DAHC: DIFFERENTIATION-AUGMENTED HYPER-CONNECTIONS VIA STRUCTURED TRANSFORMATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Hyper-Connections (HC) expand the Transformer residual state into multiple parallel streams, yet the resulting streams need not become correspondingly differentiated representations. We show that, in mHC, stream differentiation emerges gradually during training and is reorganized non-monotonically across network depth. Motivated by these observations, we propose **Differentiation-Augmented Hyper-Connections (DAHC)**, which introduces structured sources of stream differentiation both at stream creation and throughout depth. DAHC combines Differentiated Stream Expansion (DSE), which replaces replicated initialization with stream-indexed causal transformations, and Layerwise Differentiation Augmentation (LDA), which constructs depth-adapted stream candidates and integrates them through state-conditioned write-back while preserving the native mHC pathway. Empirically, DAHC reaches the final mHC training loss and validation perplexity after 7.3k and 7.5k updates, requiring 27% and 25% fewer updates, respectively. At the matched 10k-update budget, DAHC further reduces validation perplexity from 29.34 to 28.49, corresponding to a 2.90% improvement over mHC.

1 INTRODUCTION

Residual connections provide a direct pathway for information and gradients through deep networks (He et al., 2016; Vaswani et al., 2017). In Transformer, the residual state is also the persistent representation that successive blocks read and update, making its organization an architectural design axis. Recent scaling approaches explore complementary axes, from sparse parameter activation in Switch Transformer (Fedus et al., 2022) and GLaM (Du et al., 2022) to widened token representations in AltUp (Baykal et al., 2023). Hyper-Connections (HC) (Zhu et al., 2025) take a related step at the residual pathway itself, replacing one residual stream with multiple streams that each block reads, mixes, and updates. Manifold-Constrained Hyper-Connections (mHC) (Xie et al., 2026) retain this multi-stream topology while constraining stream transport to improve training stability.

However, a wider residual state does not necessarily realize a correspondingly wider set of representations. HC and mHC initialize the streams by replicating the same token representation, so multiplicity is present by construction whereas stream differentiation is not. Subsequent block computations may separate the streams, but the degree and organization of that separation are learned rather than specified by the topology. Recent stream-collapse analysis (Alimaskina et al., 2026) likewise makes uneven stream utilization an explicit concern and proposes Learned Stream Scaling (LSS) to mitigate dominant-stream behavior. These results motivate a more basic question for multi-stream residual architectures: how is their nominal state capacity realized geometrically during training, and how does that geometry change with depth?

We investigate this question along both optimization and depth. In an eight-layer, four-stream mHC training run, we track stream geometry at FFN outputs from 1k to 10k updates, averaging each diagnostic across layers. The effective stream width rises gradually, while linear CKA declines steadily and mean pairwise cosine decreases more moderately. The trajectory indicates that geometric differentiation develops over optimization, even as substantial cross-stream overlap remains. We then examine layer inputs in the pretrained 43-layer DeepSeek-V4-Flash-0731 model (DeepSeek-AI, 2026), measuring both attention and FFN interfaces. The streams are exactly aligned at the first attention interface, but their geometry changes non-monotonically through depth: effective width changes non-monotonically across depth, while cosine and CKA reveal complementary aspects of

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.

the evolving cross-stream geometry. Together, these observations suggest that stream differentiation is learned rather than supplied by the multi-stream topology itself.

Motivated by these observations, we propose **Differentiation-Augmented Hyper-Connections (DAHC)**, an extension of native mHC that explicitly introduces structured sources of stream differentiation both when the multi-stream state is created and throughout network depth. DAHC consists of two complementary components: **Differentiated Stream Expansion (DSE)** replaces the exact replicated initialization with causal, stream-indexed transformations using different receptive fields, providing structured and input-dependent differences from the point at which the streams are created; **Layerwise Differentiation Augmentation (LDA)** complements this one-time intervention by reusing the shared pre-expansion token representation at each layer, constructing depth-adapted stream candidates, and integrating them into the post-attention stream state through a gate conditioned on the current token representation. DSE therefore determines how cross-stream differences are introduced at initialization, whereas LDA provides additional stream-specific representation sources as the residual state evolves across depth. In the 8-layer setting, DAHC reaches the mHC training-loss and validation-perplexity targets at 7.3k and 7.5k updates, respectively (27% and 25% fewer updates). At a matched 10k-update budget, it lowers validation perplexity from 29.34 to 28.49 (2.90%). Our contributions are summarized as:

- We characterize stream differentiation in native mHC from along two axes: its emergence over the course of training and its non-monotonic reorganization across the depth of a pretrained model. Using cosine similarity, linear CKA, and effective stream width, we show that cross-stream geometry evolves substantially along both optimization and depth.
- We propose Differentiation-Augmented Hyper-Connections (DAHC), combining DSE for structured, stream-indexed causal initialization with LDA for state-conditioned, depth-adapted write-back at later layers, thereby supporting stream differentiation from initial expansion throughout network depth while preserving the native mHC pathway.
- We empirically evaluate DAHC in language modeling and stream-geometry analysis. In the 8-layer setting, DAHC reaches the mHC training-loss and validation-perplexity targets with 27% and 25% fewer updates, respectively, and lowers validation perplexity by 2.90% at a matched 10k-update budget, while exhibiting stronger stream differentiation under the reported geometry metrics.

