MARGINGATE: SPARSE MARGIN-TRIGGERED VERIFICATION FOR BATCH-SENSITIVE GREEDY LLM DECODING

Anonymous authors
Paper under double-blind review

ABSTRACT

Greedy BF16 decoding is commonly treated as reproducible at temperature zero. The same request can still emit a different token when the serving batch changes, because finite-precision reductions depend on the batch shape, and the user does not control the other requests. Across five models and three benchmarks, these synchronous flips occupy only 0.3–1.3% of steps. An online gate cannot see which steps flipped. The verification rate is set by $A(\tau_R)$, the fraction of token-stable steps whose top-1/top-2 margin lies under the smallest threshold that covers the flips. At complete coverage, $A(\tau_R)$ is 3.3% on Qwen2.5-14B MATH500 and 88.4% on DeepSeek-R1-Distill-Qwen-7B.

MarginGate keeps the default BF16 decode and reads the margin that forward already produced. A step with margin at least τ is committed. A smaller margin is recomputed from the committed cache. A token match keeps the tentative BF16 column. A mismatch emits the verifier token and replaces only the current column. On the Qwen2.5-14B MATH500 selection set, one HF eager run at $\tau = 4.0$ matches $120/120$ batch-1 sequences, against $6/120$ for unguarded BF16, while verifying 15.05% of steps. The same threshold matches $119/120$, $113/120$, and $118/120$ on GSM8K, SharedGPT, and HumanEval. On DeepSeek-R1-Distill-Qwen-7B, where $A(\tau_R)$ is 88.4%, verification stays frequent.

1 INTRODUCTION

Greedy decoding is commonly treated as reproducible in evaluation, regression testing, and production serving (Gu et al., 2024; Song et al., 2025; Zhang et al., 2025; Gond et al., 2026). BF16 inference can violate this assumption even at temperature zero: the same request may emit different tokens when decoded alone and in a larger batch (Yuan et al., 2025). The variation is numerical rather than stochastic. Serving engines change the batch shape as requests enter and leave (Yu et al., 2022; Kwon et al., 2023; Zheng et al., 2024; Agrawal et al., 2024; Zhong et al., 2024; Chu et al., 2025a). Users do not control the other requests, so a nominally deterministic output should not depend on them.

Two existing responses either modify every step or verify on a fixed schedule. Batch-invariant operator libraries fix the reduction order inside attention and matrix multiplies, and LayerCast upcasts linear layers, so both change the kernel at every decode step (Thinking Machines Lab, 2026; Yuan et al., 2025). Serving stacks have begun to expose these deterministic paths (vLLM Project, 2026; SGLang Project, 2026). LLM-42 keeps the default BF16 kernels and verifies a window of $K_{\text{verify}} = 64$ unverified tokens, or the finished request, because a matching state usually yields the same next token (Gond et al., 2026). The window recomputes every step inside it, including steps whose top-1/top-2 margin is large. MarginGate reads the margin the current BF16 forward already produced and calls the verifier only when the margin falls below τ.

Synchronous flips occupy 0.3–1.3% of steps across five models and three benchmarks, but the gate cannot see which of them flipped. At complete coverage, the fraction of token-stable steps under the threshold, $A(\tau_R)$, is 3.3% on Qwen2.5-14B MATH500 and 88.4% on DeepSeek-R1-Distill-Qwen-7B. On Llama and Qwen, every flip's alternative lies in the top-3, so the top-1/top-2 margin is available from the current forward; DeepSeek-R1-Distill-Qwen-7B's alternative has a longer tail. Final-layer KV deviation stays near its pre-flip floor and jumps at the first disagreeing token. The

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

jump at the first flip is the reason to repair only the current column. Earlier columns are not shown to equal the deterministic reference.

MarginGate skips a step whose top-1/top-2 margin is at least τ. It keeps the default BF16 decode and calls a verifier only when the margin falls below a threshold chosen on the MATH500 selection set. A matching token keeps the tentative BF16 column. A mismatch emits the verifier token and replaces only the current column. The timed prototype cannot write that column in place, so it rolls back to the committed prefix and re-decodes the current step. We measure sequence match of this rollback prototype against a batch-1 trajectory.

This paper makes three contributions.

- Across five models and three benchmarks, complete flip coverage admits 3.3% of stable MATH500 steps on Qwen2.5-14B and 88.4% on DeepSeek-R1-Distill-Qwen-7B, at a synchronous flip rate of 0.3–1.3%.
- MarginGate skips high-margin steps and sends only a low-margin step to the verifier. On a token match it keeps the tentative BF16 column. We measure sequence match for models with small $A(\tau_R)$.
- On the HF eager Qwen2.5-14B MATH500 selection set, the rule matches $120/120$ reference sequences, against $6/120$ for unguarded BF16, while verifying 15.05% of steps. τ is selected on this set. GSM8K, SharedGPT, and HumanEval then match $119/120$, $113/120$, and $118/120$. The rollback overhead is measured separately from $A(\tau_R)$.

2 Motivation

Selective verification pays when a signal from the current forward covers the flips while few stable steps are admitted. Rounding error makes the greedy token depend on the order of finite-precision reductions (Goldberg, 1991; Higham, 2002). Exact attention kernels accumulate products in an order that depends on the tile shape (Dao et al., 2022).

For each prompt, we place a protected request in a batch that repeats that prompt, at sizes 2, 4, 8, and 16, and compare it with a deterministic batch-invariant execution, hereafter the deterministic reference. A *batch-induced token flip* occurs when the two runs emit different tokens at the same position. Rates use *synchronous samples*, which end at the first flip in each trial; later mismatches compare different continuations and mix in the effect of the new prefix. Table 1 pools the four batch sizes. Table 16 gives the counts and two-sided 95% Wilson intervals. These batches repeat one prompt; Appendix A.5 reports heterogeneous batches.

2.1 Sparse Batch-Induced Token Flips

Table 1: Pooled synchronous token-flip rate over batch sizes 2, 4, 8, and 16.

Benchmark	Llama-3.2-1B	Llama-3.1-8B	Qwen2.5-7B	Qwen2.5-14B	DeepSeek-R1-Distill-Qwen-7B
MATH500	0.36%	0.48%	0.67%	0.47%	1.16%
GSM8K	0.45%	0.32%	0.86%	0.61%	1.02%
HumanEval	0.59%	0.51%	1.27%	0.84%	1.09%

Across five models and three benchmarks, the pooled synchronous flip rate is 0.3–1.3% (Table 1). The highest cell is Qwen2.5-7B on HumanEval, 1.27% ($108/8{,}483$), so 98.7% of the synchronous steps in that cell match the reference. DeepSeek-R1-Distill-Qwen-7B stays near 1% on every benchmark. Most synchronous steps already match. An online gate does not observe which of them flipped, so this rate is not the verification budget.

Insight. Synchronous flips occupy 0.3–1.3% of steps. Section 4.2 measures $A(\tau_R)$, the fraction of low-margin stable steps.

2.2 MARGIN AS AN OBSERVABLE RISK SIGNAL

The flip rate is not the verifier budget. On the synchronous prefix, let τ_R be the smallest margin that covers a fraction R of flips, and let

$$A(\tau_R) = \Pr[g_t^{\text{logit}} \le \tau_R \mid \text{stable}]$$

be the fraction of token-stable steps that threshold selects. On one distribution and one threshold, if the flip rate is F, flip coverage is R, and $A(\tau_R) = A$, then $r_{\text{verify}} = FR + (1 - F)A$. When F is near 1%, r_{verify} is close to A. Table 1 and Table 5 are the two sides of that identity. Qwen2.5-14B on MATH500 has a pooled synchronous flip rate of 0.47% and, at complete coverage, $A(\tau_R) = 3.3\%$. DeepSeek-R1-Distill-Qwen-7B on MATH500 has 1.16% and 88.4%. With F near 1%, r_{verify} stays near $A(\tau_R)$. Section 4.3 measures wall-clock time separately, because rollback and the kernel path also contribute. The gate still needs a score it can read from the current forward. Comparing two trajectories at every step would spend the cost selective verification is meant to avoid.

Ranking the reference token inside the protected request's logits shows where the alternative lands. On the Llama and Qwen instruct models, the alternative lies in the top-3 for every flip (Table 2). DeepSeek-R1-Distill-Qwen-7B is wider: 69.7% of its alternatives are in the top-2, 88.1% are in the top-8, and about 10% fall outside the top-50. Rank is not a margin: a top-3 alternative can still sit far from the top logit, and a lower-ranked alternative can sit close to it. Appendix B.2 reports the mean number of logits within 0.5 of the top-1.

Table 2: Cumulative percentage of token flips for which the cross-batch alternative appears in the protected request's top-K logits.

