EQUILIBRIUM UPDATES FOR RESIDUAL LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-stream residual networks combine mechanisms for retaining intermediate state and injecting sublayer output. When mixing and writing are independently parameterized, changing retention also changes the state maintained by repeated input. This paper introduces Equilibrium-coordinate Hyper-Connections (eHC), which parameterize residual writes via this conditional equilibrium using the existing write head. Given a doubly stochastic mixing matrix P, eHC writes $g\mathbf{1}+(I-P)q$, sharing a gain g and zero-mean coordinates q. This separates additive accumulation of the stream mean from target-centered disagreement dynamics, making the direct sensitivity to mixing dependent on target error. For fixed positive mixing, equilibrium coordinates are in one-to-one correspondence with signed affine writes, so eHC changes the learning of equivalent writes without changing their local expressive capacity. Controlled experiments with language models indicate improved language modeling and zero-shot transfer compared to mHC and a matched signed-write baseline with little additional training cost. Gradient ablations identify benefits of adaptive write coordinates and their gradient coupling to mixing, and retention interventions reveal that preserving the target reduces predictive sensitivity. These results suggest that parameterizing residual writes by the states they should sustain, rather than learning injection independently of retention, may improve residual networks.

1 INTRODUCTION

Residual networks update an intermediate representation at each layer (He et al., 2016a;b). Attention and feed-forward sublayers write to this state using additive connections (Vaswani et al., 2017). Hyper-Connections extend the state space by maintaining multiple residual streams and learning how to read, mix, and write to them, without needing the computational sublayers to be wider (Zhu et al., 2024). Manifold-Constrained Hyper-Connections (mHC) use doubly stochastic mixing matrices whose products preserve non-negativity and row and column sums of one across depth (Xie et al., 2026).

The equilibrium sustained by repeated input depends jointly on mixing and writing. For example, the scalar recurrence $s_{l+1} = \lambda s_l + bf$, with $0 \leq \lambda < 1$ and persistent input f, converges to $s_\infty = bf/(1-\lambda)$. Increasing λ at fixed b prolongs retention but also changes the equilibrium magnitude. Preserving the equilibrium therefore requires a compensating change in b. This dependence becomes stronger as λ approaches one, even while the recurrence remains stable. The same issue arises when a multi-stream residual layer learns both a mixing matrix and the coefficients that inject each sublayer output.

Residual scaling, initialization, and normalization address signal and gradient flow in deep neural networks (Zhang et al., 2019; Bachlechner et al., 2020; Xiong et al., 2020; Wang et al., 2022). Analyses of recurrent learning further connect persistent states to parameter sensitivity and input-injection scaling (Pascanu et al., 2013; Orvieto et al., 2023; Zucchet & Orvieto, 2024). These results motivate parameterizing a residual write through the state it sustains. In the scalar case, setting $b = (1-\lambda)a$ gives equilibrium af for every admissible λ. The parameter a then determines the equilibrium, while λ determines the convergence rate. Doubly stochastic mixing admits the same separation on the subspace of differences between streams.

We introduce *Equilibrium-coordinate Hyper-Connections* (eHC). Given the residual state X_l, mixing matrix P_l, and sublayer output f_l, eHC predicts a shared gain g_l and zero-mean coordinates q_l and

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.

updates the state as

$$X_{l+1} = P_l X_l + \big[g_l \mathbf{1} + (I - P_l) q_l\big] f_l. \tag{1}$$

Writing $X_l = \mathbf{1}\bar{x}_l + D_l$ separates additive accumulation of the stream mean $\bar{x}_l$ from the target-centered disagreement update $D_{l+1} = q_l f_l + P_l(D_l - q_l f_l)$. Each sublayer thus predicts a conditional equilibrium through its existing write head. Figure 1 illustrates how this construction separates the target from retention dynamics. The resulting write requires only one small matrix-vector product, while retaining the same computational sublayers and residual width.

This parameterization also changes how mixing is learned. At a fixed target, a mixing perturbation acts on the error $D_l - q_l f_l$ and has no direct effect when that error is zero. A target-tracking bound characterizes how changes in the target accumulate across sublayers. At fixed positive mixing, equilibrium coordinates represent exactly the same updates as signed affine writes. This correspondence defines the Signed-Direct baseline, which matches eHC in parameter count, initial function, and initial write-head Jacobian.

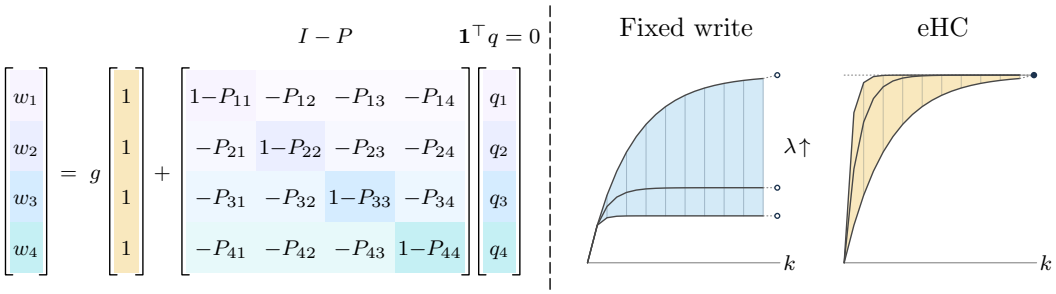


Figure 1: eHC preserves the target as retention varies. The left panel expands the four-stream write $w = g\mathbf{1} + (I - P)q$. The shared gain g contributes equally to each stream, while $I - P$ converts zero-mean target coordinates q into a write adapted to mixing P. The right panels show analytic scalar responses to a constant input from $s_0 = 0$, with k indexing updates and $\lambda \in \{0.2, 0.5, 0.8\}$ controlling retention. A fixed write $bf = 0.2$ yields equilibria $bf/(1 - \lambda)$, whereas eHC uses $(1 - \lambda)af$ with target $af = 1$. Increasing retention slows convergence while preserving the eHC target. Shading spans the responses, and circles mark their limiting states.

Experiments with 125M- and 355M-parameter language models trained on FineWeb-Edu with three paired seeds show that eHC reduces test loss relative to mHC and Signed-Direct, and performs better on two additional corpora and four zero-shot tasks. At 355M parameters, test loss is 0.0091 nats/token lower than mHC and 0.0052 lower than Signed-Direct. A forward-identical gradient ablation reveals that the write-to-mixing derivative is responsible for a 0.0016 nats/token gain at 125M. In trained models, retaining equilibrium coordinates during a mixing intervention reduces predictive sensitivity by 37.3% at the intermediate strength. A model with a deeper, parameter-matched backbone shows a 0.0066 nats/token reduction in test loss over Signed-Direct, up from 0.0041 nats/token. Training speed remains comparable to mHC, with only a 0.7% increase in step time at 355M parameters. At equal elapsed training time, eHC retains a 0.0088 nats/token loss advantage over mHC.

We make three contributions. First, we introduce equilibrium coordinates for residual writes while retaining mHC's sublayers and residual width. Second, we establish their equivalence to signed affine writes, derive a target-tracking bound, and characterize the innovation-weighted mixing derivative. Third, paired language-model comparisons isolate the benefits of adaptive coordinates and their gradient coupling, with a larger gain in a deeper, parameter-matched backbone.

2 Residual transport and repeated writes

l indexes the attention and feed-forward sublayers. $X_l \in \mathbb{R}^{n \times C}$ is the input matrix to the l-th sublayer, consisting of n residual streams of width C. Batch and sequence dimensions are omitted. We write

$$f_l = F_l(r_l^\top X_l), \qquad X_{l+1} = P_l X_l + w_l f_l, \tag{2}$$

where $r_l, w_l \in \mathbb{R}^n$ and $P_l \in \mathbb{R}^{n \times n}$ are coefficients generated from the current state. Connection maps act across streams, preserving the width C of the input to F_l. Attention remains causal along the sequence.

As in mHC, the mixing matrices satisfy

$$P_l \geq 0, \qquad P_l \mathbf{1} = \mathbf{1}, \qquad \mathbf{1}^\top P_l = \mathbf{1}^\top. \tag{3}$$

Define $J = \mathbf{1}\mathbf{1}^\top/n$ and $\Pi = I - J$, and decompose the state as

$$X_l = \mathbf{1}\bar{x}_l + D_l, \qquad \bar{x}_l = \frac{1}{n}\mathbf{1}^\top X_l, \qquad D_l = \Pi X_l. \tag{4}$$

The mean $\bar{x}_l$ describes the common component, while D_l captures disagreement among streams. Double stochasticity implies $P_l J = J P_l = J$. Decomposing the write as $w_l = g_l \mathbf{1} + b_l$, with $g_l = \mathbf{1}^\top w_l/n$ and $b_l = \Pi w_l$, therefore gives

$$\bar{x}_{l+1} = \bar{x}_l + g_l f_l, \qquad D_{l+1} = P_l D_l + b_l f_l. \tag{5}$$

Equation (5) separates the additive accumulation of the mean from the driven affine update of disagreement for the current coefficients and sublayer output.

For fixed coefficients and input f, a disagreement equilibrium satisfies $(I - P)D_\star = bf$. Its value depends jointly on the write b and the restriction of $I - P$ to the zero-mean subspace. Learning these quantities independently therefore allows changes in mixing to alter the maintained state. This observation motivates using the equilibrium itself as the write coordinate.

3 Equilibrium-coordinate hyper-connections

3.1 Learning the target of a write

Let $\mathcal{S} = \{q \in \mathbb{R}^n : \mathbf{1}^\top q = 0\}$. eHC predicts $q_l \in \mathcal{S}$ and defines

$$b_l = (I - P_l)q_l, \qquad T_l = q_l f_l. \tag{6}$$

Substituting into Equation (5) gives

$$\boxed{\bar{x}_{l+1} = \bar{x}_l + g_l f_l, \qquad D_{l+1} = T_l + P_l(D_l - T_l).} \tag{7}$$

T_l is a conditional equilibrium of the current affine update. It is computed from the current state and the existing sublayer output, and has stream–feature rank at most one. Both the sign and amplitude of q_l are learned. Different sublayers can produce different targets.