2 Related Work

Architectural Scaling. Transformer scaling increasingly allocates capacity along different axes rather than uniformly widening dense layers. Sparse Mixture-of-Experts models, including Switch Transformer (Fedus et al., 2022), GLaM (Du et al., 2022), Mixtral (Jiang et al., 2024), and DeepSeekMoE (Dai et al., 2024), expand parameter capacity while routing each token through only part of the expert set. Conditional-computation methods instead vary where and how computation is spent: Mixture-of-Depths (Raposo et al., 2024) selects token positions for processing at each layer, Mixture-of-Modules (Gong et al., 2024) dynamically assembles attention and feed-forward modules, and Mixture-of-Recursions (Bae et al., 2026) combines parameter sharing with token-dependent recursion depth. A complementary line targets representation capacity. AltUp (Baykal et al., 2023) widens the persistent token representation while processing one sub-block per layer and updating the others through prediction and correction; Variable-Width Transformers (Wu et al., 2026) allocate hidden width non-uniformly across depth. Together, these works illustrate a broader shift from uniform scaling toward structured allocation of parameters, computation, depth, and representation capacity, motivating the residual state itself as another architectural scaling axis.

Residual Stream Architectures. A complementary line redesigns the state carried across layers or the pathways through which layers read and write it. The Highway Transformer (Chai et al., 2020) introduces content-dependent gating in the skip pathway, while ResiDual (Xie et al., 2023) maintains complementary residual accumulations. DenseFormer (Pagliardini et al., 2024) aggregates representations from multiple depths, and LAuReL (Menghani et al., 2025) augments residual updates with learned transformations. Residual Matrix Transformers (RMT) (Mak & Flanigan, 2025) replace the vector residual state with an outer-product memory matrix, making state capacity a more explicit design dimension. Hyper-Connections (HC) (Zhu et al., 2025) expand a single

residual stream into parallel streams that layers read, write, and mix; Manifold-Constrained Hyper-Connections (mHC) (Xie et al., 2026) constrain the residual mixing map through Sinkhorn normalization for stable multi-stream transport. Recent HC analyses make stream utilization an explicit concern: the stream-collapse analysis introduces Learned Stream Scaling (LSS) (Alimaskina et al., 2026) to address dominant-stream behavior. Together, these works establish residual-state topology as an architectural design axis and motivate asking how differentiation among its streams emerges during optimization and is reorganized across depth.

3 Preliminaries and Observations

Hyper-Connections. Hyper-Connections (HC) (Zhu et al., 2025) replace the single residual state with N parallel streams and learn how each block reads from, mixes, and writes back to them. Let $\mathbf{X}_{\ell,t} \in \mathbb{R}^{N\times d}$ denote the stream state at token t entering block $\mathcal{F}_\ell$. An HC block is written as

$$\mathbf{Y}_{\ell,t} = \mathbf{H}^{\mathrm{res}}_{\ell,t}\mathbf{X}_{\ell,t} + \mathbf{h}^{\mathrm{post}}_{\ell,t}\mathcal{F}_\ell\Big((\mathbf{h}^{\mathrm{pre}}_{\ell,t})^\top \mathbf{X}_{\ell,t}\Big)^\top, \tag{1}$$

where $\mathbf{h}^{\mathrm{pre}}_{\ell,t}, \mathbf{h}^{\mathrm{post}}_{\ell,t} \in \mathbb{R}^N$ control stream readout and write-back, and $\mathbf{H}^{\mathrm{res}}_{\ell,t} \in \mathbb{R}^{N\times N}$ mixes the residual streams. At the initial expansion, HC replicates the original token representation, $\mathbf{X}_{0,t} = \mathbf{1}_N\mathbf{h}_{0,t}^\top$, so all streams begin with identical content; later readout, residual mixing, and write-back operations can shape their geometry differently as the streams propagate through subsequent layers.

Manifold-Constrained Hyper-Connections (mHC) (Xie et al., 2026) keep the basic HC structure while adding constraints to the stream read, mixing, and write-back mappings:

$$\begin{aligned} \mathbf{h}^{\mathrm{pre}}_{\ell,t} &= \sigma(\widetilde{\mathbf{h}}^{\mathrm{pre}}_{\ell,t}), \qquad \mathbf{h}^{\mathrm{post}}_{\ell,t} = 2\sigma(\widetilde{\mathbf{h}}^{\mathrm{post}}_{\ell,t}), \\ \mathbf{H}^{\mathrm{res}}_{\ell,t} &= \mathrm{SinkhornKnopp}(\widetilde{\mathbf{H}}^{\mathrm{res}}_{\ell,t}). \end{aligned} \tag{2}$$

where σ denotes the sigmoid function. Sinkhorn–Knopp normalization constrains the residual mixing matrix to be nonnegative and doubly stochastic, with each row and column summing to one. Consequently, the residual pathway mixes streams through convex combinations, while the pre- and post-mappings remain nonnegative. Together, these constraints define the topology-preserving mHC transport regime used as our reference for examining stream geometry across layers.