Top-K	Llama-3.2-1B	Llama-3.1-8B	Qwen2.5-7B	Qwen2.5-14B	DeepSeek-R1-Distill-Qwen-7B
top-2	94.8%	94.1%	98.9%	95.8%	69.7%
top-3	100%	100%	100%	100%	77.1%
top-8	100%	100%	100%	100%	88.1%

That identity check uses both trajectories, so an online gate cannot compute it. The quantity available from the current forward is the top-1/top-2 margin. Let $N(\eta) = |\{j : \ell_j \ge \ell_{(1)} - \eta\}|$ be the number of candidates within η of the top-1. The condition $N(\eta) > 1$ holds exactly when the margin $g_t^{\text{logit}} = \ell_{(1)} - \ell_{(2)}$ is at most η. Appendix B.2 reports $\eta = 0.5$ and the wider neighborhoods.

At $\eta = 0.5$, flip positions have 1.82–$2.31\times$ as many near-top candidates as stable positions. The ratio shows that flip positions are more crowded. It does not choose τ. $A(\tau_R)$ is the fraction of stable steps under the threshold. The verification rate counts every selected step and is reported in §4.3.

Insight. On Llama and Qwen, every flip's alternative is inside the top-3, which motivates a margin gate. DeepSeek-R1-Distill-Qwen-7B's longer tail does not fit that pattern. We report $A(\tau_R)$ at that coverage. Serving time is measured separately (§4.3).

2.3 STATE LOCALITY AT THE FIRST FLIP

For the protected request, $E_p^K = \|K_{:,p,:}^{\text{prot}} - K_{:,p,:}^{\text{ref}}\|_2$ is the final-layer key-column deviation at position p, and E_p^V is the same quantity for values. Trials are aligned by $\Delta = p - p_{\text{flip}}$. p_{flip} is the cache index of the first differing generated token after that token has been written into the cache, so columns with $\Delta < 0$ still share a token prefix, while the column at $\Delta = 0$ already takes the disagreeing token as input. The spike includes that input change and is not the cause of the flip.

For Llama-3.1-8B, K and V stay flat before the first flip and rise by $41\times$ and $44\times$ at $\Delta = 0$ (Table 3, Figure 1). Across models the final-layer spike is 12–$60\times$ for K and 17–$82\times$ for V. A per-trial regression over $\Delta < 0$ gives a median deepest-layer K slope within $|0.007|$ per token. These flat traces motivate current-column repair (§3.4); they do not show that earlier columns match the reference. Known flip positions, replaced by the reference column, return the error to the pre-flip floor (Appendix D.1). That oracle is not the serving measurement.

Insight. Final-layer deviation stays on the pre-flip floor for $\Delta < 0$, then jumps 12–82× at $\Delta = 0$. The spike includes the new input token.

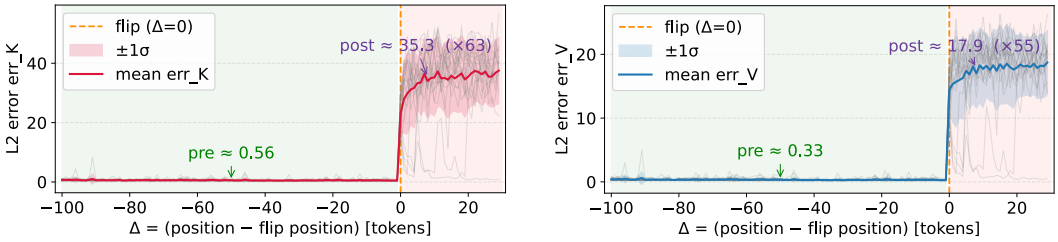


Figure 1: Llama-3.1-8B final-layer K (left) and V (right) at B=8. Δ=0 is the first disagreement.

Table 3: Flip-aligned KV-cache deviation at B=8 relative to the deterministic reference. Each column is that model's last layer. "pre" is the median pre-flip mean ℓ_2 deviation over $\Delta < 0$; "spike" is the $\Delta = 0$ deviation divided by pre.

Metric	Llama-3.2-1B	Llama-3.1-8B	Qwen2.5-7B	Qwen2.5-14B	DeepSeek-R1-Distill-Qwen-7B
K pre	0.37	0.56	1.37	1.00	0.72
K spike	60×	41×	12×	28×	23×
V pre	0.21	0.33	4.81	1.93	2.24
V spike	82×	44×	17×	30×	39×

3 Design: MarginGate

3.1 Overview

MarginGate is a per-request policy on an otherwise unchanged BF16 greedy decode. At step t the committed state is the prefix $y_{<t}$ and its cache $(K_{<t}, V_{<t})$. The BF16 engine produces logits ℓ_t, a tentative token $\tilde{y}_t$, and a tentative column $(\widetilde{K}_t, \widetilde{V}_t)$. The controller scores those logits. A margin of at least τ fast-commits the tentative transition and resumes BF16. A smaller margin hands the step to the repair engine, which recomputes it from the committed prefix and cache. A matching token accepts the tentative column. A disagreement emits the verifier token and replaces only the current column; earlier columns stay in place. Either outcome resumes BF16. Other requests in the same batch can remain on the default BF16 path. Figure 2 shows this data flow, and Algorithm 1 gives the step.

Each step ends in one of three ways. A margin of at least τ commits the tentative BF16 token and column. A smaller margin is recomputed from the committed cache $(y_{<t}, K_{<t}, V_{<t})$. A token match keeps the tentative column. A mismatch replaces only the current column. The wall-clock number in §4.3 is the rollback prototype in §3.4, which cannot perform that column write.

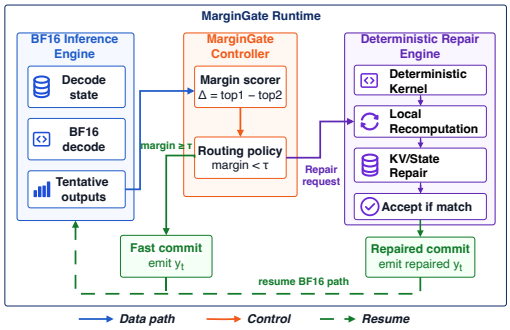


Figure 2: Data flow for one protected request.

3.2 Controller

For the protected request q, let ℓ_q be the logits from the default BF16 batched forward. The controller computes $g_t^{\text{logit}} = \ell_{q,(1)} - \ell_{q,(2)}$ and invokes the verifier when $g_t^{\text{logit}} < \tau$. $A(\tau_R)$ in §2.2 instead counts steps whose margin is at most τ_R. A step with $g_t^{\text{logit}} \geq \tau$ commits the tentative token and column. The score uses logits the BF16 forward already produced, so that step runs no second execution.

The gate uses the margin from §2.2. It does not observe the flip label, so stable low-margin steps are selected as well; §4.2 measures $A(\tau_R)$.

The verification rate $r_{\text{verify}} = \Pr[g_t^{\text{logit}} < \tau]$ is the fraction of steps sent to the verifier. It is zero on the default BF16 path and one for always-on verification. The repair rate r_{repair} counts steps whose verified token disagrees and whose current column is replaced. Verifier calls account for most of the added work; repairs record changes to the decode state.

Algorithm 1 One MarginGate decode step for a protected request.

$(\ell_t, \tilde{y}_t, \widetilde{K}_t, \widetilde{V}_t) \leftarrow$ BF16 forward
$g_t^{\text{logit}} \leftarrow \ell_{t,(1)} - \ell_{t,(2)}$
if $g_t^{\text{logit}} \geq \tau$ **then**
    commit $(\tilde{y}_t, \widetilde{K}_t, \widetilde{V}_t)$ ▷ fast commit
else
    $(y^{\text{ver}}, K^{\text{ver}}, V^{\text{ver}}) \leftarrow \text{verifier}(y_{<t}, K_{<t}, V_{<t})$ ▷ committed cache, not the reference cache
    if $y^{\text{ver}} = \tilde{y}_t$ **then**
        commit $(\tilde{y}_t, \widetilde{K}_t, \widetilde{V}_t)$ ▷ accept
    else
        commit $(y^{\text{ver}}, K^{\text{ver}}, V^{\text{ver}})$ ▷ current column
    end if
end if

3.3 Calibration

The threshold τ is chosen offline for each model and serving configuration. If two full-vocabulary logit vectors differ by at most ε in ℓ_∞, their argmaxes can differ only when the margin is at most 2ε; a tie at that boundary can still change the argmax (Thinking Machines Lab, 2026; Yuan et al., 2025). On the MATH500 selection prompts we estimate $\varepsilon_{\text{pert}}^{(50)}$ and use $2 \cdot \max(\varepsilon_{\text{pert}}^{(50)})$ only as an initial sweep scale. It is not an upper bound, and the top-50 scale is not an ℓ_∞ bound on the full vocabulary. The operating point is the smallest tested threshold in Table 10 with no observed sequence mismatch, and several selected values lie above that scale. It is then held fixed. This is an empirical operating point, not a certificate for a later prompt or workload. Section 4.6 measures the full-vocabulary scale, and Appendix A lists the scales.

