QUANTIZING LOOPED TRANSFORMERS: FEEDBACK EXPOSURE AND CALIBRATION BLINDNESS

Anonymous authors
Paper under double-blind review

ABSTRACT

Looped transformers reuse weights across recurrence steps, making low-bit quantization especially attractive. We identify two distinct failure modes of standard post-training quantization. On Huginn-3.5B, per-channel INT4 fails primarily at the non-residual loop-entry adapter, while quantizing the residual core is much less damaging. We call this *feedback exposure*: a quantized layer perturbs the recurrent state without an identity path, and the resulting error is fed back at later steps. Controlled experiments on linear filters and Mamba state-space models show that feedback exposure also occurs outside transformers. Grouped INT4 reveals a separate failure, *calibration blindness*: our one-step GPTQ baseline builds its Hessian from step-0 activations, leaving input directions used later in the recurrence nearly unweighted. Across nine checkpoints from seven looped architectures, one-step GPTQ is worse than round-to-nearest (RTN) on the primary task metric for five checkpoints. Accumulating the GPTQ Hessian across recurrence steps outperforms both one-step GPTQ and RTN on all nine checkpoints and recovers bf16-level accuracy on Huginn. These results separate two questions for PTQ on looped models: where quantization error enters the recurrence, and which states calibration sees.

1 INTRODUCTION

Looped transformers iterate a shared layer block N times, decoupling computational depth from parameter count. Huginn (Geiping et al., 2025) has 3.5B parameters but unfolds to an effective depth of 132 layers at 32 recurrences; Ouro-1.4B and Ouro-2.6B (Zhu et al., 2025) reuse 24- and 48-layer stacks for four loop steps (96 and 192 block applications), and COCONUT (Hao et al., 2024) recycles GPT-2's entire forward pass. For single-pass transformers, INT4 quantization roughly quarters the weight footprint. We test whether that win transfers to looped models where the same weights are reused across recurrence steps.

On a 100-example GSM8K diagnostic, per-channel INT4 on the loop-entry adapter and recurrent core reduces exact match from 34% in bf16 to 0% (Appendix Table 5). To localize this collapse, Figure 1a compares each configuration's token predictions with bf16 on a separate 50-prompt diagnostic. Adapter-only INT4 ends at 10.5% agreement, nearly matching 10.6% when both components are quantized; core-only INT4 retains 87.9%. COCONUT's residual GPT-2 stack behaves differently, with agreement rising from 44.8% after the base forward to 61.5% after 8 latent recurrences (Figure 1b).

Standard one-step sensitivity metrics do not explain this split. Small random input perturbations produce the least local output change at the adapter (17th of 17 projections), while Hessian trace ranks a core projection as most sensitive even though quantizing that projection alone at INT4 group size 128 changes far fewer predictions than quantizing the adapter. These metrics omit where the rounding error enters the recurrence. Section 3 isolates this mechanism; linear-filter and Mamba controls show that transition perturbations are also more damaging outside transformers.

Grouped INT4 (g=128) avoids the immediate collapse, but one-step GPTQ still underperforms round-to-nearest (RTN) on five of nine checkpoints. Its step-0 Hessian misses directions used later in the recurrence, a mismatch that persists with more calibration prompts. These mechanisms are independent. Feedback exposure identifies where perturbations are structurally dangerous, while calibration blindness identifies when GPTQ calibrates the wrong subspace. Huginn's adapter lies

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.

at their intersection. We isolate the first with controlled interventions and address the second with trajectory calibration across nine checkpoints from seven looped architectures.

2 BACKGROUND

Looped transformers. A looped transformer applies a shared block iteratively to a latent state:

$$s_{t+1} = R_\theta(s_t, e, c_t), \qquad t = 0, \ldots, N-1, \tag{1}$$

where s_t is the latent state, e is the fixed prompt embedding, and c_t is architecture-specific conditioning.

Huginn-3.5B (Geiping et al., 2025) and Recurrent-Llama-1.4B (R-Llama) (McLeish et al., 2025) route recurrence through a non-residual adapter that projects the concatenation of s_t and e to d dimensions; their recurrent cores remain residual. Other looped models reuse residual layers without a dedicated adapter (Zhu et al., 2025; Jeddi et al., 2026; Hao et al., 2024) or impose explicit stability constraints on the recurrent core (Prairie et al., 2026).

We use the local spectral radius ρ of the per-step Jacobian $\partial s_{t+1}/\partial s_t$ as a first-order stability diagnostic; $\rho > 1$ indicates linearized amplification in some direction. The echo state property requires trajectories driven by the same input to forget their initial conditions (Jaeger & Haas, 2004); uniform contractivity is a sufficient global condition, whereas our per-step ρ is a local surrogate. For GPTQ calibration, we report the rank of $H = X^\top X$ at threshold $10^{-3}\lambda_{\max}$, written $\#\{\lambda_i(H) > 10^{-3}\lambda_{\max}\}$, which counts calibration directions above a fixed relative energy threshold.

Post-training quantization. Per-channel INT4 quantizes each output row with a single scale factor. At g=128, each group of 128 consecutive columns gets its own scale, which lowers quantization error but adds storage overhead. GPTQ (Frantar et al., 2022) rounds weights column by column to minimize $\|XW^\top - X\hat{W}^\top\|_F^2$, where each row of X is a calibration activation and $H = X^\top X$ is the per-layer Hessian proxy. To spread outlier energy, QuIP# (Tseng et al., 2024) and SpinQuant (Liu et al., 2025) apply orthogonal transforms across dimensions. QuIP# adds lattice codebooks for sub-4-bit targets. AWQ (Lin et al., 2024) instead rescales channels by activation magnitude and does not use a Hessian proxy. To test whether step-0 Hessians miss deployed directions, we compare GPTQ using H from each shared layer's first invocation with GPTQ using H accumulated over all N invocations.

Quantization under iteration. Q-Diffusion (Li et al., 2023) samples calibration data across denoising timesteps, while AdaTSQ (Zhang et al., 2026) combines timestep-dependent activation policies with temporally weighted calibration for static weights. Both address activation shifts along recursively generated denoising trajectories. Looped transformers create a related shift by feeding the shared block's output back as its next latent state. For linear time-invariant systems, Quamba (Chiang et al., 2025) bounds quantized error when the transition matrix A is unperturbed. Li et al. (2026) analyze quantized monotone DEQs and show that convergence survives operator perturbation when the monotonicity margin is large enough. The checkpoints studied here come with no such certificate.

Residual robustness under reuse. Veit et al. (2016) showed that ResNets tolerate layer *deletion* at test time. Their path-ensemble argument implies that deleting one of L residual blocks removes 2^{L-1} of the 2^L paths. The argument is feedforward because each residual block appears at most once on any path. In a looped model, the same weight perturbation acts on the recurrent state at every step, so this path-counting argument does not describe how its effect evolves across recurrence.

3 FEEDBACK EXPOSURE UNDER RECURRENCE

A looped transformer applies the same quantized weights at every step of the recurrence, but not all shared layers contribute equally to the resulting damage. We separately quantize the adapter and core with per-channel INT4 in Huginn and R-Llama to isolate which layers are responsible. Grouped INT4 (g=128) is used for the sensitivity-ranking comparison in Appendix G and for the recovery experiments in Sections 5 and 6.

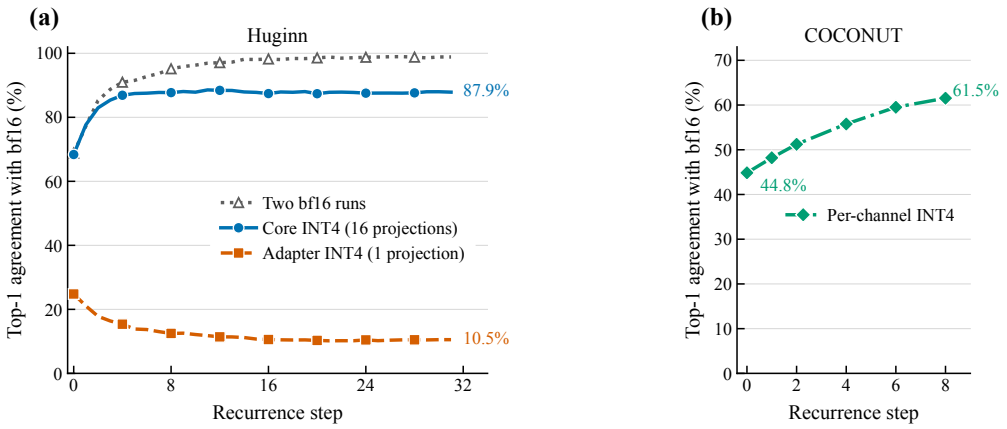

Figure 1: Per-channel INT4 feedback exposure. (a) Huginn's non-residual adapter-only agreement falls to 10.5%, while quantizing all 16 residual-core projections together retains 87.9%. Two bf16 runs with independently sampled initial states reach 98.9% agreement. (b) On 15 fixed prompts, COCONUT agreement rises from 44.8% after the quantized base forward to 61.5% after 8 latent recurrences.

Unless noted, the Huginn and R-Llama diagnostics use deterministic symmetric round-to-nearest per-channel INT4 with simulated float32 dequantization, with all other layers in bf16. Agreement is the fraction of token positions whose top-1 prediction matches bf16 at the same step, averaged equally over the first 50 GSM8K test prompts.

3.1 Controlled Injection

We first isolate the adapter and recurrent core. Adapter-only INT4 drops final-step agreement with bf16 to 10.5%, whereas quantizing all 16 core projections leaves 87.9% (Figure 1a). Agreement under core-only INT4 rises over the recurrence, and quantizing both components yields 10.6%, matching the adapter-only result.

To test whether recirculating prior state error drives this damage, we vary how much of that error reaches the adapter at the next step. This control uses 50 GSM8K training prompts disjoint from calibration and 3 recurrent-state initializations shared across all λ values. Writing s_t^λ for the intervened state, the quantized adapter receives $s_t^{\text{bf16}} + \lambda(s_t^\lambda - s_t^{\text{bf16}})$, where $\lambda = 0$ removes prior state error and $\lambda = 1$ gives the ordinary quantized rollout. Removing prior error raises final-step agreement from 10.4% to 69.2%. The relative final-state error $\|s_{32}^\lambda - s_{32}^{\text{bf16}}\|_2 / \|s_{32}^{\text{bf16}}\|_2$ grows monotonically with λ under both per-channel and grouped INT4. Quantizing the recurrent-state and fixed-embedding columns separately gives the same trend, although the feedback effect is larger for the embedding columns (Appendix E).

R-Llama-1.4B shows the same asymmetry at smaller scale: adapter-only agreement is 49.1%, compared with 74.8% for core-only (Appendix Table 4). Standard sensitivity rankings also miss this asymmetry. In Huginn's grouped-INT4 single-projection ablation, the adapter causes the most damage, although small random input perturbations rank it last and Hessian trace ranks it third (Appendix Table 16).

The adapter/core asymmetry persists under other weight perturbations (Appendix F). At the same Frobenius norm for both locations, a random rank-4 weight update leaves 71.8% agreement at the adapter and 95.8% at the core, a 24.0 pp gap. A Gaussian weight perturbation scaled separately at each layer to match its grouped-INT4 output-error norm likewise leaves lower agreement at the adapter than at the core (89.3% vs. 96.5%), despite a smaller matched norm at the adapter (3.16 vs. 8.11). By comparison, per-channel INT4 rounding at the same adapter position drops agreement to 10.5%.

Together, these controls tie the collapse to the error's position in the recurrence, its repeated feedback, and the structure of the perturbation.

By contrast, COCONUT reuses GPT-2's full transformer stack across latent steps and has no dedicated adapter, so every layer sits on an identity path. Under per-channel INT4 on the shared stack, agreement with bf16 improves from 44.8% after the quantized base forward (N=0 latent recurrences) to 61.5% after 8 latent recurrences, and KL divergence falls from 1.25 to 0.93 (Figure 1b).

3.2 Feedback Exposure and the Identity Path

At a fixed operating point, the Jacobian of one full recurrence step with respect to s_t factors as

$$J = \underbrace{\textstyle\prod_i \left(I + \frac{\partial F_i}{\partial x}\right)}_{J_{\text{core}}} \times W_s, \tag{2}$$

where F_i is the net residual update of core block i. Writing the linear adapter as $W_s s_t + W_e e + b$, quantization changes its output by $\Delta W_s s_t + \Delta W_e e$. The first term directly perturbs the state-input Jacobian, while both errors enter the next recurrent state through the same unprotected adapter output. No identity path carries s_t around the adapter. Each core factor includes the identity ($I + \partial F_i/\partial x$, with RMSNorm, attention, and MLP nonlinearities inside F_i), so each block retains an additive identity path for its input. We call a perturbation *feedback-exposed* when it acts on the recurrent state without an identity path around it.