Mean pairwise cosine. Let $\mathbf{x}_{\ell,t,n} \in \mathbb{R}^d$ denote the representation of token t in stream n at layer ℓ, where $n \in \{1,\dots,N\}$ and $N = 4$. To quantify local alignment between streams, we compute average cosine similarity for stream pairs over matched token positions:

$$\bar{c}_\ell = \frac{2}{N(N-1)}\sum_{i<j}\frac{1}{T}\sum_{t=1}^{T}\frac{\mathbf{x}_{\ell,t,i}^\top \mathbf{x}_{\ell,t,j}}{\|\mathbf{x}_{\ell,t,i}\|_2\|\mathbf{x}_{\ell,t,j}\|_2}. \tag{3}$$

Cosine measures angular alignment while ignoring magnitude. Higher cosine indicates stronger directional alignment between streams, whereas smaller values indicate greater directional distinction.

Linear CKA. Let $\mathbf{Z}_{\ell,n} \in \mathbb{R}^{T'\times d}$ contain the token-centered activations of stream n over the sampled positions. Linear CKA (Kornblith et al., 2019) between streams i and j is

$$\mathrm{CKA}(\mathbf{Z}_{\ell,i}, \mathbf{Z}_{\ell,j}) = \frac{\|\mathbf{Z}_{\ell,i}^\top \mathbf{Z}_{\ell,j}\|_F^2}{\|\mathbf{Z}_{\ell,i}^\top \mathbf{Z}_{\ell,i}\|_F\|\mathbf{Z}_{\ell,j}^\top \mathbf{Z}_{\ell,j}\|_F}. \tag{4}$$

We report the mean over stream pairs, $\overline{\mathrm{CKA}}_\ell = \frac{2}{N(N-1)}\sum_{i<j}\mathrm{CKA}(\mathbf{Z}_{\ell,i}, \mathbf{Z}_{\ell,j})$. Unlike cosine, which compares vector directions at matched token positions, CKA compares the centered representation geometry across tokens and is invariant to orthogonal changes of feature coordinates.

Effective stream width. For each stream, let $\boldsymbol{\mu}_{\ell,n} = T^{-1}\sum_t \mathbf{x}_{\ell,t,n}$. At each token, $\mathbf{V}_{\ell,t} \in \mathbb{R}^{N\times d}$ stacks the unit-normalized stream vectors $(\mathbf{x}_{\ell,t,n} - \boldsymbol{\mu}_{\ell,n})/\|\mathbf{x}_{\ell,t,n} - \boldsymbol{\mu}_{\ell,n}\|_2$. We average their Gram matrices across tokens, $\mathbf{G}_\ell = T^{-1}\sum_t \mathbf{V}_{\ell,t}\mathbf{V}_{\ell,t}^\top$, and define the participation-ratio statistic:

$$N_{\mathrm{eff},\ell} = \frac{\big(\mathrm{tr}\,\mathbf{G}_\ell\big)^2}{\mathrm{tr}(\mathbf{G}_\ell^2)}. \tag{5}$$

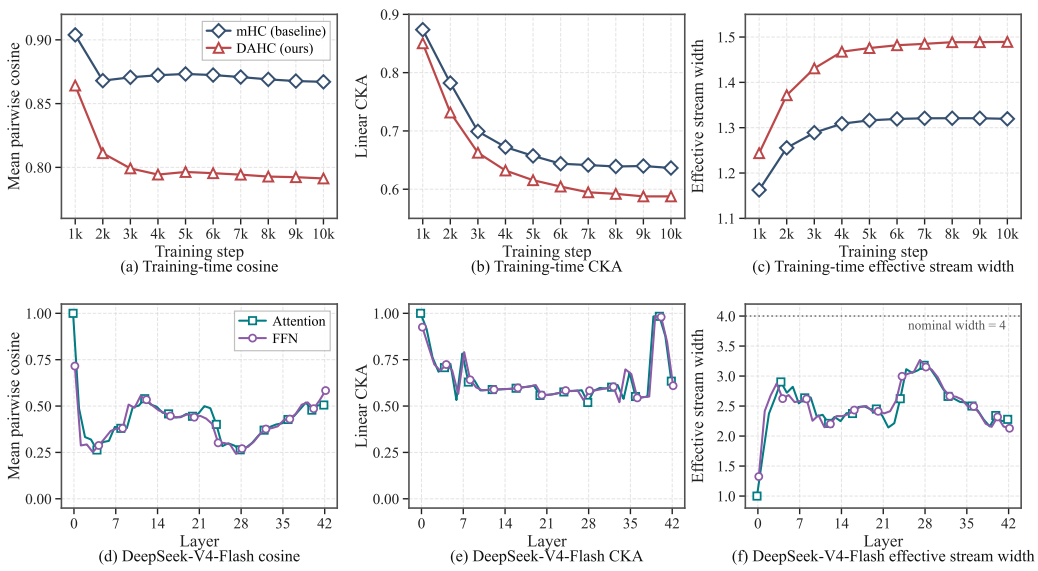


Figure 1: Stream geometry over training and depth. Top: training trajectories of three geometry metrics at MLP outputs, averaged over all layers. Bottom: layer-wise stream geometry at the attention and FFN inputs of the 43-layer DeepSeek-V4-Flash-0731 model (DeepSeek-AI, 2026).

