GATED DELTA HYPER-CONNECTIONS: SEPARATING RESIDUAL MIXING FROM STATE RETENTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Hyper-Connections (HC) expand Transformer residual pathways into multiple streams, offering a scaling dimension beyond model depth and width. Manifold-Constrained Hyper-Connections (mHC) stabilize cross-stream mixing, but mixing alone does not control how much of the carried state is retained: under its doubly stochastic constraint, the stream-average component is left unattenuated as layers add new outputs. Our measurements on trained models show that this component grows faster across depth than the remaining state, while write gates become more concentrated and lose mass during training. These observations motivate **Gated Delta Hyper-Connections (GDHC)**, which adds explicit retention control alongside the existing read/write maps by combining global decay with write-aligned erasure. Decay regulates overall retention, while erasure selectively attenuates old content along the incoming write direction, independently of write strength. Experiments with mHC and identity-mixer iHC show lower training loss at both 4B- and 7B-parameter mixture-of-experts models, with 0.55B and 0.91B active parameters, respectively, and higher average downstream scores at the larger scale. Component ablations show further gains from combining decay and erasure. GDHC also flattens activation profiles and increases median write-gate mass; common-mode RMS decreases across depth on A1B mHC and becomes nearly flat on iHC. Together, these results support explicit retention as a complementary control in multi-stream residual architectures.

1 INTRODUCTION

Residual connections (He et al., 2016) make deep networks trainable through a simple operation: carry the current representation forward and add the next layer's output. In Transformers, this lets attention and MLP blocks build on earlier computations across depth. Beyond increasing model depth or width, recent work explores the residual pathway itself as a scaling dimension (Pagliardini et al., 2024; Menghani et al., 2025). Hyper-Connections (HC) (Zhu et al., 2025a) expand this pathway into multiple streams, with learned maps that read, mix, and write their contents without widening the attention or MLP blocks. Stable propagation through repeated mixing is essential to this design. Manifold-Constrained Hyper-Connections (mHC) (Xie et al., 2026) address it by constraining the mixer to limit amplification across depth. Expanded residual pathways have also reached industrial-scale language models, including DeepSeek-V4 (DeepSeek-AI, 2026), Qwen3.8-Next (Qiu et al., 2026), GLM-5.3-Flash (Z.ai, 2026), and Hy4 (Tencent Hy Team, 2026).

Stable mixing leaves a second question open: *how much old state should remain when each layer adds new content?* An exactly doubly stochastic mixer preserves the average across residual streams, even though it can contract differences between them. Contributions to this stream-average component therefore pass through subsequent mixers without attenuation. Removing such content requires later layers to write an offsetting contribution through the same pathway used to introduce new information, coupling removal of old state with addition of new state. Thus, as in an ordinary additive residual connection, the carried state has no independent retention control separate from mixing and writing. This motivates extending the residual update with an explicit retention mechanism (Section 2).

Measurements in trained models make this question concrete. In 18- and 30-layer mHC models, layer-input absolute maxima grow by factors of 6.1 and 13.8 across depth; direct decomposition on

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.

A1B locates faster relative growth in the stream-average component (Figure 1). Meanwhile, read-gate allocation broadens during training, whereas write gates become more concentrated and lose total weight (Figure 2). Activation scale also affects the balance between retained state and new content: a fixed-size write makes a smaller relative change to a larger state. These observations motivate controlling the old state directly while allowing each layer to choose its new write independently.

Gated Delta Hyper-Connections (GDHC) provides this control through two complementary operations. An input-dependent scalar **decay** regulates overall retention, including the component preserved by mixing. Rank-one **erasure** then permits additional attenuation along the stream-space direction selected by the incoming write. Combining the two allows gentle global decay and stronger revision in the selected direction, while orthogonal components receive only the global factor. Erase strength is learned independently of write strength, separating how much to remove from how much to add. Building on gated delta updates (Yang et al., 2025; Hatamizadeh et al., 2026) and depth-wise rewriting (Zhang et al., 2026b), GDHC integrates these controls with the existing mixer and read/write maps. The same construction applies to both mHC and identity-mixer iHC (Tencent Hy Team, 2026).

We evaluate GDHC on both mHC and iHC across two model scales. On A1B (7B parameters, 0.91B active), GDHC improves average downstream scores by 0.86–1.30 points on iHC and 0.13–0.25 points on mHC across the two reported training budgets, with lower training loss in every comparison. Ablations on A500M (4B parameters, 0.55B active) show complementary effects from decay and erasure on iHC, while adding erasure on top of decay further reduces training perplexity on mHC. The internal state changes consistently with the proposed retention mechanism: on A1B mHC, the deep-to-shallow activation-maximum ratio falls from 13.8 to 1.5, and the common-mode RMS decreases across depth, alongside a larger median relative net state change. Section 5 connects these performance and diagnostic results to the two retention controls.

This work makes three contributions. ① We distinguish **state retention from cross-stream mixing** in multi-stream residual pathways, and show that mHC's preserved stream-average component accumulates disproportionately with depth. ② We build GDHC by combining **global decay and write-aligned erasure** within the existing multi-stream interface, and give a norm bound for the carried-state operator conditioned on the realized gates. ③ Experiments on mHC and iHC show **improved training and downstream performance**, while diagnostics reveal flatter depth profiles alongside larger median relative net state changes.

2 PRELIMINARIES

Each multi-stream residual update combines a carried-state transition with a new branch write. Separating these terms lets us distinguish how the base mixer propagates existing state from how successive writes accumulate new content across depth.

Multi-stream residual updates. Let L be the number of Transformer layers, C the hidden size, and n the number of residual streams. Attention and MLP sub-layers are indexed by $l = 1, \ldots, 2L$. Suppressing batch and token indices, the state is $\mathbf{x}_l \in \mathbb{R}^{n\times C}$, with one C-dimensional stream per row. Across these streams, a read map $\mathbf{H}_l^{\text{pre}} \in \mathbb{R}^{1\times n}$ forms the branch input, a write map $\mathbf{H}_l^{\text{post}} \in \mathbb{R}^{1\times n}$ distributes its output, and a mixer $\mathbf{B}_l \equiv \mathbf{H}_l^{\text{res}\top}$ propagates the old state:

$$\begin{aligned} \mathbf{o}_l &= \mathbf{H}_l^{\text{pre}}\mathbf{x}_l, & \mathbf{v}_l &= F_l(\mathrm{LN}(\mathbf{o}_l))\,, \\ \mathbf{x}_{l+1} &= \underbrace{\mathbf{B}_l\mathbf{x}_l}_{\text{carried state}} & &+\underbrace{\mathbf{H}_l^{\text{post}\top}\mathbf{v}_l}_{\text{new write}}. \end{aligned} \tag{1}$$

Here F_l is attention or an MLP, and LN is the branch's pre-normalization. The maps depend on the current token's state; attention still operates over tokens even though the token index is omitted. Importantly, pre-normalization acts only on the branch input and does not directly control the scale of the carried-state term $\mathbf{B}_l\mathbf{x}_l$.

Base mixers. HC learns an unconstrained mixer. mHC (Xie et al., 2026) uses

$$\mathbf{H}_l^{\text{pre}} = \sigma(\mathbf{u}_l^{\text{pre}}), \qquad \mathbf{H}_l^{\text{post}} = 2\sigma(\mathbf{u}_l^{\text{post}}), \qquad \mathbf{B}_l = \mathrm{SK}(\mathbf{u}_l^{\text{res}})^\top, \tag{2}$$

where unconstrained logits $\mathbf{u}_l^{\text{pre}}, \mathbf{u}_l^{\text{post}} \in \mathbb{R}^{1\times n}$ and $\mathbf{u}_l^{\text{res}} \in \mathbb{R}^{n\times n}$ are produced by a linear mapping head from the current token's normalized residual features. Here σ is the elementwise sigmoid and SK denotes Sinkhorn normalization toward a doubly stochastic matrix. iHC instead fixes $\mathbf{B}_l = \mathbf{I}$ while retaining learned read/write maps (Tencent Hy Team, 2026). We refer to this existing choice of $\mathbf{B}_l$ as the *base mixer*. Later, GDHC augments the carried-state path with explicit retention controls, yielding an *effective transition* ($\mathbf{M}_l$) distinct from the base mixer.

What constrained mixing preserves. To assess retention under constrained mixing, consider the average across streams. An exact nonnegative doubly stochastic base satisfies $\mathbf{B}_l\mathbf{1} = \mathbf{1}$ and $\mathbf{1}^\top\mathbf{B}_l = \mathbf{1}^\top$, preserving both the stream average $\bar{\mathbf{x}}_l = \mathbf{1}^\top\mathbf{x}_l/n$ and its replicated component $\mathbf{1}\bar{\mathbf{x}}_l$, called the *common mode*. Averaging Equation 1 gives

$$\bar{\mathbf{x}}_{l+1} = \bar{\mathbf{x}}_l + \frac{s(\mathbf{H}_l^{\text{post}})}{n}\,\mathbf{v}_l. \tag{3}$$

where $s(\mathbf{h}) = \sum_j h_j$ is the gate's total weight. Each branch therefore writes onto a component whose previous value is **carried forward with coefficient one**. Later writes may reinforce or cancel earlier content, but the mixer provides no independent attenuation of this component. Hence, although $\|\mathbf{B}_l\|_2 = 1$ limits amplification by the mixer itself, it does not bound the full state after repeated writes. mHC can contract the zero-mean subspace while preserving the common mode, whereas the identity base preserves every direction. Appendix A.1 extends this analysis to approximate Sinkhorn bases. Whether this preserved state actually accumulates in trained models is the empirical question addressed next.

3 Activation Growth and Gate Adaptation

Preservation identifies what mixing leaves unchanged, but not whether the preserved state actually accumulates after repeated writes. We therefore examine depth-wise state growth and track how the learned read/write interface adapts during training.

Diagnostics use four-stream A500M and A1B models, with depth snapshots taken at step 50.00K. Common-mode decomposition is reported for A1B on both mHC and iHC; detailed probe definitions and aggregation are given in Appendix A.2.