Equation (7) preserves an additive residual update of the stream mean. On the orthogonal complement, P_l acts on the target error $D_l - T_l$. For a fixed target, learning P_l changes the convergence dynamics while preserving the equilibrium.

The connection to retention can also be made mode-by-mode. Given a left eigenvector $u^\top P = \lambda u^\top$ with $u^\top \mathbf{1} = 0$ and $\lambda \in [0, 1)$ real,

$$u^\top D^+ = \lambda u^\top D + (1 - \lambda)(u^\top q) f. \tag{8}$$

The conditional equilibrium is $(u^\top q)f$, while λ governs convergence, recovering the separation illustrated in Figure 1. eHC realizes this mode-wise relation by applying P directly in the stream basis, without an eigendecomposition.

3.2 Connection parameterization and computational cost

The connection coefficients depend on the RMS-normalized residual state (Zhang & Sennrich, 2019)

$$h_l = \frac{\mathrm{vec}(X_l)}{\sqrt{\|\mathrm{vec}(X_l)\|_2^2/(nC) + \epsilon}}. \tag{9}$$

Three affine heads compute read logits $a_l \in \mathbb{R}^n$, mixing logits $S_l \in \mathbb{R}^{n \times n}$, and write-coordinate logits $z_l \in \mathbb{R}^n$. Each head has a learned scalar multiplier α on its state-dependent projection, as in $z_l = c_l^z + \alpha_l^z W_l^z h_l$. We set $r_l = \sigma(a_l)$ and apply log-space Sinkhorn normalization (Sinkhorn, 1964; Cuturi, 2013) to obtain P_l. Write-head outputs are

$$m_l = \frac{1}{n}\mathbf{1}^\top z_l, \qquad g_l = 2\sigma(m_l), \qquad q_l = z_l - m_l \mathbf{1}. \tag{10}$$

Equation (10) uses one coordinate for the shared gain and $n-1$ for the disagreement target. The connection head outputs n^2+2n scalars, the same as mHC.

The effective write $w_l = g_l\mathbf{1} + q_l - P_l q_l$ avoids forming the full target T_l. Its additional cost is one $n \times n$ matrix-vector product and $O(n)$ elementwise operations, independent of the sublayer width. Attention and feed-forward computation retain their original dimensions. Connection coefficients are evaluated in FP32, with BF16 backbone computation and stored activations and FP32 accumulation in the residual update. The corresponding gradients reuse quantities already required by the residual update, as derived in Appendix A.10.

4 Properties of equilibrium updates

We analyze the affine residual update conditional on the current input, connection coefficients, and sublayer output. The following results characterize target tracking, the space of representable writes, and the dependence of the mixing gradient on target error.

4.1 Target tracking

Proposition 1 (Local target contraction)**.** *Let P be doubly stochastic, $q \in \mathcal{S}$, and $T = qf$. eHC satisfies*

$$D^+ - T = P(D-T), \qquad \|D^+ - T\|_F \le \|D-T\|_F. \tag{11}$$

The identity follows directly from Equation (7), and the norm bound follows from $\|P\|_2 \le \sqrt{\|P\|_1\|P\|_\infty} = 1$. For strictly positive P, the restriction to $\mathcal{S}$ is contractive.

The targets can differ at different depths. Let $e_l = D_l - T_l$ and $\rho_l = \|P_l - J\|_2$. Then

$$e_{l+1} = P_l e_l + T_l - T_{l+1}. \tag{12}$$

If $\rho_l \le \rho < 1$ and $\|T_{l+1} - T_l\|_F \le \delta$, repeated substitution yields

$$\|e_L\|_F \le \rho^L \|e_0\|_F + \delta \frac{1-\rho^L}{1-\rho}. \tag{13}$$

The bound separates decay of the initial error from the accumulated effect of target variation across depth. Appendix A gives the corresponding result for nonuniform ρ_l.

4.2 Affine write capacity

Proposition 2 (Equilibrium coordinates for signed writes)**.** *If P is strictly positive and doubly stochastic, every write vector w with mean $g = \mathbf{1}^\top w/n \in (0,2)$ has a unique representation*

$$w = g\mathbf{1} + (I-P)q, \qquad q = (I - P + J)^{-1}(w - g\mathbf{1}) \in \mathcal{S}. \tag{14}$$

Strict positivity implies that eigenvalue one is simple. On $\mathcal{S}$, the operator $I-P$ is invertible. Adding J gives the identity on the common direction and Equation (14) follows.

At fixed positive mixing, Proposition 2 establishes a bijection between equilibrium coordinates and signed affine writes with the same range of shared gains. Equation (14) establishes this equivalence analytically, whereas eHC predicts q directly. The correspondence also motivates the Signed-Direct baseline in Section 5, which has an independent write parameterization with matched initialization and write-head scale.

4.3 Differentiation with respect to mixing

Proposition 3 (Innovation-weighted mixing)**.** *Let X, q, g, f be fixed and dP such that $dP\mathbf{1} = 0$ and $\mathbf{1}^\top dP = 0$. The direct variation of an eHC update is*

$$dX^+_{\mathrm{eHC}} = dP(D-T). \tag{15}$$

For an independent write vector held fixed, the corresponding variation is $dX^+_{\mathrm{direct}} = dPD$.

Table 1: Backbone configurations, including four-stream connection parameters.

Model	Blocks	Width	FFN	Heads	Parameters	Tokens
Base-125	12	768	2,048	12	125.36M	1.074B
Base-355	24	1,024	2,688	16	355.13M	2.147B
Deep-125	24	512	1,952	8	125.27M	1.074B

Differentiating, we obtain $dX^+ = dPX - dPqf$. The constraint $dP\mathbf{1} = 0$ eliminates the common component. If $G = \partial\mathcal{L}/\partial X^+$, the unprojected coefficient adjoint is

$$\nabla_P \mathcal{L} = G(X - T)^\top. \tag{16}$$

Composing the adjoint with the Sinkhorn Jacobian yields the mixing-logit gradient. When $D = T$, the direct derivative vanishes. Thus, an admissible mixing perturbation preserves the represented target.

For the scalar update $s^+ = \lambda s + (1 - \lambda)af$, the equilibrium derivative is $\partial s_\infty/\partial\lambda|_a = 0$. At fixed b in $s^+ = \lambda s + bf$, it is $bf/(1 - \lambda)^2$. eHC incorporates the compensating write into its parameterization. The gradient ablation isolates the effect of this compensation on learning, while the retention intervention tests its effect on predictions.

4.4 Composition across depth

Consider a sequence of sublayers with the same conditional target T (possibly different mixing matrices). Iterative application of Equation (7) yields

$$D_L = T + P_{L-1} \cdots P_0(D_0 - T). \tag{17}$$

Any admissible sequence preserves T. A perturbation of a single mixing matrix acts on the propagated initial error $D_0 - T$, where the other matrices determine its contribution to the output. The derivative and its norm bound are in Appendix A.5. When targets change, Equation (12) adds their successive differences.

These identities motivate complementary tests of learning and prediction. Removing the write-to-mixing gradient isolates the optimization effect of Equation (15). Increasing retention with either a recomputed or fixed write tests whether preserving the target reduces predictive sensitivity. Both comparisons use the sublayers and residual states of the primary language models.

5 Experiments

5.1 Models, comparisons, and training

We train decoder-only Transformers with the LLaMA block architecture (Touvron et al., 2023), pre-normalization, rotary embeddings (Su et al., 2021), SwiGLU feed-forward layers (Shazeer, 2020), and causal multi-head attention. All models use tied input and output embeddings (Press & Wolf, 2017) and the Pythia tokenizer (Biderman et al., 2023). Multi-stream models replicate the token embeddings across four streams and apply the language-model head to their RMS-normalized final mean.

We draw pretraining data from the ten-billion-token sample of FineWeb-Edu (Penedo et al., 2024). Content hashes of normalized documents define disjoint training, development, and test partitions, with 2^{22} prediction tokens in each held-out partition. We remove duplicate documents and exclude training documents that share a 13-gram with evaluation data, following substring-based decontamination (Lee et al., 2022). Within each model size and seed, all methods receive the same tokens in the same order. Table 1 gives the model dimensions and training-token budgets.

The baselines are mHC, whose write is $w = 2\sigma(z)$, and Signed-Direct, defined by

$$m = \mathrm{mean}(z), \quad g = 2\sigma(m), \quad q = \Pi z, \qquad w = g\mathbf{1} + 0.1q. \tag{18}$$

Signed-Direct uses the same write-head outputs as eHC but maps q to the write through a fixed coefficient. The 125M comparison additionally includes a standard single-stream Transformer with 123.59M parameters and the same computational backbone.

Table 2: FineWeb-Edu test loss in nats/token, averaged over three paired seeds ($\pm$ standard deviation).

Method	Base-125	Base-355
Standard residual	3.3566 ± 0.0021	n/a
mHC	3.3467 ± 0.0017	3.1527 ± 0.0016
Signed-Direct	3.3435 ± 0.0018	3.1488 ± 0.0016
eHC	$\mathbf{3.3394} \pm 0.0021$	$\mathbf{3.1436} \pm 0.0020$

All multi-stream methods share the initial mixing, read, and gain

$$P_0 = 0.9I + 0.1J, \qquad r_0 = \mathbf{1}/n, \qquad g_0 = 1. \tag{19}$$

Across successive sublayers, q_0 cycles through permutations of $0.1(-3, -1, 1, 3)^\top$, with common initial write $w_0 = \mathbf{1} + (I - P_0)q_0$. Each parameterization realizes these coefficients through its static biases. State-dependent projection matrices start at zero, and their learned scalar multipliers start at 0.01. eHC and Signed-Direct also share the initial write-head Jacobian

$$\left.\frac{\partial w}{\partial z}\right|_0 = \frac{1}{2}J + 0.1\Pi. \tag{20}$$

Signed-Direct retains the map $0.1\Pi = I - P_0$ throughout training, whereas eHC adapts it to the learned mixing matrix. Paired models share backbone parameters at initialization and therefore begin with the same language-model function.