For four streams, $N_{\text{eff}} = 1$ when the normalized stream representations are identical and $N_{\text{eff}} = 4$ when they are mutually orthogonal. Intermediate values summarize how concentrated the stream-wise geometry is: smaller values indicate stronger redundancy, whereas larger values indicate a greater effective number of represented directions within the four-stream state at each layer.

Observation 1: Stream differentiation develops progressively during optimization. We first examine how cross-stream geometry evolves during training in an eight-layer, four-stream mHC model with hidden size $d = 768$ and sequence length 1024. At checkpoints from 1k to 10k training steps, we evaluate the model on the validation split associated with the training corpus and compute the three geometry metrics from stream states at feed-forward network (FFN) outputs, averaging uniformly across layers and validation forwards.

As shown in the top row of Figure 1, the native mHC exhibits a clear but gradual evolution of stream geometry over training. Effective stream width increases, while mean pairwise CKA decreases steadily and cosine similarity declines more moderately before largely stabilizing. Since the standard mHC expansion begins from exact replicas, this trajectory indicates that geometric distinctions among streams are acquired through optimization rather than supplied at initialization. Taken together, the metrics show this differentiation developing gradually, while substantial cross-stream overlap persists across the full 10k-step training horizon of this eight-layer mHC run evaluated here.

Observation 2: Stream differentiation is reorganized non-monotonically across depth. We next examine stream geometry across depth in the pretrained 43-layer DeepSeek-V4-Flash-0731 model (DeepSeek-AI, 2026), using 33 WikiText-103 validation sequences (Merity et al., 2017) of length 4096. We measure the four-stream states at the attention and FFN inputs of each layer, averaging statistics across sequences; linear CKA uses a fixed set of sampled token positions. Unlike the training analysis at FFN outputs, this layer-input analysis includes the first attention interface, where the streams are still exactly aligned, and follows their changing geometry at both attention and FFN inputs across all 43 layers, from the initial alignment to the final FFN interface.

The bottom row of Figure 1 shows that the resulting differentiation is strongly depth-dependent. The streams begin in exact agreement at the first attention interface and separate rapidly in the early layers, but this separation does not increase monotonically with depth. Instead, effective stream width repeatedly expands and contracts, while both cosine and CKA exhibit substantial changes in alignment across different depth ranges. Later layers also show that the three metrics need not move

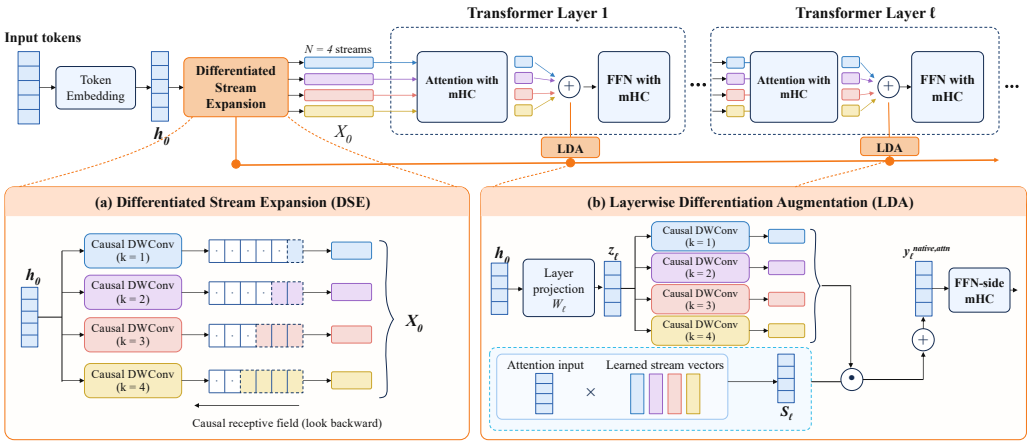


Figure 2: Overview of Differentiation-Augmented Hyper-Connections (DAHC). DAHC augments native mHC at stream creation and throughout depth. DSE replaces replicated initialization with stream-indexed causal transformations. LDA constructs depth-adapted stream candidates from $\mathbf{h}_0$ and integrates them through state-conditioned write-back.

together: streams can exhibit highly similar centered representation geometry under CKA while remaining only moderately aligned in their matched-token directions. Thus, stream differentiation is not a fixed property established once and propagated unchanged through the network; rather, its geometry is continually reorganized by subsequent computation and stream mixing.

