the long-budget campaign. μ_{div} and out_std are measured on held-out observations at matched training steps against the fp32 twin. Success follows each task's native checker; return is the accumulated per-step environment reward.

item	canonical value
optimizer	AdamW: lr 10^{-3}, weight decay 10^{-6}, grad clip 1.0
batch size	256
training steps	6000 (small-matrix cells); 1k/2k/4k/8k (ladder cells); 150k–300k (long-budget campaign)
ACT KL weight β	10^{-2}; β=10 only in the production-β cell (P1-1)
twin protocol	paired initialization and paired data order per seed; the fp32 twin is the reference
seeds	42/43/44 (cliff cells add 45)
LockedAdamW	warmup 100 steps (Table 2 cells) or 200 (T9 cells); clamp $[0.05, 3.0]$
probe	50-step twin; $\hat{v}$ ratio ≥ 1.2 →LockedAdamW; kl_div_late ≥ 0.05 → bypass; else nothing
abort rule	50 consecutive NaN steps terminate a run (recorded as NaN-abort)
fp8 simulator	fp32 accumulation followed by an explicit output cast (round-to-nearest-even)

Table 38: The canonical protocol. Every number in the paper comes from runs under this table plus the stated per-cell deviation.

cell family	dataset	policy / teacher
ACT Square, Can	robomimic `low_dim_abs` (ph, 200 demos)	ACT (Zhao et al., 2023); teacher: diffusion policy (ep1750)
ACT Kitchen	robomimic `kitchen_demos_multitask`	ACT (Zhao et al., 2023)
Lift	robomimic Lift ph (`low_dim_v141`)	ACT (Zhao et al., 2023)
MetaWorld	9 tasks, state observations (Yu et al., 2021)	TD-MPC2 official checkpoints (`<task>.pt`)
ladder families	per official recipes	Octo, RDT, $\pi_{0.5}$, GR00T, OpenVLA-OFT, CogACT, SmolVLA (official releases; manifest)

Table 39: Datasets, teachers, and model sources. Links and checksums live in the supplementary manifests.

Assets. Datasets, teachers, and model sources are tabulated in Table 39; links and checksums are in the supplementary manifests.

Rounding and accumulation. Native bf16 autocast casts with round-to-nearest-even (RNE). The fp8 simulator models hardware matmul as fp32 accumulation followed by an explicit output cast, and the cast-simulation arm reproduces the native bare-bf16 collapse cell-for-cell (Appendix E); a reduction-precision switch (fp32 vs reduced accumulation in the GEMM path) reproduces the untreated trajectory bit-for-bit on both seeds (15 decimal places), so the bias is sourced at the casts, not the accumulator (`p0_3_rz_hypothesis.json`). UPH's stochastic-rounding dither is injected on the incoming activation gradient, before the backward matmuls. Per-cell batch size and accumulation follow the canonical protocol (Table 38) and the archived manifests.

Manifests and the claim-to-record map. Every evidence paragraph cites its source record inline (`*.json`/`*.jsonl`); the archive index in the supplementary material maps each record to its run manifest, and each record carries its harness fingerprint (GPU model, CUDA/cuBLAS versions, batch and accumulation configuration). Code, data, and weight manifests include all HF links.

Anonymous code. UPH, LockedAdamW, the probe, and all experiment scripts are provided with the supplementary material; the probe and the figures regenerate from the archived records alone (the reproduction guide is included).