Spectral radius. For each prompt, we evaluate the bf16 and INT4 Jacobians at its bf16 state after 32 recurrence steps. The mean ρ is 1.064 for bf16 and 1.145 after per-channel adapter INT4; the INT4 map has $\rho > 1$ on all 30 prompts, versus 9/30 for bf16.

Identity-bypass control. To isolate the bypass, we hold each prompt's state and the bf16/INT4 adapter weights fixed, then apply the same small state perturbations with and without adding s around the adapter. The bypass reduces the RMS difference between the bf16 and INT4 one-step changes by 37.4% (41.7% after normalizing each difference by its bf16 response magnitude; Appendix K).

R-Llama fails differently from Huginn. Per-channel RTN reduces GSM8K accuracy to 0%, but the measured trajectory displacement stabilizes quickly. Starting from the bf16 state after 32 recurrence steps, a diagnostic rollout advances the bf16 and quantized trajectories for 32 more updates and measures $\|s_t^{\text{q}} - s_t^{\text{bf16}}\|_2$. For both per-channel and g=128 INT4, mean displacement is within 1% of its final value after 4 updates. The final norm is 1082 under per-channel INT4 and 300 at g=128 (Appendix L). This finite-horizon saturation is consistent with a displaced fixed point: the measured trajectory remains bounded without returning to the bf16 state.

Feedback exposure in IIR filters. An infinite-impulse-response (IIR) filter isolates the same structural distinction in a linear recurrence. Its matrix A feeds the previous state into the next step, while B injects the current input and C, D produce the output. The poles are the eigenvalues of A, so only quantizing the feedback transition can change stability. Quantizing A moves at least one pole outside the unit circle in all 4 tested Butterworth, Chebyshev, and elliptic filters at 10 bits or below; quantizing B, C, D leaves every pole unchanged across all 6 tested precisions (Appendix M).

Feedback exposure in SSMs. Mamba (Gu & Dao, 2023) shows the same distinction between perturbing the state transition and perturbing components outside it. On Mamba-2.8B, let $d_t = \|h_t^{\text{INT4}} - h_t^{\text{bf16}}\|_2 / \|h_t^{\text{bf16}}\|_2$ denote the relative SSM-state error at token t. Under symmetric INT4 round-to-nearest on `A_log`, the ratio of median d_{32} to median d_8 ranges from 1.30 to 1.71 across 5 perturbed layers and 10 prompts. Perturbing `out_proj` changes the block output but leaves the same layer's SSM state unchanged because the projection follows the scan. The recurrence explains this behavior: $A = -\exp(A_{\log})$ and $\bar{A}_t = \exp(\Delta_t A)$, so quantizing `A_log` changes the multiplier on h_{t-1} at every token even though each diagonal entry of $\bar{A}_t$ lies in $(0, 1)$. In a Vim-S 4-bit activation ablation, QMamba (Li et al., 2025) reports a 66.8-point top-1 drop from quantizing $\bar{A}_t$, the largest among its tested SSM activations, while Bi-Mamba (Tang et al., 2025) retains the SSM parameters at full precision and binarizes the projections. Both results are consistent with feedback exposure (Appendix N).

Mitigating feedback-exposed layers. In Huginn at g=128, the adapter is 1 of 17 shared projections. In a 100-example GSM8K ablation, keeping only the adapter in bf16 raises exact-match accuracy

from 25% to 32%, recovering 7 of the 9 points lost to quantization (Appendix Table 12). At per-channel INT4, adapter-only Hadamard rotation with trajectory calibration reaches 29.4% exact match on all 1,319 GSM8K test examples, compared with 34.9% in bf16 (Appendix D). Huginn's localized feedback path permits these projection-specific fixes. Ouro and LoopFormer distribute recurrence across the full stack, so we evaluate trajectory calibration instead (Section 4).

4 Calibration Blindness from Step-0 Hessians

One-step GPTQ underperforms round-to-nearest on five of nine checkpoints (Table 2). This grouped-INT4 failure has a different cause from the per-channel collapse in Section 3. The Hessian solver can optimize the wrong subspace even when immediate collapse is absent. Our one-step GPTQ baseline rounds weights to minimize a Hessian-weighted proxy $\mathrm{tr}(\Delta W\, H_0\, \Delta W^\top)$, where H_0 is built from step-0 activations alone. For layers whose input shifts across recurrence, the step-0 proxy leaves most directions weakly penalized.

Table 1: Per-layer thresholded Hessian rank at $10^{-3}\lambda_{\max}$ on 50 GSM8K prompts for Huginn-3.5B. The adapter's step-0 Hessian covers only 43 of 10,560 input dimensions. Accumulating 32 recurrence steps raises this to 335.

Layer	Dimension	1-step rank	1-step %	32-step rank	Gain
Adapter	10,560	**43**	**0.41%**	**335**	**7.8×**
Core Wqkv (avg)	5,280	∼1,050	20%	∼1,450	1.3×
`mlp.proj` (block 3)	17,920	65	0.36%	89	1.4×

Of 10,560 adapter input dimensions, only 43 exceed the $10^{-3}\lambda_{\max}$ threshold in H_0. Greedy column compensation can reduce the step-0 proxy while leaving error along directions that the proxy weights weakly. Later recurrence steps activate additional directions, so a small step-0 proxy does not guarantee low error along the deployed trajectory. Core Wqkv starts at 20% coverage and gains only $1.3\times$ when later-step activations are included, compared with $7.8\times$ for the adapter.

1,000-prompt control. At step 0, the adapter concatenates a noise-initialized recurrent state h with the content-dependent prompt embedding e. Across 1,000 prompts, these inputs still occupy a narrow step-0 subspace. To compare sample counts, we use the row-averaged Hessian $\widetilde{H} = 2X^\top X/m$ for m activation rows; this scaling leaves thresholded rank unchanged. Rank falls from 43 at 50 prompts to 20 at 1,000 while the top eigenvalue changes by less than 0.3%, as near-threshold eigenvalues shrink with more samples. The top-32 eigenspaces of the uncentered recurrent-state second moments at steps 0 and 31 are nearly orthogonal across 2 initialization seeds (mean squared cosine of principal angles: 0.57–0.63%; Appendix O), confirming that step-0 data does not cover later-state directions.

Per-step rank. Holding the same 2,919 prompt-token positions across steps, Huginn's rank rises from 43 at step 0 to 259 at step 1 and 363 at step 4, ruling out sample accumulation. Ouro and LoopFormer also show large first-step jumps, while COCONUT changes little (Appendix Table 19).

Step-0 rank versus trajectory shift. Block-3 `mlp.proj` has similarly low step-0 rank (0.36%, Table 1) but only $1.4\times$ rank gain, versus $7.8\times$ for the adapter. The contrast shows that low step-0 rank alone does not measure trajectory shift.

Ouro-1.4B has no Huginn-style adapter, yet its layer-0 `down_proj` has rank 1/5,632 (0.02%) at step 0, while per-step Hessians at steps 1–3 individually reach rank 974–1,194 (Appendix Table 18). Step 0 contributes 99.5% of the 4-step trace sum, leaving the later directions below the accumulated Hessian's relative cutoff and its thresholded rank at 1. Their covariance still changes GPTQ's solution: 4-step calibration improves PPL from 14.67 to 12.97 (Table 2). Calibration blindness therefore extends beyond Huginn. Huginn's adapter is distinctive because it is both feedback-exposed and calibration-blind.

Method dependence. AWQ (Lin et al., 2024) selects per-channel scales from activation magnitudes, which are stable across recurrence steps (Pearson $r = 0.89$ between step 0 and step 31 on the Huginn

adapter). AWQ is insensitive to calibration step (0.01 PPL range vs. 0.29 for GPTQ), but step-0 AWQ improves by only 0.06 PPL over RTN with the same asymmetric quantizer, comparable to AWQ's 0.01–0.18 PPL gains on standard LLMs at the same precision (Lin et al., 2024) (Appendix Table 13). Hessian-based solvers exploit the covariance subspace that per-channel magnitudes miss, but their reliance on H makes them vulnerable to step-0 mismatch.

4.1 Cross-Architecture Rank Survey

Rank gain is the accumulated-to-step-0 rank ratio for one shared layer, measuring how much subspace coverage grows after step 0. Figure 2 reports the maximum for each checkpoint and plots recovery separately on the vertical axis. LoopFormer and COCONUT start from comparable minimum step-0 rank ratios (1.5% for LoopFormer `c_fc` and 1.4% for COCONUT `c_proj`; Appendix Table 18), yet later steps add far more directions for LoopFormer. All nine checkpoints in Table 2 have a maximum rank gain of at least $2\times$, so later steps add above-threshold directions across every architecture tested.

Rank gain is large when a layer's step-0 input differs qualitatively from its later-step input. Huginn's adapter concatenates a noise-initialized state with the prompt embedding. Ouro's layer-0 sees raw embeddings before any transformer block has shaped them. LoopFormer's iteration-conditioned adaLN (Jeddi et al., 2026) changes layer inputs across loops. Among its GPTQ-quantized projections, the lowest-rank `c_fc` gains $4.1\times$. By contrast, COCONUT reuses GPT-2's full residual stack across latent steps, and its per-step ranks barely shift ($43 \rightarrow 50$).

Model-level averages hide low-rank layers. Ouro-2.6B's worst layer (0.02%) is masked by 47 near-full-rank layers, so an average would not flag it.

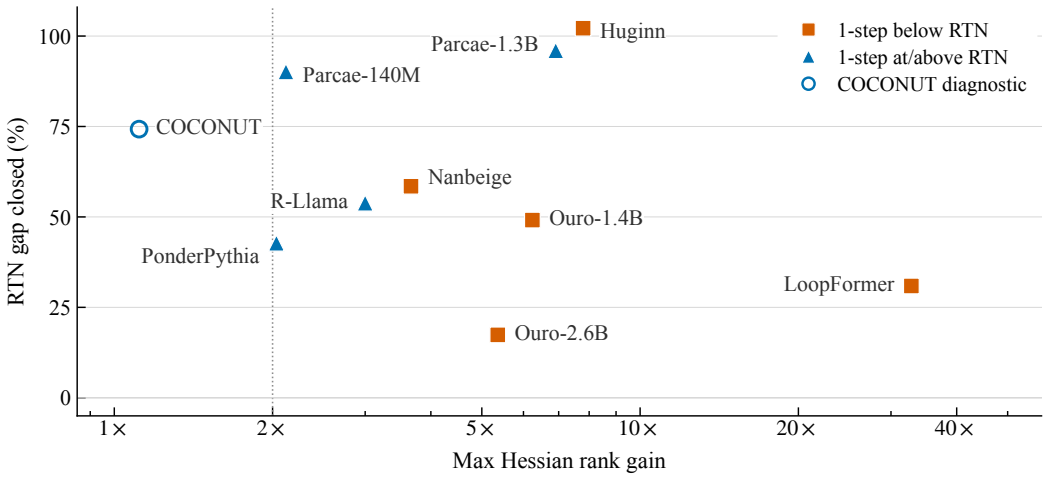

Figure 2: Maximum accumulated-to-step-0 Hessian rank ratio across shared layers versus the RTN-to-full-precision gap closed by trajectory calibration. Marker shape shows whether one-step GPTQ outperforms RTN. The hollow COCONUT point is an in-sample top-1 diagnostic (n=30); all 9 filled points have rank gain $\geq 2\times$.

5 Trajectory Calibration

Instead of calibrating at step 0 alone, we accumulate the Hessian across the full rollout. The step-0 Hessian assigns low relative weight to directions that later recurrence steps activate (Section 4). On the adapter, 99.6% of input directions fall below the reported $10^{-3}\lambda_{\max}$ threshold, and GPTQ's column compensation can shift rounding error toward them. With the state trajectory fixed at the bf16 reference, the reconstruction proxy over the rollout is

$$L_{\text{deploy}} \approx \sum_{t=0}^{N-1} \operatorname{tr}\big(\Delta W\, H_t\, \Delta W^{\top}\big) = \operatorname{tr}\big(\Delta W\, H_{\text{deploy}}\, \Delta W^{\top}\big), \tag{3}$$

where $H_t = X_t^\top X_t$ and $H_{\text{deploy}} = \sum_t H_t$. One-step GPTQ computes column compensation from a damped step-0 Hessian: damping regularizes low-energy directions but cannot supply their later-step activation covariance. Accumulating across the rollout adds that covariance, so the solver penalizes rounding error along directions used during deployment.