3.1 Activation growth is strongest in the preserved component

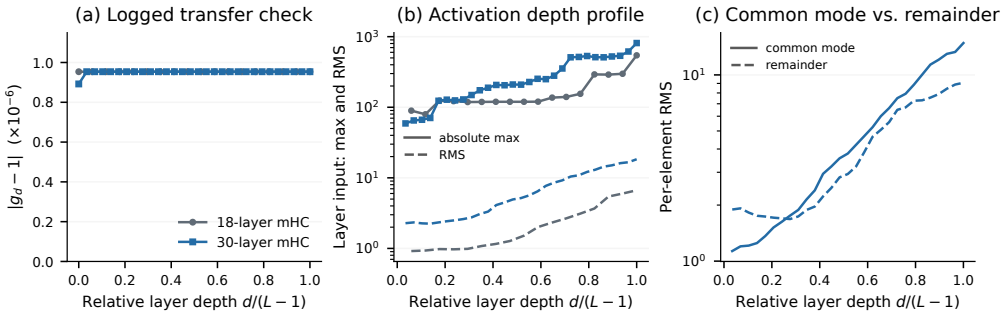

Figure 1: **Activation growth is strongest in the component preserved by mixing.** (a) the logged transfer statistic remains near one; (b) maxima and RMS increase across depth; (c) on A1B, common-mode RMS grows faster than the remainder. Grey denotes A500M and blue A1B. State profiles omit the embedding-facing layer and use $d = 1, \ldots, L-1$; panel (a) is an implementation diagnostic (Appendix A.2).

Layer-input maxima rise from 89.5 to 544 in A500M and from 59 to 812 in A1B (Figure 1(b)). Defining $R = A_{L-1}/A_1$, where A_d is the absolute maximum at layer d, gives depth ratios of 6.1 and 13.8. RMS grows by 7.3× and 8.1×, respectively, showing that the growth is not confined to isolated activation outliers. The logged transfer-column statistic remains within 10^{-6} of one (panel (a)), providing an auxiliary check of the expected mixing behavior.

Which component grows. Overall scale does not reveal where the growth occurs. We decompose $\mathbf{x}_l = \mathbf{1}\bar{\mathbf{x}}_l + \mathbf{x}_l^{\perp}$ into the common mode and the zero-mean remainder, and measure their per-element RMS separately (Figure 1(c)). On A1B, the common-mode depth ratio exceeds that of the remainder by a factor of 1.59 on iHC and 2.73 on mHC (Table 5). Its energy share rises from 0.413 to 0.636 on iHC and from 0.264 to 0.721 on mHC. The growth is therefore disproportionately concentrated in the component preserved by mixing.

3.2 Write gates become more concentrated during training

Beyond state scale at a fixed checkpoint, the read/write maps reveal how the interface adapts during training. For a positive gate $\mathbf{h} \in \mathbb{R}_+^n$, mass $s(\mathbf{h})$ measures total gate weight, while its *effective width* measures how that weight is distributed:

$$n_{\text{eff}}(\mathbf{h}) = \frac{\left(\sum_j h_j\right)^2}{\sum_j h_j^2} \in [1, n]. \tag{4}$$

Width ranges from concentration on one stream (1) to equal allocation (n). Reported values are the effective width of the token-averaged gate.[1] These statistics describe gate allocation; the resulting branch write also depends on $\mathbf{v}_l$.

During A1B training, median attention read width increases from its configured initial value of 1.31 to 2.13, while write width falls from 4.00 to 2.00 and write-gate mass from 4.00 to 1.24 (Figure 2). A500M follows the same pattern, reaching widths of 2.04 for reading and 1.54 for writing, with write mass 0.89; MLP branches behave similarly. Thus, as training progresses, reads become more distributed across streams while writes become more selective and carry less total gate weight.[2]

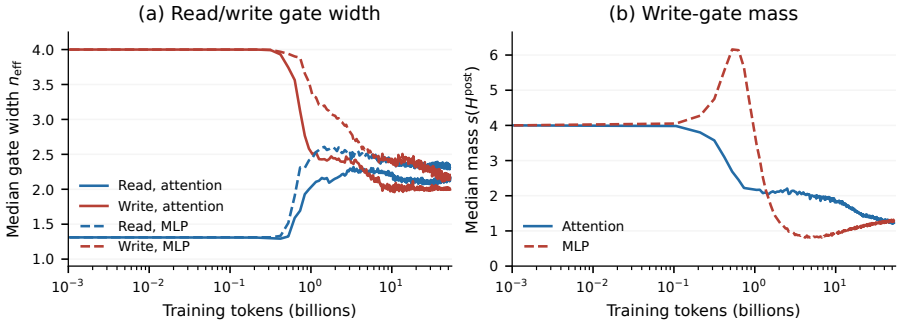

Figure 2: **Read gates broaden as write gates concentrate.** A1B mHC attention and MLP branches: (a) effective widths of the token-averaged gates, defined in Equation 4, and (b) write-gate mass. Curves show medians over layers 1–28; aggregation is specified in Appendix A.2.

Across depth, the trend is different from the decline over training: at the final checkpoint, deeper layers have larger write-gate mass. Its depth correlation ranges from $r = +0.88$ to $+0.97$ across models and branches.

Implications for residual updates. These observations motivate controlling carried state separately from the new write. The accumulation of preserved state calls for a global retention control, while increasingly selective writes motivate stronger revision along the direction where new content is introduced. We therefore combine global attenuation with write-aligned erasure, leaving orthogonal components subject only to the global factor. Independent erase and write strengths further separate how much old content to remove from how much new content to add.

[1] Each logged per-stream value is a mean over the tokens of a microbatch, over monitored microbatches, and over devices. Later mentions shorten this to gate width.

[2] A released DeepSeek-V4-Flash (DeepSeek-AI, 2026) checkpoint shows a qualitatively similar allocation pattern, with broad reads and some highly concentrated writes (Appendix A.12).

4 GATED DELTA HYPER-CONNECTIONS

GDHC inserts explicit retention control into the carried-state path after base mixing and before the new branch write, while leaving the existing read/write interface unchanged.

4.1 THE RESIDUAL UPDATE WITH RETENTION CONTROL

For either base $\mathbf{B}_l$, let $a_l \in (0, 1]$ control global retention, $\beta_l \in [0, 1]$ control erasure, and $\mathbf{k}_l \in \mathbb{R}^n$ be a unit vector derived from the write map. GDHC defines the effective carried-state transition as:

$$\boxed{\mathbf{M}_l = a_l \mathbf{E}_l \mathbf{B}_l, \qquad \mathbf{E}_l = \mathbf{I} - \beta_l \mathbf{k}_l \mathbf{k}_l^\top} \tag{5}$$

$$\mathbf{x}_{l+1} = \mathbf{M}_l \mathbf{x}_l + \mathbf{H}_l^{\text{post}\top} \mathbf{v}_l. \tag{6}$$

Writing $\mathbf{y}_l = \mathbf{B}_l \mathbf{x}_l$ separates the update into three terms:

$$\mathbf{x}_{l+1} = \underbrace{a_l \mathbf{y}_l}_{\text{global retention}} \underbrace{- a_l \beta_l \mathbf{k}_l (\mathbf{k}_l^\top \mathbf{y}_l)}_{\text{directional erasure}} + \underbrace{\mathbf{H}_l^{\text{post}\top} \mathbf{v}_l}_{\text{new write}}. \tag{7}$$

Setting $a_l = 1$ and $\beta_l = 0$ exactly recovers the base update in Equation 1; Figure 3 illustrates where the two retention controls enter the residual path.

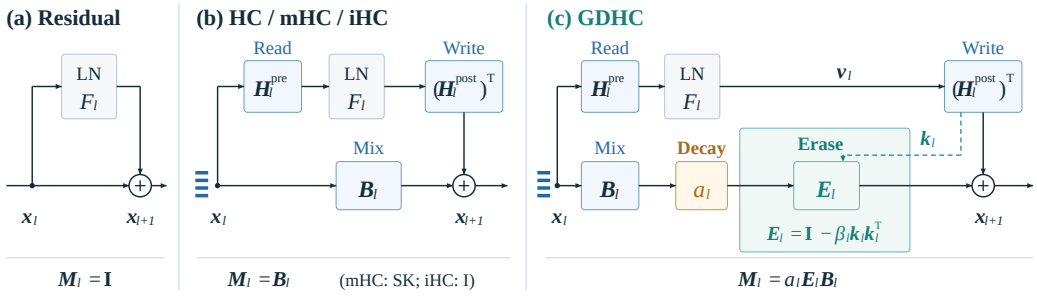

Figure 3: **Retention control enters the carried-state path.** Carried-state operators for (a) an ordinary residual connection, (b) hyper-connections with a base mixer, and (c) GDHC, which composes that mixer with global decay a_l and erasure $\mathbf{E}_l$ along the write direction $\mathbf{k}_l$. Read and write maps are the same in (b) and (c), and the branch reads the unmodified state; $a_l = 1$ at sub-layers that carry no decay head.

4.2 GLOBAL DECAY REGULATES OVERALL RETENTION

In Equation 3, the stream average persists with coefficient one. Global decay makes that coefficient controllable: under decay alone, a common-mode input $\mathbf{1}\mathbf{q}^\top$ is scaled to $a_l \mathbf{1}\mathbf{q}^\top$ before the new write is added. For the global-RMS parameterization, define $\tilde{\mathbf{x}}_l = \text{vec}(\mathbf{x}_l) \in \mathbb{R}^{nC}$ and $r_l = 1/(\text{RMS}(\tilde{\mathbf{x}}_l) + \varepsilon)$, with $\varepsilon = 10^{-6}$. A scalar head gives

$$a_l = \sigma\big(\alpha_l^{\text{dec}} r_l \boldsymbol{\theta}_l^{\text{dec}\top} \tilde{\mathbf{x}}_l + b_l^{\text{dec}}\big), \qquad \boldsymbol{\theta}_l^{\text{dec}} \in \mathbb{R}^{nC}. \tag{8}$$

Each decay head has its own projection, scale, and bias. Initialization keeps retention close to the base update: zero projections, scales of 0.01, and biases of 6 give $a_l \approx 0.9975$. Appendix A.2 details normalization and optimization.

4.3 WRITE-ALIGNED ERASURE ENABLES SELECTIVE REVISION

Each branch writes new content along a single direction in stream space. GDHC aligns erasure with this direction, allowing stronger revision where new content is introduced while leaving orthogonal components subject only to global decay. Define

$$\mathbf{k}_l = \frac{\mathbf{H}_l^{\text{post}\top}}{\|\mathbf{H}_l^{\text{post}}\|_2}, \qquad w_l = \|\mathbf{H}_l^{\text{post}}\|_2. \tag{9}$$

Writing the new contribution as $w_l \mathbf{k}_l \mathbf{v}_l$ and projecting the update onto $\mathbf{k}_l$ yields

$$\mathbf{k}_l^\top \mathbf{x}_{l+1} = a_l(1-\beta_l)\mathbf{k}_l^\top \mathbf{B}_l \mathbf{x}_l + w_l \mathbf{v}_l. \tag{10}$$

Along $\mathbf{k}_l$, *old-state retention* is controlled by $a_l(1-\beta_l)$, while the *new write* is controlled independently by w_l. In orthogonal directions, the mixed state is affected only by a_l. Reusing the write direction provides a data-dependent erase key without introducing additional key parameters.