3.4 Repair Engine

The online verifier reads the protected request's committed prefix and cache, $(y_{<t}, K_{<t}, V_{<t})$, and recomputes only step t. That cache can already contain numerical error or a missed token. Given this state and a fixed kernel configuration, the recomputed token is deterministic. A deterministic result on this cache can still differ from the batch-1 trajectory, which uses its own cache. We measure sequence match against that trajectory.

Table 4: Difference from LLM-42.

	LLM-42	MarginGate
Trigger	K=64 / end	margin $< \tau$
Match	replace KV	keep BF16
Mismatch	repair window	current column
Skipped	covered	not rechecked

If $y_t^{\text{ver}} = \tilde{y}_t$, the engine accepts the tentative BF16 column and resumes. Otherwise it emits y_t^{ver} and replaces $(\widetilde{K}_t, \widetilde{V}_t)$ with the verifier's current column. Earlier columns stay in place. The bounded pre-flip deviations in §2.3 motivate that choice; they do not prove those columns match the reference. LLM-42 instead replaces the verified window's KV even when the tokens match.

The evaluated serving APIs do not expose a mid-generation column write. The timed prototype rolls back to the committed prefix and decodes the current step again. It does not replay the request from the first token, and it does not revisit steps the gate skipped. The $+75.2\%$ in §4.3 is that rollback, not Algorithm 1's column write. The $120/120$ MATH500 match is this prototype.

4 EVALUATION

4.1 SETUP

The core experiments select each threshold on the MATH500 selection set. For each model, τ_{100} is the smallest tested threshold with no observed sequence mismatch in that sweep. The MATH500 full-match values therefore describe selected operating points rather than performance on an independent held-out MATH500 set. We keep each selected threshold fixed for transfer to GSM8K, SharedGPT, and HumanEval (Hendrycks et al., 2021; Cobbe et al., 2021; Chiang et al., 2023; Chen et al., 2021). All runs use greedy BF16 decoding on dense open-weight models and compare a protected request in a batched run with the deterministic reference. Sampling, beam search, and speculative decoding change the control flow and lie outside this evaluation (Leviathan et al., 2023; Chen et al., 2023), as does mixture-of-experts serving, where expert residency and precision move with the batch (Shen et al., 2025; Chu et al., 2025b). Unless otherwise noted, batches replicate the same prompt to isolate the effect of batch shape; Appendix A.5 evaluates heterogeneous batches. Timing and sequence-match measurements use a 1024 output-token cap. A sequence match is exact agreement with the reference output.

The deterministic reference in §2 is the batch-invariant execution that labels synchronous flips. Sequence match on the 1024-token runs compares the protected request with a batch-1 greedy trajectory. The online verifier reads the protected request's committed cache.

The evaluated checkpoints are Llama-3.2-1B-Instruct, Llama-3.1-8B-Instruct, Qwen2.5-7B-Instruct, Qwen2.5-14B-Instruct, and DeepSeek-R1-Distill-Qwen-7B; elsewhere we use the shorter display names Llama-3.2-1B, Llama-3.1-8B, Qwen2.5-7B, Qwen2.5-14B, and DeepSeek-R1-Distill-Qwen-7B. These checkpoints use grouped-query attention (Ainslie et al., 2023; Meta AI, 2024; Guo et al., 2025). The synchronous diagnostics in §2 stop at the first token flip. All of these measurements are produced by the evaluation harness in this project: it loads the checkpoint, builds the homogeneous batch, runs greedy BF16 decoding, applies the margin gate or the $K_{\text{verify}} = 64$ window, and writes the sequence-match and timing records reported below.

We compare four strategy classes. *BF16 default* applies no intervention. *Global interventions*, represented by `batch_invariant_ops` (Thinking Machines Lab, 2026) and LayerCast (Yuan et al., 2025), patch reductions or linear layers at every step; our Batch-Invariant Ops measurements use the SGLang implementation of the Thinking-Machines kernels. *Always-on verification* (LLM-42 (Gond et al., 2026)) retains the default path and verifies when $K_{\text{verify}} = 64$ unverified tokens have accumulated, or when the request finishes. *Sparse verification* (MarginGate) invokes that verifier only when $g_t^{\text{logit}} < \tau$. On a token match it keeps the tentative BF16 column; LLM-42 replaces the verified window's KV even when the tokens match.

4.2 WHEN SPARSE VERIFICATION APPLIES

Figure 3 draws each model and benchmark at B=8 as a segment from R=90% to R=100%. The open marker is the stable-step rate at 90% flip coverage, and the filled marker covers every observed flip. The horizontal coordinate is the synchronous flip rate.

Both Llama models, and Qwen2.5-14B on MATH500 and GSM8K, cover every synchronous flip while selecting 2.4–6.4% of stable steps (Table 5). Qwen2.5-7B on MATH500 moves from 3.6% at 90% coverage to 39.2% at complete coverage, and DeepSeek-R1-Distill-Qwen-7B on MATH500 moves from 40.9% to 88.4%. Qwen2.5-14B needs 3.3% of stable MATH500 steps for complete coverage and 29.3% on HumanEval. A small flip rate is not sufficient when $A(\tau_R)$ is large. $A(\tau_R)$ is sepa-

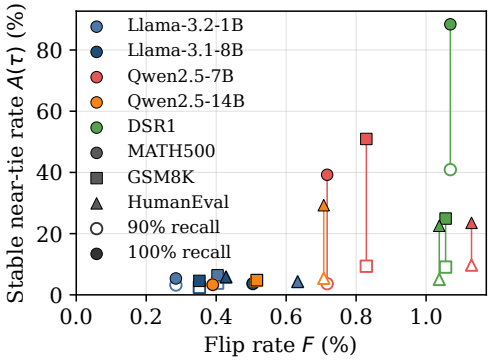


Figure 3: Synchronous prefix at B=8. Each segment runs from R=90% to R=100%.

rate from the overhead in §4.3. $A(\tau_R)$ is defined in §2.2 and counted on the synchronous prefix in §2.

Table 5: Stable near-tie rate $A(\tau_R)$ at B=8.

	MATH500		GSM8K		HumanEval	
Model	90%	100%	90%	100%	90%	100%
Llama-3.2-1B	3.2%	5.4%	3.8%	6.4%	4.3%	4.3%
Llama-3.1-8B	3.7%	3.7%	2.4%	4.6%	5.8%	5.8%
Qwen2.5-7B	3.6%	39.2%	9.4%	50.9%	9.7%	23.5%
Qwen2.5-14B	3.3%	3.3%	4.8%	4.8%	5.4%	29.3%
DeepSeek-R1-Distill-Qwen-7B	40.9%	88.4%	9.1%	24.9%	5.0%	22.6%

On the teacher-forced 192-token traces, both the batch-1 and batched runs consume the batch-1 token at every step, including steps after an earlier disagreement. In-sample complete coverage costs 2.66%, 2.86%, and 4.38% verification for Llama-3.2-1B, Llama-3.1-8B, and Qwen2.5-14B, against 73.35% and 93.96% for Qwen2.5-7B and DeepSeek-R1-Distill-Qwen-7B. Here 3.3% is $A(\tau_R)$ on the synchronous prefix, 4.38% is the verification rate on teacher-forced 192-token rows, and 15.05% is the verification rate of free 1024-token generation.

4.3 Sequence Agreement and Serving Cost

Table 6: Qwen2.5-14B MATH500 at B=8, one pass of 120 prompts. Each overhead uses that row's own BF16 time.

Method	Stack	BF16 (s)	Overhead	Match
BF16	HF eager	70.15	0%	6/120
MarginGate	HF eager, rollback	70.15	+75.2% (+23.9% excl. catch-up)	120/120

On one HF eager pass, Qwen2.5-14B at $\tau = 4.0$ matches 120/120 reference sequences, against 6/120 for BF16, while verifying 15.05% of steps. Unguarded BF16 matches 6 of the 120 batch-1 sequences, so 114 sequences differ in at least one token. That pass uses one A6000 at $B = 8$. Its mean BF16 time is 70.15 seconds; the rollback prototype costs +75.2% against that time, and +23.9% is the same pass with single-token cache catch-up removed (Table 6). LLM-42 costs +47.5% against a vLLM BF16 mean of 24.95 seconds, and Batch-Invariant Ops costs +129% against an SGLang BF16 mean of 22.67 seconds. The timing logs for LLM-42 and Batch-Invariant Ops do not record sequence match.

$\tau = 4.0$ is the smallest tested threshold at which sequence match reaches 120/120. At $\tau = 2.0$, the verification rate is 8.95% and sequence match is 99.2%. Raising the threshold to $\tau = 8.0$ increases verification from 15.05% to 24.60% with no further matched sequence. Appendix A.1 gives the full sweep. Serving points for the other models, including entries marked †, stay in Appendices A.2 and A.3.