Quantized-trajectory control. At grouped INT3 (g=128), a stronger stress test than the main INT4 setting, recollecting Huginn's Hessians on the quantized rollout changes final-step top-1 agreement with bf16 from 84.59% to 84.95% on 20 prompts held out from calibration (prompt-cluster bootstrap 95% CI, $[-2.31, 3.10]$ pp). The grouped-INT4 reconstruction proxy is also nearly constant in magnitude along the bf16 rollout (maximum-to-minimum ratio $1.01\times$ for Huginn and $1.02\times$ for Ouro-1.4B), even as the active directions change (Appendix J).

Consistent with the role of missing directions, COCONUT's thresholded Hessian rank changes little across steps (maximum gain $1.1\times$; Section 4.1), and trajectory accumulation improves its in-sample top-1 agreement by only 0.8 percentage points.

Single late-step calibration. On Huginn WikiText-2, step-31-only calibration reaches 15.07 PPL under the same simulated grouped-INT4 setting, compared with 15.36 for one-step and 14.89 for full accumulation. It captures 62% of the one-step-to-full improvement. Step 31 has higher thresholded rank (361 vs. 335 at $10^{-3}\lambda_{\max}$), so thresholded rank alone does not determine recovery.

Calibration horizon. The gain appears only at the full 32-step horizon in the matched 100-example Huginn sweep. Exact-match accuracy is 25% at N_{cal}=1 and 8, then 30% at N_{cal}=32, compared with 28% for RTN and 32% for bf16 (Figure 3). Because horizon curves differ across architectures, as the Ouro and Parcae sweeps show (Appendix Table 7), the main table uses each model's deployed recurrence depth.

6 Evaluation

Sections 3 and 4 identify feedback exposure and calibration blindness under recurrent weight reuse. We test the first with controlled interventions and the second with rank and recovery results across nine checkpoints from Huginn, R-Llama, Ouro, LoopFormer, Parcae, PonderPythia (Zeng et al., 2026), and Nanbeige (Nanbeige Lab et al., 2026).

Setup. All recovery rows use deterministic simulated grouped-INT4 (g=128). Huginn and R-Llama calibrate on the first 50 GSM8K training questions and report the disjoint full test set (n=1,319, 8-shot, greedy). The other checkpoints use WikiText-2 perplexity. Appendix Tables 11 and 9 give separate bits-per-byte reruns and supporting GSM8K/LAMBADA results. Table 2 gives each GPTQ horizon; full details are in Appendix Q.

6.1 Performance Recovery

If the failure comes from directions underweighted by the step-0 Hessian, accumulating Hessians along the recurrence should recover performance. Table 2 tests this prediction across nine checkpoints.

One-step GPTQ underperforms RTN on five checkpoints; on LoopFormer, the deficit is 10.81 PPL. Trajectory calibration improves every row: Huginn returns to bf16-level accuracy, and the others close 17–96% of the RTN-to-base gap. On Huginn, N-step calibration with rotation also remains within 2 pp of bf16 on ARC-C and HellaSwag (Appendix Table 8).

Architecture and scale. Parcae has no adapter, providing a test of calibration blindness without Huginn's loop-entry adapter. At 140M, trajectory calibration closes 90% of the RTN-to-base gap. At 1.3B, it lowers PPL from 27.35 with one-step GPTQ to 20.84, compared with 51.30 for RTN and 19.57 for f32. One projection falls to 0.03% step-0 rank despite the model's $\sim$16% core-block average (Appendix Table 18). LAMBADA accuracy confirms the ranking (Appendix Table 9).

On Ouro-2.6B, trajectory calibration improves PPL by 1.99 over one-step GPTQ and 0.32 over RTN. RTN is already within 1.84 PPL of base. GSM8K confirms the ordering: 79.8% for 4-step calibration, 78.8% for RTN, and 72.6% for one-step (Appendix Table 9).

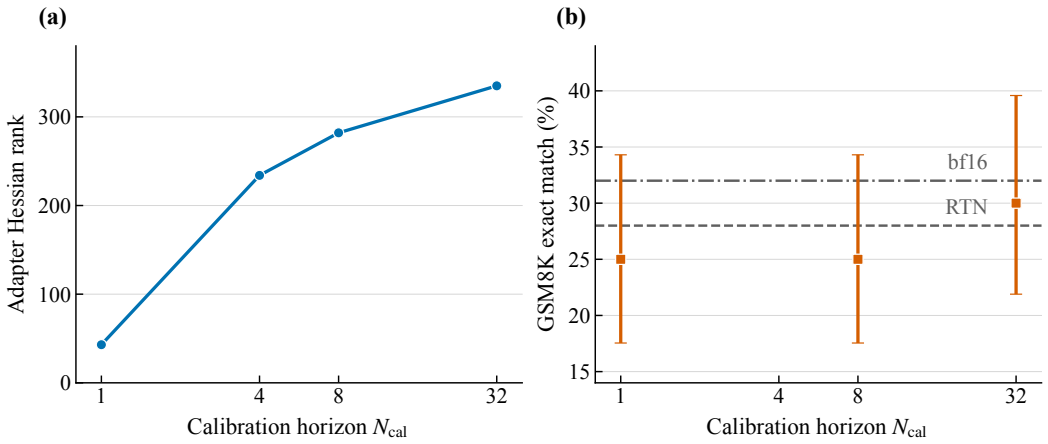

Figure 3: Huginn-3.5B calibration horizon (grouped INT4, g=128). (a) Adapter Hessian rank. (b) Exact match on GSM8K test IDs 0–99, held out from calibration, with Wilson 95% intervals and matched bf16/RTN references. Panels use separate runs.

Table 2: Simulated grouped-INT4 recovery (g=128). N_{cal} is the calibration horizon; gap closed is $(N\text{-step} - RTN)/(Base - RTN)$, computed from unrounded results. Base is bf16 except Parcae (f32). The best quantized result in each row is bold.

Checkpoint	N_{cal}	**Base**	**RTN**	**1-step**	N**-step**	**Gap closed**
Full GSM8K exact-match accuracy ↑						
Huginn 3.5B[a]	32	34.9	27.9	26.9	**35.0**	102%
R-Llama 1.4B	32	50.2	39.5	40.3	**45.3**	54%
Full WikiText-2 perplexity ↓						
Ouro 1.4B	4	11.51	14.38	14.67	**12.97**	49%
Ouro 2.6B[b]	4	10.10	11.94	13.61	**11.62**	17%
LoopFormer 278M	8	31.80	35.10	45.91	**34.08**	31%
Parcae 140M	8	38.46	46.08	41.49	**39.21**	90%
Parcae 1.3B	8	19.57	51.30	27.35	**20.84**	96%
PonderPythia 2.8B	4	11.75	12.61	12.51	**12.24**	43%
Nanbeige4.2-3B-Base[c]	2	10.50	13.13	13.23	**11.59**	58%

[a] No rotation; exact counts are 460/1,319 for bf16, 368/1,319 for RTN, and 462/1,319 for N-step, giving $94/92 = 102\%$. [b] The 17% closure corresponds to 1.99 PPL over one-step GPTQ and 0.32 over RTN. [c] Unrounded Nanbeige PPL is 10.5033/13.1284/11.5932 for bf16/RTN/N-step, giving 58.48%; the model retains its native 131,072-token context.

On PonderPythia-2.8B, four-pass calibration lowers PPL from 12.51 to 12.24 (RTN: 12.61). Nanbeige4.2-3B extends the result to 4.17B total parameters: one-step GPTQ is worse than RTN, while 2-pass calibration closes 58% of the RTN-to-bf16 gap at its deployed depth of two.

Additional bit widths. On WikiText-2, per-channel INT8 leaves Huginn within 0.01 PPL of bf16 (Appendix D). On the full GSM8K test set at grouped INT3, exact match rises from 5.84% with one-step GPTQ to 20.70% with full-trajectory calibration. RTN reaches 8.11%, while equal-row trajectory calibration reaches 17.44% with the same activation-row budget as one-step (Appendix Table 10).

6.2 Real-Kernel Evaluation

Trajectory calibration preserves the inference graph. On a single NVIDIA A100 at batch size 1, median eager latency for a 32-step Huginn forward on a 7-token prompt is 303.9 ms in bf16 and 250.9 ms with the Marlin W4A16 kernel (g=128), a 1.21× speedup (paired-bootstrap 95% CI, 1.20–1.23×). Across inputs of 1, 7, and 16 tokens, the speedup ranges from 1.21× to 1.23×. The 17

shared projections use 3.27 GB for bf16 weights and 0.86 GB in Marlin format, a $3.80\times$ reduction. With the same quantized weights, exact-match accuracy on the full GSM8K test set is 34.0% under Marlin and 35.0% under deterministic simulated INT4 (Appendix D).

7 Related Work

Hyperloop Transformers studies loop-level hyper-connections. In one weight-only INT4 experiment, it accumulates GPTQ Hessians across loops without a step-0 comparison or subspace analysis (Zeitoun et al., 2026). LoopQ attributes failure to loop-dependent distributions, state reuse, and recursive error; it learns loop-dependent scales and transforms, a transition adapter, and logit/hidden-state distillation under W4A4/W4A8 (Fang et al., 2026). Jim et al. (2026) report architecture-dependent INT4 fragility in small recursive reasoners. Our causal analysis separates the mechanisms. Rank and subspace measurements identify directions missed by step-0 Hessians, while controlled changes to prior-error feedback and the identity path isolate error recirculation and residual protection.

Calibration mismatch in iterative systems. Q-Diffusion samples denoising timesteps, while AdaTSQ uses timestep-dependent activation policies and temporally weighted calibration (Li et al., 2023; Zhang et al., 2026). In a looped transformer, the same operator generates and processes later states, so a step-0 Hessian can miss directions used during deployment (Section 4).

Quantized recurrence. Quamba's contractive-LTI bound assumes an unperturbed A (Chiang et al., 2025). Our IIR and Mamba controls perturb the transition itself; in the IIR control, quantizing B, C, and D leaves the pole spectrum fixed (Section 3). QMamba ablates SSM components, while Bi-Mamba keeps SSM parameters at full precision and binarizes input/output projections (Li et al., 2025; Tang et al., 2025). Monotone DEQ theory bounds operator perturbations under a certified monotonicity margin (Li et al., 2026). Precision highways address quantization during training, at the cost of retraining (Park et al., 2018).

Residual connections and error propagation. Residual-network stability and path-ensemble analyses do not cover repeated re-entry of the same perturbation (Haber & Ruthotto, 2018; Veit et al., 2016). Our Jacobian analysis extends the stability view to discrete recurrence. The identity-bypass control isolates residual protection under fixed evaluation states and adapter weights (Section 3.2).

PTQ methods. GPTQ, AWQ, SpinQuant, QuIP#, and SmoothQuant (Frantar et al., 2022; Lin et al., 2024; Liu et al., 2025; Tseng et al., 2024; Xiao et al., 2023) typically use single-pass rather than recurrence-conditioned calibration. With step-0 calibration, GPTQ and LDLQ (Chee et al., 2023) are equivalent adaptive-rounding algorithms that minimize $\mathrm{tr}(\Delta W\, H\, \Delta W^\top)$, so both inherit H's rank deficiency. QuIP#'s Hadamard rotation is orthogonal and rank-preserving, so it does not restore missing directions in H (Appendix Table 12). AWQ selects per-channel scales from activation magnitudes, which are more stable across recurrence steps (Section 4, Appendix Table 13).

8 Conclusion

Conventional calibration-based PTQ uses activations from a single forward pass. In looped transformers, weight reuse turns quantization into a feedback problem: rounding error becomes dangerous when it enters the recurrent state transition without an identity path. IIR and Mamba controls likewise show greater damage when quantization changes the recurrent transition.

Calibration faces a parallel problem. A step-0 Hessian can miss the directions used later in the rollout, and one-step GPTQ underperforms RTN on five of nine checkpoints. Accumulating Hessians along the recurrence trajectory recovers performance in every tested case.

Limitations. Our evaluation covers nine checkpoints up to 4.17B parameters. Trajectory calibration does not always recover full-precision performance, and rank gain alone does not predict recovery magnitude.

We expect the same failure modes in larger looped models that retain unprotected feedback paths or step-dependent activation subspaces. For looped models, a quantizer that never sees the loop should not be trusted to quantize it.

AI USE STATEMENT

Generative AI tools assisted with hypothesis refinement, experimental design, implementation and debugging, result analysis, literature search, and manuscript editing. The authors reviewed the relevant literature, inspected and tested AI-assisted code, verified reported results against experiment artifacts, and take responsibility for all content in this work.

REFERENCES

Jerry Chee, Yaohui Cai, Volodymyr Kuleshov, and Christopher De Sa. QuIP: 2-bit quantization of large language models with guarantees. In *Advances in Neural Information Processing Systems*, volume 36, pp. 4396–4429. Curran Associates, Inc., 2023. doi: 10.52202/075280-0196. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/0df38cd13520747e1e64e5b123a78ef8-Abstract-Conference.html.