Each comparison uses three paired seeds, a context length of 2,048, and 2^{17} prediction tokens per AdamW update (Loshchilov & Hutter, 2019). The 125M and 355M models train for 8,192 and 16,384 updates, with peak learning rates of 6×10^{-4} and 3×10^{-4}, respectively. After 256 warmup updates, the learning rate decays to one tenth of its peak on a cosine schedule shared by all parameter groups. Weight decay is 0.1 for backbone matrices and zero for connection heads. Training uses BF16 computation, FP32 optimizer states, and FlashAttention-2 (Dao, 2023). Appendix B specifies the data partitions, optimization, and evaluation protocols.

The primary metric is token-weighted test negative log-likelihood. Loss confidence intervals use within-seed differences between each baseline and eHC. Downstream comparisons preserve the same prompts and examples across methods, with paired resampling of seeds and evaluation items. Development loss is measured every 512 updates, and Figure 2 traces the paired differences over the final three quarters of training. Final test losses use the complete held-out test partition.

5.2 Language modeling and transfer

eHC achieves the lowest test loss at both model sizes (Table 2). At 125M parameters, the reductions relative to mHC and Signed-Direct are 0.0073 and 0.0041 nats/token, with paired 95% confidence intervals of $[0.0051, 0.0095]$ and $[0.0024, 0.0058]$. At 355M parameters, these reductions increase to 0.0091 and 0.0052 nats/token. The advantage over Signed-Direct persists despite matched signed-write capacity and initial write-head Jacobians, supporting a benefit from adapting the coordinate map to learned mixing.

The advantage over Signed-Direct grows during training (Figure 2). At 355M parameters, the loss gap increases from 0.0019 after one quarter of training to 0.0034 halfway through and 0.0052 at completion. The 125M gap similarly grows from 0.0022 to 0.0041 over the same fractions of training. These differences develop from a matched initial function and local write-head scale as the two parameterizations are learned. Appendix D gives the absolute held-out trajectories.

The gains extend to WikiText-103 (WT103) and English C4 (Merity et al., 2016; Raffel et al., 2020), and to zero-shot HellaSwag (HS), PIQA, ARC-Easy, and LAMBADA (Zellers et al., 2019; Bisk et al., 2020; Clark et al., 2018; Paperno et al., 2016). With the same pretrained checkpoints, eHC has lower perplexity on both corpora and higher accuracy on all four tasks (Table 3). Its accuracy gains over Signed-Direct are 0.4, 0.3, 0.5, and 0.7 percentage points, respectively. The equal-weight mean gain is 0.5 points, with a paired bootstrap interval of $[0.2, 0.8]$. These improvements require no task-specific adaptation and follow the ordering observed in FineWeb-Edu test loss.

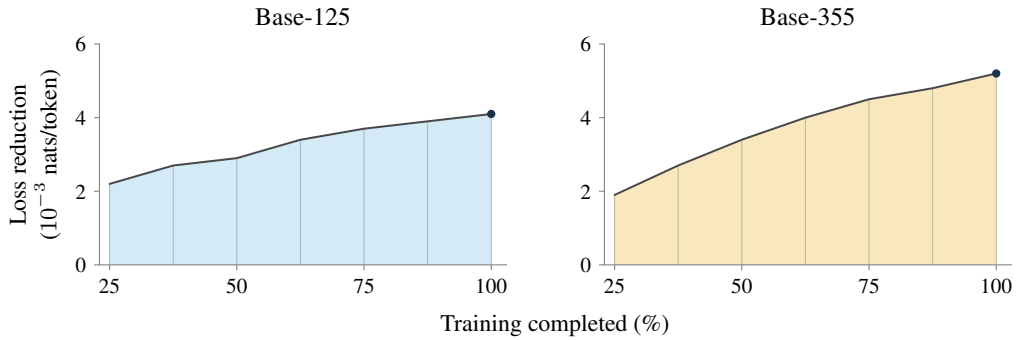


Figure 2: eHC's held-out loss advantage grows during pretraining. Shaded heights show reductions relative to Signed-Direct, averaged over three paired seeds. Both panels use the same axis scales.

Table 3: Cross-corpus perplexity and zero-shot accuracy of Base-355 models.

	Perplexity ↓		Accuracy (%) ↑				
Method	WT103	C4	HS	PIQA	ARC-E	LAMBADA	Mean
mHC	34.18	29.57	30.1	58.5	41.8	22.4	38.2
Signed-Direct	33.97	29.41	30.4	58.8	42.2	22.9	38.6
eHC	**33.78**	**29.25**	**30.8**	**59.1**	**42.7**	**23.6**	**39.1**

5.3 Adaptive coordinates and the backward path

To isolate the contribution of the mixing derivative in Proposition 3, we train eHC-Detached with the write

$$w = g\mathbf{1} + \big(I - \mathrm{sg}(P)\big)q. \tag{21}$$

where sg denotes stop-gradient. At identical parameters, eHC-Detached has the same forward map as eHC, but excludes the derivative of the write with respect to P.

Allowing the coordinate map to follow P reduces loss from 3.3435 to 3.3410 (Table 4). Restoring the gradient through this adaptive write yields a further 0.0016 nats/token reduction, with a paired 95% interval of $[0.0004, 0.0028]$. Because eHC and eHC-Detached coincide in the forward pass at identical parameters, their difference after training isolates the contribution of the write-to-mixing gradient.

5.4 Changing retention without rewriting the target

We test predictive sensitivity to retention in trained Base-355 models on held-out natural text. In the final quarter of residual sublayers, the mixing matrix is perturbed towards the identity according to

$$P_\eta = (1-\eta)P + \eta I, \qquad \eta \in \{0.01, 0.025, 0.05\}. \tag{22}$$

The *equilibrium-preserving* intervention recomputes the write $w_\eta = g\mathbf{1} + (I - P_\eta)q$. The *fixed-write* intervention uses the same P_η but the original write $g\mathbf{1} + (I - P)q$. For a fixed local input, the change in state is

$$\delta X^+_{\mathrm{eq}} = \eta(I-P)(D-T), \qquad \delta X^+_{\mathrm{fixed}} = \eta(I-P)D. \tag{23}$$

The perturbed state propagates through all subsequent sublayers. Predictive KL divergence from the unperturbed model is measured over the last 32 positions of each of 64 held-out sequences and normalized by the number of scored positions and η^2.

The equilibrium-preserving/fixed-write sensitivity ratios are 0.611, 0.627, and 0.654 across increasing intervention strengths (Figure 3). At $\eta = 0.025$, preserving the target reduces predictive KL by 37.3%, with ratio interval $[0.582, 0.674]$. The corresponding local state changes are compared through

$$\mathcal{R} = \frac{\sum_{l,t} \|(I - P_{l,t})(D_{l,t} - T_{l,t})\|_F^2}{\sum_{l,t} \|(I - P_{l,t})D_{l,t}\|_F^2 + \epsilon}, \tag{24}$$

Table 4: Write-coordinate ablations on Base-125. Gains are relative to Signed-Direct.

Method	Coordinate map	Mixing derivative	Test loss	Gain vs. Signed
Signed-Direct	$I - P_0$	$dP\,D$	3.3435	0.0000
eHC-Detached	$I - P$	$dP\,D$	3.3410	0.0025
eHC	$I - P$	$dP(D - T)$	**3.3394**	**0.0041**

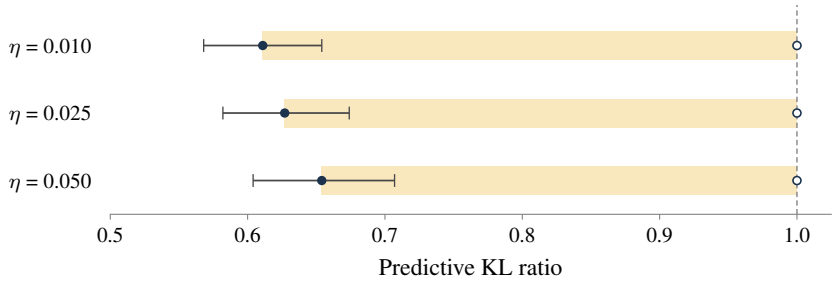


Figure 3: Equilibrium-preserving writes reduce predictive sensitivity across intervention strengths η. Filled circles and bars show the equilibrium-preserving/fixed-write KL ratio and 95% confidence intervals. Open circles mark the fixed-write reference at one, and shading spans the reduction from this reference.

The measured value is 0.607, consistent with the reduction in predictive sensitivity. Thus, both the local state response and the final predictive distribution are less sensitive to retention changes when the write preserves the current target.

5.5 Depth and training efficiency

To examine how the benefit varies with depth, we compare the 12-block Base-125 and 24-block Deep-125 backbones in Table 1. Their parameter counts are closely matched, and both receive the same training-token budget. Signed-Direct and eHC are evaluated in each backbone using three paired seeds.

eHC's advantage over Signed-Direct increases from 0.0041 to 0.0066 nats/token in the deeper backbone (Table 5). The interaction between depth allocation and parameterization is 0.0025, with a paired 95% interval of $[0.0009, 0.0041]$. Under a matched parameter budget, equilibrium coordinates therefore provide a larger improvement in the model with twice as many residual updates.

eHC adds 0.7% to step time and 0.1 GB to peak allocated memory relative to mHC (Table 6). At equal elapsed training time, its loss remains 0.0088 below mHC and 0.0046 below Signed-Direct. This small overhead is consistent with a correction confined to the low-dimensional connection coefficients and their gradients, while backbone computation and the full residual-state update retain the same form.

6 Related work

Residual architecture and optimization. Additive identity paths are direct paths for signal and gradient propagation in residual networks (He et al., 2016a;b). Highway Networks learn how much transformed and carried state to use in each update (Srivastava et al., 2015). An ensemble interpretation of residual networks shows that much of the gradient flow in residual networks comes from paths shorter than the full network (Veit et al., 2016). DenseNet concatenates earlier features to make them available to later layers without summing them into a single state (Huang et al., 2017). Initialization and normalization control the scale of these updates. Fixup rescales initial weights, ReZero starts residual gates at zero, and DeepNet combines residual scaling with initialization in deep Transformers (Zhang et al., 2019; Bachlechner et al., 2020; Wang et al., 2022). Pre-normalization controls the initial gradient magnitudes that otherwise motivate learning-rate warmup (Xiong et al.,

Table 5: Parameter-matched depth comparison. Gains are relative to Signed-Direct.