4.4 Fixed-Threshold Transfer

With Qwen2.5-14B fixed at $\tau = 4.0$, GSM8K matches 119/120, SharedGPT 113/120, and HumanEval 118/120 (Table 7). Verification rises from 15.05% to 19.38–25.58%. SharedGPT's sequence-match 95% Wilson interval is 88.5–97.2%. A missed flip can change the suffix, so token recall and sequence match differ. Appendix A.4 reports the within-stack overheads, and Llama-3.1-8B at 120/120, 118/120, and 117/120. The 120/120 MATH500 result selects τ. The 119/120, 113/120, and 118/120 counts are what that fixed threshold does on the other three workloads.

Table 7: Qwen2.5-14B at $\tau = 4.0$. MATH500 selects the threshold; other rows hold it fixed.

Dataset	BF16 match	MarginGate match	Verification
MATH500	6/120	120/120	15.05%
GSM8K	5/120	119/120	21.16%
SharedGPT	2/120	113/120	25.58%
HumanEval	2/120	118/120	19.38%

4.5 Flip Coverage at Low Verification Rates

Oracle repair copies the deterministic-reference column at known flip positions (Figure 4). No log compares those columns with the online verifier's columns. For $B \in \{2, 4, 8, 16\}$, the oracle replaces the current KV column with the deterministic reference at each known token flip, and only when the token changes. Fewer than 2.4 repairs per sequence recover sequence match for every measured model and batch size. Appendix D.1 shows the corresponding traces.

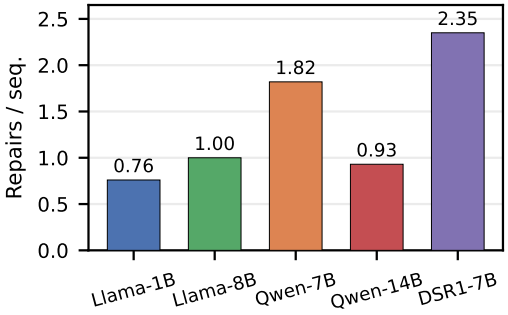


Figure 4: Copies of the deterministic-reference column at known flips. Not the rollback timing in §4.3.

The comparison is which score covers every observed flip at the lowest verification rate. On Llama-3.1-8B and Qwen2.5-14B, margin's zero-miss rates are 2.9% and 4.4%; entropy needs 18.2% and 14.5%, and the best cluster-count rule needs 8.1% and 7.4% (Table 8). Appendix B reports the remaining scores and model families.

Table 8: Zero-miss verification rate on 192-token traces.

Model	Margin	Entropy	Best cluster
Llama-3.1-8B	2.9%	18.2%	8.1%
Qwen2.5-14B	4.4%	14.5%	7.4%

To test whether calibration preserves this coverage on new prompts, each fold holds out five complete prompts and selects the smallest scalar-margin threshold covering every flip in the remaining prompts. Both Llama traces contain 10 prompts (two folds, 1,920 token rows); Qwen2.5-7B, Qwen2.5-14B, and DeepSeek-R1-Distill-Qwen-7B contain 30 prompts (six folds, 5,760 token rows). Each token row is evaluated out of fold exactly once, and verification rates are pooled across these rows. These are the same 192-token rows as Appendix B; holding out prompts changes the threshold, not the flip count. Llama-3.1-8B covers 8/8 held-out flips and Qwen2.5-14B covers 25/25, so the two-sided 95% Wilson lower bounds are 67.6% and 86.7% (Table 22). The intervals treat flips as independent and omit dependence inside a prompt. This row is local flip coverage on that trace set. Sequence match on the separate 120-prompt, 1024-token workload is reported in §4.3.

Qwen2.5-14B covers all 25 held-out flips at 4.38% verification on these traces. Qwen2.5-7B verifies 67.31% of steps and misses one of 61 flips. DeepSeek-R1-Distill-Qwen-7B verifies 88.84% of steps and still misses one of 98 flips. Margin stays cheap only when $A(\tau_R)$ is small. Appendix C uses this same 30-prompt DeepSeek-R1-Distill-Qwen-7B trace.

4.6 Full-Vocabulary Perturbation Envelope

The 2ε rule is a sufficient condition on two logit vectors. We teacher-force one shared prefix through a batch-1 reference and a batched SDPA run on the first 32 MATH500 prompts and 64 steps of Qwen2.5-7B and Qwen2.5-14B. Prompts 0–15 estimate ε, and prompts 16–31 are held out, giving 1,024 steps. Llama-3.1-8B weights were not available for this check. At homogeneous B=8, the

calibration median is 0.31. Every recorded flip has batched margin at most 2ε. Verifying at $2\hat{\varepsilon}_{\mathrm{p99}}$ covers all 6 held-out flips on Qwen2.5-7B and all 7 on Qwen2.5-14B, at 17.7% and 32.5% of those steps. The serving thresholds $\tau{=}8$ and $\tau{=}4$ also cover those flips, at 50.2% and 36.8%. About 1% of held-out steps exceed the calibration 99th percentile. Section 4.3 reports the 120-sequence match on a separate HF eager run, so this envelope does not bound that run. Appendix A.6 gives the calibration ε in Table 15, together with the CDF, position, and top-50 checks.

5 Discussion

Llama-3.2-1B, Llama-3.1-8B, and Qwen2.5-14B on MATH500 and GSM8K cover every synchronous flip while selecting 2.4–6.4% of stable steps (Table 5). Qwen2.5-7B on MATH500 moves from 3.6% at 90% coverage to 39.2% at complete coverage. Qwen2.5-14B needs 3.3% of stable MATH500 steps for complete coverage and 29.3% on HumanEval. DeepSeek-R1-Distill-Qwen-7B on MATH500 moves from 40.9% to 88.4%. These cells are $A(\tau_R)$, the fraction of token-stable steps under the covering threshold. They sit apart from the 0.3–1.3% synchronous flip rate, which is why a small flip rate can still leave verification frequent.

The $120/120$ match is the Qwen2.5-14B MATH500 selection set, and $\tau = 4.0$ is chosen there. GSM8K, SharedGPT, and HumanEval then match $119/120$, $113/120$, and $118/120$. SharedGPT's sequence-match 95% Wilson interval is 88.5–97.2%. A missed flip can change the suffix, so sequence match on a 1024-token output is stricter than flip recall on the same run. Llama-3.1-8B transfer, at $120/120$, $118/120$, and $117/120$, stays in Appendix A.4.

On the HF eager pass, rollback costs $+75.2\%$ against the 70.15 second BF16 mean, and $+23.9\%$ is that same pass with single-token cache catch-up removed. LLM-42 costs $+47.5\%$ against its vLLM BF16 mean of 24.95 seconds, and Batch-Invariant Ops costs $+129\%$ against its SGLang BF16 mean of 22.67 seconds. Those two timing logs do not record sequence match. Each overhead uses the BF16 time of its own stack.

Oracle repair recovers sequence match with fewer than 2.4 column replacements per sequence for $B \in \{2, 4, 8, 16\}$. It copies the deterministic-reference column at known flips, and no log compares those columns with the online verifier. Verifying at $2\hat{\varepsilon}_{\mathrm{p99}}$ covers $6/6$ and $7/7$ held-out flips at 17.7% and 32.5% of the SDPA steps, which is a sufficient condition on that teacher-forced trace. Appendix C is a separate DeepSeek-R1-Distill-Qwen-7B diagnostic and leaves the 88.4% MATH500 cell unchanged.

6 Conclusion

Batch-induced token flips show that greedy BF16 output depends on the batch even at temperature zero. Those flips occupy 0.3–1.3% of synchronous steps, and a gate that sees only the current forward cannot tell which steps flipped. The verification rate follows $A(\tau_R)$, the fraction of token-stable steps under the margin threshold that covers the flips. At complete coverage this fraction is 3.3% on Qwen2.5-14B MATH500 and 88.4% on DeepSeek-R1-Distill-Qwen-7B. MarginGate reads the top-1/top-2 margin the BF16 forward already produced, commits a high-margin step, and sends a low-margin step to a verifier that reads the committed cache. A match keeps the tentative column. A mismatch replaces only the current column.

On the HF eager Qwen2.5-14B MATH500 selection set, the rule matches $120/120$ batch-1 sequences, against $6/120$ for unguarded BF16, at a 15.05% verification rate. τ is chosen on that set. GSM8K, SharedGPT, and HumanEval then match $119/120$, $113/120$, and $118/120$. The measured cost is the rollback time of this HF eager run. Where $A(\tau_R)$ is large, as on DeepSeek-R1-Distill-Qwen-7B, the same rule keeps verification frequent.

AI use statement

We used generative AI to organize the manuscript and edit language. The authors reviewed those edits and are responsible for the final content.