Hung-Yueh Chiang, Chi-Chih Chang, Natalia Frumkin, Kai-Chiang Wu, and Diana Marculescu. Quamba: A post-training quantization recipe for selective state space models. In *International Conference on Learning Representations*, 2025. URL https://iclr.cc/virtual/2025/poster/28449.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021.

EleutherAI. lm-evaluation-harness. GitHub repository, 2026. URL https://github.com/EleutherAI/lm-evaluation-harness/tree/889c656bddc5b1d5b53381d68a1ac0b5bedfec78. Commit 889c656bddc5b1d5b53381d68a1ac0b5bedfec78; accessed 2026-09-05.

Rui Fang, Hsi-Wen Chen, and Ming-Syan Chen. LoopQ: Quantization for recursive transformers, 2026. URL https://arxiv.org/abs/2605.16343.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. GPTQ: Accurate post-training quantization for generative pre-trained transformers, 2022. URL https://arxiv.org/abs/2210.17323.

Elias Frantar, Roberto L. Castro, Jiale Chen, Torsten Hoefler, and Dan Alistarh. MARLIN: Mixed-precision auto-regressive parallel inference on large language models. In *Proceedings of the 30th ACM SIGPLAN Annual Symposium on Principles and Practice of Parallel Programming*, pp. 239–251, 2025. doi: 10.1145/3710848.3710871.

Jonas Geiping, Sean McLeish, Neel Jain, John Kirchenbauer, Siddharth Singh, Brian R. Bartoldson, Bhavya Kailkhura, Abhinav Bhatele, and Tom Goldstein. Scaling up test-time compute with latent reasoning: A recurrent depth approach. In *Advances in Neural Information Processing Systems 38*, pp. 46242–46293, 2025. doi: 10.52202/085713-1380.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces, 2023. URL https://arxiv.org/abs/2312.00752.

Eldad Haber and Lars Ruthotto. Stable architectures for deep neural networks. *Inverse Problems*, 34(1):014004, 2018. doi: 10.1088/1361-6420/aa9a90. URL https://doi.org/10.1088/1361-6420/aa9a90.

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. Training large language models to reason in a continuous latent space. *arXiv preprint arXiv:2412.06769*, 2024.

Herbert Jaeger and Harald Haas. Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *Science*, 304(5667):78–80, 2004.

Ahmadreza Jeddi, Marco Ciccone, and Babak Taati. LoopFormer: Elastic-depth looped transformers for latent reasoning via shortcut modulation. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=RzYXb5YWBs.

Pearse Jim, Steven Kolawole, Opegbemi Matthias Busoye, Glory Bagai, and Virginia Smith. What survives when you compress a recursive reasoner for the edge?, 2026. URL https://arxiv.org/abs/2606.26488.

James Li, Philip H. W. Leong, and Thomas Chaffey. Quantization robustness of monotone operator equilibrium networks. *arXiv preprint arXiv:2603.10562*, 2026.

Xiuyu Li, Yijiang Liu, Long Lian, Huanrui Yang, Zhen Dong, Daniel Kang, Shanghang Zhang, and Kurt Keutzer. Q-Diffusion: Quantizing diffusion models. In *ICCV*, 2023.

Yinglong Li, Xiaoyu Liu, Jiacheng Li, Ruikang Xu, Yinda Chen, and Zhiwei Xiong. QMamba: Post-training quantization for vision state space models. *arXiv preprint arXiv:2501.13624*, 2025.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. AWQ: Activation-aware weight quantization for on-device LLM compression and acceleration. In *Proceedings of Machine Learning and Systems*, volume 6, pp. 87–100, 2024. URL https://proceedings.mlsys.org/paper_files/paper/2024/hash/42a452cbafa9dd64e9ba4aa95cc1ef21-Abstract-Conference.html.

Zechun Liu, Changsheng Zhao, Igor Fedorov, Bilge Soran, Dhruv Choudhary, Raghuraman Krishnamoorthi, Vikas Chandra, Yuandong Tian, and Tijmen Blankevoort. SpinQuant: LLM quantization with learned rotations. In *International Conference on Learning Representations*, 2025.

Sean McLeish, Ang Li, John Kirchenbauer, Dayal Singh Kalra, Brian R. Bartoldson, Bhavya Kailkhura, Avi Schwarzschild, Jonas Geiping, Tom Goldstein, and Micah Goldblum. Teaching pretrained language models to think deeper with retrofitted recurrence, 2025. URL https://arxiv.org/abs/2511.07384.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *International Conference on Learning Representations*, 2017.

Nanbeige Lab, Chen Yang, Chengrui Huang, Fufeng Lan, Hanhui Chen, Hao Zhou, Huatong Song, Jiaqi Cao, Jiaying Zhu, Jinlin Niu, Kai Wang, Lisheng Huang, Qiliang Liang, Ran Le, Ruixiang Feng, Shuang Sun, Tao Gu, Tao Zhang, Tianyu Luo, Yang Song, Yun Xing, Yuntao Wen, Ziyao Xu, Zongchao Chen, and Zongqiang Li. Nanbeige4.2-3B: Unlocking agentic capabilities in a compact model, 2026. URL https://arxiv.org/abs/2607.22083.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Ngoc Quan Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. The LAMBADA dataset: Word prediction requiring a broad discourse context. In Katrin Erk and Noah A. Smith (eds.), *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1525–1534, Berlin, Germany, 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1144. URL https://aclanthology.org/P16-1144/.

Eunhyeok Park, Dongyoung Kim, Sungjoo Yoo, and Peter Vajda. Precision highway for ultra low-precision quantization. *arXiv preprint arXiv:1812.09818*, 2018.

Hayden Prairie, Zachary Novack, Taylor Berg-Kirkpatrick, and Daniel Y. Fu. Parcae: Scaling laws for stable looped language models. *arXiv preprint arXiv:2604.12946*, 2026.

Eduardo D. Sontag. Input to state stability: Basic concepts and results. In Paolo Nistri and Gianna Stefani (eds.), *Nonlinear and Optimal Control Theory: Lectures given at the C.I.M.E. Summer School held in Cetraro, Italy June 19–29, 2004*, volume 1932 of *Lecture Notes in Mathematics*, pp. 163–220. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008. ISBN 978-3-540-77653-6. doi: 10.1007/978-3-540-77653-6_3. URL https://doi.org/10.1007/978-3-540-77653-6_3.

Shengkun Tang, Liqun Ma, Haonan Li, Mingjie Sun, and Zhiqiang Shen. Bi-Mamba: Towards accurate 1-bit state space model. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=CKQ4AgoRQm.

Albert Tseng, Jerry Chee, Qingyao Sun, Volodymyr Kuleshov, and Christopher De Sa. QuIP#: Even better LLM quantization with hadamard incoherence and lattice codebooks. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 48630–48656. PMLR, 2024. URL https://proceedings.mlr.press/v235/tseng24a.html.

Andreas Veit, Michael J. Wilber, and Serge Belongie. Residual networks behave like ensembles of relatively shallow networks. In *Advances in Neural Information Processing Systems*, volume 29, pp. 550–558. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/hash/37bc2f75bf1bcfe8450a1a41c200364c-Abstract.html.

Guangxuan Xiao, Ji Lin, Mickael Seznec, Hao Wu, Julien Demouth, and Song Han. SmoothQuant: Accurate and efficient post-training quantization for large language models. In *ICML*, 2023.

Abbas Zeitoun, Lucas Torroba-Hennigen, and Yoon Kim. Hyperloop transformers, 2026. URL https://arxiv.org/abs/2604.21254.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019.

Boyi Zeng, Shixiang Song, Siyuan Huang, Yixuan Wang, He Li, Ziwei He, Xinbing Wang, Zhiyu Li, and Zhouhan Lin. PonderLM: Pretraining language models to ponder in continuous space. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=UrM4MNRYZm.

Shaoqiu Zhang, Zizhong Ding, Kaicheng Yang, Junyi Wu, Xianglong Yan, Xi Li, Bingnan Duan, Jianping Fang, and Yulun Zhang. AdaTSQ: Pushing the pareto frontier of diffusion transformers via temporal-sensitivity quantization. *arXiv preprint arXiv:2602.09883*, 2026.

Rui-Jie Zhu, Zixuan Wang, Kai Hua, Tianyu Zhang, Ziniu Li, Haoran Que, Boyi Wei, Zixin Wen, Fan Yin, He Xing, Lu Li, Jiajun Shi, Kaijing Ma, Shanda Li, Taylor Kergan, Andrew Smith, Xingwei Qu, Mude Hui, Bohong Wu, Qiyang Min, Hongzhi Huang, Xun Zhou, Wei Ye, Jiaheng Liu, Jian Yang, Yunfeng Shi, Chenghua Lin, Enduo Zhao, Tianle Cai, Ge Zhang, Wenhao Huang, Yoshua Bengio, and Jason Eshraghian. Scaling latent reasoning via looped language models. *arXiv preprint arXiv:2510.25741*, 2025.

A Architecture Diagram

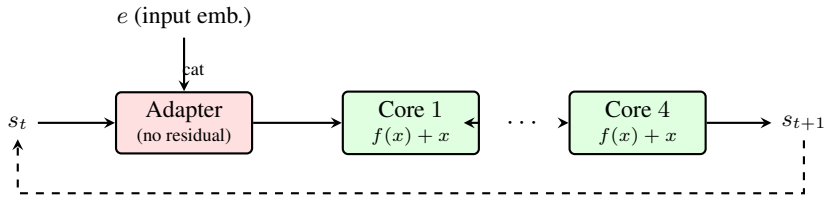

Figure 4: Huginn recurrence loop. The adapter projects $\mathrm{cat}(s_t, e)$ back to the model width with no residual path. Adapter perturbations therefore enter the recurrence without an identity bypass, while core residual paths provide an identity channel that can partially buffer small perturbations.

B Trajectory Calibration Algorithm

Algorithm 1 Trajectory calibration for looped transformers.

Require: bf16 model, prompts $\mathcal{D}$, recurrence depth N

for each shared layer ℓ **do**

  $H_\ell \leftarrow \sum_{t=0}^{N-1} (X_t^\ell)^\top X_t^\ell$

  $\hat{W}_\ell \leftarrow \mathrm{GPTQ}(W_\ell, H_\ell)$ ▷ unchanged column solver

end for

C Supporting Tables

Tables 3–12 collect supplementary numeric results referenced in Sections 3 and 6.

Table 3: Controlled injection for Huginn-3.5B (50 prompts; data from Figure 1). Arms 1–3 use deterministic per-channel INT4 FakeQuant on the listed layers, with all other layers bf16. Arm 4 adds one fixed Gaussian weight-noise draw with the same Frobenius norm as the adapter's per-channel INT4 weight error. Two bf16 runs with independently sampled initial states reach 98.9% agreement.

Arm	Quantized layers	Top-1 (N=32)	Top-1 endpoints (N=1 → 32)
0	None (bf16 control)	98.9%	68.8% → 98.9%
1	Adapter only	**10.5%**	24.8% → 10.5%
2	Core only (16 projections)	**87.9%**	68.4% → 87.9%
3	All (adapter + core)	10.6%	25.0% → 10.6%
4	Gaussian noise (norm-matched)	25.0%	49.3% → 25.0%

Table 4: Controlled injection for R-Llama-1.4B (50 prompts, per-channel INT4 FakeQuant). The adapter/core split replicates on a second architecture: adapter-only INT4 drops agreement to 49.1%, while core-only preserves 74.8%. The gap is narrower than Huginn (25.7 pp vs. 77.4 pp). Two bf16 runs with independently sampled initial states reach 98.7% agreement.

Arm	Quantized layers	Top-1 (N=32)
0	None (bf16 control)	98.7%
1	Adapter only	**49.1%**
2	Core blocks only (6 layers)	**74.8%**
3	All (adapter + core)	34.6%

Table 5: Huginn-3.5B per-channel INT4 diagnostic. Top-1 agreement uses 20 prompts disjoint from calibration. GSM8K uses the first 100 test examples. For trajectory GPTQ, those 100 GSM8K inputs are also used without labels for Hessian calibration, so the corresponding task score is diagnostic.

Configuration	**Top-1 (N=32)**	**GSM8K strict/flex**
bf16	$\sim$100%	34% / 41%
Per-channel INT4, no rotation	8.5%	0% / 0%
Per-channel RTN + rotation	82.6%	13% / 15%
Per-channel N-step + rotation	**92.7%**	**24% / 25%**

Table 6: Huginn-3.5B calibration-horizon diagnostics (deterministic grouped-INT4 FakeQuant, g=128, no rotation). The upper block reports Hessian rank and top-1 agreement. The lower block is a separate held-out GSM8K evaluation on test examples 0–99, held fixed across rows. For comparison, the separate step-31-only Hessian has rank 361 (3.42%), while the 32-step accumulated Hessian has rank 335. Each is evaluated against its own relative threshold (Section 5).