Erase strength is predicted independently of the write magnitude:

$$\beta_l = \beta_{\max}\sigma\left(\alpha_l^{\mathrm{er}} r_l \boldsymbol{\theta}_l^{\mathrm{er}\top} \tilde{\mathbf{x}}_l + b_l^{\mathrm{er}}\right). \tag{11}$$

As with decay, each sub-layer has its own projection, scale, and bias. We use $\beta_{\max} = 1$, so erasure attenuates the selected component without reversing its sign. Zero projections, scales of 0.01, and bias $b_l^{\mathrm{er}} = -6$ initialize erasure near zero, keeping GDHC close to the base update. Appendix A.11 considers $\beta_{\max} = 2$.

4.4 Operator bound and practical choices

Erasure can introduce negative off-diagonal entries, taking the effective transition outside the doubly stochastic set. Its norm therefore needs a separate bound. Conditioned on realized gates, $\mathbf{E}_l$ has eigenvalue $1-\beta_l$ along $\mathbf{k}_l$ and 1 in the orthogonal complement, so $\|\mathbf{E}_l\|_2 = 1$ for $n > 1$ and $\beta_l \in [0,1]$. With an exact doubly stochastic or identity base,

$$\|\mathbf{M}_l\|_2 \le a_l \|\mathbf{B}_l\|_2 \le a_l. \tag{12}$$

Hence the carried-state transition is non-expansive, and becomes contractive when $a_l < 1$. Equation 12 bounds the carried-state path with realized gates, not the full nonlinear block Jacobian or state magnitude after new writes. Appendix A.1 treats finite Sinkhorn normalization.

Decay placement. Across successive carried-state transitions, scalar factors multiply, making decay frequency as important as each gate's value. Over 36 sub-layers, constant factors of 0.9975 and 0.95 yield products of approximately 0.914 and 0.158. GDHC therefore controls this frequency with stride s: a decay head is active when $(l-1) \bmod s = 0$, with $a_l = 1$ elsewhere. Erasure remains available at every sub-layer. Experiments compare dense decay ($s = 1$) with the default $s = 5$.

5 Experiments

Evaluation proceeds in three steps: compare GDHC with both base mixers, isolate the effects of decay and erasure, and test whether the resulting state dynamics match the proposed retention mechanism. A1B provides downstream results and diagnostics at two training budgets, while A500M supports component and decay-placement ablations.

5.1 Evaluation protocol

Models. Experiments use decoder-only mixture-of-experts Transformers trained from scratch on a proprietary corpus. A1B has 30 layers, 7B total parameters, and 0.91B active parameters; A500M has 18 layers, 4B total, and 0.55B activated. Both use hidden size 1024, rotary position embeddings (Su et al., 2024), gated linear units (Shazeer, 2020), per-head query–key normalization (Henry et al., 2020), and sparsely gated MoE layers (Shazeer et al., 2017) with top-5 routing over 256 experts plus one shared expert.

Optimization. Training uses Muon with momentum 0.95, peak learning rate 10^{-3}, weight decay 0.1, global gradient clipping at 1.0, and bf16 computation with fp32 master gradients. A warmup-stable-decay schedule starts with 750 warmup steps. Each step processes 2^{20} tokens at sequence length 8192. All HC-based models use four residual streams and the same single-stream initialization for the mapping heads as mHC (Xie et al., 2026), under which read and write gates initially depend only on their static biases. Within each base/variant pair, the corpus, seed, data order, optimizer, and learning-rate schedule are identical. DDL instead uses four value channels per feature (Appendix A.10). Table 4 and Appendix A.2 provide the remaining settings.

Evaluation. Matched checkpoint exports are evaluated on ten tasks spanning exams, knowledge, science, and reasoning (Appendix A.6); group and overall scores are unweighted task means. Training cross-entropy is smoothed as $\mathrm{EMA} \leftarrow 0.99\,\mathrm{EMA} + 0.01\,\mathrm{loss}$; reported training PPL is $\exp(\mathrm{EMA})$. A1B is evaluated at 52.43B tokens and 83.89B tokens, and A500M 52.43B tokens. Each variant is trained once; the A500M mHC baseline is repeated once with the same seed and configuration to assess execution variability. Matched base/variant runs share the same seed and data order, enabling paired training-loss comparisons. Baselines include mHC (Xie et al., 2026) and iHC (Tencent Hy Team, 2026), augmented by GDHC, as well as DDL (Zhang et al., 2026b), which applies rank-one edits in feature space rather than stream space.

5.2 Performance with learned and identity mixers

Across both evaluated budgets, GDHC improves the average downstream score of each base (Table 1). At 52.43B and 83.89B tokens, GDHC raises the unweighted ten-task mean over iHC by 1.30 and 0.86 points, and over mHC by 0.13 and 0.25 points, respectively; these gains use unrounded task means. Task-level wins are $8/10$ and $7/10$ for iHC, and $6/10$ and $7/10$ for mHC. GDHC_{iHC} achieves the highest overall score at both checkpoints. Notably, the mHC downstream gain increases at the later evaluated budget rather than contracting.

The gains vary across task groups. Exam tasks show the most consistent pattern: GDHC_{iHC} outperforms iHC on all four exams at both budgets, while GDHC_{mHC} improves over mHC on three of four. Knowledge and reasoning results are more mixed, indicating that the aggregate gains are distributed unevenly across tasks.

At 52.43B and 83.89B tokens, DDL obtains overall scores of 25.64 and 27.78, respectively. Both GDHC variants exceed these scores at each budget: GDHC_{iHC} by 2.45 and 2.39 points, and GDHC_{mHC} by 1.89 and 0.75 points. Training losses show the same ordering: all four multi-stream configurations are below DDL by 0.053–0.059 at 52.43B tokens and by 0.036–0.043 at 83.89B tokens. Appendix A.10 describes the reproduced DDL update and its feature-space edit.

GDHC lowers training loss at both budgets; at 83.89B tokens the gaps are -0.0047 on iHC and -0.0014 on mHC. Appendix Table 6 reports the trajectory.

Table 1: **GDHC improves average downstream scores on both bases at both budgets.** A1B task scores are percentages. Avg. is the unweighted mean within a group; Overall averages all ten tasks. Bold marks each column's best score within a budget, and shaded rows identify the two GDHC variants.

	Exams					Knowledge				Science	Reasoning			
Method	CMMLU	MMLU	Redux	Pro	Avg.	Hotpot	PopQA	CSimpleQA	Avg.	GPQA	Wino	BBH	Avg.	Overall
52.43B tokens (step 50.00K)														
DDL	32.86	35.50	36.99	15.44	30.20	14.71	11.54	7.10	11.12	22.87	49.57	29.86	39.72	25.64
iHC	35.98	37.87	39.71	16.21	32.44	15.66	**12.54**	**8.22**	**12.14**	22.70	49.17	29.81	39.49	26.79
mHC	35.94	38.14	38.90	15.31	32.07	15.41	11.72	7.31	11.48	**29.44**	**52.80**	29.00	**40.90**	27.40
GDHC_{iHC}	**37.70**	**41.41**	**42.02**	**16.64**	**34.44**	**16.04**	12.33	7.88	12.08	25.76	51.22	**29.91**	40.57	**28.09**
GDHC_{mHC}	36.51	38.53	39.50	15.18	32.43	15.39	11.79	**8.22**	11.80	28.82	51.78	29.58	40.68	27.53
83.89B tokens (step 80.00K)														
DDL	37.24	40.46	41.44	18.64	34.45	15.94	11.18	7.55	11.56	26.55	48.70	30.10	39.40	27.78
iHC	40.38	43.27	45.57	17.97	36.80	16.40	12.34	8.94	**12.56**	26.26	49.88	**32.15**	41.02	29.32
mHC	38.36	40.39	42.01	17.53	34.57	15.69	**12.51**	8.69	12.30	27.53	50.75	29.37	40.06	28.28
GDHC_{iHC}	**42.98**	**45.92**	**47.29**	**18.96**	**38.79**	**16.83**	11.30	**9.09**	12.41	25.66	**51.78**	31.92	**41.85**	**30.17**
GDHC_{mHC}	38.89	40.50	42.67	16.95	34.75	16.19	11.57	8.85	12.20	**28.96**	49.80	30.96	40.38	28.53

5.3 Adding erasure and varying decay placement

At fixed dense placement, adding erasure after decay lowers training PPL a further 0.007 on iHC and 0.011 on mHC, following reductions of 0.013 and 0.014 from decay alone (Table 2). The iHC erase-only completes the four-way comparison: its PPL is 5.860, compared with 5.863 for decay alone and 5.856 for their combination (Appendix A.9). Thus, decay and erasure provide complementary gains, while erasure also improves the overall performance when added on top of decay.

Switching decay placement from dense to $s = 5$ uses 8 rather than 36 decay sites and lowers PPL by another 0.001 on iHC and 0.008 on mHC. The iHC difference corresponds to only about 0.00017

Table 2: **Adding erasure improves on decay alone on both bases.** A500M training PPL at step 50.00K (52.43B tokens). The mHC entry is the mean and sample standard deviation of two runs with identical configuration, which describe variability between two same-seed executions; the mHC Δ column is relative to that mean, while every other entry is a single run and the iHC Δ column is relative to the single iHC run. Dense rows isolate the addition of erasure; $s = 5$ also changes decay placement. All runs share the seed and data order.

identity base, $\mathbf{B}_l = \mathbf{I}$			doubly stochastic base, $\mathbf{B}_l = \mathrm{SK}(\mathbf{u}_l^{\mathrm{res}})^\top$		
Variant	PPL	Δ	Variant	PPL	Δ
iHC	5.876	—	mHC	5.875 ± 0.005	—
+ decay, dense	5.863	-0.013	+ decay, dense	5.861	-0.014
+ decay + erase, dense	5.856	-0.020	+ decay + erase, dense	5.850	-0.025
GDHC$_{\text{iHC}}$, s=5	**5.855**	**-0.021**	**GDHC$_{\text{mHC}}$, s=5**	**5.842**	**-0.033**

in log loss, below the practical resolution discussed in Appendix A.7. Stride 5 therefore retains comparable iHC performance with substantially fewer decay sites, while slightly improving mHC PPL, and is used in the main A1B comparisons.