Reproducibility Statement

Algorithm 1 states the decode rule. Section 4.1 lists the checkpoints, the three executions, and how τ is selected. The timed HF eager run and the fixed-threshold transfer are in §4.3 and §4.4. Appendix E points to the flip counts, the held-out folds, the envelope protocol, and the cells marked provisional.

References

Amey Agrawal, Nitin Kedia, Ashish Panwar, Jayashree Mohan, Nipun Kwatra, Bhargav Gulavani, Alexey Tumanov, and Ramachandran Ramjee. Taming Throughput-Latency tradeoff in LLM inference with Sarathi-Serve. In *18th USENIX Symposium on Operating Systems Design and Implementation (OSDI 24)*, pp. 117–134. USENIX Association, 2024. URL https://www.usenix.org/conference/osdi24/presentation/agrawal.

Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebrón, and Sumit Sanghai. GQA: Training generalized multi-query transformer models from multi-head checkpoints. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 4895–4901. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.298. URL https://aclanthology.org/2023.emnlp-main.298/.

Charlie Chen, Sebastian Borgeaud, Geoffrey Irving, Jean-Baptiste Lespiau, Laurent Sifre, and John Jumper. Accelerating large language model decoding with speculative sampling. *arXiv preprint arXiv:2302.01318*, 2023.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing GPT-4 with 90%* ChatGPT quality. https://lmsys.org/blog/2023-03-30-vicuna/, 2023.

Kexin Chu, Zixu Shen, Sheng-Ru Cheng, Dawei Xiang, Ziqin Liu, and Wei Zhang. MCaM: Efficient LLM inference with multi-tier KV cache management. In *Proceedings of the 45th IEEE International Conference on Distributed Computing Systems*, pp. 571–581, 2025a. doi: 10.1109/ICDCS63083.2025.00062.

Kexin Chu, Dawei Xiang, Zixu Shen, Yiwei Yang, Zecheng Liu, and Wei Zhang. Dynamic expert quantization for scalable mixture-of-experts inference. arXiv:2511.15015, 2025b. URL https://arxiv.org/abs/2511.15015.

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. *arXiv preprint arXiv:2110.14168*, 2021.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

David Goldberg. What every computer scientist should know about floating-point arithmetic. *ACM Computing Surveys*, 23(1):5–48, 1991.

Raja Gond, Aditya K. Kamath, Ramachandran Ramjee, and Ashish Panwar. LLM-42: Enabling determinism in LLM inference with verified speculation. arXiv:2601.17768, 2026. URL https://arxiv.org/abs/2601.17768.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. MiniLLM: Knowledge distillation of large language models. In *International Conference on Learning Representations*, 2024.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and DeepSeek-AI. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645:633–638, 2025. doi: 10.1038/s41586-025-09422-z.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Nicholas J. Higham. *Accuracy and Stability of Numerical Algorithms*. SIAM, 2nd edition, 2002.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, pp. 611–626, 2023.

Yaniv Leviathan, Matan Kalman, and Yossi Matias. Fast inference from transformers via speculative decoding. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 19274–19286. PMLR, 2023. URL https://proceedings.mlr.press/v202/leviathan23a.html.

Meta AI. The Llama 3 herd of models. https://arxiv.org/abs/2407.21783, 2024.

SGLang Project. Deterministic inference. https://sgl-project.github.io/advanced_features/deterministic_inference.html, 2026.

Zixu Shen, Kexin Chu, Yifan Zhang, Dawei Xiang, Runxin Wu, and Wei Zhang. ExpertFlow: Adaptive expert scheduling and memory coordination for efficient MoE inference. arXiv:2510.26730, 2025. URL https://arxiv.org/abs/2510.26730.

Yifan Song, Guoyin Wang, Sujian Li, and Bill Yuchen Lin. The good, the bad, and the greedy: Evaluation of LLMs should not ignore non-determinism. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 4195–4206. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.naacl-long.211. URL https://aclanthology.org/2025.naacl-long.211/.

Thinking Machines Lab. batch_invariant_ops: Batch-invariant implementations of reduction operations. https://github.com/thinking-machines-lab/batch_invariant_ops, 2026.

vLLM Project. Batch invariance. https://docs.vllm.ai/en/v0.19.0/features/batch_invariance/, 2026.

Gyeong-In Yu, Joo Seong Jeong, Geon-Woo Kim, Soojeong Kim, and Byung-Gon Chun. Orca: A distributed serving system for Transformer-based generative models. In *16th USENIX Symposium on Operating Systems Design and Implementation (OSDI 22)*, pp. 521–538, 2022.

Jiayi Yuan, Hao Li, Xinheng Ding, Wenya Xie, Yu-Jhe Li, Wentian Zhao, Kun Wan, Jing Shi, Xia Hu, and Zirui Liu. Understanding and mitigating numerical sources of nondeterminism in LLM inference. In *Advances in Neural Information Processing Systems*, volume 38, pp. 188045–188077, 2025. doi: 10.52202/085713-5653. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/f80094a824ba5912d4a2de169c404a40-Abstract-Conference.html.

Ziyang Zhang, Xinheng Ding, Jiayi Yuan, Rixin Liu, Huizi Mao, Jiarong Xing, and Zirui Liu. Deterministic inference across tensor parallel sizes that eliminates training-inference mismatch. arXiv:2511.17826, 2025. URL https://arxiv.org/abs/2511.17826.

Lianmin Zheng, Liangsheng Yin, Zhiqiang Xie, Chuyue Sun, Jeff Huang, Cody Hao Yu, Shiyi Cao, Christos Kozyrakis, Ion Stoica, Joseph E. Gonzalez, Clark Barrett, and Ying Sheng. SGLang: Efficient execution of structured language model programs. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-2000.

Yinmin Zhong, Shengyu Liu, Junda Chen, Jianbo Hu, Yibo Zhu, Xuanzhe Liu, Xin Jin, and Hao Zhang. DistServe: Disaggregating prefill and decoding for goodput-optimized large language model serving. In *18th USENIX Symposium on Operating Systems Design and Implementation (OSDI 24)*, pp. 193–210. USENIX Association, 2024. URL https://www.usenix.org/conference/osdi24/presentation/zhong-yinmin.

A Operating Points and Additional Serving Results

Table 9 estimates $\varepsilon_{\text{pert}}^{(50)}$ on the MATH500 selection prompts from the protected request's top-50 logits. It is not an ℓ_∞ bound. The quantity $2 \cdot \max(\varepsilon_{\text{pert}}^{(50)})$ only sets the sweep range in Table 10. τ_{100} is the smallest tested threshold with no observed mismatch, then held fixed. A dagger marks a cell that is not a completed public-API rollback measurement: sequence match when the latency run did not confirm agreement, and overhead only when that latency is itself provisional. Those cells are excluded from the main claims.

Table 9: Empirical top-50 perturbation scale and selected threshold for each model. The sweep scale is $2 \cdot \max(\varepsilon_{\text{pert}}^{(50)})$.

Model	med. $\varepsilon^{(50)}$	max $\varepsilon^{(50)}$	sweep scale	selected τ_{100}
Llama-3.2-1B	0.25	0.52	1.03	$1.0^\dagger$
Llama-3.1-8B	0.33	0.91	1.81	$4.0^\dagger$
Qwen2.5-7B	1.66	4.21	8.42	$8.0^\dagger$
Qwen2.5-14B	0.50	1.13	2.26	4.0
DeepSeek-R1-Distill-Qwen-7B	0.51	3.88	7.75	$16.0^\dagger$

A.1 Full Threshold Sweep

Table 10 lists every tested threshold. Sequence-match rate is agreement with the deterministic reference, and overhead is relative to BF16. Bold rows are the selected points. Every Llama-3.2-1B match rate is provisional, as is its selected overhead. Llama-3.1-8B and Qwen2.5-7B have provisional full-match cells where the overhead was measured. DeepSeek-R1-Distill-Qwen-7B's selected verification rate, full match, and overhead are provisional, and that overhead exceeds LLM-42.

Table 10: Full MATH500 threshold sweep for the five evaluated models.