Rank and top-1 diagnostic

N_{cal}	**Hessian rank**	**Rank (% of dim.)**	**Top-1 (32-step)**
1	43	0.41%	88.2%
4	234	2.22%	90.9%
8	282	2.67%	**92.5%**
32	**335**	**3.17%**	**92.5%**
RTN baseline			88.0%

Clean held-out GSM8K, same 100 test examples

Configuration	**Strict**	**Flexible**
bf16	32/100	38/100
RTN	28/100	32/100
N_{cal}=1	25/100	28/100
N_{cal}=8	25/100	**33/100**
N_{cal}=32	**30/100**	31/100

Table 7: Calibration horizons across architectures (grouped-INT4 FakeQuant, g=128). Each row reports one calibration horizon. Ouro-1.4B obtains 88% of the N=1-to-N=4 PPL improvement by N_{cal}=2. Parcae-1.3B improves sharply by N_{cal}=4; at N_{cal}=2, one layer has only 3 eigenvalues above the rank threshold and PPL is worse than RTN.

Model	N_{cal}	**WikiText-2 PPL**	**LAMBADA acc**
Ouro 1.4B	1	14.67	—
	2	13.17	—
	3	13.05	—
	4	**12.97**	—
Parcae 1.3B	1	27.35	0.341
	2	99.59[†]	0.326
	4	20.98	**0.430**
	6	**20.62**	—
	8	20.84	0.416

[†] At N_{cal}=2, block-0 `mlp.proj` has only 3/6,144 eigenvalues above $10^{-3}\lambda_{\max}$, and GPTQ gives worse PPL than RTN (99.59 versus 51.30). At N_{cal}=4, PPL recovers to 20.98. PPL is not monotone thereafter, increasing slightly from 20.62 at N_{cal}=6 to 20.84 at N_{cal}=8.

Table 8: Auxiliary Huginn/R-Llama benchmarks for g=128 with trajectory calibration plus adapter rotation. ARC-C and HellaSwag are accuracy (higher is better); WikiText-2 is perplexity (lower is better). The main-table rows (Table 2) isolate trajectory calibration without rotation. For comparison, Section 5 reports 14.89 PPL for the non-rotated Huginn trajectory-calibration run.

Model	Config	ARC-C (%)	HellaSwag (%)	WikiText-2 PPL
Huginn 3.5B	bf16	37.4%	66.6%	14.02
	N-step + rotation	37.0%	65.3%	14.54
R-Llama 1.4B	bf16	37.1%	46.1%	30.49
	N-step + rotation	36.3%	45.2%	32.64

Table 9: Supporting metrics for Ouro-2.6B and Parcae-1.3B. Table 2 uses WikiText-2 perplexity as the primary metric for these checkpoints, and the GSM8K and LAMBADA results preserve the same method ordering.

Checkpoint	Metric	RTN	1-step	N-step
Ouro-2.6B	GSM8K strict/flex (n=1,319)	78.8 / 81.5	72.6 / 74.6	**79.8 / 82.8**
Parcae-1.3B	LAMBADA accuracy	0.3117	0.3410	**0.4157**

Parcae-1.3B f32 LAMBADA accuracy is 0.4584.

Table 10: Full-test Huginn GSM8K at grouped INT3 (g=128; n=1,319). Equal-row calibration uses the one-step activation-row budget distributed across all 32 recurrence steps.

Configuration	Strict (%)	Flexible (%)
bf16	34.87 (460/1,319)	42.23 (557/1,319)
RTN	8.11 (107/1,319)	8.64 (114/1,319)
1-step GPTQ	5.84 (77/1,319)	6.44 (85/1,319)
Equal-row trajectory GPTQ	17.44 (230/1,319)	17.82 (235/1,319)
Full-trajectory GPTQ	**20.70** (273/1,319)	**23.28** (307/1,319)

Full trajectory improves over one-step by 14.9 pp strict and 16.8 pp flexible (paired-bootstrap 95% CIs [12.66, 17.06] and [14.56, 19.18]; exact McNemar $p = 9.35 \times 10^{-39}$ and 2.72×10^{-45}, respectively). Against equal-row calibration, the gains are 3.3 pp strict and 5.5 pp flexible; the strict paired-bootstrap 95% CI is [0.83, 5.69], with exact McNemar $p = 0.00975$.

Table 11: Direct bits per byte on WikiText-2 for the seven perplexity checkpoints in Table 2; lower is better. Base is bf16 except for Parcae (f32). The best quantized result in each row is bold. All Nanbeige conditions retain the checkpoint's native 131,072-token context configuration.

Checkpoint	Base	RTN	1-step	N-step
Ouro 1.4B	0.6591	0.7193	0.7238	**0.6912**
Ouro 2.6B	0.6240	0.6689	0.7014	**0.6582**
LoopFormer 278M	0.9334	0.9600	1.0327	**0.9516**
Parcae 140M	0.9847	1.0326	1.0053	**0.9899**
Parcae 1.3B	0.8032	1.0514	0.8895	**0.8311**
PonderPythia 2.8B	0.6646	0.6837	0.6817	**0.6757**
Nanbeige4.2-3B-Base	0.6345	0.6947	0.6968	**0.6611**

Table 12: Component ablation on Huginn-3.5B (g=128, n=100). Scores are diagnostic because GPTQ uses these unlabeled inputs for Hessian calibration. Bold marks the feedback-protection and best-quantized results.

Configuration	Strict (%)	Flex (%)
RTN (no rotation, no GPTQ)	25%	28%
1-step GPTQ (rotated variant)	25%	28%
Adapter-bf16 + RTN (no rotation)	**32%**	**35%**
Adapter N-step + rotation (core 1-step)	23%	29%
N-step all layers (no rotation)	30%	37%
N-step all layers (rotated variant)	**34%**	**39%**
bf16	34%	41%

Rotated rows bundle adapter rotation with the calibration variant used in that ablation. "Adapter N-step + rotation" applies trajectory calibration and rotation to the adapter while the core uses one-step calibration.

Table 13: Calibration-step sensitivity on Huginn-3.5B WikiText-2 (g=128, deterministic FakeQuant). Bold marks the best result within each method family.

Configuration	WikiText-2 PPL
RTN g=128	14.98
GPTQ	
Step 0 (1-step)	15.36
Step 31	15.07
32-step	**14.89**
AWQ	
Step 0	14.66
Step 31	**14.65**
All steps	**14.65**

AWQ uses per-channel activation magnitudes, while GPTQ uses $H = X^\top X$. GPTQ and RTN are symmetric. AWQ is asymmetric (asymmetric RTN: 14.72 PPL), so absolute values are not directly comparable across method families.

D Rotation and Marlin Implementation Checks

All rotation comparisons use adapter-only rotation (`--no-core-rotation`). The simulated evaluations are deterministic, while the Marlin deployment diagnostic below uses the real kernel. Table 14 reports the per-channel diagnostic where rotation matters most.

Table 14: Hadamard vs. random orthogonal at per-channel INT4 with N-step GPTQ (both at block size 512). Top-1 agreement uses 20 prompts disjoint from calibration. The GSM8K task evaluations include the unlabeled inputs used for Hessian calibration and are reported as diagnostics.

Config	Top-1 (N=32)	GSM8K (n=100) strict/flex	GSM8K (n=1,319) strict/flex
Random orthogonal	**91.0%**	23% / 24%	26.9% / 27.5%
Hadamard	90.7%	**29% / 33%**	**29.4% / 34.0%**
bf16	∼100%	34% / 41%	34.9% / 42.2%

At INT8 per-channel RTN, Hadamard (block size 512) and random orthogonal both match bf16-level perplexity (14.02), while Hadamard with block size 32 gives 14.03. The QR fallback for partial Hadamard blocks removes the remaining INT8 perplexity difference; all tested INT8 rotation variants are within 0.01 PPL of bf16.

At grouped deployment precision, Marlin and deterministic FakeQuant use identical quantized weights and scales across all 17 shared projections. The no-rotation runs calibrate on 50 GSM8K

training questions and evaluate all 1,319 test questions. Marlin reaches 34.0%/34.3% strict/flexible accuracy, versus 35.0%/35.3% for FakeQuant; the paired differences are $-1.0/-1.1$ percentage points (exact McNemar $p = 0.28/0.24$).

The latency benchmark uses 10 warmups per arm and 50 paired bf16/Marlin trials in alternating order at each input length.

We also compare every shared Marlin projection against its dequantized grouped-INT4 reference at 1, 7, and 16 input tokens. Across these 51 comparisons, the maximum relative ℓ_2 error is 1.35×10^{-3}. A 32-step full forward at each input length executes the expected 544 Marlin calls.

E Extended Controlled Injection Analysis

Varying feedback strength. For each bf16 state s_t^{bf16} and current intervention state s_t^{λ}, we feed the quantized recurrence $s_t^{\mathrm{bf16}} + \lambda(s_t^{\lambda} - s_t^{\mathrm{bf16}})$. Thus $\lambda = 0$ removes prior state error before each update, while $\lambda = 1$ reproduces the natural quantized rollout. We use GSM8K training prompts 70–119 and 3 recurrent-state initializations shared across $\lambda \in \{0, 0.25, 0.5, 0.75, 1\}$.

We measure final relative state error as $\|s_{32}^{\lambda} - s_{32}^{\mathrm{bf16}}\|_2 / \|s_{32}^{\mathrm{bf16}}\|_2$. With the full adapter quantized, the geometric mean of its prompt-level $\lambda = 1$-to-$\lambda = 0$ ratio is $1.989\times$ under per-channel INT4 (prompt-bootstrap 95% CI, $[1.980, 1.998]$) and $2.085\times$ at g=128 ($[1.971, 2.229]$). Under per-channel INT4, cutting feedback raises final-step top-1 agreement with bf16 from 10.4% to 69.2%. Mean final state error increases monotonically with λ in both settings, as does every prompt-level curve.

For a prespecified equal-width split, we quantize the full adapter once, retain its quantized values in either the 5,280 recurrent-state columns or 5,280 embedding columns, and leave the other half in bf16. Both arms show feedback amplification, but the embedding-half ratio is larger: $2.010\times$ versus $1.358\times$ under per-channel INT4, and $2.264\times$ versus $2.065\times$ at g=128. Both errors enter the recurrence through the same adapter output, so this split does not compare feedback-exposed with protected paths.

Adapter contribution under full INT4. Arm 1 (adapter INT4, core bf16) and Arm 3 (both INT4) produce nearly identical top-1 agreement (10.5% vs. 10.6%). Core INT4 noise adds little on top of adapter error. The adapter term remains dominant whether or not the core blocks are quantized, consistent with the controlled injection result in the main text (Table 3).

F Perturbation-Type Generalization

The feedback-exposure mechanism is not specific to quantization noise. Table 15 reports separate weight-perturbation controls at N=32. The rank-4 rows use a common target norm ($\|\Delta W\|_F = 8.59$) at both locations to compare adapter and core perturbations at fixed size; their core arms split this norm between block-0 `mlp.fc` and `mlp.proj`. The trained directions use 50 next-token cross-entropy updates on the GSM8K training split before rescaling. For the Gaussian row, 5 random-input measurements of grouped-INT4 output error give target norms of 3.16 for the adapter and 8.11 for block-0 `mlp.fc`. Each perturbed forward is paired with its bf16 reference using the same initial-state seed.

Table 15: Perturbation-type generalization under the perturbation scales defined above. Top-1 agreement with bf16 at N=32. The trained rank-4 update is evaluated on 60 prompts; the random rank-4 update on 60 prompts for each of 3 seeds; Gaussian noise on 10 prompts for each of 3 seeds.

Perturbation	Adapter	Core	Gap (Core − Adapter)
Trained rank-4 update	13.4%	77.8%	**64.4 pp**
Random rank-4 update	71.8%	95.8%	**24.0 pp**
Gaussian noise	89.3%	96.5%	**7.2 pp**

Each Gaussian seed samples one fixed weight perturbation and reuses it for all 10 prompts. The bf16 and perturbed forwards use the same initial state.

The random rank-4 update preserves a 24.0 pp adapter/core gap (71.8% vs. 95.8%), and the trained update widens the gap to 64.4 pp (13.4% vs. 77.8%). The Gaussian control gives the same ordering even though its adapter perturbation has the smaller norm (3.16 vs. 8.11). Core-block perturbations are less damaging in all three settings, consistent with buffering from the residual path.