Allocation	Parameters	Signed-Direct	eHC	Gain
12 blocks, width 768	125.36M	3.3435	**3.3394**	0.0041
24 blocks, width 512	125.27M	3.3554	**3.3488**	0.0066

Table 6: Base-355 training efficiency and held-out loss at a common elapsed time.

Method	Step (ms)	Peak memory (GB)	Tokens/s	Common-time loss
mHC	612.4	31.8	214,030	3.1530
Signed-Direct	610.8	31.7	214,591	3.1488
eHC	616.5	31.9	212,607	3.1442

2020). Hyper-Connections expand the residual state and learn its read, write, and mixing coefficients (Zhu et al., 2024). mHC constrains mixing to doubly stochastic matrices, which preserve nonnegativity and unit row and column sums under composition (Xie et al., 2026).

Recurrent memory and timescales. Long horizon gradient propagation relies on products of recurrent Jacobians (Pascanu et al., 2013) and identity initialization gives a useful starting transition for rectified recurrent networks (Le et al., 2015). Time rescaling invariance allows for a derivation of recurrent gates and suggests timescale-aware initialization (Tallec & Ollivier, 2018). HiPPO builds recurrent memory by projecting past inputs onto polynomial bases (Gu et al., 2020). Structured state-space models make these transitions computationally feasible, with diagonal parameterizations achieving strong long-sequence performance with appropriate initialization (Gu et al., 2022a;b). LRU combines diagonal recurrence with input normalization (Orvieto et al., 2023). Additional analysis of long-memory parameter sensitivity connects it to input correlations and recurrence coordinates (Zucchet & Orvieto, 2024). Input-dependent state-space coefficients in Mamba allow for selective retention (Gu & Dao, 2024). State-space duality expresses related computations through structured attention matrices (Dao & Gu, 2024). Mega learns moving-average components with gated attention (Ma et al., 2023). xLSTM combines exponential gates with scalar and matrix memories and their normalization states (Beck et al., 2024).

Associative and equilibrium updates. Linear attention can be computed recurrently by accumulating the outer product of keys and values (Katharopoulos et al., 2020). Viewing the state as fast weights leads to delta-rule updates that subtract the retrieved value before writing a new association (Schlag et al., 2021). DeltaNet parallelizes this sequence of updates using a compact representation of products of Householder matrices (Yang et al., 2024). Gated Delta Networks combine the delta rule with gated forgetting to enable both association-specific revision and decay of the full memory (Yang et al., 2025). Deep Delta Learning performs target-based updates across network depth (Zhang et al., 2026). Deep Equilibrium Models learn representations by solving for fixed points of weight-tied nonlinear transformations and use implicit differentiation to train the system (Bai et al., 2019).

7 Discussion and Conclusion

mHC constrains residual transport through doubly stochastic mixing, yet the disagreement equilibrium under repeated input still depends on how writing interacts with that transport. eHC retains these constraints and specifies a conditional target for disagreement, adapting the write to preserve that target as retention changes. It improves language modeling and transfer over mHC with unchanged computational sublayers and comparable training speed.

For residual architectures with learned transport, these findings favor parameterizing the state to be maintained and adapting input injection to the learned retention dynamics. The direct mixing gradient is weighted by the current target error. Comparisons with Signed-Direct and eHC-Detached indicate that both the adaptive coordinates and their gradient coupling contribute beyond the freedom to use signed coefficients. Studying how targets vary and how disagreement contracts across depth could help explain the larger gain over Signed-Direct in the parameter-matched deeper backbone.

AI USE STATEMENT

We used large language models (LLMs) to improve grammar and correct spelling errors in the manuscript. The authors take full responsibility for the final content of this work.

REFERENCES

Thomas Bachlechner, Bodhisattwa Prasad Majumder, Huanru Henry Mao, Garrison W. Cottrell, and Julian McAuley. ReZero is all you need: Fast convergence at large depth. *arXiv preprint arXiv:2003.04887*, 2020.

Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. Deep equilibrium models. In *Advances in Neural Information Processing Systems*, 2019.

Maximilian Beck, Korbinian Pöppel, Markus Spanring, Andreas Auer, Oleksandra Prudnikova, Michael Kopp, Günter Klambauer, Johannes Brandstetter, and Sepp Hochreiter. xLSTM: Extended long short-term memory. In *Advances in Neural Information Processing Systems*, 2024.

Stella Biderman, Hailey Schoelkopf, Quentin Anthony, et al. Pythia: A suite for analyzing large language models across training and scaling. In *Proceedings of the International Conference on Machine Learning*, 2023.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. PIQA: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

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

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems*, 2013.

Tri Dao. FlashAttention-2: Faster attention with better parallelism and work partitioning. *arXiv preprint arXiv:2307.08691*, 2023.

Tri Dao and Albert Gu. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. In *Proceedings of the International Conference on Machine Learning*, 2024.

Yann N. Dauphin, Angela Fan, Michael Auli, and David Grangier. Language modeling with gated convolutional networks. In *Proceedings of the International Conference on Machine Learning*, 2017.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *Conference on Language Modeling*, 2024.

Albert Gu, Tri Dao, Stefano Ermon, Atri Rudra, and Christopher Ré. HiPPO: Recurrent memory with optimal polynomial projections. In *Advances in Neural Information Processing Systems*, 2020.

Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations*, 2022a.

Albert Gu, Ankit Gupta, Karan Goel, and Christopher Ré. On the parameterization and initialization of diagonal state space models. In *Advances in Neural Information Processing Systems*, 2022b.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European Conference on Computer Vision*, 2016b.

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*, 2016a.

Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are RNNs: Fast autoregressive transformers with linear attention. In *Proceedings of the International Conference on Machine Learning*, 2020.

Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. Deduplicating training data makes language models better. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2022.

Quoc V. Le, Navdeep Jaitly, and Geoffrey E. Hinton. A simple way to initialize recurrent networks of rectified linear units. *arXiv preprint arXiv:1504.00941*, 2015.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Xuezhe Ma, Chunting Zhou, Xiang Kong, Junxian He, Liangke Gui, Graham Neubig, Jonathan May, and Luke Zettlemoyer. Mega: Moving average equipped gated attention. In *International Conference on Learning Representations*, 2023.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*, 2016.

Antonio Orvieto, Samuel L. Smith, Albert Gu, Anushan Fernando, Caglar Gulcehre, Razvan Pascanu, and Soham De. Resurrecting recurrent neural networks for long sequences. In *Proceedings of the International Conference on Machine Learning*, pp. 26670–26698, 2023.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Ngoc Quan Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. The LAMBADA dataset: Word prediction requiring a broad discourse context. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2016.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In *Proceedings of the International Conference on Machine Learning*, 2013.

Guilherme Penedo, Hynek Kydlíček, Loubna Ben Allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro von Werra, and Thomas Wolf. The FineWeb datasets: Decanting the web for the finest text data at scale. *arXiv preprint arXiv:2406.17557*, 2024.

Ofir Press and Lior Wolf. Using the output embedding to improve language models. In *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics*, 2017.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.

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

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

Richard Sinkhorn. A relationship between arbitrary positive matrices and doubly stochastic matrices. *The Annals of Mathematical Statistics*, 35(2):876–879, 1964.

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

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. RoFormer: Enhanced transformer with rotary position embedding. *arXiv preprint arXiv:2104.09864*, 2021.

Corentin Tallec and Yann Ollivier. Can recurrent neural networks warp time? In *International Conference on Learning Representations*, 2018.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, et al. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

Andreas Veit, Michael Wilber, and Serge Belongie. Residual networks behave like ensembles of relatively shallow networks. In *Advances in Neural Information Processing Systems*, 2016.

Hongyu Wang, Shuming Ma, Li Dong, Shaohan Huang, Dongdong Zhang, and Furu Wei. DeepNet: Scaling transformers to 1,000 layers. *arXiv preprint arXiv:2203.00555*, 2022.

Zhenda Xie, Yixuan Wei, Huanqi Cao, et al. mHC: Manifold-constrained hyper-connections. *arXiv preprint arXiv:2512.24880*, version 2, 2026.

Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tie-Yan Liu. On layer normalization in the transformer architecture. In *Proceedings of the International Conference on Machine Learning*, 2020.

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*, 2024.

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

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

Hongyi Zhang, Yann N. Dauphin, and Tengyu Ma. Fixup initialization: Residual learning without normalization. In *International Conference on Learning Representations*, 2019.

Yifan Zhang, Yifeng Liu, Mengdi Wang, and Quanquan Gu. Deep delta learning. *arXiv preprint arXiv:2601.00417*, version 4, 2026.

Biao Zhang and Rico Sennrich. Root mean square layer normalization. In *Advances in Neural Information Processing Systems*, 2019.

Defa Zhu, Hongzhi Huang, Zihao Huang, Yutao Zeng, Yunyao Mao, Banggu Wu, Qiyang Min, and Xun Zhou. Hyper-connections. *arXiv preprint arXiv:2409.19606*, 2024.

Nicolas Zucchet and Antonio Orvieto. Recurrent neural networks: Vanishing and exploding gradients are not the end of the story. In *Advances in Neural Information Processing Systems*, 2024.

A PROOFS AND COMPUTATIONAL IDENTITIES

A.1 NOTATION AND TENSOR CONVENTIONS

For a minibatch of N sequences of length L, the residual tensor has shape $N \times L \times n \times C$. Connection heads act independently at each token on its n streams of width C. Suppressing the batch and sequence indices gives $X \in \mathbb{R}^{n \times C}$, a read vector $r^\top X \in \mathbb{R}^{1 \times C}$, and a sublayer output $f \in \mathbb{R}^{1 \times C}$. Attention within F couples tokens causally along the sequence, while stream mixing and the outer-product write act separately at each position.

Table 7: Notation for one residual update. Common and disagreement components are orthogonal to each other under the Frobenius inner product. The connection vectors are columns, and the computational outputs are rows.