τ	Verification rate	Sequence-match rate	Overhead
Llama-3.2-1B	*(LLM-42 =+69.9%)*		
0.25	1.22%	85.0%†	+3.7%
0.5	2.62%	99.2%†	+4.9%
1.0	**4.90%**	**100.0%**†	**+9.3%**†
2.0	8.17%	100.0%†	+16.2%
Llama-3.1-8B	*(LLM-42 =+64.6%)*		
0.5	4.14%	97.5%	+6.9%
1.0	7.41%	98.3%	+11.9%
2.0	11.75%	98.3%	+18.5%
4.0	**18.56%**	**100.0%**†	**+29.0%**
Qwen2.5-7B	*(LLM-42 =+50.7%)*		
1.0	7.21%	75.0%†	+12.8%
2.0	12.07%	85.8%†	+21.0%
4.0	18.64%	95.8%†	+32.0%
8.0	**29.15%**	**100.0%**†	**+49.8%**
Qwen2.5-14B	*(LLM-42 =+47.5%)*		
1.0	4.97%	99.2%	+8.1%
2.0	8.95%	99.2%	+14.3%
4.0	**15.05%**	**100.0%**	**+23.9%**
8.0	24.60%	100.0%	+38.9%
DeepSeek-R1-Distill-Qwen-7B	*(LLM-42 =+72.0%)*		
1.0	12.63%	48.3%	+21.3%
2.0	19.69%	75.0%	+28.2%
4.0	27.44%	90.8%	+37.1%
8.0	37.08%	97.5%	+51.8%
16.0	**49.50%**†	**100.0%**†	**+84.5%**†

A.2 Additional Model Operating Points

Table 11 is the B=8 MATH500 latency for every model except Qwen2.5-14B, which is Table 6. The measured overheads exclude the daggered cells: Llama-3.2-1B's +9.3%, and DeepSeek-R1-Distill-Qwen-7B's 49.50% verification rate and +84.5% overhead. Sequence match for Llama-3.1-8B and Qwen2.5-7B remains the provisional full-match row in Table 10.

Table 11: MATH500 latency at B=8 on one A6000 for the models other than Qwen2.5-14B. † marks provisional entries.

Model	τ	Verification rate	BF16 (s)	Batch-Invariant	LayerCast	LLM-42	MarginGate
Llama-3.2-1B	1.0	4.90%	2.22	+119%	+133.4%	+69.9%	+9.3%†
Llama-3.1-8B	4.0	18.56%	12.27	+143%	+354.3%	+64.6%	+29.0%
Qwen2.5-7B	8.0	29.15%	12.79	+129%	+346.9%	+50.7%	+49.8%
DeepSeek-R1-Distill-Qwen-7B	16.0	49.50%†	23.06	+133%	+345.1%	+72.0%	+84.5%†

A.3 Batch-Size Scaling

Table 12 extends the comparison from B=8 to B=64. Sequence match at $B \neq 8$ is not reported. BF16 seconds are the SGLang baseline. LLM-42 overheads use vLLM BF16, and MarginGate's +23.9% excludes HF eager cache catch-up, so these percentages are not multiples of the BF16 column. The LayerCast percentages in Table 11 come from a vLLM run of 10 prompts and 192 tokens.

Table 12: Batch-size scaling on MATH500. BF16 seconds are the SGLang baseline. LLM-42 overheads are relative to vLLM BF16, and MarginGate's $+23.9\%$ excludes HF eager cache catch-up.

	Llama-3.1-8B			Qwen2.5-14B		
B	BF16 (s)	LLM-42	MarginGate	BF16 (s)	LLM-42	MarginGate
8	12.27	$+64.6\%$	$+29.0\%$	22.67	$+47.5\%$	$+23.9\%$
16	13.10	$+69.0\%$	$+45.4\%$	23.96	$+57.9\%$	$+36.6\%$
32	13.98	$+73.8\%$	$+61.6\%$	25.96	$+65.5\%$	$+50.0\%$
64	16.90	$+106.5\%$	$+73.6\%$	30.48	$+122.7\%$	$+69.7\%$

A.4 Complete Transfer Results

Thresholds are selected on MATH500 and held fixed. Llama-3.1-8B matches $120/120$, $118/120$, and $117/120$ transfer requests and is slower than LLM-42 on SharedGPT. Its MATH500 match remains provisional.

Table 13: Transfer with the MATH500 threshold held fixed. Each row has 120 requests. BF16 seconds are the SGLang baseline; LLM-42 uses vLLM BF16, and MarginGate's $+23.9\%$ excludes HF eager cache catch-up.

Dataset	BF16 (s)	LLM-42	MarginGate	BF16 match	MG match	Verify
Llama-3.1-8B, $\tau = 4.0$						
MATH500	12.27	$+64.6\%$	$+29.0\%$	$7/120$	$120/120^{\dagger}$	18.56%
GSM8K	5.71	$+123.3\%$	$+30.3\%$	$12/120$	$120/120$	19.3%
SharedGPT	13.38	$+24.6\%$	$+36.8\%$	$5/120$	$118/120$	23.5%
HumanEval	12.66	$+38.3\%$	$+28.0\%$	$8/120$	$117/120$	18.0%
Qwen2.5-14B, $\tau = 4.0$						
MATH500	22.67	$+47.5\%$	$+23.9\%$	$6/120$	$120/120$	15.05%
GSM8K	12.04	$+49.0\%$	$+33.5\%$	$5/120$	$119/120$	21.16%
SharedGPT	19.29	$+44.0\%$	$+40.4\%$	$2/120$	$113/120$	25.58%
HumanEval	20.67	$+32.5\%$	$+29.9\%$	$2/120$	$118/120$	19.38%

On the transfer workloads in Table 13, the two-sided 95% Wilson intervals are 96.9–100.0% (GSM8K), 94.1–99.5% (SharedGPT), and 92.9–99.2% (HumanEval) for Llama-3.1-8B, and 95.4–99.9% (GSM8K), 88.5–97.2% (SharedGPT), and 94.1–99.5% (HumanEval) for Qwen2.5-14B.

A.5 Heterogeneous Batches

Mixed MATH500 batches use left padding and per-row position IDs, with each content sequence starting at position 0. Recall at $\tau{=}0.25$ and $\tau{=}0.5$ is a synchronous diagnostic; the selected threshold is $\tau{=}4.0$. The match rows are a separate run of 120 requests in 30 batches of four. They copy the reference KV column. No log compares that column with the on-demand verifier, and these rows are not the latency in §4.

Table 14: Llama-3.1-8B on heterogeneous MATH500 batches. Recall rows sit below τ=4.0. Match rows copy the reference column. Verification and repair share the 23,040-step denominator; misses are a count.

Metric	Homogeneous	Heterogeneous
Synchronous flip rate	0.48%	0.53%
Recall @ $\tau = 0.25$, diagnostic	99%	95%
Recall @ $\tau = 0.5$, diagnostic	100%	100%
BF16 matches, no repair	–	42/120
MarginGate matches, oracle repair	–	120/120
Verification, share of 23,040 steps	–	39.2%
Repair, share of 23,040 steps	–	0.51%
Missed disagreements, count	–	0

A.6 Perturbation Envelope Figures

The split, the median ε=0.31, and the statement that this envelope does not bound the HF eager run are in §4.6. Table 15 records the calibration percentiles and the held-out coverage.

Table 15: Homogeneous B=8 SDPA envelope: held-out token flips on 1,024 steps.

	Calibration ε		$2\hat{\varepsilon}_{\text{p99}}$		Serving τ	
Model	p99	max	held-out flips	verify	τ	verify
Qwen2.5-7B	1.00	1.25	6/6	17.7%	8	50.2%
Qwen2.5-14B	1.74	3.60	7/7	32.5%	4	36.8%

Figures 5–7 are the checks omitted from Table 15. On the B=8 calibration split, the maximizing coordinate falls outside the reference top-50 on 59.7% of Qwen2.5-7B steps and 66.7% of Qwen2.5-14B steps. The top-50 99th percentiles are 0.88 and 1.24, against 1.00 and 1.74 over the full vocabulary. Heterogeneous B=8 batches have 99th percentiles 1.00 and 1.88.

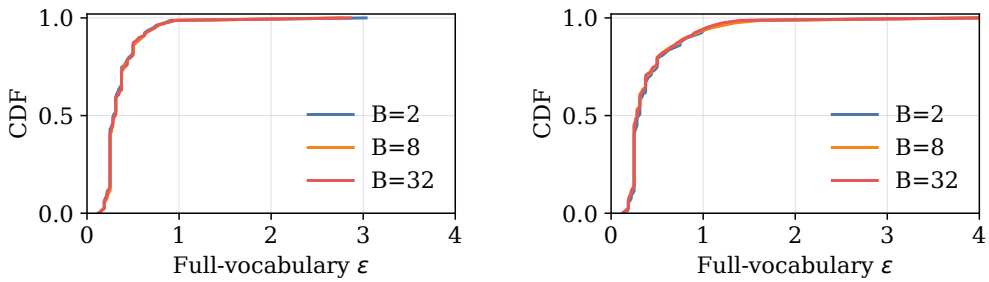


Figure 5: Full-vocabulary ε CDF on homogeneous batches, Qwen2.5-7B (left) and Qwen2.5-14B (right). Colors are batch sizes $B \in \{2, 8, 32\}$. The horizontal axis stops at 4.