4 Proposed Method

Challenges. The observations in Section 3 reveal two challenges for multi-stream residual representations. First, standard mHC creates streams from exact replicas, leaving their differentiation to emerge entirely through optimization. Second, the resulting stream geometry is continually reorganized across depth, so differences introduced only at the network entrance need not remain sufficient for later layers. These observations motivate architectural support for stream differentiation both at stream creation and throughout network depth, rather than only at the initial expansion.

Overview. We propose **Differentiation-Augmented Hyper-Connections (DAHC)**, which augments the native mHC pathway with two complementary components. Differentiated Stream Expansion (DSE) replaces replicated initialization with stream-indexed causal transformations, introducing structured, input-dependent differences when the streams are created. Layerwise Differentiation Augmentation (LDA) provides additional depth-adapted stream candidates at later layers and integrates them through state-conditioned write-back. DSE therefore determines how stream differentiation begins, while LDA supplies additional stream-specific representation sources as the residual state evolves. The native mHC read–mix–write pathway is retained throughout.

4.1 Differentiated Stream Expansion

Standard mHC creates multiple residual streams by replicating the same token representation, so the expansion operation itself introduces no stream-specific representation structure. Motivated by the gradual development of stream differentiation observed in Section 3, we instead introduce structured differences directly when the stream state is created. We instantiate this idea with a family of stream-indexed causal transformations. Specifically, the n-th stream is initialized as

$$\mathbf{X}_{0,:,n} = \mathcal{T}_n(\mathbf{h}_0) = \mathrm{CausalDWConv}_{k_n}(\mathbf{h}_0), \qquad n = 1, \ldots, N, \tag{6}$$

where $\mathbf{h}_0 \in \mathbb{R}^{T \times d}$ is the token representation before stream expansion and k_n specifies the causal receptive field of stream n. The depthwise convolution operates along the sequence dimension, so position t depends only on positions up to t. Using different receptive fields provides a lightweight, input-dependent source of variation across streams while preserving autoregressive causality.

For the four-stream setting, we use $(k_1, k_2, k_3, k_4) = (1, 2, 3, 4)$, with the $k_1 = 1$ branch kept as the identity. Thus, one stream preserves the original token representation, while the others view the same input sequence through progressively different local contexts. DSE therefore replaces exact replication with a structured differentiated starting state without assigning predefined semantic roles.

4.2 Layerwise Differentiation Augmentation

Because the cross-stream geometry continues to be reorganized throughout depth, a one-time intervention at the network entrance may be insufficient. Rather than deriving all later stream candidates from the already-evolving multi-stream state, we Layerwise Differentiation Augmentation (LDA), which re-reads the shared representation $\mathbf{h}_0$ at every layer. Since this common source needs to be interpreted differently at different depths, we first apply a layer-specific projection and then reuse the stream-indexed receptive-field design to construct one candidate for each stream.

$$\mathbf{z}_\ell = \mathbf{h}_0 \mathbf{W}_\ell, \qquad \mathbf{W}_\ell \in \mathbb{R}^{d \times d}. \tag{7}$$

$$\mathbf{c}_{\ell,:,n} = \mathcal{T}_{\ell,n}(\mathbf{z}_\ell) = \mathrm{CausalDWConv}_{k_n}(\mathbf{z}_\ell), \qquad n = 1, \ldots, N. \tag{8}$$

where $\mathcal{T}_{\ell,n}$ is layer-specific, with convolutional parameters not shared across layers and receptive-field assignments $(k_1, \ldots, k_N)$ inherited from DSE. For $k_n > 1$, the causal filters are initialized as moving averages and learned thereafter, while $k_n = 1$ remains the identity. $\mathbf{W}_\ell$ provides depth-specific adaptation and $\mathcal{T}_{\ell,n}$ introduces structured variation across streams.

State-conditioned write-back. The additional candidates need not be incorporated with equal strength at every token, layer, and stream. We therefore condition their write-back on the current token state. Let $\mathbf{q}_{\ell,t} \in \mathbb{R}^d$ denote the representation passed to self-attention at layer ℓ, and let $\mathbf{g}_{\ell,n} \in \mathbb{R}^d$ be a learned gating vector. We define

$$\begin{aligned} s_{\ell,t,n} &= \alpha \left[1 + \tanh\left(\frac{\langle \mathbf{q}_{\ell,t}, \mathbf{g}_{\ell,n} \rangle}{\sqrt{d}} \right) \right], \\ \mathbf{Y}^{\mathrm{attn}}_{\ell,:,n} &= \mathbf{Y}^{\mathrm{native,attn}}_{\ell,:,n} + s_{\ell,:,n} \odot \mathbf{c}_{\ell,:,n}, \end{aligned} \tag{9}$$

where $\alpha > 0$ controls the overall write-back scale and $s_{\ell,t,n} \in (0, 2\alpha)$. The gate allows the evolving model state to adaptively determine how strongly each stream-specific candidate contributes at a given token and depth, rather than applying a fixed coefficient across tokens or layers.