Symbol	Shape	Definition
X	$n \times C$	Residual state
r, w	n	Read and effective write vectors
P	$n \times n$	Stream-mixing matrix
J	$n \times n$	Common-mode projector $\mathbf{1}\mathbf{1}^\top/n$
Π	$n \times n$	Disagreement projector $I - J$
$\bar{x}$	$1 \times C$	Stream mean $\mathbf{1}^\top X/n$
D	$n \times C$	Disagreement state ΠX
q	n	Zero-mean equilibrium coordinate
g	scalar	Common-mode write gain
f	$1 \times C$	Computational sublayer output
T	$n \times C$	Conditional target qf
e	$n \times C$	Target error $D - T$
U	$n \times (n-1)$	Orthonormal basis for $\mathcal{S}$
A	$(n-1) \times (n-1)$	Disagreement transport $U^\top PU$

The decomposition $X = \mathbf{1}\bar{x} + D$ satisfies

$$\|X\|_F^2 = n\|\bar{x}\|_2^2 + \|D\|_F^2, \qquad \|D\|_F^2 = \frac{1}{2n}\sum_{i,j=1}^{n} \|X_i - X_j\|_2^2. \tag{25}$$

The first identity follows from the fact that $\mathbf{1}^\top D = 0$. The second follows by expanding the pairwise squared distances. The disagreement energy measures the variation in the streams independently of the common component.

A.2 THE INVARIANT SUBSPACES OF RESIDUAL MIXING

Let $U \in \mathbb{R}^{n \times (n-1)}$ have orthonormal columns spanning $\mathcal{S}$, and let $B = [\mathbf{1}/\sqrt{n}, U]$. By double stochasticity,

$$B^\top PB = \begin{pmatrix} 1 & 0 \\ 0 & A \end{pmatrix}, \qquad A = U^\top PU. \tag{26}$$

In particular, $\|P - J\|_2 = \|A\|_2 \leq 1$. To prove strict contraction for positive P, take nonzero $x \in \mathcal{S}$. Such an x is nonconstant. By strict convexity,

$$\left(\sum_j P_{ij} x_j\right)^2 < \sum_j P_{ij} x_j^2.$$

Summing over i and using the column sums gives $\|Px\|_2^2 < \|x\|_2^2$. The unit sphere in $\mathcal{S}$ is compact, so $\|A\|_2 < 1$. For fixed P and fixed T,

$$D_L - T = P^L(D_0 - T), \qquad \|D_L - T\|_F \leq \|A\|_2^L \|D_0 - T\|_F. \tag{27}$$

These conclusions hold for nonsymmetric mixing matrices without needing a common eigenbasis for all layers.

A.3 Energy decrease around a fixed target

Let $e = D - T$ and keep P and T fixed during the update. Proposition 1 yields $e^+ = Pe$. The decrease of the squared target error can be written as

$$\|e\|_F^2 - \|e^+\|_F^2 = \operatorname{tr}\big(e^\top (I - P^\top P) e\big)\,. \tag{28}$$

Since each column of e lies in $\mathcal{S}$, only the restriction of $P^\top P$ to this subspace contributes. With $\rho = \|P - J\|_2$, Equation (28) gives

$$\|e\|_F^2 - \|e^+\|_F^2 \geq (1 - \rho^2)\|e\|_F^2. \tag{29}$$

This is a quadratic Lyapunov function for the conditional disagreement update. Its reference point is the target generated by the current sublayer. For varying targets, their increments appear in Equation (12).

A.4 Stream permutations and the choice of coordinates

Let R be a permutation matrix on streams. We relabel the state and coefficients as

$$X' = RX, \qquad P' = RPR^\top, \qquad q' = Rq, \qquad r' = Rr.$$

The read is unchanged, $(r')^\top X' = r^\top X$. Because $R\mathbf{1} = \mathbf{1}$,

$$P'X' + \big[g\mathbf{1} + (I - P')q'\big] f = R\left(PX + \big[g\mathbf{1} + (I - P)q\big] f\right). \tag{30}$$

The target-centered update respects stream relabeling. The same identity holds for input-dependent connection heads after applying the corresponding input and output permutations to their parameters.

Expressing disagreement in the orthonormal basis U of $\mathcal{S}$ reduces the update to $n - 1$ stream coordinates. Let $H = U^\top D$, $a = U^\top q$, and $A = U^\top P U$. Then

$$H^+ = AH + (I - A)af = af + A(H - af). \tag{31}$$

Replacing U with UO, for any orthogonal $(n-1) \times (n-1)$ matrix O, changes only the coordinates of this expression. eHC retains the original stream basis and obtains q by centering the write-head output.

A.5 Composition at a shared target

If all conditional targets are T, then unrolling Equation (7) yields Equation (17). The perturbation of P_k with all other coefficients and T fixed is

$$dD_L = P_{L-1} \cdots P_{k+1}\, dP_k\, P_{k-1} \cdots P_0 (D_0 - T). \tag{32}$$

The products on both sides of the perturbation act on the zero-mean subspace. For $\rho_i = \|P_i - J\|_2$,

$$\|dD_L\|_F \leq \left(\prod_{i \neq k} \rho_i\right) \|dP_k\|_2 \|D_0 - T\|_F. \tag{33}$$

The derivative is proportional to the initial error relative to T. When $D_0 = T$, every admissible perturbation of the mixing sequence preserves the target. With varying targets, the additional terms are their successive differences in Equation (12).

A.6 Moving targets

From $e_l = D_l - T_l$ and $\Delta T_l = T_{l+1} - T_l$, Equation (7) yields $e_{l+1} = P_l e_l - \Delta T_l$. Unrolling,

$$e_L = P_{L-1} \cdots P_0 e_0 - \sum_{k=0}^{L-1} P_{L-1} \cdots P_{k+1} \Delta T_k. \tag{34}$$

All terms are in the zero-mean subspace. With $\rho_l = \|P_l - J\|_2$,

$$\|e_L\|_F \leq \left(\prod_{i=0}^{L-1} \rho_i\right) \|e_0\|_F + \sum_{k=0}^{L-1} \left(\prod_{i=k+1}^{L-1} \rho_i\right) \|\Delta T_k\|_F. \tag{35}$$

Uniform bounds $\rho_l \leq \rho < 1$ and $\|\Delta T_l\|_F \leq \delta$ give Equation (13) by a geometric sum.

A.7 Coordinate equivalence and matched initialization

In the basis of Equation (26), we have

$$B^\top (I - P + J)B = \begin{pmatrix} 1 & 0 \\ 0 & I - A \end{pmatrix}.$$

For $b = w - g\mathbf{1} = U\beta$, the unique zero-mean solution is $q = U(I-A)^{-1}\beta$. This proves Proposition 2. For a Signed-Direct write, we have $b = \gamma q_{\mathrm{dir}}$ with $\gamma = 0.1$. The relation is

$$q_{\mathrm{eHC}} = (I - P + J)^{-1}\gamma q_{\mathrm{dir}}. \tag{36}$$

At $P_0 = 0.9I + 0.1J$, we have $(I - P_0)q = 0.1q$ for every $q \in \mathcal{S}$. The initial coordinate maps are the same.

The write head defines $g = 2\sigma(m)$, $m = \mathbf{1}^\top z/n$, and $q = \Pi z$. At fixed P, its Jacobian is

$$\frac{\partial w_{\mathrm{eHC}}}{\partial z} = 2\sigma(m)(1 - \sigma(m))J + (I - P)\Pi. \tag{37}$$

At $m = 0$ and $P = P_0$, this is $\frac{1}{2}J + 0.1\Pi$. Signed-Direct has the same expression but with $(I - P)\Pi$ replaced by 0.1Π, so Equation (20) holds.

A.8 Differentials of the equilibrium coordinates

The relation $b = (I - P)q$ implies

$$db = (I - P)\, dq - dP\, q. \tag{38}$$

For feasible variations on $\mathcal{S}$, we can solve for dq to obtain

$$dq = (I - P + J)^{-1}(db + dP\, q). \tag{39}$$

Holding an independent write b fixed allows changes in mixing to shift the represented equilibrium. Holding q fixed instead induces the compensating write variation $db = -dP\, q$, which eHC includes directly in its parameterization.

The total differential of Equation (7) for simultaneous changes in the state, target and mixing is

$$dD^+ = P\, dD + (I - P)\, dT + dP(D - T), \qquad dT = (dq)f + q\, df. \tag{40}$$

The three terms describe the effects of changes in the incoming state, the conditional target, and mixing, respectively. The final term is the innovation-weighted derivative of Proposition 3. Together, they determine how retention interventions propagate to predictions in Appendix C.

A.9 Scalar response and continuous-depth interpretation

For a fixed real mode with retention $\lambda \in [0, 1)$ and target coordinate a, we have

$$s_{l+1} = \lambda s_l + (1 - \lambda)a f_l.$$

For an arbitrary input sequence, repeated substitution gives

$$s_L = \lambda^L s_0 + (1 - \lambda)a \sum_{k=0}^{L-1} \lambda^{L-1-k} f_k. \tag{41}$$

For a constant input $f_k = f$, we have

$$s_L = \lambda^L s_0 + (1 - \lambda^L)af. \tag{42}$$

The equilibrium gain is a, and the initial error decays by a factor of λ^L over depth L.

For a harmonic input, the transfer function of Equation (41) is

$$H(z) = \frac{(1 - \lambda)a}{z - \lambda}, \qquad H(1) = a. \tag{43}$$

The zero-frequency gain is a, and the pole is λ. For a scalar target t_l that varies with depth, the error obeys $e_{l+1} = \lambda e_l + t_l - t_{l+1}$, the one-dimensional counterpart of the target-tracking bound.

The affine update also admits a continuous-depth interpretation. Let L be a balanced graph Laplacian with nonpositive off-diagonal entries and $L\mathbf{1} = 0$, $\mathbf{1}^\top L = 0$. Keeping the target fixed over an interval of length h, we have the differential equation

$$\frac{dD(t)}{dt} = -L\big(D(t) - T\big) \tag{44}$$

which has solution

$$D(t+h) = T + \exp(-hL)\big(D(t) - T\big). \tag{45}$$

Because $\exp(-hL)$ is nonnegative and doubly stochastic, Equation (45) defines a continuous-depth family of equilibrium updates. This interpretation uses a graph Laplacian, whereas the learned mixing matrices in eHC are parameterized directly by Sinkhorn normalization.