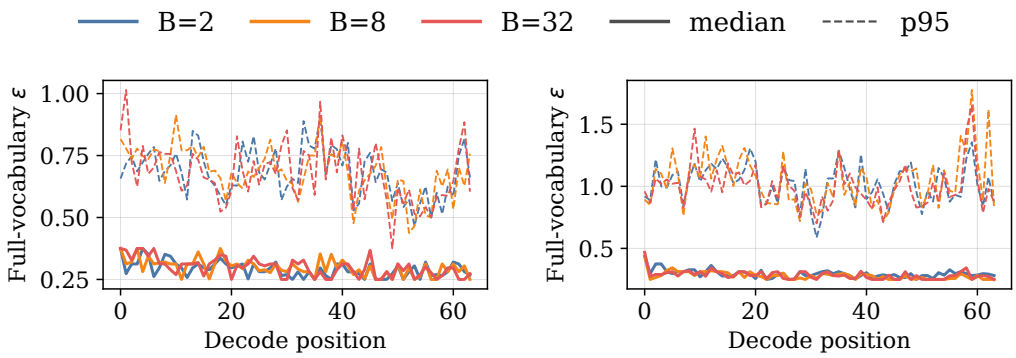


Figure 6: Full-vocabulary ε against decode position, Qwen2.5-7B (left) and Qwen2.5-14B (right). Colors are batch sizes; solid lines are medians and dashed lines are 95th percentiles.

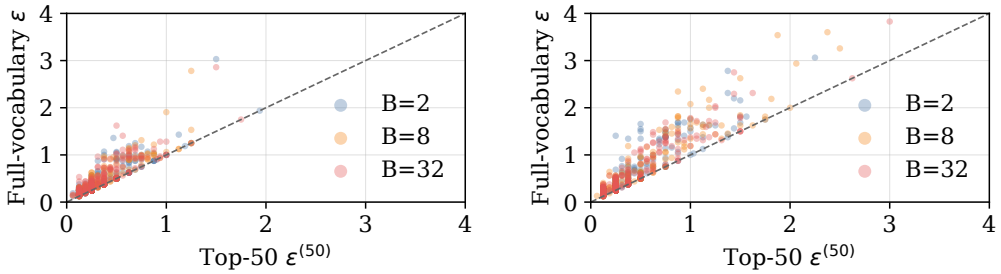


Figure 7: Top-50 perturbation against the full-vocabulary perturbation, Qwen2.5-7B (left) and Qwen2.5-14B (right). Colors are batch sizes; the dashed line is $y{=}x$.

B Additional Evidence for the Margin Signal

B.1 Raw Flip Counts and Uncertainty

Table 16 gives the counts and Wilson intervals behind Table 1.

Table 16: Raw synchronous token-flip counts and two-sided 95% Wilson intervals for the rates summarized in Table 1.

Model	Benchmark	Flips / synchronous steps	Rate (95% CI)
Llama-3.2-1B	MATH500	58/16,003	0.36% (0.28–0.47%)
	GSM8K	64/14,336	0.45% (0.35–0.57%)
	HumanEval	75/12,786	0.59% (0.47–0.73%)
Llama-3.1-8B	MATH500	68/14,082	0.48% (0.38–0.61%)
	GSM8K	54/16,655	0.32% (0.25–0.42%)
	HumanEval	71/13,997	0.51% (0.40–0.64%)
Qwen2.5-7B	MATH500	87/12,933	0.67% (0.55–0.83%)
	GSM8K	91/10,543	0.86% (0.70–1.06%)
	HumanEval	108/8,483	1.27% (1.06–1.53%)
Qwen2.5-14B	MATH500	72/15,190	0.47% (0.38–0.60%)
	GSM8K	76/12,508	0.61% (0.49–0.76%)
	HumanEval	91/10,862	0.84% (0.68–1.03%)
DeepSeek-R1-Distill-Qwen-7B	MATH500	109/9,375	1.16% (0.96–1.40%)
	GSM8K	111/10,874	1.02% (0.85–1.23%)
	HumanEval	110/10,129	1.09% (0.90–1.31%)

B.2 NEAR-TOP LOGIT COUNTS

Table 17 reports the mean number of logits within η of the top-1. The η=0.5 columns are the slice cited from the main text. $N(\eta) > 1$ exactly when the margin is at most η; the mean count is a separate summary.

Table 17: Mean number of logits within η of the top-1 logit at stable and flip positions. Ratio is the flip mean divided by the stable mean.

	$\eta = 0.25$			$\eta = 0.5$			$\eta = 1.0$			$\eta = 2.0$		
Model	Stable	Flip	Ratio	Stable	Flip	Ratio	Stable	Flip	Ratio	Stable	Flip	Ratio
Llama-3.2-1B	1.06	2.21	2.08×	1.12	2.59	2.31×	1.25	2.95	2.36×	1.61	4.81	2.99×
Llama-3.1-8B	1.07	2.13	2.00×	1.12	2.24	1.99×	1.25	2.66	2.13×	1.54	3.72	2.42×
Qwen2.5-7B	1.04	1.86	1.80×	1.07	2.05	1.92×	1.14	2.34	2.06×	1.31	3.02	2.31×
Qwen2.5-14B	1.04	2.17	2.09×	1.07	2.31	2.16×	1.14	2.71	2.38×	1.30	3.65	2.81×
DeepSeek-R1-Distill-Qwen-7B	1.05	1.78	1.70×	1.09	1.99	1.82×	1.19	2.29	1.93×	1.43	3.12	2.18×

B.3 ALTERNATIVE RISK SIGNALS

Each score is computed from the current BF16 top-20 logits. The matched rate in the recall column is the raw-margin verification rate that covers every observed flip. Zero-miss is the rate that score needs to cover all of them. Both runs consume the batch-1 token, including after an earlier disagreement. Table 22 uses these same rows and changes only the threshold, by holding out prompts. Among training-free scores, raw margin has the lowest zero-miss rate on Llama-3.1-8B and Qwen2.5-14B. On Qwen2.5-7B, margin normalized by the top-20 width reaches zero-miss at 68.4%, against 73.4% for raw margin; prompt-level HGB reaches 43.8%. On DeepSeek-R1-Distill-Qwen-7B, margin normalized by the top-20 standard deviation reaches 87.7%, against 94.0% for raw margin. Those HGB rates are cross-validation. The in-sample rates 10.2% and 7.9% in Appendix C are a different summary.

Table 18: Risk-signal comparison on Llama-3.1-8B: 10 prompts, 1,920 token rows, and 8 flips.

Score	AUC	Recall @ 2.9%	Zero-miss rate
Raw margin	0.988	100.0%	2.9%
Normalized / width	0.983	87.5%	5.1%
Normalized / std.	0.984	75.0%	4.6%
Top-20 entropy	0.910	37.5%	18.2%
Cluster within 0.5	0.967	50.0%	8.1%
Cluster within 1.0	0.942	37.5%	15.3%
Top-1/top-3 gap	0.815	37.5%	50.4%
HGB, prompt CV	0.867	87.5%	98.2%
Random (expected)	0.500	2.9%	100.0%

Table 19: Risk-signal comparison on Qwen2.5-14B: 30 prompts, 5,760 token rows, and 25 flips.

Score	AUC	Recall @ 4.4%	Zero-miss rate
Raw margin	0.984	100.0%	4.4%
Normalized / width	0.983	96.0%	6.0%
Normalized / std.	0.983	96.0%	6.5%
Top-20 entropy	0.934	44.0%	14.5%
Cluster within 0.5	0.970	68.0%	7.4%
Cluster within 1.0	0.952	56.0%	12.3%
Top-1/top-3 gap	0.888	52.0%	41.3%
HGB, prompt CV	0.937	96.0%	99.8%
Random (expected)	0.500	4.4%	100.0%

Table 20: Risk-signal comparison on Qwen2.5-7B: 30 prompts, 5,760 token rows, and 61 flips.

Score	AUC	Recall @ 73.4%	Zero-miss rate
Raw margin	0.944	100.0%	73.4%
Normalized / width	0.927	100.0%	68.4%
Normalized / std.	0.927	100.0%	68.9%
Top-20 entropy	0.925	98.4%	75.2%
Cluster within 0.5	0.882	96.7%	100.0%
Cluster within 1.0	0.881	96.7%	100.0%
Top-1/top-3 gap	0.903	98.4%	77.0%
HGB, prompt CV	0.941	100.0%	43.8%
Random (expected)	0.500	73.4%	100.0%

Table 21: Risk-signal comparison on DeepSeek-R1-Distill-Qwen-7B: 30 prompts, 5,760 token rows, and 98 flips.

Score	AUC	Recall @ 94.0%	Zero-miss rate
Raw margin	0.923	100.0%	94.0%
Normalized / width	0.906	100.0%	88.2%
Normalized / std.	0.906	100.0%	87.7%
Top-20 entropy	0.879	99.0%	95.7%
Cluster within 0.5	0.850	99.0%	100.0%
Cluster within 1.0	0.867	99.0%	100.0%
Top-1/top-3 gap	0.852	99.0%	96.7%
HGB, prompt CV	0.904	100.0%	88.1%
Random (expected)	0.500	94.0%	100.0%

Table 22: Held-out margin calibration on the 192-token MATH500 traces (10 prompts for Llama; 30 for Qwen and DeepSeek-R1-Distill-Qwen-7B).