The gated candidates are inserted after the native attention-side mHC mapping and before the subsequent FFN-side mHC block. The original mHC read–mix–write pathway remains unchanged; in particular, the auxiliary term is not itself passed through the native post-mapping or Sinkhorn normalization. We therefore do not require the complete augmented update to lie on the original mHC manifold. Instead, LDA retains the constrained mHC backbone while providing later layers with an additional, adaptively integrated source of stream-indexed representations.

4.3 Causality and Computational Cost

Both DSE and LDA use causal convolutions along the sequence dimension, so the representation at position t depends only on positions up to t, and the proposed pathways introduce no future-token information. For an L-layer model, LDA introduces one layer-specific projection $\mathbf{W}_\ell \in \mathbb{R}^{d \times d}$, resulting in $O(Ld^2)$ additional parameters and $O(LTd^2)$ computation for a sequence of length T. The stream-indexed depthwise convolutions add $O(Ld \sum_{n=1}^{N} k_n)$ parameters and $O(LTd \sum_{n=1}^{N} k_n)$ computation, while the state-conditioned gates require $O(LNd)$ parameters and $O(LTNd)$ computation. DSE applies the same stream-indexed causal transformations only once at the initial expansion, and therefore contributes substantially less overhead. For the four-stream setting with $(k_1, k_2, k_3, k_4) = (1, 2, 3, 4)$, the convolution is linear in sequence length and hidden dimension.

5 Experiments

5.1 Experimental Setup

Dataset. All pretraining experiments use DCLM-Baseline 1.0 (Li et al., 2024), a large-scale web-text corpus derived from Common Crawl and curated through deduplication, heuristic cleaning,

and model-based quality filtering. The released baseline contains approximately 3.8T tokens and serves as the common pretraining source for all experiments in this work. Training sequences are sampled from fixed DCLM shard lists, while evaluation is performed on a held-out validation split from the same data distribution. We use a fixed random seed of 42 throughout. Across compared configurations, we keep the data shards, sampling procedure, and evaluation protocol unchanged to minimize variation in data exposure and ensure that observed differences are attributable primarily to the model architecture design rather than to changes in the training data.

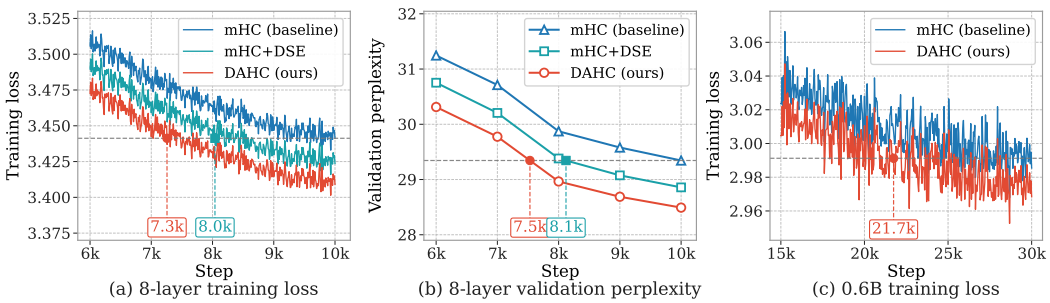
Models and baselines. All experiments use the same decoder-only Transformer backbone, with multi-head self-attention (Vaswani et al., 2017), rotary positional embeddings (RoPE) (Su et al., 2024), SwiGLU FFN (Shazeer, 2020), and four-stream mHC residual pathways. We use the Qwen3 tokenizer (Qwen Team, 2025), whose vocabulary contains approximately 152K entries and is held fixed across configurations. We compare three configurations. **mHC** denotes the native Manifold-Constrained Hyper-Connections baseline (Xie et al., 2026), where streams are initialized by replication and updated through the standard constrained read–mix–write operators. **mHC+DSE** replaces the replicated initialization with Differentiated Stream Expansion, isolating the effect of structured differentiation at stream creation. **DAHC** denotes the full model, combining DSE with Layerwise Differentiation Augmentation throughout depth. Since DSE and LDA introduce different parameter overheads, we adjust the FFN intermediate dimension of each configuration to approximately match total parameter counts. This setup reduces model size as a confounding factor, while other backbone and training settings remain fixed unless otherwise specified.

Implementation details. All models are trained with MindSpeed-LLM on an Ascend 910C cluster using bfloat16 precision. Unless otherwise specified, we use a global batch size of 1024, Adam with $(\beta_1, \beta_2) = (0.9, 0.95)$, weight decay 0.1, gradient clipping at 1.0, and a cosine learning-rate schedule. The Transformer backbone uses rotary positional embeddings, SwiGLU feed-forward layers, and Flash Attention. Model depth, hidden dimension, and FFN width vary across scales and are reported with the corresponding experiments. For most experiments, we use sequence length 1024 and train for 10,000 updates, corresponding to approximately 10.5B training tokens. For the largest 0.6B-scale, 12-layer setting, we increase the sequence length to 4096 and train for 30,000 updates, yielding approximately 125.8B token positions. All runs use the same data and optimization pipeline described above, with architecture-specific settings varied only where noted.