5.4 State Dynamics under Retention Control

We next ask whether the performance gains coincide with the internal behavior targeted by GDHC: limiting accumulated carried-state growth while preserving the ability to revise the residual state.

Activation profiles. On A1B mHC at 52.43B tokens, GDHC lowers the last retained activation maximum from 812 to 366 while the first rises from 59 to 246; the deep-to-shallow ratio falls from 13.8 to 1.5 (Figure 4(a)). GDHC thus flattens the depth profile rather than uniformly shrinking the residual state.

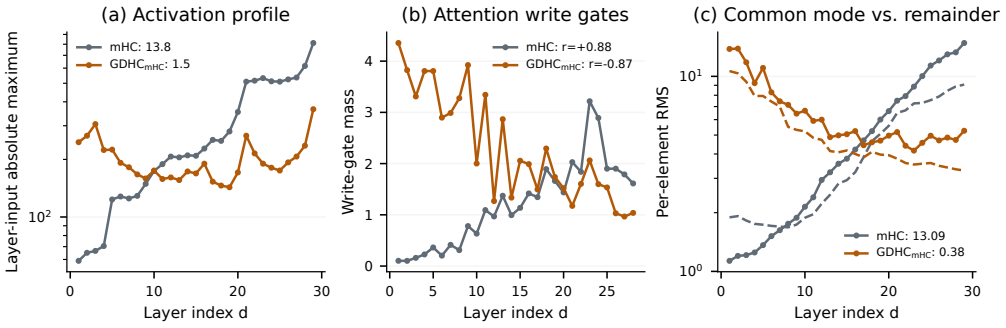

Figure 4: **GDHC flattens activation growth and changes write.** At step 50.00K: (a) the activation-maximum ratio falls from 13.8 to 1.5; (b) write-gate mass shifts toward shallower layers; (c) common-mode RMS decreases across depth. Solid and dashed lines in (c) denote the common mode and remainder.

Common-mode measurements directly revisit Section 3.1 and the component most directly affected by retention control. Under GDHC, the deep-to-shallow RMS ratio falls from 13.09 to 0.38 on mHC and from 13.40 to 1.05 on iHC (Table 5), turning strong baseline accumulation into a decline on mHC and a nearly flat profile on iHC. Importantly, this does not correspond to uniformly smaller activations: depth-averaged common-mode RMS increases from 5.40 to 6.53 on mHC while decreasing from 6.88 to 5.67 on iHC. The consistent effect is therefore reduced accumulation across depth, matching the intended role of explicit retention control.

Gate allocation. The change extends beyond state magnitude: the learned residual interface also reallocates reading and writing. Median attention write-gate mass rises from 1.24 to 2.03 and read width from 2.13 to 2.87, while the correlation of write mass with depth reverses from $+0.88$ to -0.87 (Figure 4(b)). Similar trends persist at 83.89B tokens and on the identity base (Table 7). Thus, explicit retention control changes not only how much state is carried across depth, but also how the model allocates subsequent reads and writes.

Relative state change. To characterize local state revision, we measure net state change relative to the sub-layer input (Equation 19). At 52.43B tokens, the median attention-layer ratio rises from 0.200 to 0.273 on mHC and from 0.147 to 0.289 on iHC, with the same pattern at the later checkpoint. Flatter depth profiles therefore coexist with larger median relative net state changes under GDHC.

6 Related Work

Expanding and stabilizing residual pathways. Residual connections (He et al., 2016) provide a direct path through depth. DenseFormer (Pagliardini et al., 2024), DeepCrossAttention (Heddes et al., 2025), and MUDDFormer (Xiao et al., 2025) increase its flexibility through learned aggregation and expanded states. HC introduces multiple streams with learned read, write, and mixing maps (Zhu et al., 2025a); Frac-Connections reduces their memory traffic (Zhu et al., 2025b); and mHC constrains mixing for stable propagation (Xie et al., 2026). Subsequent work studies mixer parameterizations (Yang & Gao, 2026; Zhou et al., 2026; Dandachi & Diggs-Galligan, 2026; Gu et al., 2026; Sengupta et al., 2026) and the feasible set itself (Wang & Qiu, 2026; Liu et al., 2026), as well as stream expansion (Zhang et al., 2026a), collapse (Alimaskina et al., 2026), and causal importance (Peng et al., 2026). GDHC is orthogonal to these mixer designs: it retains the chosen base mixer and adds explicit control over carried-state retention.

Updating representations across depth. Highway Networks gate shortcut retention (Srivastava et al., 2015), while MRLA (Fang et al., 2023), DenseFormer (Pagliardini et al., 2024), Attention Residuals (Kimi Team, 2026), and Delta Attention Residuals (Luo et al., 2026) modify how information is aggregated across layers. Deep Delta Learning (DDL) applies gated rank-one rewriting in feature space, with erase and write coupled through a shared strength (Zhang et al., 2026b). GDHC instead preserves the existing multi-stream mixer and read/write interface, applies the rank-one edit in stream space, and parameterizes decay, erase, and write strengths separately. Section 5.2 provides the corresponding empirical comparison.

Decay and delta rules in sequence models. Fast-weight programmers (Schlag et al., 2021) and DeltaNet (Yang et al., 2024b) use structured memory updates. Gated linear attention (Yang et al., 2024a) and Gated DeltaNet (Yang et al., 2025) introduce adaptive forgetting; Kimi Linear uses finer-grained decay (Kimi Team, 2025); and Gated DeltaNet-2 decouples erase and write gates (Hatamizadeh et al., 2026). GDHC adapts these primitives to multi-stream residual pathways: retention acts in stream space, its directional erase key is tied to the current write map, and the resulting transition composes with an existing base mixer.

7 Conclusion

Stable residual mixing does not by itself determine how much state should persist across depth. Structural analysis identifies a preserved stream-average component under doubly stochastic mixing, and direct measurements show that it accumulates disproportionately in trained models. GDHC makes retention explicit through global decay and write-aligned erasure while preserving the existing mixer and read/write interface. Across both mHC and iHC, GDHC yields lower training loss and higher downstream averages at the larger scale; diagnostics further show flatter depth profiles alongside larger median relative net state changes. These results position retention alongside mixing as a useful design dimension for multi-stream residual networks.

Reproducibility Statement

Section 4 defines the update, gate equations, and decay placement. Appendix A.1 derives the operator properties; Appendices A.2–A.7 document implementation, diagnostics, checkpoints, and evaluation metadata. The pre-training corpus is proprietary.

AI Use Statement

Generative AI tools assisted with language editing, manuscript organization, figure prototyping, and analysis and plotting code. The authors' research workflow produced the architecture, training

runs, evaluation data, and reported measurements. The authors remain responsible for verifying the claims, numerical results, and final submitted artifacts.

Ethics Statement

This study modifies Transformer residual pathways and evaluates research checkpoints using pre-training loss and public benchmarks. It includes no new human-subject study or model-weight release. The checkpoints may inherit biases and data-related limitations from the pre-training corpus; the evaluations do not determine whether GDHC changes those properties.

References

Ekaterina Alimaskina, Gleb Molodtsov, and Aleksandr Beznosikov. Analyzing stream collapse in hyper-connections: From diagnosis to mitigation. *arXiv preprint arXiv:2606.03483*, 2026.

Torque Dandachi and Sophia Diggs-Galligan. go-mHC: Direct parameterization of manifold-constrained hyper-connections via generalized orthostochastic matrices. *arXiv preprint arXiv:2604.02309*, 2026.

DeepSeek-AI. DeepSeek-V4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026.

Yanwen Fang, Yuxi Cai, Jintai Chen, Jingyu Zhao, Guangjian Tian, and Guodong Li. Cross-layer retrospective retrieving via layer attention. In *International Conference on Learning Representations*, 2023.

Aryo Pradipta Gema, Joshua Ong Jun Leang, Giwon Hong, Alessio Devoto, Alberto Carlo Maria Mancino, Rohit Saxena, Xuanli He, Yu Zhao, Xiaotang Du, Mohammad Reza Ghasemi Madani, et al. Are we done with MMLU? In *Conference of the North American Chapter of the Association for Computational Linguistics*, 2025.

Riccardo Grazzi, Julien Siems, Arber Zela, Jörg K. H. Franke, Frank Hutter, and Massimiliano Pontil. Unlocking state-tracking in linear RNNs through negative eigenvalues. In *International Conference on Learning Representations*, 2025.

Yuxuan Gu, Wuyang Zhou, Huijun Xing, and Danilo Mandic. TEMPER: Tensorized efficient manifold-constrained parameterization for expressive residual routing. *arXiv preprint arXiv:2608.07851*, 2026.

Ali Hatamizadeh, Yejin Choi, and Jan Kautz. Gated DeltaNet-2: Decoupling erase and write in linear attention. *arXiv preprint arXiv:2605.22791*, 2026.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Yancheng He, Shilong Li, Jiaheng Liu, Yingshui Tan, Weixun Wang, Hui Huang, Xingyuan Bu, Hangyu Guo, Chengwei Hu, Boren Zheng, et al. Chinese SimpleQA: A chinese factuality evaluation for large language models. In *Annual Meeting of the Association for Computational Linguistics*, 2025.

Mike Heddes, Adel Javanmard, Kyriakos Axiotis, Gang Fu, MohammadHossein Bateni, and Vahab Mirrokni. DeepCrossAttention: Supercharging transformer residual connections. In *International Conference on Machine Learning*, 2025.

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

Alex Henry, Prudhvi Raj Dachapally, Shubham Shantaram Pawar, and Yuxuan Chen. Query-key normalization for transformers. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 4246–4253, 2020.

Kimi Team. Kimi Linear: An expressive, efficient attention architecture. *arXiv preprint arXiv:2510.26692*, 2025.

Kimi Team. Attention residuals. *arXiv preprint arXiv:2603.15031*, 2026.

Haonan Li, Yixuan Zhang, Fajri Koto, Yifei Yang, Hai Zhao, Yeyun Gong, Nan Duan, and Timothy Baldwin. CMMLU: Measuring massive multitask language understanding in chinese. In *Findings of the Association for Computational Linguistics: ACL*, 2024.

Zhaoyi Liu, Haichuan Zhang, and Ang Li. Beyond the Birkhoff polytope: Spectral-sphere-constrained hyper-connections. *arXiv preprint arXiv:2603.20896*, 2026.