Model	Prompts	Held-out flips	Recall (95% CI)	Verification rate
Llama-3.2-1B	10	8/8	100.0% (67.6–100.0%)	2.66%
Llama-3.1-8B	10	8/8	100.0% (67.6–100.0%)	2.86%
Qwen2.5-7B	30	60/61	98.4% (91.3–99.7%)	67.31%
Qwen2.5-14B	30	25/25	100.0% (86.7–100.0%)	4.38%
DeepSeek-R1-Distill-Qwen-7B	30	97/98	99.0% (94.4–99.8%)	88.84%

B.4 Additional Model Families

Granite-3.3-2B-Instruct and Mistral-7B-Instruct-v0.3 use the same 192-token MATH500 traces. Zero-miss is the margin rule's verification rate at complete flip coverage.

Table 23: Risk-signal profiles for two additional model families. Zero-miss rate is the smallest verification rate covering all observed flips; stable near-tie rate is the fraction of stable tokens below the selected threshold.

Model	Flip rate	Zero-miss rate	Stable near-tie rate
Granite-3.3-2B-Instruct	0.52%	3.8%	3.2%
Mistral-7B-Instruct-v0.3	0.70%	6.4%	5.7%

C Exploratory Learned-Gate Analysis for DeepSeek-R1-Distill-Qwen-7B

This comparison is not part of MarginGate. Six-fold prompt-level cross-validation on 30 prompts holds out five prompts per fold. Near 95% recall, the held-out gates miss 4 of 98 flips with top-K logits and 5 of 100 flips when KV summaries are added. In-sample, HGB lowers zero-miss from 94.0% to 10.2% and from 93.1% to 7.9%. Those two rates measure separability on the training examples. The traces were collected separately, so the flip counts and the scalar-margin rates differ. "CV rate @ 95% recall" is the held-out verification rate at approximately 95% flip recall.

Table 24: DeepSeek-R1-Distill-Qwen-7B learned-gate results with top-K logit features: 30 prompts, 5,760 token rows, and 98 flips.

Gate	In-sample zero-miss	CV zero-miss	CV rate @ 95% recall	Misses of 98
Scalar margin	94.0%	–	–	–
Logistic reg.	74.0%	98.6%	42.0%	4
HGB-small	10.2%	87.7%	40.5%	4
RF-small	13.1%	97.8%	38.1%	4

Table 25: DeepSeek-R1-Distill-Qwen-7B learned-gate results with top-K logits and KV state summaries: 30 prompts, 5,760 token rows, and 100 flips.

Gate	In-sample zero-miss	CV zero-miss	CV rate @ 95% recall	Misses of 100
Scalar margin	93.1%	–	–	–
Logistic reg.	43.3%	98.1%	34.0%	5
HGB-small	7.9%	98.2%	49.5%	5
RF-small	8.5%	100.0%	36.1%	5

D Additional KV Cache Diagnostics

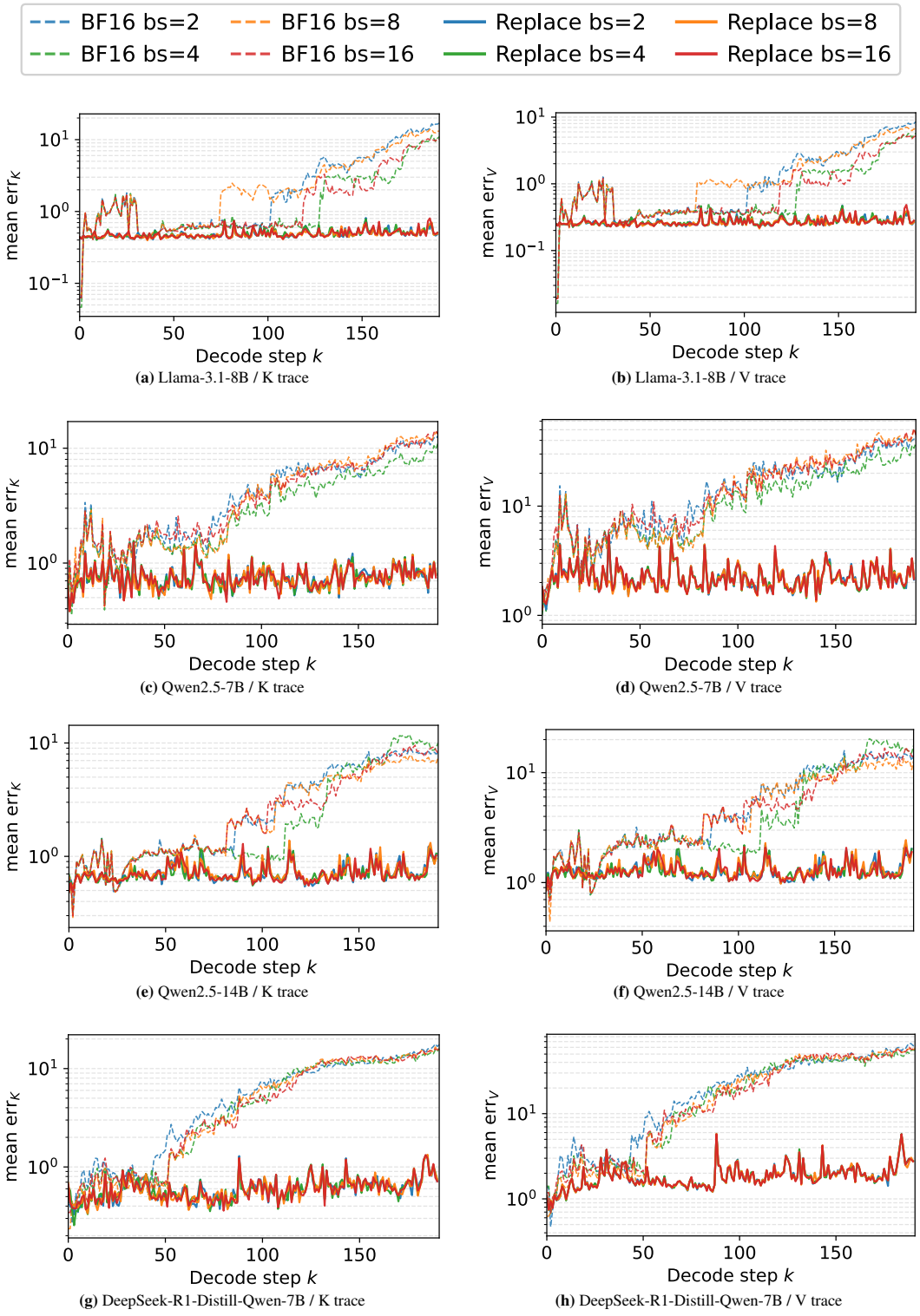
D.1 Oracle Repair Traces

The oracle uses the true token-flip positions. Figure 4 counts the repairs that recover sequence match. Figure 8 isolates current-column repair and is not the serving rollback. With repair, K and V remain at the pre-flip floor for $B \in \{2, 4, 8, 16\}$.

E Reproducibility Checklist

The items below locate the recorded protocol. This manuscript does not include an anonymous code archive.

- **Models.** The five checkpoints and their short names are in §4.1.
- **Decode.** Runs are greedy BF16 with a 1024-token cap. Sampling, beam search, and speculative decoding are outside the evaluation.
- **Data and threshold.** MATH500 selects τ. GSM8K, SharedGPT, and HumanEval keep that threshold fixed (§4.4).
- **Batches.** Unless noted, a batch repeats one prompt. Mixed MATH500 batches are in Appendix A.5.
- **Metrics.** Synchronous flip rates use the prefix in §2. Sequence match compares a protected request with a batch-1 greedy trajectory. The online verifier reads the committed cache.
- **Compute.** The HF eager timing pass uses one A6000 at B=8 (§4.3). Other stacks report their own BF16 times.



(a) Llama-3.1-8B / K trace

(b) Llama-3.1-8B / V trace

(c) Qwen2.5-7B / K trace

(d) Qwen2.5-7B / V trace

(e) Qwen2.5-14B / K trace

(f) Qwen2.5-14B / V trace

(g) DeepSeek-R1-Distill-Qwen-7B / K trace

(h) DeepSeek-R1-Distill-Qwen-7B / V trace

Figure 8: Oracle current-column repair traces for four models and four batch sizes. Solid curves apply oracle repair; dashed curves show unrepaired BF16 trajectories. The y-axis is logarithmic.

- **Uncertainty.** Raw flip counts and Wilson intervals are in Appendix B. Held-out margin folds are in Table 22.
- **Boundaries.** Cells marked † are provisional. The SDPA envelope in Appendix A.6 is not a serving guarantee. Oracle repair in Appendix D.1 copies the deterministic-reference column and is not the timed rollback.