G Sensitivity Metric Comparison

Table 16 compares three per-projection sensitivity measures. *Ablation sensitivity* quantizes a single projection to grouped INT4 (g=128) while all other projections remain at bf16, then reports the fraction of per-token top-1 predictions that disagree with the all-bf16 baseline, averaged across all 32 recurrence steps. Higher values indicate that the projection's quantization error changes more predictions. Random-direction gain is the arithmetic mean of $\|f(x+\delta) - f(x)\|_F / \|\delta\|_F$ over 10 prompts and 10 Gaussian perturbations per prompt, using the last captured layer input from a 16-step rollout and $\delta = 10^{-3}\|x\|_F z$ for $z \sim \mathcal{N}(0, I)$. The third measure is $\mathrm{tr}(H)$, where $H = X^\top X$ is the activation Hessian used by GPTQ.

Table 16: Top five projections by ablation sensitivity from rankings across 17 shared projections under single-projection grouped INT4 ablation (g=128; 10 prompts, averaged across 32 steps). Ablation sensitivity is the fraction of per-token top-1 predictions that disagree with bf16 (0 = no effect, 1 = all predictions change). The adapter is a $2.3\times$ outlier by ablation but ranks last by random-direction gain and third by Hessian trace.

Projection	Ablation sens.	Random-direction rank	Hessian-trace rank
transformer.adapter	**0.118** ($2.3\times$ next)	**17/17 (last)**	**3/17**
core_block.3.mlp.fc	0.051	4/17	4/17
core_block.3.mlp.proj	0.044	14/17	1/17
core_block.1.mlp.proj	0.041	15/17	2/17
core_block.2.mlp.fc	0.040	5/17	5/17

H Group-Size Dependence

Table 17: Group-size sweep (simulated INT4, full model, 30 prompts). Grouped settings retain high final-step agreement, while per-channel quantization collapses to 7.6%. Bold marks the collapsed per-channel endpoint.

Group size	N=1	N=8	N=32
g=32	68.3%	90.6%	88.4%
g=64	68.8%	90.5%	87.5%
g=128	68.7%	87.7%	86.5%
g=256	69.4%	86.6%	83.9%
Per-channel	18.8%	7.8%	**7.6%**

All grouped configurations maintain 84–88% top-1 at N=32, with marginal improvement below g=128. Per-channel quantization drops to 7.6%, a $10\times$ gap. The per-channel collapse is consistent with the adapter's $\rho > 1$ result in Section 3.2.

Perturbation scale. Input-to-state stability relates bounded disturbances to bounded state responses (Sontag, 2008). For a locally contractive map with contraction modulus $L < 1$, a uniform per-step perturbation δ_q has a geometric bound proportional to $\delta_q/(1-L)$. The bound explains why a modest increase in δ_q can move a near-boundary system from bounded error to basin escape.

I Cross-Architecture Rank Survey

Table 18: Cross-architecture Hessian-rank survey at $10^{-3}\lambda_{\max}$. The bottleneck has the lowest step-0 rank ratio among GPTQ-quantized shared layers. Max rank gain is the largest accumulated-to-step-0 ratio across shared layers. Bold marks checkpoints where one-step GPTQ is worse than RTN.

Architecture	Bottleneck layer	Step-0 rank	Max rank gain	1-step vs. RTN	Gap closed
Ouro 1.4B	layer-0 down_proj	0.02%	6.2×[†]	**worse**	49%
Ouro 2.6B	layer-0 down_proj	0.02%	5.4×[†]	**worse**	17%
Nanbeige4.2-3B-Base	layer-1 down_proj	0.02%	3.67×	**worse**	58%
R-Llama 1.4B	adapter	0.02%	3.0×	better	54%
Parcae 1.3B	core.7 mlp.proj	0.03%	6.9×	better	96%
Huginn 3.5B	core.3 mlp.proj	0.36%	7.8×	**worse**	102%
Parcae 140M	core.1 mlp.proj	0.88%	2.1×	better	90%
COCONUT 124M	h.2 mlp.c_proj	1.40%	1.1×	better	74%[§]
LoopFormer 278M	c_fc (block 0)[‡]	1.51%	32.8×	**worse**	31%
PonderPythia 2.8B	layer-0 dense_4h_to_h	2.76%	2.03×	better	43%

For Huginn, `mlp.proj`@3 has the lowest step-0 rank ratio (0.36%; gain $1.4\times$), while the adapter has the maximum rank gain ($7.8\times$). Gap closed is $(N\text{-step} - RTN)/(Base - RTN)$ and is computed from unrounded scores.

[†] Max rank gain is from a non-bottleneck layer. For both Ouro models, the step-0 Hessian contributes at least 99.5% of the 4-step trace sum, and adding steps 1–3 does not raise the bottleneck's thresholded rank above 1 at $10^{-3}\lambda_{\max}$. Per-step Hessians at steps 1–3 individually have rank 974–1,194 of 5,632 (1.4B) and 1,010–1,335 of 5,632 (2.6B).

[‡] The $32.8\times$ maximum belongs to `attn.c_proj` in block 0 (rank 50 → 1,640); the GPTQ-relevant c_fc bottleneck gains $4.1\times$. adaLN modulation layers have rank 1 at step 0 (gain $6.0\times$) and use RTN fallback.

[§] COCONUT uses in-sample top-1 agreement on 30 prompts (fp32 FakeQuant, 4 latent tokens). One-step GPTQ already closes 70% of this gap; the trajectory contribution is $+0.8$ pp top-1 ($+4$ pp gap-closed).

Parcae-1.3B's bottleneck (core.7 mlp.proj, 0.03%) drives the per-layer calibration failure even though one-step GPTQ beats RTN at model level; the remaining layers are well-conditioned enough to compensate. Ouro-2.6B's bottleneck has step-0 rank 1/5,632 while the other 47 layers are near full rank; a model average would not flag it. Mamba is not included because the SSM experiments measure component divergence rather than a step-0 Hessian-rank sweep.

Table 19: Step-1 Hessian ranks for the 8 checkpoints with retained per-step traces. For $H_t = X_t^\top X_t$, each count is the number of eigenvalues above $10^{-3}\lambda_{\max}(H_t)$; denominators are input dimensions. Rows use the rank-survey bottleneck. Huginn instead uses its adapter, where its maximum rank gain occurs.

Checkpoint	Screened layer	Step 0	Step 1
1-step GPTQ worse than RTN			
Ouro 1.4B	layer-0 `down_proj`	1/5,632	974/5,632
Ouro 2.6B	layer-0 `down_proj`	1/5,632	1,010/5,632
Huginn 3.5B	adapter	43/10,560	259/10,560
LoopFormer 278M	`c_fc` block 0	31/2,048	154/2,048
1-step GPTQ better than RTN			
COCONUT 124M	h.2 `mlp.c_proj`	43/3,072	45/3,072
R-Llama 1.4B	adapter	1/4,096	2/4,096
Parcae 1.3B	core.7 `mlp.proj`	2/6,144	4/6,144
Parcae 140M	core.1 `mlp.proj`	27/3,072	21/3,072

A separate Huginn control holds the same 2,919 prompt-token positions across steps and gives ranks 361–363 at steps 7, 15, and 31, each slightly above the 32-step accumulated rank of 335.

J FROZEN-TRAJECTORY PROXY STABILITY

The deployment-weighted reconstruction proxy in Eq. 3 fixes the state trajectory at the bf16 reference. We evaluate fixed per-channel and g=128 RTN perturbations with the per-step proxy $\mathrm{tr}(\Delta W\, H_t\, \Delta W^\top)$ on bf16 activations, summed over all 17 shared projections (50 GSM8K prompts, Huginn-3.5B). This directly tests whether the proxy's magnitude is stable across the reference rollout.

At g=128, the per-step proxy varies by only $1.01\times$ across all 32 steps (709.5 at step 0 to 702.5 at step 31), showing that its magnitude is nearly constant along the bf16 rollout at deployment precision. At per-channel, the proxy varies $1.12\times$, with a $1.12\times$ increase from step 0 to step 1, then gradually stabilizes. Since the matched adapter-only per-channel diagnostic has $\rho > 1$ on all 30 prompts, we use the proxy there as a diagnostic rather than an exact loss estimate.

Step 0 accounts for only 3.0% (per-channel) and 3.2% (g=128) of the cumulative deployment proxy $\sum_t \mathrm{tr}(\Delta W\, H_t\, \Delta W^\top)$, close to the uniform expectation of 3.125%. The proxy magnitude is roughly equal across steps; the calibration gap is directional, not scalar. Only 43 of 10,560 adapter input directions exceed the step-0 threshold (0.41%; Table 1), so single-step GPTQ optimizes rounding over a narrow subspace even though per-step error magnitude is uniform.

Cross-architecture check. On Ouro-1.4B (4 recurrence steps, 50 WikiText-2 sequences), the g=128 per-step proxy varies by $1.02\times$ across steps, and step 0 accounts for 25.2% of the total (uniform expectation: 25.0%).

Closed-loop INT3 check. Recollecting the Hessians from the quantized rollout changes final-step agreement with bf16 by $+0.37$ pp (prompt-cluster bootstrap 95% CI, $[-2.31, 3.10]$ pp). The open-loop arm collects Hessians for all 17 shared linears over the 32-step bf16 rollout and then applies deterministic group-128 INT3 FakeQuant. The closed-loop arm recollects them from that quantized rollout and requantizes the same linears. On 20 calibration-held-out GSM8K training prompts, agreement is 922/1,090 (84.59%) open-loop and 926/1,090 (84.95%) closed-loop.

K JACOBIAN MEASUREMENT DETAILS

The spectral radius ρ is the magnitude of the dominant eigenvalue of the recurrence Jacobian $J = \partial s_{t+1}/\partial s_t$. For the matched experiment below, we losslessly promote the fixed states and weights to FP32, compute exact forward-mode Jacobian-vector products, and estimate the largest-magnitude Ritz value with implicitly restarted Arnoldi, without materializing J.

Matched spectral-radius protocol. For each of 30 prompts, we form the bf16 states after 32 applications of the no-bypass and bypass recurrences from the same initial state. At each operating state, we evaluate four local maps: bf16 and deterministic per-output-row INT4 adapter weights, each without and with the identity bypass. The bf16 reference states and checkpoint/rounded weights are promoted losslessly into the shared FP32 surrogate; all four maps share the same state and FP32 core, with dtype-dependent normalization epsilon pinned to its bf16-effective value.

We use the same two Arnoldi start vectors in every arm and estimate the largest-magnitude Ritz value with $k = 4$, $\mathrm{ncv} = 64$, tolerance 10^{-5}, and a 4,096-matvec cap, requiring relative Ritz residual below 10^{-3} and agreement between starts within 2×10^{-3}. All 480 arm/state/start estimates satisfy these criteria; the largest residual is 9.23×10^{-6} and the largest start disagreement is 9.34×10^{-6}.

Spectral radius vs. spectral norm. Arnoldi estimates the eigenvalue of largest magnitude, $\rho(J) = |\lambda_{\max}|$, rather than the largest singular value $\sigma_1(J) = \|J\|_2$. For directions with nonzero projection onto the dominant eigenspace, ρ sets the asymptotic exponential rate, while non-normality can still produce finite-step transient growth. The spectral norm governs worst-case single-step amplification but can overestimate asymptotic behavior when J is non-normal. We use ρ because the experiment concerns repeated application of the same local map; σ_1 remains a complementary single-step worst-case diagnostic.

Matched spectral result. The effect of the identity bypass on the INT4-induced change in ρ depends on the operating point. At the no-bypass bf16 state, mean ρ is 1.064 for bf16 and 1.145

after adapter INT4; $\rho > 1$ on 9/30 and 30/30 prompts, respectively. The signed INT4 shifts without and with the bypass are $+0.080$ and $+0.101$, giving a difference-in-differences of $+0.021$ (prompt-bootstrap 95% CI, $[-0.061, 0.121]$). At the bypass bf16 state, the corresponding shifts are -0.340 and -0.041 (difference-in-differences $+0.299$, 95% CI, $[0.204, 0.393]$). The main text therefore uses the no-bypass $\rho > 1$ result and the fixed-weight identity-bypass comparison.

Matched identity-path control. Let C denote the full four-block residual core and a_M the bf16 or deterministic per-output-row INT4 adapter. We reuse one adapter pair in the full recurrence maps $F_{M,0}(s) = C(a_M([s, e]))$ and $F_{M,1}(s) = C(s + a_M([s, e]))$. For a shared state s and unit direction v_k, let $r_{M,i}(s, v_k; \epsilon) = [F_{M,i}(s + \epsilon\|s\|_2 v_k) - F_{M,i}(s - \epsilon\|s\|_2 v_k)]/(2\epsilon\|s\|_2)$ and $D_i = [\sum_k \|r_{Q,i} - r_{B,i}\|_2^2 / \sum_k \|r_{B,i}\|_2^2]^{1/2}$.