Cheng Luo, Zefan Cai, and Junjie Hu. Delta attention residuals. *arXiv preprint arXiv:2605.18855*, 2026.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Annual Meeting of the Association for Computational Linguistics*, 2023.

Gaurav Menghani, Ravi Kumar, and Sanjiv Kumar. LAuReL: Learned augmented residual layer. In *International Conference on Machine Learning*, 2025.

Matteo Pagliardini, Amirkeivan Mohtashami, François Fleuret, and Martin Jaggi. DenseFormer: Enhancing information flow in transformers via depth weighted averaging. In *Advances in Neural Information Processing Systems*, 2024.

William Peng, Josheev Rai, Kevin Tseng, Siwei Wang, and Sean Wu. Ablate and rescue: A causal analysis of residual stream hyper-connections. *arXiv preprint arXiv:2603.14833*, 2026.

Zihan Qiu, Zekun Wang, Xiao Li, Yanpeng Li, Yang Xu, Yixuan Wang, Huaqing Zhang, Rui Men, Bochao Mao, Chengruidong Zhang, et al. On the design of Qwen3.8-Next architecture: Evaluation, efficiency, and training stability. *arXiv preprint arXiv:2608.30320*, 2026.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. GPQA: A graduate-level google-proof Q&A benchmark. In *Conference on Language Modeling*, 2024.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. WinoGrande: An adversarial Winograd schema challenge at scale. In *AAAI Conference on Artificial Intelligence*, 2020.

Imanol Schlag, Kazuki Irie, and Jürgen Schmidhuber. Linear transformers are secretly fast weight programmers. In *International Conference on Machine Learning*, 2021.

Biswa Sengupta, Jinhua Wang, and Leo Brunswic. JPmHC dynamical isometry via orthogonal hyper-connections. *arXiv preprint arXiv:2602.18308*, 2026.

Noam Shazeer. GLU variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2017.

Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. Highway networks. *arXiv preprint arXiv:1505.00387*, 2015.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. RoFormer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, and Jason Wei. Challenging BIG-Bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL*, 2023.

Tencent Hy Team. Hy4 preview. https://huggingface.co/tencent/Hy4-preview, 2026. Model card. Accessed 2026-09-06.

Chenrui Wang and Yixuan Qiu. Accelerating Birkhoff projection for manifold-constrained hyper-connections. *arXiv preprint arXiv:2606.07574*, 2026.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. In *Advances in Neural Information Processing Systems*, 2024.

Da Xiao, Qingye Meng, Shengping Li, and Xingyuan Yuan. MUDDFormer: Breaking residual bottlenecks in transformers via multiway dynamic dense connections. In *International Conference on Machine Learning*, 2025.

Zhenda Xie, Yixuan Wei, Huanqi Cao, Chenggang Zhao, Chengqi Deng, Jiashi Li, Damai Dai, Huazuo Gao, Jiang Chang, Kuai Yu, et al. mHC: Manifold-constrained hyper-connections. In *International Conference on Machine Learning*, 2026.

Songlin Yang, Bailin Wang, Yikang Shen, Rameswar Panda, and Yoon Kim. Gated linear attention transformers with hardware-efficient training. In *International Conference on Machine Learning*, 2024a.

Songlin Yang, Bailin Wang, Yu Zhang, Yikang Shen, and Yoon Kim. Parallelizing linear transformers with the delta rule over sequence length. In *Advances in Neural Information Processing Systems*, pp. 115491–115522, 2024b.

Songlin Yang, Jan Kautz, and Ali Hatamizadeh. Gated delta networks: Improving Mamba2 with delta rule. In *International Conference on Learning Representations*, 2025.

Yongyi Yang and Jianyang Gao. mHC-lite: You don't need 20 Sinkhorn-Knopp iterations. *arXiv preprint arXiv:2601.05732*, 2026.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Conference on Empirical Methods in Natural Language Processing*, 2018.

Z.ai. GLM-5.3-Flash. https://huggingface.co/zai-org/GLM-5.3-Flash, 2026. Model card. Accessed 2026-09-06.

Xiangdong Zhang, Xiaohan Qin, Sunan Zou, Tuo Dai, Xiaoming Shi, Huaijin Wu, Yebin Yang, Zhuo Xia, Shaofeng Zhang, Lin Yao, et al. xHC: Expanded hyper-connections. *arXiv preprint arXiv:2607.14530*, 2026a.

Yifan Zhang, Yifeng Liu, Mengdi Wang, and Quanquan Gu. Deep delta learning. *arXiv preprint arXiv:2601.00417*, 2026b.

Wuyang Zhou, Yuxuan Gu, Giorgos Iacovides, and Danilo Mandic. KromHC: Manifold-constrained hyper-connections with Kronecker-product residual matrices. In *International Conference on Machine Learning*, 2026.

Defa Zhu, Hongzhi Huang, Zihao Huang, Yutao Zeng, Yunyao Mao, Banggu Wu, Qiyang Min, and Xun Zhou. Hyper-connections. In *International Conference on Learning Representations*, 2025a.

Defa Zhu, Hongzhi Huang, Jundong Zhou, Zihao Huang, Yutao Zeng, Banggu Wu, Qiyang Min, and Xun Zhou. Frac-connections: Fractional extension of hyper-connections. *arXiv preprint arXiv:2503.14125*, 2025b.

A Additional Analysis and Experimental Details

A.1 Analysis of the retention controls

The analysis first identifies the component preserved by mixing, then quantifies directional erasure and extends the norm bound to approximate bases.

The component preserved by mixing. For a nonnegative doubly stochastic $\mathbf{B}_l$, decompose the stream state as

$$\mathbf{x}_l = \mathbf{1}\bar{\mathbf{x}}_l + \mathbf{x}_l^{\perp}, \qquad \bar{\mathbf{x}}_l = \tfrac{1}{n}\mathbf{1}^\top \mathbf{x}_l, \qquad \mathbf{1}^\top \mathbf{x}_l^{\perp} = 0. \tag{13}$$

The equality $\mathbf{B}_l\mathbf{1} = \mathbf{1}$ fixes the common-mode component, while $\mathbf{1}^\top\mathbf{B}_l = \mathbf{1}^\top$ keeps the zero-mean subspace invariant. A doubly stochastic matrix is a convex combination of permutation matrices, giving spectral norm at most one. Its fixed all-ones direction makes that norm exactly one. Premultiplying the full residual update by $\mathbf{1}^\top/n$ gives Equation 3: the old average persists and the write adds a new contribution. Successive writes can reinforce or cancel one another, so the preserved average establishes a missing attenuation control, not inevitable growth.

How much erasure removes. For a unit key $\mathbf{k}$ and $\mathbf{E} = \mathbf{I} - \beta\mathbf{k}\mathbf{k}^\top$,

$$\|\mathbf{E}\mathbf{Y}\|_F^2 = \|\mathbf{Y}\|_F^2 - \beta(2-\beta)\|\mathbf{k}^\top\mathbf{Y}\|_2^2. \tag{14}$$

Expanding $\mathbf{E}^\top\mathbf{E}$ gives $\mathbf{I} - \beta(2-\beta)\mathbf{k}\mathbf{k}^\top$, which yields the identity. For $0 \le \beta \le 2$, the edit is non-expansive. Removal is greatest at $\beta = 1$ and vanishes at both endpoints: $\beta = 2$ reflects the selected direction without reducing its norm. At any fixed gate, the energy removed also depends on the state's projection onto the key.

Alignment with the common mode. A positive write gate gives the tied key a nonzero projection onto the stream-average direction. Let $\mathbf{u} = \mathbf{1}/\sqrt{n}$ and $\mathbf{k} = \mathbf{h}^\top/\|\mathbf{h}\|_2$ for a positive gate $\mathbf{h}$. Then

$$(\mathbf{u}^\top\mathbf{k})^2 = \frac{\left(\sum_j h_j\right)^2}{n\sum_j h_j^2} = \frac{n_{\text{eff}}(\mathbf{h})}{n}, \tag{15}$$

so squared alignment equals gate width divided by n. For a nonzero pure common-mode input $\mathbf{Y} = \mathbf{1}\mathbf{q}^\top$, Equation 14 gives

$$\frac{\|\mathbf{E}\mathbf{Y}\|_F^2}{\|\mathbf{Y}\|_F^2} = 1 - \beta(2-\beta)\,\frac{n_{\text{eff}}(\mathbf{h})}{n}. \tag{16}$$

At $\beta = 1$, erasure removes the fraction $n_{\text{eff}}(\mathbf{h})/n$ of this input's energy. Concentrated writes therefore limit the fraction reachable by a single erase direction, whereas global decay acts on the entire component. When the key is not parallel to $\mathbf{1}$, erasure can also move part of the initially common-mode state into the zero-mean subspace; Equation 16 measures the energy of the full edited state. The identity applies to each realized gate.

The identity $\mathbf{k}_l^\top\mathbf{E}_l = (1-\beta_l)\mathbf{k}_l^\top$, together with the write $w_l\mathbf{k}_l\mathbf{v}_l$, gives Equation 10. The old coefficient along the key is $a_l(1-\beta_l)$; the new contribution has coefficient w_l. On the orthogonal complement, erasure is the identity and only global decay acts on the mixed state. These statements condition on the current key, gates, and base. Their derivatives must also be included when studying the full block Jacobian.

Approximate Sinkhorn bases. For a nonnegative approximate base, define

$$\epsilon_r = \|\mathbf{B}_l\mathbf{1} - \mathbf{1}\|_\infty, \qquad \epsilon_c = \|\mathbf{1}^\top\mathbf{B}_l - \mathbf{1}^\top\|_\infty. \tag{17}$$

Since $\|\mathbf{E}_l\|_2 = 1$ for a unit key, $n > 1$, and any $\beta_l \in [0,2]$, $\|\mathbf{B}_l\|_2 \le \sqrt{\|\mathbf{B}_l\|_1\|\mathbf{B}_l\|_\infty} \le \sqrt{(1+\epsilon_c)(1+\epsilon_r)}$, and

$$\|\mathbf{M}_l\|_2 \le a_l\sqrt{(1+\epsilon_c)(1+\epsilon_r)}. \tag{18}$$

Strict contraction requires the right-hand side to be below one, which needs both residuals rather than the single aggregate reported in Figure 1(a).

A.2 Implementation and diagnostic definitions

Algorithm 1 implements the order described in Section 4: mix the old state, apply the two retention controls, and add the original branch write. The gate inputs use global RMS normalization and the mapping heads come from the chosen base. The rank-one correction is evaluated without materializing an erase matrix.