A.10 Forward and backward evaluation in connection space

Algorithm 1 evaluates the equilibrium correction using the connection coefficients, avoiding the full target matrix $T = qf$. The layer index is omitted.

Algorithm 1 Equilibrium-coordinate residual update.

Require: State X, sublayer F, and connection-head parameters
1: $h \leftarrow \mathrm{RMSNorm}(\mathrm{vec}(X))$
2: $a, S, z \leftarrow \mathrm{ConnectionHeads}(h)$
3: $r \leftarrow \sigma(a), \quad P \leftarrow \mathrm{Sinkhorn}_{20}(S)$
4: $m \leftarrow \mathrm{mean}(z), \quad g \leftarrow 2\sigma(m), \quad q \leftarrow z - m\mathbf{1}$
5: $f \leftarrow F(r^\top X)$
6: $w \leftarrow g\mathbf{1} + q - Pq$
7: **return** $PX + wf$ ▷ Fused residual mixing and write

For the backward pass, define $G = \partial\mathcal{L}/\partial X^+$ and $u = Gf^\top \in \mathbb{R}^n$. The eHC coefficient adjoints are

$$\nabla_P\mathcal{L} = GX^\top - uq^\top, \qquad \nabla_q\mathcal{L} = (I - P)^\top u, \tag{46}$$

$$\nabla_g\mathcal{L} = \mathbf{1}^\top u, \qquad \nabla_f\mathcal{L} = w^\top G. \tag{47}$$

The direct state adjoint is $P^\top G$, with additional chain-rule contributions through the connection heads and the read. The write-logit adjoint is

$$\nabla_z\mathcal{L} = \Pi\nabla_q\mathcal{L} + \frac{2\sigma(m)(1-\sigma(m))}{n}(\nabla_g\mathcal{L})\mathbf{1}. \tag{48}$$

The additional mixing-gradient term $-uq^\top$ is an $n \times n$ outer product. Its factor u and the standard mixing adjoint $GX^\top$ are already required by the residual update. Omitting this term gives eHC-Detached, whose forward map remains identical to eHC at the same parameters.

A.11 Read and normalization adjoints

Let $v = r^\top X$ be the input to the computational sublayer and $K = \partial\mathcal{L}/\partial v \in \mathbb{R}^{1\times C}$ its adjoint. Differentiating the read gives

$$\nabla_r\mathcal{L} = XK^\top, \qquad \nabla_X^{\mathrm{read}}\mathcal{L} = rK. \tag{49}$$

These terms are added to the direct residual-state adjoint and gradients through the connection heads.

For routing normalization, let $x = \mathrm{vec}(X)$, $d = nC$, and $s = (\|x\|_2^2/d + \epsilon)^{1/2}$. If $a = \partial\mathcal{L}/\partial h$ for $h = x/s$, then the routing input adjoint is

$$\nabla_x^{\mathrm{route}}\mathcal{L} = \frac{a}{s} - \frac{x(x^\top a)}{ds^3}. \tag{50}$$

The adjoint a combines contributions from the three connection heads. Equation (50) then gives their contribution to the residual-state gradient. These read and normalization derivatives are shared by eHC, eHC-Detached, and Signed-Direct.

A.12 LOG-SPACE MATRIX SCALING AND ITS ADJOINT

For a log-matrix S, one column normalization and one row normalization are

$$A_{ij} = S_{ij} - \log \sum_k \exp(S_{kj}), \tag{51}$$

$$B_{ij} = A_{ij} - \log \sum_k \exp(A_{ik}). \tag{52}$$

A numerically stable log-sum-exp evaluation avoids forming the unnormalized exponential matrix. Twenty alternating column and row normalizations define the final mixing coefficients $P_{ij} = \exp(B_{ij})$.

For the row-normalization adjoint $V = \partial \mathcal{L}/\partial B$, we have $p_{ij} = \exp(A_{ij})/\sum_k \exp(A_{ik})$. The input adjoint is

$$\frac{\partial \mathcal{L}}{\partial A_{ij}} = V_{ij} - p_{ij} \sum_k V_{ik}. \tag{53}$$

For the column-normalization adjoint, where $p_{ij} = \exp(S_{ij})/\sum_k \exp(S_{kj})$ and the upstream adjoint is U, the expression is

$$\frac{\partial \mathcal{L}}{\partial S_{ij}} = U_{ij} - p_{ij} \sum_k U_{kj}. \tag{54}$$

The exponential map contributes an elementwise factor of P to the coefficient adjoint. Composing this derivative with the row and column normalization adjoints gives the mixing-logit gradient. In eHC, the coefficient adjoint is $GX^\top - uq^\top$, with all normalization derivatives shared by the compared parameterizations.

A.13 FINITE SINKHORN ITERATIONS

Sinkhorn normalization (Sinkhorn, 1964; Cuturi, 2013) consists of alternating column and row normalization in log space for 20 iterations, with row normalization at the end. Denote the resulting matrix by $\widetilde{P}$ and let

$$c^\top = \frac{1}{n}\mathbf{1}^\top(\widetilde{P} - I).$$

For zero-mean q, the shared component residual is given by

$$\bar{x}^+ - \bar{x} - gf = c^\top(X - qf). \tag{55}$$

If $\widetilde{P}\mathbf{1} = \mathbf{1}$, then $c^\top\mathbf{1} = 0$ and

$$\bar{x}^+ - \bar{x} - gf = c^\top(D - T), \qquad D^+ = T + \Pi\widetilde{P}\Pi(D - T). \tag{56}$$

These identities characterize the finite-iteration update through its row-normalized coefficient matrix. Sinkhorn normalization and effective-write calculations use FP32 throughout.

For a matrix with row residual $r = \widetilde{P}\mathbf{1} - \mathbf{1}$, the projected state identity is

$$D^+ = T + \Pi\widetilde{P}\Pi(D - T) + \Pi r\,\bar{x}. \tag{57}$$

When the final row normalization condition holds, the last term is zero. The mean residual obeys

$$\|\bar{x}^+ - \bar{x} - gf\|_2 \le \|c\|_2 \|D - T\|_F. \tag{58}$$

These bounds relate deviations from unit row and column sums to errors in the common and disagreement components of the residual state.

B EXPERIMENTAL DETAILS

B.1 BACKBONE AND OPTIMIZATION

Attention and feed-forward projections are bias-free, with SwiGLU feed-forward blocks (Dauphin et al., 2017; Shazeer, 2020) and attention head dimension 64. Rotary embeddings use base 10,000 without context-extension scaling, and RMSNorm uses $\epsilon = 10^{-6}$. No dropout is applied. The GPT-NeoX vocabulary is padded to 50,304 entries, with unused entries excluded when scoring predictions.

Backbone matrix weights are initialized from $\mathcal{N}(0, 0.02^2)$. Attention output and feed-forward down-projection weights are additionally scaled by $1/\sqrt{2B}$, where B is the number of Transformer blocks, and normalization scales start at one. The four residual streams share the same initial token embeddings. Final predictions use the tied output head applied to their normalized mean.

AdamW uses moment coefficients $(\beta_1, \beta_2) = (0.9, 0.95)$ and $\epsilon = 10^{-8}$, with global gradient-norm clipping at 1.0. Weight decay is 0.1 for backbone matrices and zero for embeddings, normalization parameters, and connection heads. All parameters follow the same schedule of 256 warmup updates and cosine decay to one tenth of the peak learning rate. The peak is 6×10^{-4} for Base-125 and Deep-125 and 3×10^{-4} for Base-355.

All models are trained on a single NVIDIA H100 80GB GPU with seeds 11, 23, and 37. Base-125 and Deep-125 accumulate gradients over four microbatches of 16 sequences, while Base-355 accumulates eight microbatches of 8 sequences. At context length 2,048, both settings yield 2^{17} prediction tokens per optimizer update. Training does not use activation checkpointing.

B.2 BATCH CONSTRUCTION AND OPTIMIZER SCHEDULE

The prediction-token count per optimizer update for Base-125 and Deep-125 is

$$1 \times 16 \times 4 \times 2048 = 131{,}072.$$

For Base-355, it is

$$1 \times 8 \times 8 \times 2048 = 131{,}072.$$

In both expressions, the factors represent the number of GPUs, sequences per microbatch, accumulation steps, and prediction positions per sequence. Base-125 and Deep-125 receive 8,192 optimizer updates, and Base-355 receives 16,384. Thus, methods within each model size are matched in both input tokens and optimizer updates.

Let $u \in \{1, \dots, U\}$ be the optimizer-update index, $u_w = 256$, and $\eta_{\max}$ the maximum learning rate. The schedule is

$$\eta(u) = \begin{cases} \eta_{\max} u / u_w, & u \le u_w, \\ \eta_{\max} \left[0.1 + 0.45 \left(1 + \cos\left(\pi \frac{u - u_w}{U - u_w} \right) \right) \right], & u > u_w. \end{cases} \tag{59}$$

Backbone and connection parameters share this schedule. Global-norm clipping is applied to the accumulated gradient before each optimizer update, with no weight decay on connection-head biases, multipliers, or projection matrices.

B.3 CONNECTION INITIALIZATION

The read and mixing biases are $c_i^r = \mathrm{logit}(1/n)$ and $c_{ij}^P = \log(P_{0,ij})$. The write-head bias is $c^z = q_0$ for eHC and Signed-Direct and $c_i^z = \mathrm{logit}(w_{0,i}/2)$ for mHC. All state-dependent connection projections start at zero, with their three learned scalar multipliers per residual sublayer initialized to 0.01.

Successive sublayers use cyclic permutations of $0.1(-3, -1, 1, 3)^\top$ for q_0, giving the initial effective write $\mathbf{1} + 0.1 q_0$. The nonuniform write introduces disagreement among streams, but the uniform read depends only on their mean. Paired methods therefore share the same initial language-model function despite these stream differences.