Metrics. We evaluate language modeling using validation loss and perplexity (PPL), with lower values indicating better predictive performance. For representation analysis, we use the three stream-geometry diagnostics introduced in Section 3: mean pairwise cosine similarity, linear CKA, and effective stream width N_{eff}. Cosine and CKA measure complementary forms of cross-stream similarity, while N_{eff} summarizes the effective dimensionality of the stream geometry. These geometry metrics are used to analyze representation dynamics, cross-stream differentiation, and their evolution over training and depth, rather than as direct measures of language-modeling quality.

5.2 Main Results

Small-scale comparison. We first compare mHC, mHC+DSE, and DAHC on the 8-layer model under approximately matched parameter budgets. Although the model contains roughly 0.3B parameters in total, a substantial fraction resides in the untied input embedding and language-modeling head, leaving a comparatively smaller Transformer backbone. As shown in Figure 3(a)–(b), both mHC+DSE and DAHC consistently improve the optimization trajectory over native mHC, with DAHC yielding the strongest results. DAHC reaches the final 10k-step mHC training loss and validation perplexity at approximately 7.3k and 7.5k steps, respectively, requiring about 25–27% fewer optimization steps; mHC+DSE reaches the same targets at approximately 8.0k and 8.1k steps. The advantage also remains at the matched 10k-step budget: validation perplexity decreases from 29.34 for mHC to 28.86 for mHC+DSE and 28.49 for DAHC, corresponding to reductions of 1.64% and 2.90%, respectively. DAHC further improves perplexity by 1.28% over mHC+DSE, indicating that DSE already provides a measurable gain while LDA contributes additional benefit and further improves both optimization efficiency and final language-modeling performance.



(a) 8-layer training loss (b) 8-layer validation perplexity (c) 0.6B training loss

Figure 3: Language-modeling optimization across model scales. (a)–(b) On the 8-layer model, mHC+DSE and DAHC reach the 10k-step mHC training-loss and validation PPL levels earlier. (c) At the 0.6B scale, DAHC reaches the 30k-step mHC training-loss level earlier. Horizontal dashed lines denote mHC baseline levels, and vertical dashed lines mark the corresponding crossing steps.

Table 1: Validation perplexity across model depths. Δ denotes the relative reduction over mHC.

Depth	d	mHC	Ours	Δ
8	768	30.32	29.73	-1.97%
8	1024	27.26	26.72	-1.98%
24	768	24.98	24.44	-2.15%
24	1024	22.72	22.11	$\mathbf{-2.70}\%$

Larger-scale comparison. We next examine whether this advantage persists with a substantially larger Transformer backbone and a longer training horizon. As shown in Figure 3(c), the augmented model reaches the final 30k-step mHC training loss after approximately 21.7k steps, reducing the required optimization steps by about 28%. At the same 30k-step budget, it also achieves a lower final training loss, decreasing from 2.9855 to 2.9684. Importantly, the earlier baseline crossing is maintained despite the increase in backbone capacity and training compute, indicating that the optimization advantage is not limited to the smaller model setting. The lower matched-budget loss further suggests that the benefit is retained throughout extended training rather than arising only from faster early-stage convergence. Together, these results provide additional evidence that the proposed augmentation remains effective as the Transformer backbone is scaled up.

5.3 Further Experiments

Performance across model depths. We further examine whether the benefit of the proposed augmentation is specific to a particular network depth. Table 1 compares mHC with the augmented model using 8- and 24-layer Transformers at hidden dimensions of 768 and 1024. Across all four configurations, the augmented model achieves lower validation perplexity than the corresponding mHC baseline. For the 8-layer models, perplexity is reduced by approximately 2.0% at both hidden sizes. The improvement persists in the deeper 24-layer models, with reductions of 2.15% and 2.70% for hidden dimensions 768 and 1024, respectively. These results indicate that the observed advantage is not restricted to the depth used in our primary experiment, but remains present when the Transformer stack is made substantially deeper. Together with the larger-scale result in Section 5.2, this provides additional evidence that the augmentation remains effective across different backbone configurations, model depths, and capacity regimes considered here.

Layerwise stream-geometry dynamics. We first revisit the aggregate training-time geometry in Figure 1, where the top row averages the three diagnostics over all layers. Compared with native mHC, DAHC consistently exhibits lower mean pairwise cosine similarity and linear CKA together with a larger effective stream width throughout training. Thus, at the model level, DAHC develops stronger cross-stream differentiation under all three diagnostics. We next examine where this aggregate difference arises by resolving the same metrics at individual layers and additionally including mHC+DSE to separate the effects of differentiated initialization and layerwise augmentation.