Algorithm 1 One GDHC sub-layer: retention control followed by the original write.

Input: State $\mathbf{x} \in \mathbb{R}^{n \times C}$, sub-layer index l, base type, decay-active flag d, erase limit $\beta_{\max}$.

1. Compute $\mathbf{H}^{\text{pre}}, \mathbf{H}^{\text{post}}$ and $\mathbf{B}$ from the base mapping heads; set $\mathbf{B} \leftarrow \mathbf{I}$ for iHC.
2. Set $\mathbf{y} \leftarrow \mathbf{B}\mathbf{x}$.
3. Set $\tilde{\mathbf{x}} \leftarrow \text{vec}(\mathbf{x})$ and $r \leftarrow 1/(\text{RMS}(\tilde{\mathbf{x}}) + \varepsilon)$, $\varepsilon = 10^{-6}$.
4. Set $\mathbf{k} \leftarrow \mathbf{H}^{\text{post}\top} / \max(\|\mathbf{H}^{\text{post}}\|_2, 10^{-12})$.
5. Compute $\beta \leftarrow \beta_{\max}\sigma(\alpha_l^{\text{er}} r \boldsymbol{\theta}_l^{\text{er}\top}\tilde{\mathbf{x}} + b_l^{\text{er}})$.
6. If d is true, set $a \leftarrow \sigma(\alpha_l^{\text{dec}} r \boldsymbol{\theta}_l^{\text{dec}\top}\tilde{\mathbf{x}} + b_l^{\text{dec}})$; otherwise set $a \leftarrow 1$.
7. Apply $\mathbf{y} \leftarrow a[\mathbf{y} - \beta\mathbf{k}(\mathbf{k}^\top\mathbf{y})]$.
8. Compute $\mathbf{v} \leftarrow F(\text{LN}(\mathbf{H}^{\text{pre}}\mathbf{x}))$.
9. Return $\mathbf{y} + \mathbf{H}^{\text{post}\top}\mathbf{v}$.

Table 3: **Carried-state operators and parameter costs.** Bounds condition on realized gates, exact bases, and $\beta \in [0, 2]$. Base counts assume flattened mapping projections and omit biases/scales; GDHC counts include all added head parameters. Here $P = nC$, and $\mathbb{1}_s$ marks a sub-layer that carries a decay head.

Method	Base $\mathbf{B}_l$	Effective $\mathbf{M}_l$	Norm bound	Parameters per sub-layer
HC	unconstrained	$\mathbf{B}_l$	no fixed bound	$n(n+2)P$ projection weights
mHC	Sinkhorn	$\mathbf{B}_l$	1	$n(n+2)P$ projection weights
iHC	$\mathbf{I}$	$\mathbf{I}$	1	$2nP$ projection weights
GDHC	either base	$a_l\mathbf{E}_l\mathbf{B}_l$	$\leq a_l$	$+(1+\mathbb{1}_s)(P+2)$

Table 4: **Model and training settings for the reported comparisons.** The main GDHC configuration uses decay stride 5, a write-tied erase key, and $\beta_{\max} = 1$.

Run	Comparison	L	C	Heads	Experts/top-k	n	Seq.	Budget
A1B	iHC $\leftrightarrow$ GDHC$_{\text{iHC}}$	30	1024	16	256/5	4	8192	52.43B, 83.89B
A1B	mHC $\leftrightarrow$ GDHC$_{\text{mHC}}$	30	1024	16	256/5	4	8192	52.43B, 83.89B
A500M	iHC $\leftrightarrow$ GDHC$_{\text{iHC}}$	18	1024	8	256/5	4	8192	52.43B
A500M	mHC $\leftrightarrow$ GDHC$_{\text{mHC}}$	18	1024	8	256/5	4	8192	52.43B
GDHC switches		decay + erase; stride 5; tied key; $\beta_{\max} = 1$						
MoE widths		expert width 512; first layer dense with width 2048; vocabulary 201K, untied embeddings						
Parameters		A1B: 7B total, 0.91B activated; A500M: 4B total, 0.55B activated						
Shared training		seed 42; Muon (momentum 0.95); lr 10^{-3}; WSD; warmup 750 weight decay 0.1; grad clip 1.0; bf16 with fp32 master gradients						
Tokens per step		1.05M (2^{20}) for both A1B and A500M						
Gate numerics		logits in fp32; RMS floor $\varepsilon = 10^{-6}$; decay bias 6; erase bias -6						

Separate projection heads compute decay and erasure without changing the base mapping-head output shapes. Projections start at zero, scales at 0.01, and biases at 6 for decay and -6 for erasure; no head parameters are shared across sub-layers. All added parameters use AdamW: projections are excluded from Muon by name, and scalar scales and biases by rank. Sub-layers without a decay head still apply erasure. Loading a base checkpoint therefore requires initializing the added heads.

Gate logits are computed in fp32. Normalization uses $r_l = 1/(\text{RMS}(\tilde{\mathbf{x}}_l) + 10^{-6})$, with the constant outside the square root. The erase key uses $\mathbf{k}_l = \mathbf{H}_l^{\text{post}\top} / \max(\|\mathbf{H}_l^{\text{post}}\|_2, 10^{-12})$ for numerical protection. The unit-key equalities above apply when this floor is inactive; otherwise $\|\mathbf{k}_l\|_2 \leq 1$ and the non-expansive bounds still hold for $\beta_l \in [0, 2]$. All main comparisons use the write-tied key; the independently projected option is not evaluated in the main tables. For the counts in Table 3, a flattened input of size $P = nC$ projects to $n^2 + 2n$ mixing/read/write logits, or $2n$ read/write logits on iHC. Each added scalar head has P projection weights, one scale, and one bias.

Layer-input maxima. A_d is the absolute maximum over the input to Transformer layer d, taken from the attention branch of that layer. The embedding-facing layer $d = 0$ is omitted, and $R = A_{L-1}/A_1$ compares the last and first retained layers. This differs from $R_{\text{peak}} = \max_{d \geq 1} A_d/A_1$ in Appendix A.11. Both are max-reductions over tokens and devices, and describe activation scale.

RMS distinguishes broad scale changes from isolated extremes. On mHC at step 50.00K, the ratio $\max/\mathrm{RMS}$ decreases from 97.6 to 81.6 across depth in A500M but increases from 25.9 to 44.3 in A1B; growing maxima therefore do not imply the same change in relative outlier severity at both scales.

Logged transfer statistic. For the attention branch of layer d, the logged statistic g_d in Figure 1(a) summarizes the largest transfer-column sum: $\lambda_j = \sum_i (\mathbf{B}_l)_{ij} = \sum_i (\mathbf{H}_l^{\mathrm{res}})_{ji}$ and g_d averages $\max_j \lambda_j$ over tokens. This is a column sum of $\mathbf{B}_l$, equivalently a row sum of the stored $\mathbf{H}_l^{\mathrm{res}}$. An exact doubly stochastic base gives $g_d = 1$.

Gate summaries. For each branch and layer, read and write gates are averaged over tokens, monitored microbatches, and participating devices before computing gate mass and width. Reported width therefore measures the effective width of the averaged gate rather than the mean token-wise width. Layer medians and Pearson depth correlations use $d = 1, \ldots, L-2$.

Net state change per sub-layer. A separate probe records the size of each sub-layer's actual change to the residual state,

$$\rho_l = \frac{\mathrm{RMS}(\mathbf{x}_{l+1} - \mathbf{x}_l)}{\mathrm{RMS}(\mathbf{x}_l)}, \tag{19}$$

Both RMS values are computed over the $n \times C$ state in fp32. By Equation 6, the change is $\mathbf{D}_l + \mathbf{W}_l$, where $\mathbf{D}_l = (\mathbf{M}_l - \mathbf{I})\mathbf{x}_l$ is the change to the carried state and $\mathbf{W}_l = \mathbf{H}_l^{\mathrm{post}\top}\mathbf{v}_l$ is the new write. Only on unmodified iHC does $\mathbf{D}_l = \mathbf{0}$, making ρ_l a pure write-to-input ratio. Elsewhere it includes both terms and their alignment through the cross term in $\|\mathbf{D}_l + \mathbf{W}_l\|_F^2$. Thus ρ_l measures total local revision, not the magnitude of the branch write in isolation.

Reported values are medians over layers $1, \ldots, L-2$, denoted $\tilde{\rho}$, from single-microbatch snapshots on A1B. Table 7 reports attention. MLP medians at 52.43B are 0.144, 0.195, 0.177, and 0.180 for iHC, GDHC$_{\mathrm{iHC}}$, mHC, and GDHC$_{\mathrm{mHC}}$, respectively; at 83.89B they are 0.150, 0.217, 0.181, and 0.201. Both bases show larger median net changes under GDHC, with a smaller separation on mHC and less consistent MLP depth trends.

Common-mode split. The same monitor pass records the decomposition of the state entering each sub-layer. With $\mathbf{x}_l$ reshaped to $n \times C$,

$$\mathbf{x}_l = \mathbf{1}\bar{\mathbf{x}}_l + \mathbf{x}_l^{\perp}, \qquad \bar{\mathbf{x}}_l = \tfrac{1}{n}\mathbf{1}^{\top}\mathbf{x}_l, \qquad \mathbf{1}^{\top}\mathbf{x}_l^{\perp} = \mathbf{0}, \tag{20}$$

The terms are orthogonal, giving $\|\mathbf{x}_l\|_F^2 = n\|\bar{\mathbf{x}}_l\|_2^2 + \|\mathbf{x}_l^{\perp}\|_F^2$. Measurements report the per-element RMS of each term and the common-mode energy share $n\|\bar{\mathbf{x}}_l\|_2^2/\|\mathbf{x}_l\|_F^2$, computed per token in fp32 and then averaged over tokens. As with the activation profiles, the embedding-facing layer $d = 0$, which replicates one vector across streams, is omitted. The split is reported for A1B.

Table 5: **GDHC flattens common-mode growth, with base-dependent mean RMS.** A1B attention at step 50.00K. Growth compares the last and first retained layers; the ratio divides common-mode growth by remainder growth. The mean columns average over layers $1, \ldots, L-1$; $1/n = 0.25$ is the isotropic reference for the energy share. Shading identifies GDHC.