On GSM8K training prompts 70–119, 3 recurrent-state initializations shared across conditions, and 32 shared directions, the primary $\epsilon = 0.01$ comparison at the no-bypass bf16 step-32 state gives mean $D_0 = 1.22249$ and mean $D_1 = 0.713216$. The geometric mean of the prompt-level D_1/D_0 ratios corresponds to 41.66% attenuation (bootstrap 95% CI, 41.50–41.82%; all 50 prompt effects share the sign; Holm-adjusted two-sided sign-flip $p = 3.0 \times 10^{-5}$). Across the 150 prompt-initialization units, mean RMS mismatch before dividing by the bf16 response falls from 1.0169 to 0.6366. Under prompt-cluster aggregation, this is 37.40% attenuation.

The direction holds at both bf16 operating points and all tested radii ($\epsilon \in \{0.005, 0.01, 0.02\}$; 24.1–60.0% attenuation). Because the untrained bypass changes the bf16 rollout (10.82% final-step token agreement), this experiment supports only the local structural claim.

L Error Amplification: R-Llama

To measure R-Llama's trajectory displacement on 30 GSM8K prompts, we run the bf16 model for 32 recurrence steps, then start both the bf16 and quantized recurrences from that state and record $\|s_t^{\mathrm{q}} - s_t^{\mathrm{bf16}}\|_2$ after each of 32 further updates.

Under per-channel INT4 (adapter only, core bf16), final mean displacement is 1082. The median per-prompt geometric mean of the 31 consecutive displacement ratios is 1.003. Under grouped INT4 (g=128), the corresponding values are 300 and 1.006. The ratio is slightly higher at g=128 because the per-channel curve begins closer to its plateau, leaving less relative growth over the measured window. Median per-update amplification is largest over the first three transitions and then approaches 1, with small fluctuations around it. After 4 updates, both curves are within 1% of their final displacement. The saturation pattern is consistent with a displaced fixed point $h_q^* \neq h_{\mathrm{bf16}}^*$.

The displacement magnitude tracks quantization precision: 1082 for per-channel INT4 versus 300 at g=128, a 3.6× reduction that matches the weight-error norm ratio. The same ordering appears on GSM8K: per-channel RTN gives 0% accuracy, while g=128 RTN retains 39.5% (Section 3.2; Table 2).

M IIR Filter Experimental Details

Filter design. Four discrete-time IIR filter designs are created using `scipy.signal`: Butterworth order 4, Chebyshev Type I order 4, Chebyshev Type I order 6, and Elliptic order 4. The three order-4 designs use a normalized passband of $[0.2, 0.4]$. The order-6 Chebyshev uses $[0.3, 0.5]$ with 1 dB ripple to push poles closer to the unit circle ($|z| = 0.984$, analogous to Huginn's ρ near 1). These designs span a range of pole configurations: Butterworth places poles on a circle (maximally flat magnitude response), Chebyshev clusters poles near the unit circle for sharper rolloff, and the Elliptic filter balances passband and stopband ripple.

State-space conversion. Each filter's transfer function (b, a) is converted to controllable canonical state-space form (A, B, C, D) via `scipy.signal.tf2ss`. The matrix $A \in \mathbb{R}^{n \times n}$ is the state-transition (feedback) matrix; B, C, and D are the input/output matrices. The system dynamics are $x[n+1] = Ax[n] + Bu[n]$, $y[n] = Cx[n] + Du[n]$. Poles are the eigenvalues of A; the system is stable if and only if all poles lie inside the unit circle ($|z| < 1$).

Quantization methodology. Each matrix is quantized independently via uniform signed round-to-nearest at six bit widths: 4, 6, 8, 10, 12, and 16 bits. Two conditions are tested: (1) quantize the transition matrix A only, and (2) quantize B, C, and D jointly while leaving A fixed. With $q_{\max} = 2^{b-1} - 1$, the quantizer uses $\Delta = \max|M_{ij}|/q_{\max}$ and $M_q = \Delta\,\mathrm{clip}(\mathrm{round}(M/\Delta), -q_{\max} - 1, q_{\max})$. Stability is assessed by computing the eigenvalues of the (quantized or unquantized) A matrix and checking whether $|z|_{\max} > 1$.

Table 20: IIR stability under quantization. $|z|_{\max}$ is the largest pole magnitude; rows with **A-quant stable? = No** are unstable ($|z|_{\max} > 1$). Bold rows mark each filter's stability boundary. BCD quantization leaves poles unchanged, while A-quantization destabilizes low-bit regimes.

Filter	Bits	$\lvert z\rvert_{\max}$ (orig)	$\lvert z\rvert_{\max}$ (A-quant)	$\lvert z\rvert_{\max}$ (BCD-quant)	A-quant stable?
Butterworth 4	4	0.915	2.593	0.915	No
	6	0.915	1.287	0.915	No
	8	0.915	1.134	0.915	No
	10	0.915	**1.011**	0.915	**No**
	12	0.915	**0.934**	0.915	**Yes**
	16	0.915	0.916	0.915	Yes
Chebyshev I, 4	4	0.962	4.575	0.962	No
	6	0.962	1.441	0.962	No
	8	0.962	1.141	0.962	No
	10	0.962	**1.062**	0.962	**No**
	12	0.962	**0.950**	0.962	**Yes**
	16	0.962	0.965	0.962	Yes
Chebyshev I, 6	4	0.984	5.645	0.984	No
	6	0.984	1.922	0.984	No
	8	0.984	1.602	0.984	No
	10	0.984	1.202	0.984	No
	12	0.984	1.134	0.984	No
	16	0.984	**1.012**	0.984	**No**
Elliptic 4	4	0.970	4.569	0.970	No
	6	0.970	1.439	0.970	No
	8	0.970	1.140	0.970	No
	10	0.970	**1.002**	0.970	**No**
	12	0.970	**0.995**	0.970	**Yes**
	16	0.970	0.967	0.970	Yes

BCD quantization does not perturb $|z|_{\max}$ beyond machine epsilon at any precision, consistent with the fact that $\det(zI - A)$ is independent of B, C, D. A-quantization pushes poles past the unit circle at ≤ 10 bits for all four designs. The order-6 Chebyshev boundary case (original $|z|_{\max} = 0.984$) remains unstable even at 16 bits, while the Butterworth design (original $|z|_{\max} = 0.915$) stabilizes at 12 bits. This is consistent with the stability-margin interpretation used for Huginn: a system close to the recurrence boundary can cross it under a small perturbation.

N Mamba SSM Experimental Details

Methodology. Each component is tested under two perturbation conditions. (1) *FakeQuant*: simulated symmetric INT4 quantization (per-tensor for A_log since d_{state}=16 < g=128; per-group g=128 for 2D weights; biases at full precision). FakeQuant is deterministic and uses each component's actual quantization error norm. (2) *Gaussian*: norm-matched Gaussian noise at the median INT4 error across components (22.84 for 130M, 15.09 for 2.8B), ensuring every component receives exactly the same perturbation magnitude regardless of parameter count. The Gaussian condition tests whether the component ordering persists under a second perturbation family. Across the 5 Mamba-130M layers, the `A_log` INT4 weight-error norm ranges from 71.1 to 96.0; the Gaussian control uses the common 22.84 norm for every component.

Measurement procedure. For each condition (component × layer), we run the model on 10 prompts of approximately 100 tokens each. At every timestep t of the SSM scan, we record the

hidden state h_t under the perturbed model and the hidden state h_t^{ref} under the unperturbed model, computing the L2 divergence $\|h_t - h_t^{\text{ref}}\|_2$. Five layers are tested (0, 6, 12, 18, 23 for 130M; 0, 16, 32, 48, 63 for 2.8B). FakeQuant runs once (deterministic); Gaussian runs with three random seeds per layer.

Results. Across perturbation types and model sizes, divergence from `A_log` starts at zero before growing through the scan. With $h_{-1} = 0$, the perturbed $\bar{A}$ term has no state contribution at t=0. Under INT4 FakeQuant, the measured growth is 1.28–2.00× at 130M and 1.30–1.71× at 2.8B. `in_proj` and `x_proj` produce divergence immediately, but its magnitude and trajectory vary across layers and model sizes. `out_proj` produces zero measured SSM-state divergence because it sits after the scanned recurrence.

Connection to Mamba quantization literature. Quamba's error bound assumes the transition matrix A is unperturbed. Our experiment perturbs A directly and measures the resulting divergence. QMamba (Li et al., 2025) reports component-level sensitivity and attributes it to the long-tailed distribution of $\bar{A}_t$. Bi-Mamba (Tang et al., 2025) keeps SSM parameters at full precision and binarizes the input/output projections (Section 7). These results provide SSM-side evidence for feedback exposure. Mamba's single forward scan already observes the timestep distribution used at deployment, so it is outside our calibration-blindness test.

Identity injection sweep. To test whether adding an identity term to the SSM recurrence reduces `A_log` sensitivity, we modify the recurrence to $h_t = \alpha \cdot h_{t-1} + \bar{A} \cdot h_{t-1} + \bar{B} \cdot x_t$, where $\alpha = 0$ recovers the original model. We sweep 9 values of α from 0 to 0.5 on Mamba-130M layer 12, with 5 prompts and 3 seeds per condition. Representative rows are shown in Table 21. We add Gaussian noise to layer 12's `A_log` at $\|\Delta W\|_F = 37.18$, using layer 0's per-row INT4 error norm as the reference magnitude.

Table 21: Identity injection in Mamba SSM. Adding $\alpha \cdot h_{t-1}$ to the recurrence reduces `A_log` sensitivity monotonically from $\alpha \geq 0.15$, consistent with the Huginn identity-bypass result. Intermediate values interpolate smoothly between the shown rows.

α	**Divergence at last** t	**vs.** α=0
0.00	0.127	baseline
0.10	0.130	+2%
0.15	0.120	−6%
0.25	0.111	−13%
0.50	**0.101**	−20%

From $\alpha = 0.15$ onward, divergence drops monotonically, reaching a 20% reduction at $\alpha = 0.5$, with diminishing returns beyond $\alpha \approx 0.25$. In near-marginal Huginn, the matched identity-bypass control reduces the discrepancy between bf16 and INT4 responses to small state perturbations by 41.7%.

O Eigenvalue Spectrum Analysis

Figure 5 plots the full eigenvalue spectrum of the row-normalized calibration Hessian $\widetilde{H} = 2X^\top X/m$, where m is the number of activation rows, for the adapter and a representative core layer (Wqkv, block 0) under three conditions: 50 prompts at step 0, 1,000 prompts at step 0, and 50 prompts accumulated across 32 steps.

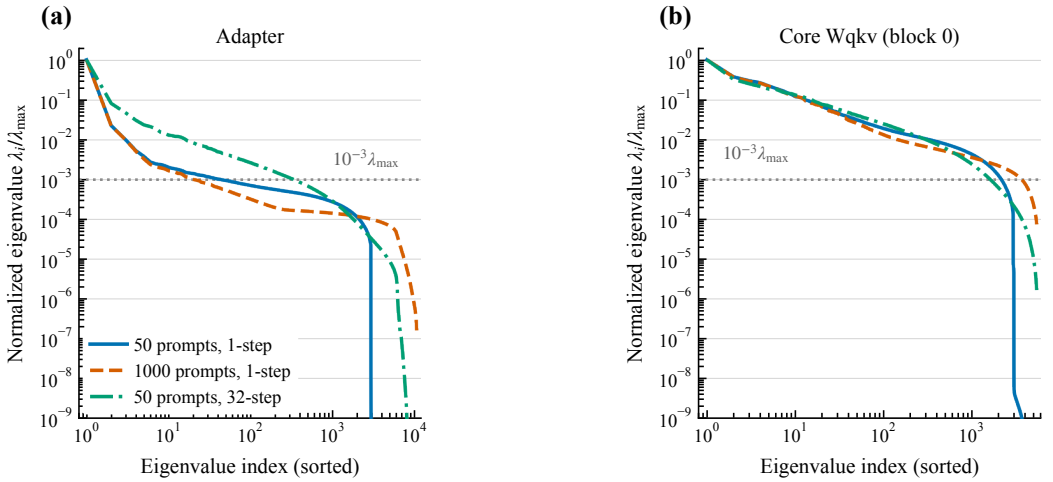

Figure 5: Eigenvalue spectrum of the row-normalized calibration Hessian $\widetilde{H} = 2X^\top X/m$. (a) Adapter ($d$=10,560): the spectrum concentrates in 2–3 dominant eigenvalues at step 0, with a sharp drop exceeding 2 orders of magnitude by index 5. Increasing prompts from 50 to 1,000 (dashed) *deflates* near-threshold eigenvalues below the $10^{-3}\lambda_{\max}$ cutoff, while $\lambda_{\max}$ is stable (1.00× ratio). Accumulating 32 steps (dash-dot) lifts the bulk spectrum, populating 335 directions above threshold. (b) Core Wqkv block 0 (d=5,280): the spectrum decays gradually with no sharp cutoff, and 2,111 eigenvalues exceed the threshold at 50 prompts, 49× more than the adapter. The core layer has a much broader step-0 spectrum than the adapter, while 32-step accumulation substantially broadens the adapter spectrum.