Table 8: Optimization settings shared across paired methods. Width-dependent learning rates are based on the model configurations in Table 1. Routing heads include read, mixing, and write-coordinate heads.

Setting	Value
Optimizer	AdamW
Moment coefficients	$(0.9, 0.95)$
Optimizer numerical constant	10^{-8}
Global gradient-norm threshold	1.0
Backbone matrix weight decay	0.1
Embedding, normalization, and routing weight decay	0
Backbone and activation precision	BF16
Routing and optimizer-state precision	FP32
Training context	2,048 tokens
Prediction tokens per update	131,072
Warmup	256 updates
Final-to-peak learning-rate ratio	0.1
Attention implementation	FlashAttention-2
Training seeds	11, 23, 37

For vocabulary size V, block count B, width C, and feed-forward width M, the single-stream parameter count is

$$VC + B(4C^2 + 3CM + 2C) + C.$$

The connection heads require

$$2B\big[(nC + 1)(n^2 + 2n) + 3\big].$$

The three additional scalars per sublayer are the learned multipliers of the state-dependent projections. Connection-head RMS normalization has no learned affine parameters.

The initial values for the first four-stream sublayer are

$$P_0 = \begin{pmatrix} 0.925 & 0.025 & 0.025 & 0.025 \\ 0.025 & 0.925 & 0.025 & 0.025 \\ 0.025 & 0.025 & 0.925 & 0.025 \\ 0.025 & 0.025 & 0.025 & 0.925 \end{pmatrix}, \qquad q_0 = \begin{pmatrix} -0.3 \\ -0.1 \\ 0.1 \\ 0.3 \end{pmatrix}, \qquad w_0 = \begin{pmatrix} 0.97 \\ 0.99 \\ 1.01 \\ 1.03 \end{pmatrix}. \tag{60}$$

Cyclic permutations of q_0 induce the same permutations of w_0 across sublayers. With $r_0 = \mathbf{1}/4$ and $g_0 = 1$, the initial read and mean update satisfy $r_0^\top X = \bar{x}$ and $\bar{x}^+ = \bar{x} + f$. Induction over sublayers establishes the shared initial prediction function.

The matched write-head Jacobian has eigenvalue $1/2$ along $\mathbf{1}$ and 0.1 on $\mathcal{S}$, giving eHC and Signed-Direct the same first-order scaling of common and disagreement writes. Their write-head parameters also use the same learning rate.

B.4 Data construction

Document-level partitions are determined by SHA-256 hashes after Unicode NFKC normalization and whitespace collapse. Interpreting the first eight digest bytes as a big-endian integer and taking its remainder modulo 1000 assigns residues 0 and 1 to development and test, respectively, and all other residues to training. Each distinct normalized document is retained once.

Development and test documents are ordered separately by content hash, with 2^{22} prediction tokens retained in each partition. Decontamination excludes any training document sharing a contiguous 13-token string with either held-out partition, the selected cross-corpus evaluation documents, or the downstream questions and candidate answers. Matching uses the token strings verbatim.

Within each seed, training documents are ordered by the hash of the seed concatenated with the normalized-content digest. The original text is tokenized with an end-of-text marker between documents, then packed into sequences of 2,048 prediction positions with one-token lookahead. Attention is causal within each sequence, with no state carried between sequences. Base-355 uses an extension of the corresponding Base-125 token stream. A single training process consumes this stream in the same order for every paired method.

B.5 HELD-OUT LOSS AND DOWNSTREAM EVALUATION

FineWeb-Edu loss uses the same sequence packing as training. Development loss is evaluated every 512 updates on a fixed subset of 2^{18} tokens, while final evaluation covers the complete held-out partitions.

WikiText-103 evaluation uses the raw test split and preserves article boundaries. English C4 validation documents are selected in increasing content-hash order until they provide 2^{20} prediction tokens. Both corpora use document-bounded windows of length 2048 and stride 1024, scoring only previously unseen positions after the first window. Perplexity is the exponential of the aggregate token-weighted negative log-likelihood.

All downstream evaluations are zero-shot. HellaSwag uses the public validation split and ranks candidate endings by token-length-normalized conditional log probability given the activity context. PIQA validation and ARC-Easy test use the same scoring rule for candidate solutions and answer texts. Their prompts place the goal or question after "Question: " and begin a new line with "Answer:" before the candidate continuation. For LAMBADA, the final word of each test passage is withheld and predicted by greedy decoding, with at most 16 generated tokens. Accuracy requires an exact final-word match after trimming trailing whitespace. The four tasks receive equal weight in mean accuracy.

B.6 LOSS, PERPLEXITY, AND CANDIDATE SCORES

For scored token positions $\mathcal{I}$, the language-model loss is

$$L = -\frac{1}{|\mathcal{I}|} \sum_{t \in \mathcal{I}} \log p_\theta(x_t \mid x_{<t}), \qquad \mathrm{PPL} = \exp(L). \tag{61}$$

Cross-corpus windows preserve document boundaries. The first window contributes all available targets, and subsequent windows advance by $1{,}024$ tokens and contribute only the newly exposed suffix. Each target is therefore counted once in Equation (61).

For a multiple choice prompt x and candidate tokens $y_1, \ldots, y_K$, the candidate score is

$$s(y \mid x) = \frac{1}{K} \sum_{k=1}^{K} \log p_\theta(y_k \mid x, y_{<k}). \tag{62}$$

The highest-scoring candidate determines the prediction. Prompt tokens condition the prediction but do not contribute to the score. HellaSwag, PIQA, and ARC-Easy use this rule with their respective contexts, while LAMBADA uses greedy generation and final-word matching.

Table 9: Evaluation corpora and tasks for pretrained checkpoints. Perplexity is the exponential of token-weighted loss. Multiple-choice accuracy uses the candidate with the highest mean token log probability.

Evaluation	Split or selection	Metric
FineWeb-Edu	Content-hash test partition	Token NLL
WikiText-103	Raw test split	Perplexity
C4	Hash-selected English validation documents	Perplexity
HellaSwag	Public validation split	Continuation accuracy
PIQA	Public validation split	Solution accuracy
ARC-Easy	Public test split	Answer accuracy
LAMBADA	Public test split	Final-word accuracy

B.7 INTERVENTIONS AND UNCERTAINTY

The Base-125 ablation compares Signed-Direct's fixed disagreement map $I - P_0$ with the adaptive map in eHC and eHC-Detached (Table 4). eHC-Detached uses Equation (21) with the same initialization, optimizer, training examples, and seeds as eHC. The Base-125 models also form the shallower comparison in Table 5.

Retention interventions use 64 held-out sequences selected by content hash and affect the last quarter of residual sublayers, counting attention and feed-forward updates separately. At each selected sublayer, P, q, g, r, and f are conditioned on its current input before P is replaced by P_η. The perturbed state then propagates through the remaining sublayers. Predictive KL uses all valid vocabulary entries at the last 32 positions of each sequence.

Let b be a baseline. Paired loss differences are $d_s = L_{b,s} - L_{\mathrm{eHC},s}$. Confidence intervals for loss contrasts are Student-t intervals over three seeds. Downstream intervals are based on 10,000 hierarchical bootstrap replicates, resampling seeds and examples, but maintaining the method pairing. Retention intervals are based on resampling seeds and complete sequences. The depth interaction is computed within each paired seed before forming its interval.

B.8 Training efficiency

Step time is averaged over 200 updates after 20 warmup updates, with measurements repeated across three independent process launches. Peak allocated memory is measured with the peak counter reset after optimizer-state initialization. Batch dimensions, attention, data loading, precision, and optimizer computation are matched across methods.

For each model size, the common elapsed-time comparison uses the shortest completed training time among the multi-stream methods. Each method is evaluated at its latest checkpoint at or before that time. Measured training time excludes validation and checkpoint saving.

Table 10: Training comparisons. Each configuration uses three paired seeds. Retention interventions and timing measurements reuse the resulting checkpoints.

Configuration	Methods	Tokens per run
Base-125	Standard, mHC, Signed-Direct, eHC	2^{30}
Base-355	mHC, Signed-Direct, eHC	2^{31}
Base-125	eHC-Detached	2^{30}
Deep-125	Signed-Direct, eHC	2^{30}

C Retention interventions and predictive sensitivity

C.1 Local changes in retention

The intervention $P_\eta = (1-\eta)P + \eta I$ preserves the doubly stochastic conditions for $\eta \in [0,1]$. On a fixed real mode with eigenvalue λ, it changes retention to $\lambda_\eta = (1-\eta)\lambda + \eta$. In eHC, retaining the current q leads to

$$(I - P_\eta)q = (1-\eta)(I-P)q. \tag{63}$$

The new disagreement write compensates for the increase in retention.

At a fixed local input, the two interventions differ by

$$X^+_{\mathrm{eq}}(\eta) - X^+(0) = \eta(I-P)(D-T), \tag{64}$$

$$X^+_{\mathrm{fixed}}(\eta) - X^+(0) = \eta(I-P)D. \tag{65}$$

Both interventions leave the common update unchanged since $\mathbf{1}^\top(I-P) = 0$. Their difference is

$$X^+_{\mathrm{eq}}(\eta) - X^+_{\mathrm{fixed}}(\eta) = -\eta(I-P)T. \tag{66}$$

This is the target-dependent part of the write adjustment.

C.2 Propagation to the output distribution

Let $h_l(X_l, \eta)$ be an intervened residual sublayer, and $\mathcal{J}_l$ be its state Jacobian at $\eta = 0$. We denote $B_l = \partial h_l/\partial\eta|_0$. The first-order state response $V_l = \partial X_l/\partial\eta|_0$ is given by

$$V_{l+1} = \mathcal{J}_l V_l + B_l. \tag{67}$$

Here, $B_l = 0$ outside the selected sublayers. At a selected sublayer, Equation (65) yields either $(I-P_l)(D_l - T_l)$ or $(I-P_l)D_l$. Unrolling, we obtain

$$V_L = \sum_{k\in\mathcal{A}} \mathcal{J}_{L-1}\cdots\mathcal{J}_{k+1}B_k, \tag{68}$$

where $\mathcal{A}$ is the set of intervened sublayers. The Jacobians in the unrolling include the read, target-generation, and computational paths of subsequent sublayers.