	Common-mode RMS			Growth over depth			Common-mode share		
Run	d=1	d=L−1	mean	common	rem.	ratio	mean	d=L−1	r_{depth}
iHC	1.25	16.71	6.88	13.40	8.44	1.59	0.596	0.636	+0.71
mHC	1.13	14.83	5.40	13.09	4.79	2.73	0.555	0.721	+0.87
GDHC$_{\mathrm{iHC}}$	6.10	6.38	5.67	1.05	1.11	0.94	0.574	0.554	+0.01
GDHC$_{\mathrm{mHC}}$	13.82	5.26	6.53	0.38	0.31	1.23	0.614	0.717	+0.40

A.3 The same two diagnostics on the identity base

Figure 5 repeats the two observations on unmodified iHC, showing that they also occur without learned Sinkhorn mixing. Because the identity base materializes no mixing matrix, it has no logged transfer statistic corresponding to Figure 1(a).

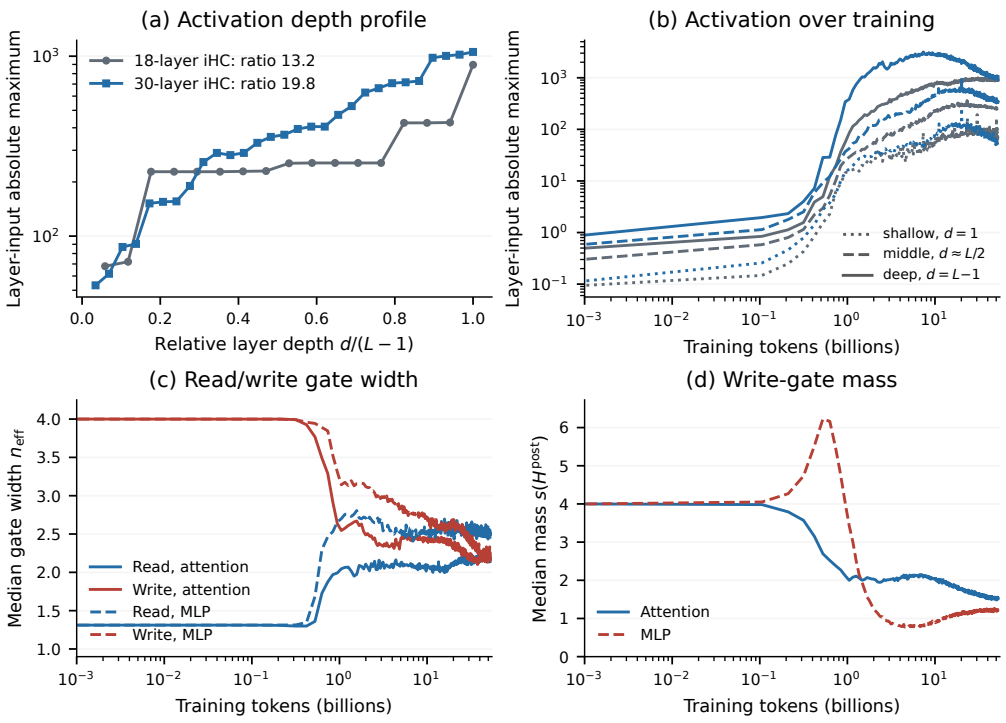

Figure 5: **Activation growth and gate adaptation also occur on iHC.** (a) Activation maxima across depth at step 50.00K. (b) Shallow, middle, and deep maxima over training; grey denotes A500M and blue A1B. (c–d) A1B gate widths and write-gate mass, corresponding to Figure 2.

At step 50.00K, layer-input maxima rise from 68.0 to 896 in A500M and from 53.25 to 1056 in A1B, giving depth ratios of 13.2 and 19.8. Both exceed the corresponding mHC ratios. During training, maxima rise early and subsequently decline while retaining the shallow-to-deep ordering in the logged trajectories.

Gate allocation shifts in the same direction as on mHC. A1B attention read width rises from 1.31 to 2.14, while write width falls from 4.00 to 2.18 and write mass from 4.00 to 1.53. The MLP endpoints are 2.48, 2.26, and 1.24, respectively. Both bases therefore exhibit the activation and allocation patterns that motivate testing explicit retention.

A.4 Can initialization account for gate adaptation?

The gate trajectories in Section 3.2 suggest a simple alternative to changing the update rule: initialize the read gate closer to its later allocation. One exploratory run shifts the static read bias for this purpose, adding no retention control. It ends 0.0015 behind the mHC and iHC references at step 12.89K, so the shifted initialization does not reproduce what retention control provides.

A.5 A1B configurations and loss trajectory

The A1B GDHC configuration enables both retention controls with decay stride $s = 5$ and decay and erase biases of 6 and -6. These choices keep the initial update close to the base while allowing the two controls to adapt during training.

Paired runs begin with similar losses: the reported step-one gap is 1.7×10^{-4}, with larger divergence appearing near step 100. Table 6 tracks both pairs at six checkpoints, extending beyond the two used for downstream evaluation. Training continues past step 80.00K.

Across the first and last checkpoints, the iHC loss gap widens from -0.0027 to -0.0047. On mHC it peaks at -0.0070 near 39.32B tokens and narrows to -0.0014. These are differences between smoothed losses; the 0.0013 standard deviation of raw per-step differences is not an uncertainty

Table 6: **GDHC lowers A1B training loss at all six checkpoints.** Values are EMA(0.99) cross-entropy; Δ uses unrounded values and is negative when GDHC improves. Shaded checkpoints match the downstream budgets in Table 1.

		identity base			doubly stochastic base		
Step	Tokens	iHC	GDHC$_{\text{iHC}}$	Δ	mHC	GDHC$_{\text{mHC}}$	Δ
12.50K	13.11B	1.8121	1.8094	−0.0027	1.8129	1.8083	−0.0046
25.00K	26.22B	1.7441	1.7406	−0.0035	1.7453	1.7406	−0.0047
37.50K	39.32B	1.7205	1.7175	−0.0030	1.7246	1.7176	−0.0070
50.00K	52.43B	1.7084	1.7040	−0.0044	1.7102	1.7059	−0.0043
60.00K	62.92B	1.7054	1.7008	−0.0046	1.7073	1.7035	−0.0038
80.00K	83.89B	1.6881	1.6834	−0.0047	1.6901	1.6886	−0.0014

estimate for them. Appendix A.7 discusses variation within the single paired trajectory. All A1B checkpoints use decay stride 5, selected in the A500M study of Section 5.3.

Table 7: **Activation and update profiles remain distinct at both A1B checkpoints.** $R = A_{L-1}/A_1$ summarizes activation maxima. Read width, write mass, and $\tilde{\rho}$ are attention-layer medians; $r^{\text{mass}}_{\text{depth}}$ and r^{ρ}_{depth} are the depth correlations of write mass and net state change. Gate/update summaries use layers 1–28. Equation 19 defines net state change; Appendix A.2 gives sampling details.

	Activation			Attention gates			Net state change	
Method	A_1	A_{L-1}	R	Read width	Write mass	$r^{\text{mass}}_{\text{depth}}$	$\tilde{\rho}$	r^{ρ}_{depth}
52.43*B tokens (step* 50.00*K)*								
iHC	53.25	1056	19.83	2.14	1.53	+0.93	0.147	+0.12
mHC	59.0	812	13.76	2.13	1.24	+0.88	0.200	−0.02
GDHC$_{\text{iHC}}$	160	370	2.31	2.98	2.02	−0.35	0.289	−0.76
GDHC$_{\text{mHC}}$	246	366	1.49	2.87	2.03	−0.87	0.273	−0.73
83.89*B tokens (step* 80.00*K)*								
iHC	51.25	780	15.22	2.19	1.46	+0.93	0.162	+0.17
mHC	39.5	736	18.63	2.08	1.08	+0.89	0.205	0.00
GDHC$_{\text{iHC}}$	146	246	1.68	3.00	1.75	−0.32	0.319	−0.71
GDHC$_{\text{mHC}}$	248	416	1.68	2.91	1.85	−0.88	0.299	−0.72

A500M and A1B differ in both depth (18 versus 30 layers) and attention heads (8 versus 16), so their comparison does not isolate depth. Table 4 records the shared geometry, MoE widths, and optimization settings.

A.6 Downstream tasks and evaluation protocol

Table 1 reports every task individually. Exams comprise CMMLU (Li et al., 2024), MMLU (Hendrycks et al., 2021), MMLU-Redux (Gema et al., 2025), and MMLU-Pro (Wang et al., 2024). Knowledge comprises HotpotQA (Yang et al., 2018), PopQA (Mallen et al., 2023), and Chinese SimpleQA (He et al., 2025). GPQA forms the science group (Rein et al., 2024); WinoGrande (Sakaguchi et al., 2020) and BBH (Suzgun et al., 2023) form the reasoning group. All scores are percentages. Overall weights each task equally, rather than weighting the four groups equally.

The exams, WinoGrande, and BBH use five-shot prompts without chain-of-thought, GPQA uses ten-shot prompts without chain-of-thought, and the three open-ended knowledge tasks use ten-shot prompts with chain-of-thought and LLM-judge scoring. The judge is DeepSeek-V4-Flash (DeepSeek-AI, 2026), applied identically to every configuration.

A.7 Interpreting small training-loss differences

Training-loss improvements (Appendix A.5) are a few thousandths of a nat, while the late-training single-step loss standard deviation is approximately 0.09. Pairing the data order helps because the

two losses move together: over a 100-step window the paired losses have Pearson $r = 0.9999$, with per-step difference standard deviations of 0.0013 for A1B and 0.0016 for the A500M sweep. These quantities describe variation within a pair of training trajectories, not uncertainty across independent runs.

Under independent white noise, an EMA with new-sample weight $\alpha = 0.01$ would reduce standard deviation by $\sqrt{\alpha/(2-\alpha)} \approx 0.071$. Training losses are temporally correlated, so this calculation does not provide a confidence interval. In one dense-placement comparison, the smoothed differences are -0.0030, -0.0032, and -0.0029 at steps 33.50K, 34.00K, and 34.50K. Their local spread illustrates why differences of a few ten-thousandths should not be interpreted as established improvements. A repeated A500M mHC run with an identical configuration gives 1.771352 against 1.770078 at step 50.00K, a difference of 0.0013 in log loss and 0.008 in PPL.

A.8 Diagnostic effects of decay and erasure

Figure 6 shows the learned controls for A1B GDHC$_{\text{mHC}}$ at step 50.00K. Decay remains mild at most decay-carrying sub-layers, with $1 - a < 0.02$, while the largest layer-mean attention erase gate is about 0.88. The two scales distinguish global attenuation from the strength assigned to the selected direction; realized energy removal additionally depends on key–state alignment (Equation 14).