Table 22 reports the adapter Hessian rank across multiple thresholds, plus effective rank (erank) and stable rank, under the three calibration conditions. One representative core layer is included for comparison.

Table 22: Hessian rank across thresholds τ relative to $\lambda_{\max}$, with effective rank (erank) and stable rank. Core Wqkv from block 0 provides the core-layer comparison. The focal τ=10^{-3} column is bold.

Condition	$\mathbf{10^{-1}}$	$\mathbf{10^{-2}}$	$\mathbf{10^{-3}}$	$\mathbf{10^{-4}}$	$\mathbf{10^{-5}}$	$\mathbf{10^{-6}}$
Adapter (d = 10,560)						
50p, 1-step	1	2	**43**	2,120	2,919	2,919
1000p, 1-step	1	2	**20**	3,047	7,462	9,621
50p, 32-step	1	16	**335**	1,795	4,796	6,213
Core Wqkv@0 (d = 5,280)						
50p, 1-step	12	323	**2,111**	2,870	2,870	2,919
1000p, 1-step	12	142	**3,588**	5,227	5,280	5,280
50p, 32-step	15	297	**1,587**	3,611	4,890	5,280

Condition	**erank**	**Stable rank**
Adapter		
50p, 1-step	51	1.8
1000p, 1-step	80	1.8
50p, 32-step	137	2.7
Core Wqkv@0		
50p, 1-step	912	16.6
1000p, 1-step	1,548	16.8
50p, 32-step	705	15.1

For $p_i = \lambda_i / \sum_j \lambda_j$, effective rank is $\exp(-\sum_i p_i \log p_i)$, and stable rank is $\mathrm{tr}(H)/\lambda_{\max}$. Counts at $\tau \le 10^{-4}$ for the 50-prompt 1-step condition are capped by the sample-limited rank of H ($\sim$ 2,919).

The adapter's maximum eigenvalue changes by $< 0.3\%$ between 50 and 1,000 prompts, so the rank decrease from 43 to 20 is not a threshold artifact from a growing $\lambda_{\max}$. Instead, near-threshold eigenvalues shrink below the cutoff as the sample count rises. Core layers behave differently. Their rank increases from 2,111 to 3,588 because the step-0 inputs occupy more directions in input space. Threshold-free measures preserve the same ordering. Adapter erank is 51 vs. 912 for core Wqkv ($17.9\times$ gap), and stable rank is 1.8 vs. 16.6 ($9.2\times$ gap). Accumulating 32 steps lifts the adapter rank by $7.8\times$ ($43 \to 335$) as recurrence steps 1–31 raise more input directions above the threshold.

Step-0 and late-step subspaces. On GSM8K training prompts 70–119, we decompose the adapter inputs at steps 0 and 31 under 2 recurrent-state initialization seeds. After restricting the comparison to the changing 5,280-dimensional recurrent-state half, we form uncentered second moments. The mean squared cosine of the principal angles between their top-32 eigenspaces at steps 0 and 31 is 0.571% and 0.631% for the two seeds. The fixed prompt-embedding half is identical across steps and is excluded. Comparing the seeds directly at k=64, mean squared-cosine overlap is 99.71% for the step-31 subspaces but only 1.25% for the step-0 subspaces. The step-31 subspace is reproducible across seeds, while the step-0 subspace is neither aligned with it nor stable across initializations.

Adapter output magnitude. Adapter output norms support the identity-path interpretation in Section 3.2: $\|\operatorname{Adapter}(\operatorname{cat}(s_t, e))\|_2 / \|s_t\|_2 = 0.415 \pm 0.044$ at steady state (steps 1–31, 50 prompts). The adapter output is non-negligible relative to s_t. Adding a residual path $s_t + \operatorname{Adapter}(\operatorname{cat}(s_t, e))$ gives the state an identity channel around that term.

P Step-Window Ablation

Algorithm 1 accumulates $H = \sum_{t=0}^{N-1} H_t$ over all recurrence steps. We test whether excluding step 0 improves GPTQ quality on 6 architectures (Table 23).

Table 23: Step-0 exclusion ablation. WikiText-2 PPL (↓) for grouped-INT4 g=128. Δ is skip-0 minus N-step; negative means skip-0 is better.

Model	N	RTN	N-step	Skip-0	Δ
Ouro 2.6B	4	11.94	11.62	**11.10**	**−0.52**
Parcae 1.3B	8	51.30	20.84	**20.58**	**−0.26**
Huginn 3.5B	32	14.98	**14.89**	14.92	+0.03
R-Llama 1.4B	32	34.02	**33.46**	33.65	+0.19
LoopFormer 278M	8	35.10	**34.08**	34.33	+0.25
Ouro 1.4B	4	14.38	**12.97**	13.78	+0.81

Step 0 activations come from prompt embeddings, before recurrence has shaped the state. Table 23 shows the effect of excluding them is architecture-dependent. Ouro-2.6B (-0.52 PPL) and Parcae-1.3B (-0.26) improve under skip-0. Ouro-2.6B's bottleneck has step-0 trace $7{,}249\times$ larger than step-1, so the accumulated Hessian is dominated by the step-0 subspace (rank 1). The Hessian assigns near-zero cost to the 1,010–1,335 directions that later steps contribute, so column compensation can push rounding error into those directions while reducing step-0 reconstruction error (Section 5). Excluding step-0 forces the solver to penalize error in those directions. LoopFormer (+0.25) and Ouro-1.4B (+0.81) degrade: step 0 contributes directions that the remaining steps do not fully cover (Table 18). Huginn (+0.03) and R-Llama (+0.19) do not benefit.

For Ouro-2.6B, downweighting rather than excluding step 0 confirms the pattern: $H = \alpha H_0 + (1-\alpha) H_{\text{rest}}$ degrades monotonically as α grows from 0 to 0.25, with PPL 11.10, 11.28, 11.36, 11.47, 11.60. For Ouro-1.4B the optimum is α=0.15 (PPL 12.93 vs. 12.97 at $\alpha = 1/N$). Skip-0 is a per-checkpoint tuning knob; full accumulation is the default.

Limits of the per-layer proxy. Both the deployment proxy (Eq. 3) and GPTQ treat each layer independently. At step t, though, activations X_t already carry rounding errors from steps 0 through $t-1$. The frozen-trajectory proxy fixes X_t at the bf16 reference and drops these cross-step terms. On both Ouro scales, skip-0 reduces per-layer $\operatorname{tr}(\Delta W\, H_3\, \Delta W^\top)$ nearly everywhere (335/336 layers

at 2.6B, 168/168 at 1.4B). PPL improves only at 2.6B. The per-layer sum ignores this coupling, so small proxy gains need not translate into lower model-level perplexity.

Q Experimental Setup

Benchmarks. GSM8K (Cobbe et al., 2021) (8-shot chain-of-thought, n=1,319) is the primary generation benchmark for Huginn/R-Llama. Strict match counts exact normalized numeric answers; flex match accepts extractor-tolerant equivalent numeric answers, and intervals are Wilson 95% confidence intervals where shown. WikiText-2 (Merity et al., 2017) perplexity is the primary metric for Ouro, LoopFormer, Parcae, PonderPythia, and Nanbeige and is evaluated on the standard test split. LAMBADA (Paperno et al., 2016) accuracy is reported as supporting evidence for Parcae. ARC-Challenge (Clark et al., 2018) (0-shot, 1,172 problems) and HellaSwag (Zellers et al., 2019) (0-shot, 10,042 problems) use log-likelihood scoring. Per-token fidelity is measured as top-1 agreement with bf16 and KL divergence across all recurrence steps. Benchmark evaluations use the `lm-eval-harness` framework (EleutherAI, 2026).

Table 24: Wilson 95% confidence intervals for the main GSM8K rows evaluated on all 1,319 test examples, computed from the exact underlying counts.

Model	Config	Strict (%)	Wilson 95% CI
Huginn 3.5B	bf16	34.9	[32.4, 37.5]
Huginn 3.5B	RTN	27.9	[25.5, 30.4]
Huginn 3.5B	1-step GPTQ	26.9	[24.6, 29.4]
Huginn 3.5B	N**-step GPTQ**	**35.0**	[32.5, 37.6]
R-Llama 1.4B	bf16	50.2	[47.5, 52.9]
R-Llama 1.4B	RTN	39.5	[36.9, 42.2]
R-Llama 1.4B	1-step GPTQ	40.3	[37.6, 42.9]
R-Llama 1.4B	N**-step GPTQ**	**45.3**	[42.6, 48.0]

Quantization and hardware. All experiments run on a single NVIDIA A100 (CUDA 12.4, PyTorch 2.6). Main-table quantization uses deterministic dequantized grouped-INT4 FakeQuant (g=128), isolating calibration effects from integer-kernel behavior. Huginn/R-Llama use a local implementation of GPTQ's column-update rule (Frantar et al., 2022); Ouro, LoopFormer, Parcae, PonderPythia, and Nanbeige use architecture-specific implementations of the same rule. Parcae quantizes `core_block` only.

PonderPythia uses 3 pondering updates plus a final shared-stack prediction pass and quantizes all 128 shared-stack projections plus its untied output projection (129 total). Nanbeige applies its 22-layer stack twice and quantizes 154 shared-stack projections plus the output projection (155 total; 3.66B weights); in both GPTQ arms the output projection uses the same RTN weights. Input embeddings remain bf16 in both models. The Marlin W4A16 kernel (Frantar et al., 2025) is used for the real-kernel speed measurement (Section 6.2). INT8 rows, where referenced in appendix ablations, use simulated per-channel symmetric quantization (FakeQuant with fp32 arithmetic). Mamba experiments use deterministic INT4 FakeQuant plus norm-matched Gaussian controls. On Huginn, logged Hessian collection took approximately 15 minutes for one-step calibration and 25 minutes for 32-step trajectory calibration on one A100; the calibration change adds no inference-time cost.

Third-party assets. All third-party checkpoints, datasets, and tools are accessed from upstream sources; none are redistributed with this submission. The table records the license metadata stated by those sources and marks checkpoints whose cards state no license terms. Links point to exact revisions when a pinned revision is available; the table displays shortened hashes.

Table 25: Calibration and solver details for the recovery rows. All rows use deterministic FakeQuant with g=128, no act-order, and unchanged inference graphs within each row.

Rows	Calibration data	Hessian collection	Solver
Huginn/R-Llama	First 50 GSM8K training questions	1 or 32 recurrence steps	GPTQ, 1% damping for core and 10% for adapter
Ouro	WikiText-2 train, 128×2048 tokens	1 or 4 recurrence steps	Column-wise GPTQ, 0.01 damping
LoopFormer	WikiText-2 train, 128×1024 tokens	1 or 8 loop iterations	Column-wise GPTQ, 0.01 damping
Parcae	WikiText-2 train, 128×1024 tokens	1 or 8 core iterations	Column-wise GPTQ, 0.01 damping
PonderPythia	WikiText-2 train, 128×2048 tokens	First pass or all 4 shared-stack passes	Column-wise GPTQ, 0.01 damping
Nanbeige	WikiText-2 train, 128×2048 tokens	First pass or both shared-stack passes	Column-wise GPTQ, 0.01 damping

Table 26: Third-party asset licenses. No third-party checkpoints, datasets, or tooling are redistributed with this submission.

Asset / source	License
Checkpoints	
Huginn-3.5B (`0f0fa0ba`); R-Llama-1.4B (`a5f6f126`); Ouro-1.4B (`574fa66c`); Ouro-2.6B (`1ed04250`)	Apache-2.0 (pinned checkpoint cards).
PonderPythia-2.8B (`5bd111c1`); Nanbeige4.2-3B-Base (`4a38e817`)	Apache-2.0 (pinned checkpoint cards).
Mamba-130M; Mamba-2.8B	Checkpoint terms not stated; code Apache-2.0.
LoopFormer-278M (`e7c87640`); Parcae-140M (`236739ce`); Parcae-1.3B (`517f8e6a`)	Checkpoint terms not stated; code MIT.
COCONUT-GPT2-124M	Checkpoint terms not stated; code MIT.
Benchmarks	
GSM8K	MIT.
WikiText-2 (Salesforce; EleutherAI document-level)	CC BY-SA 3.0; Salesforce source card also lists GFDL.
LAMBADA	Modified MIT.
ARC-Challenge	CC BY-SA 4.0.
HellaSwag	MIT.
Tooling	
`lm-eval-harness`	MIT.
IST-DASLab GPTQ; Marlin	Apache-2.0.
PyTorch	BSD-3-Clause.
HuggingFace Transformers	Apache-2.0.