Let $z(\eta)$ denote the final vocabulary logits of a scored token, and $p(\eta) = \mathrm{softmax}(z(\eta))$. Let $v = dz/d\eta|_0$. A second-order approximation yields

$$D_{\mathrm{KL}}(p(0)\,\|\,p(\eta)) = \frac{\eta^2}{2} v^\top \left[\mathrm{diag}(p(0)) - p(0)p(0)^\top\right] v + O(\eta^3). \tag{69}$$

The linear term vanishes since the two distributions coincide at zero intervention. The quadratic form is the output logit response in the softmax Fisher metric.

Equations (68) and (69) clarify the use of predictive KL divided by η^2. The local energy ratio in Equation (24) measures the state changes injected before propagation through later sublayers. The predictive ratio additionally includes those sublayers and the vocabulary head.

C.3 Scored positions and aggregation

Each of the 64 held-out sequences contributes its last 32 prediction positions, for a total of $2{,}048$ scored positions per seed. Let

$$\mathcal{K}_a(\eta) = \frac{1}{|\mathcal{E}|\eta^2}\sum_{t\in\mathcal{E}} D_{\mathrm{KL}}(p_0(\cdot \mid x_{<t})\,\|\,p_{a,\eta}(\cdot \mid x_{<t}))\,, \tag{70}$$

where $a \in \{\mathrm{eq}, \mathrm{fixed}\}$ denotes an intervention. The reported response compares $\mathcal{K}_{\mathrm{eq}}(\eta)$ to $\mathcal{K}_{\mathrm{fixed}}(\eta)$ at the same intervention strength. The position count and η^2 factor cancel in this comparison.

Both intervention conditions use the same sequences, scored positions, and valid vocabulary entries. Coefficients are conditioned on the current input at each selected sublayer, and later sublayers operate on the perturbed state. Confidence intervals preserve these pairings when resampling seeds and complete sequences.

D ADDITIONAL LEARNING AND INTERVENTION CURVES

Figure 4 places the paired loss differences in the context of the absolute held-out trajectories at the seven checkpoints used in Figure 2. Each point averages three paired seeds. Within a seed, all methods receive the same ordered token stream, and Signed-Direct and eHC begin with the same prediction function.

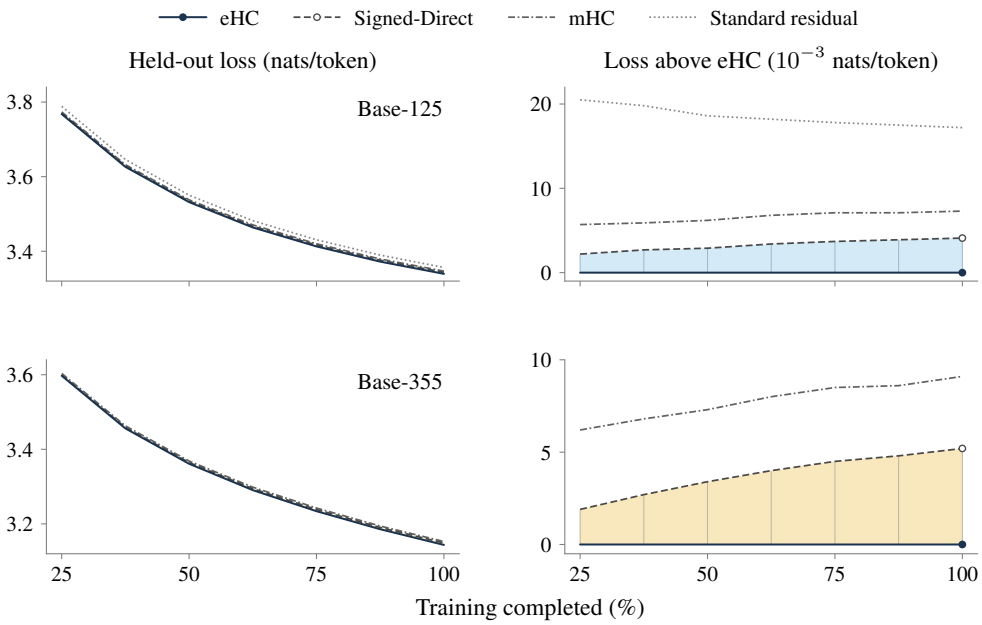


Figure 4: Held-out learning trajectories for Base-125 (top) and Base-355 (bottom). Absolute loss appears on the left and loss above eHC on the right, evaluated at the same seven checkpoints and averaged over three paired seeds. Shading highlights the gap between Signed-Direct and eHC.

D.1 CHECKPOINT LOSSES

Tables 11 and 12 give the losses underlying Figure 4. Progress is measured as a fraction of each model's training-token budget. Comparisons within a row match both training progress and input order for each seed.

Table 11: Base-125 held-out checkpoint losses in nats/token. Values are the average of the three paired seeds. The final row is the completed training token allocation.

Training completed (%)	Standard residual	mHC	Signed-Direct	eHC
25.0	3.7884	3.7736	3.7701	3.7679
37.5	3.6461	3.6322	3.6290	3.6263
50.0	3.5503	3.5379	3.5346	3.5317
62.5	3.4820	3.4706	3.4672	3.4638
75.0	3.4307	3.4200	3.4166	3.4129
87.5	3.3896	3.3792	3.3760	3.3721
100.0	3.3566	3.3467	3.3435	3.3394

Table 12: Base-355 held-out checkpoint losses in nats/token. The three methods all have the same computational backbone and all receive the same tokens in each seed.

Training completed (%)	mHC	Signed-Direct	eHC
25.0	3.6035	3.5992	3.5973
37.5	3.4628	3.4587	3.4560
50.0	3.3685	3.3646	3.3612
62.5	3.2985	3.2945	3.2905
75.0	3.2425	3.2385	3.2340
87.5	3.1943	3.1905	3.1857
100.0	3.1527	3.1488	3.1436

The paired trajectories separate progressively over training. From one quarter of the token budget to completion, eHC's advantage over Signed-Direct grows from 0.0022 to 0.0041 nats/token at Base-125 and from 0.0019 to 0.0052 at Base-355. The intermediate checkpoints in both tables show the same increasing trend, with every comparison based on matched training progress.

D.2 Predictive response to retention changes

Figure 5 shows the predictive KL values underlying the sensitivity ratios in Figure 3. A common scale sets the fixed-write value at $\eta = 0.025$ to one. This normalization preserves the ratio between intervention conditions at every strength.

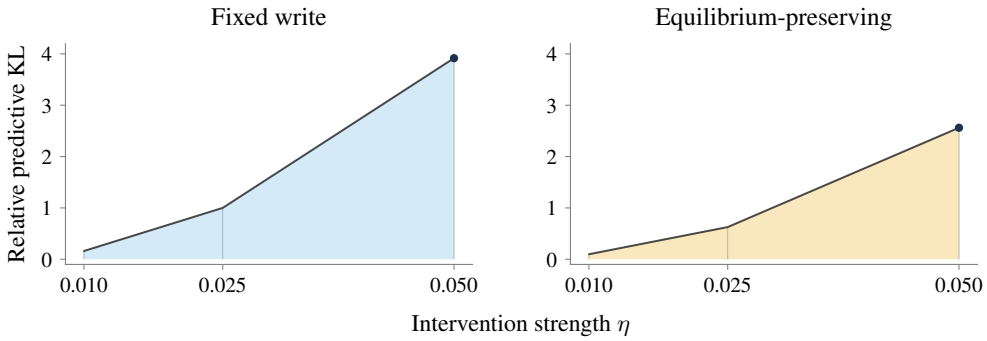


Figure 5: Predictive KL under the two retention interventions, shown on identical axes. Shaded heights give the relative KL values. The fixed-write value at $\eta = 0.025$ sets a common scale for all intervention strengths.

Table 13: Predictive sensitivity under retention interventions. Confidence intervals describe the ratio of equilibrium-preserving to fixed-write sensitivity. The corresponding local energy ratio in Equation (24) is 0.607.

η	Fixed-write KL	Equilibrium KL	Ratio	95% interval
0.010	0.161	0.098	0.611	$[0.568, 0.654]$
0.025	1.000	0.627	0.627	$[0.582, 0.674]$
0.050	3.916	2.561	0.654	$[0.604, 0.707]$

D.3 PAIRED LOSS DIFFERENCES

Table 14: Paired loss effects and their 95% intervals, in nats/token. The depth interaction is the Deep-125 effect minus the Base-125 effect.

Comparison	Difference	95% interval
Base-125 mHC minus eHC	0.0073	$[0.0051, 0.0095]$
Base-125 Signed-Direct minus eHC	0.0041	$[0.0024, 0.0058]$
Base-125 eHC-Detached minus eHC	0.0016	$[0.0004, 0.0028]$
Depth interaction	0.0025	$[0.0009, 0.0041]$

At Base-355, eHC reduces loss by 0.0091 nats/token relative to mHC and by 0.0052 relative to Signed-Direct. At equal elapsed training time, the reductions remain 0.0088 and 0.0046, respectively. Both comparisons use the same held-out examples and token-weighted loss.

D.4 DOWNSTREAM COMPARISONS BY TASK

Table 15 expresses the zero-shot results as paired accuracy differences in percentage points, complementing the absolute accuracies and cross-corpus perplexities in Table 3.

Table 15: Zero-shot accuracy differences at Base-355. Each entry subtracts the indicated baseline from eHC. The reported equal-weight mean improvement over Signed-Direct is 0.5 percentage points, with a 95% interval of $[0.2, 0.8]$.

Task	eHC minus mHC	eHC minus Signed-Direct
HellaSwag	0.7	0.4
PIQA	0.6	0.3
ARC-Easy	0.9	0.5
LAMBADA	1.2	0.7

Each method is evaluated on all four tasks using the same pretrained checkpoint and fixed candidate-scoring rules. Benchmark items do not enter model selection. The accuracy gains are accompanied by lower perplexity on WikiText-103 and C4, where eHC reaches 33.78 and 29.25, respectively.