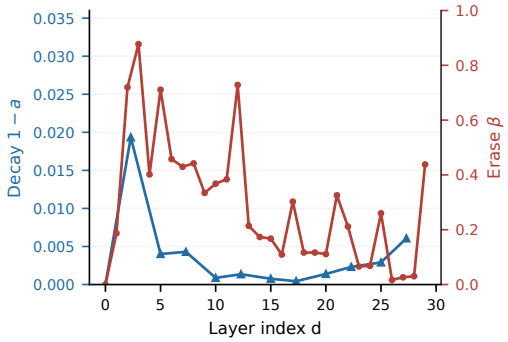

Figure 6: **Mild decay accompanies stronger directional erase gates.** A1B GDHC$_{\text{mHC}}$ at step 50.00K: triangles show $1 - a$ at the sub-layers that carry a decay head (left axis), and circles show attention β (right axis). Attention and MLP decay positions are offset horizontally.

A500M diagnostic snapshots distinguish the controls in Figure 7. On the identity base, scalar decay yields $\mathbf{M}_l = a_l \mathbf{I}$, with equal transfer-column sums and zero off-diagonal entries. Erasure introduces column-sum variation and negative off-diagonal entries; these effects also appear when the controls are combined. For positive write gates, the identity-base off-diagonal entries are $-a_l \beta_l k_i k_j \leq 0$. Gate widths differ across settings. Here λ_j denotes a signed transfer-column sum, not an eigenvalue.

A.9 Isolating the two controls

The iHC erase-only run completes the component comparison at fixed dense placement (Table 8). All four runs share the seed, data order, and schedule and are evaluated at step 50.00K. This complements the decay-first comparisons on both bases in Table 2.

Table 8: **Combining the controls improves on either alone.** A500M iHC training PPL at 52.43B tokens with dense decay where enabled. Δ is relative to iHC; the last column is the additional PPL reduction from enabling the other control.

Variant	PPL	Δ	Additional Δ
iHC	5.876	—	—
+ decay only	5.863	−0.013	−0.007
+ erase only	5.860	−0.016	−0.004
+ decay + erase	**5.856**	**−0.020**	—

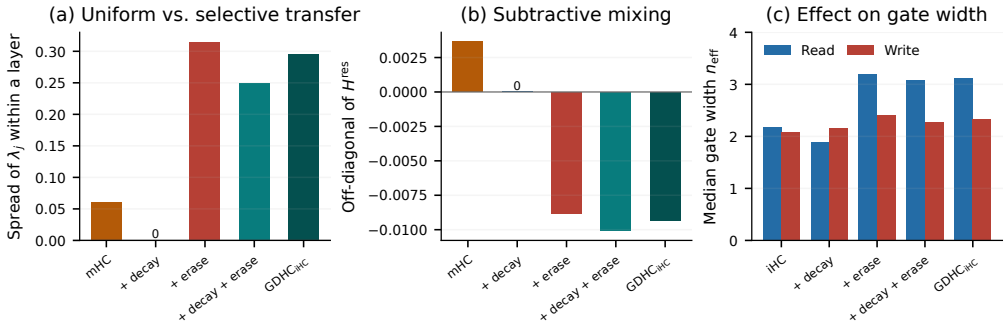

Figure 7: **Decay is uniform; erasure changes transfer by direction.** A500M attention snapshots at step 50.00K (52.43B tokens), medians over interior layers: (a) within-layer spread of signed transfer-column sums, (b) off-diagonal entries of the effective transition (plotted as H^{res}), and (c) identity-base read/write widths. The decay-only identity is zero in (a) and (b) by construction. Variant names follow Table 2; the + arms are on the identity base.

Each control lowers PPL on its own, and their combination improves by 0.007 over decay alone and 0.004 over erasure alone. Erasure gives the larger individual gain in this setting. The joint reduction of 0.020 is smaller than the sum of the individual reductions (0.029), so the measured PPL gains are sub-additive. This establishes a benefit from combining the controls on iHC; the mHC erase-only remains untested.

A.10 What the DDL control reproduces

For the DDL rows of Table 1, the reproduced Deep Delta Learning (Zhang et al., 2026b) control is that paper's expanded residual state instantiation of the Delta-Res block, carrying $d_v = 4$ value channels rather than the scalar value limit $d_v = 1$, with $\beta_{\max} = 2$, β initialized at 1, and a zero-initialized β head so that step-zero behavior is token independent. Its read-side compressor uses a causal short convolution of kernel size 1, that is, purely position-wise reading, rather than the width the reference implementation uses. Model width, depth, expert configuration, sequence length, seed, and data order match the other A1B runs. The residual interfaces differ in their readout, rank-one edit, and coupling between erase and write strengths.

GDHC carries $n = 4$ residual streams and composes its stream-space erase key $\mathbf{k}_l \in \mathbb{R}^n$ with the base mixer. DDL instead expands each hidden feature into $d_v = 4$ value channels. Its state $\mathbf{X}_l \in \mathbb{R}^{C \times d_v}$ is edited along a unit feature-space key $\mathbf{k} \in \mathbb{R}^C$: $\mathbf{X}_{l+1} = (\mathbf{I} - \beta \mathbf{k}\mathbf{k}^\top)\mathbf{X}_l + \beta \mathbf{k}\mathbf{v}^\top = \mathbf{X}_l + \beta \mathbf{k}(\mathbf{v}^\top - \mathbf{k}^\top \mathbf{X}_l)$, where $\mathbf{v} \in \mathbb{R}^{d_v}$. Both designs therefore carry $4C$ residual values per token, but edit different axes.

The reported DDL setting uses $\beta = 2\sigma(\cdot) \in (0, 2)$. Acting on the old state, $\mathbf{I} - \beta \mathbf{k}\mathbf{k}^\top$ approaches the identity as $\beta \to 0$, an orthogonal projection as $\beta \to 1$, and a Householder reflection as $\beta \to 2$; the new write is added separately. DDL ties erase and write strength through β, whereas GDHC parameterizes decay, erase, and write separately. The comparison matches training tokens and the backbone settings listed above, but the two methods have different residual readouts and need not match in parameters or compute. Its scores compare complete implementations, not the editing axis alone.

A.11 Per-stream decay and a wider erase range

Replacing scalar decay with per-stream decay changes reported PPL from 5.842 to 5.841 (approximately 0.00017 in log loss); widening $\beta_{\max}$ from 1 to 2 within the per-stream setting leaves PPL unchanged at the displayed precision. The latter run has a higher endpoint activation ratio (0.61 versus 0.52) and peak ratio (1.10 versus 1.07). These comparisons do not establish a benefit from per-stream decay or the wider range, so scalar decay with $\beta_{\max} = 1$ remains the default. The different Q_a values summarize logged decay means; for per-stream decay their product is neither a token-wise product nor a bound on the product of transition operators.

Table 9: **Per-stream decay and a wider erase range show no resolved PPL gain.** A500M mHC at step 50.00K; the mHC entry and the Δ column follow Table 2. $R_{\text{end}} = A_{L-1}/A_1$, $R_{\text{peak}} = \max_{d\geq 1} A_d/A_1$; Q_a multiplies logged decay means and is not an operator bound. The 0.001 spread among the three variants is smaller than the 0.008 difference between the two mHC executions.

Variant	PPL	Δ PPL	R_{end}	R_{peak}	Q_a
mHC	5.875 ± 0.005	—	6.08	6.08	1.0000
GDHC$_{\text{mHC}}$, scalar decay, $s = 5$	5.842	-0.033	0.43	1.08	0.9723
per-stream decay	5.841	-0.034	0.52	1.07	0.3603
per-stream decay, $\beta_{\max} = 2$	5.841	-0.034	0.61	1.10	0.3635

For $\beta > 1$, the conditioned erase operator $\mathbf{E}_l$ has a negative eigenvalue $1 - \beta$ along the unit key, reflecting rather than attenuating that direction at $\beta = 2$; the norm bound in Equation 12 still holds for an exact base. This is the regime that extends state-tracking expressivity in linear recurrences (Grazzi et al., 2025).

A.12 A QUALITATIVE SNAPSHOT FROM DEEPSEEK-V4-FLASH

Figure 8 compares gate allocation in the released 43-layer DeepSeek-V4-Flash mHC checkpoint (DeepSeek-AI, 2026). Reads usually distribute weight across several streams, while some writes assign more than 90% of their gate mass to one stream. This snapshot complements the internally measured trajectories in Section 3.2.

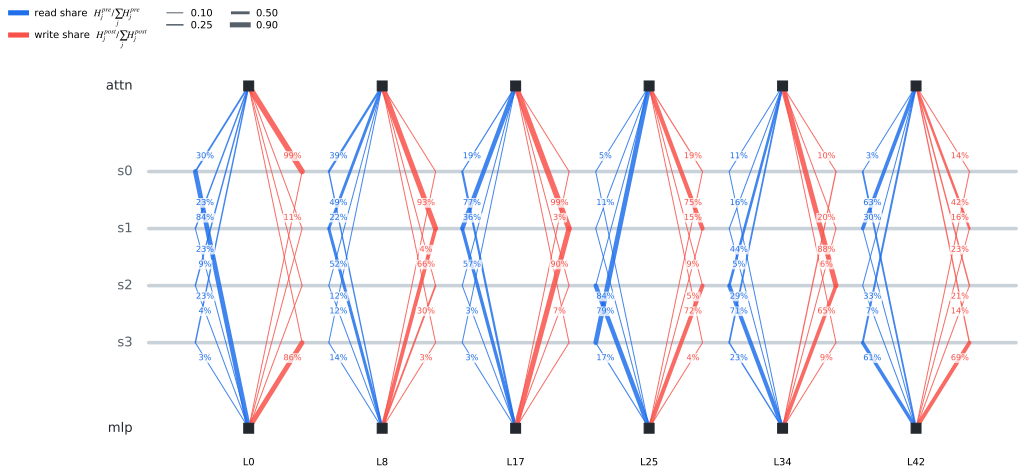

Figure 8: **Read/write allocation in DeepSeek-V4-Flash.** Six layers from the 43-layer mHC checkpoint show attention (top) and MLP (bottom). Blue and red edges denote normalized read and write shares. Shares below 3% are unlabeled; normalization is local to each node.

A.13 LIMITATIONS

The present experiments cover 0.55B and 0.91B activated parameters and up to 83.89B training tokens. The main remaining step is validation at larger model and data scales to test whether the benefits persist with greater capacity and longer training.

Sensitivity to stream width and decay placement remain to be quantified. Alternative erase directions and matched retention parameterizations would help distinguish the roles of the tied key and the added transition flexibility.