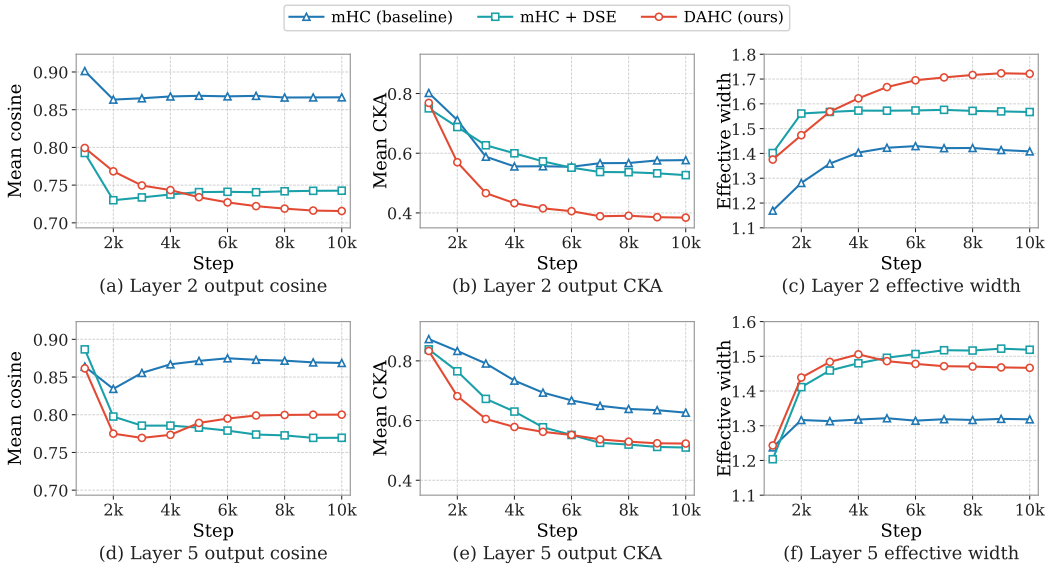


Figure 4: Layer-resolved stream geometry during training. We track mean pairwise cosine similarity, linear CKA, and effective stream width at the outputs of Layers 2 (top) and 5 (bottom) for native mHC, mHC+DSE, and DAHC. Both augmented models produce geometry distinct from native mHC, while their relative differences vary across depth.

At the relatively shallow Layer 2, Figure 4(a)–(c) shows that the distinction is already pronounced. Both mHC+DSE and DAHC move substantially away from native mHC early in training: cosine similarity and CKA decrease, while effective stream width increases. DSE therefore provides a clear source of stream differentiation beyond replicated mHC initialization. The full DAHC model further strengthens this trend, ending training with the lowest cosine similarity and CKA and the largest effective stream width among the three configurations. This shallow-layer behavior helps explain the stronger layer-averaged differentiation observed in Figure 1.

At the deeper Layer 5, shown in Figure 4(d)–(f), both mHC+DSE and DAHC exhibit geometry that is more differentiated than native mHC, particularly in CKA and effective stream width. However, the ordering between DSE and DAHC is no longer uniform across metrics: their trajectories become closer, and DSE is slightly more differentiated under some diagnostics near the end of training. This indicates that LDA does not simply increase stream separation by a fixed amount at every layer. Instead, its effect depends on the evolving layer state and is reshaped by deeper computation. Together with the aggregate results in Figure 1, these layer-resolved trajectories suggest that DAHC strengthens stream differentiation overall while retaining depth-dependent geometry rather than enforcing uniform separation throughout the network.

6 Conclusion

This work reframes multi-stream residual design as a question of how representational differences are formed and supplied across the network, rather than simply how many streams are instantiated. Our analysis shows that the geometry of mHC streams is shaped by optimization and remains dynamically reconfigured through depth, motivating architectural mechanisms that act at both stream creation and later computation. We proposed Differentiation-Augmented Hyper-Connections (DAHC), which combines Differentiated Stream Expansion (DSE) and Layerwise Differentiation Augmentation (LDA) to introduce structured sources of stream differentiation at stream creation and throughout depth while retaining the native mHC pathway. In the 8-layer setting, DAHC reaches the mHC training-loss and validation-perplexity targets with 27% and 25% fewer updates, respectively, and lowers validation perplexity from 29.34 to 28.49 (2.90%) at a matched 10k-update budget. Overall, our results suggest that how multi-stream residual representations differentiate and evolve is an important architectural design dimension.

AI USE STATEMENT

We used generative AI tools to assist with manuscript drafting and language editing. We did not use them to generate data, formulate claims, design or implement experiments, interpret results, or create figures. All AI-assisted content was reviewed by the authors, who take full responsibility for the final manuscript.

ETHICS STATEMENT

This work is a foundational study of language-model representations and does not involve human subjects, annotators, user data, or personally identifiable information. We use publicly available datasets and model resources and release no new sensitive data. We are not aware of ethical concerns beyond the limitations and potential biases of the underlying language-model data.

REPRODUCIBILITY STATEMENT

The model, metrics, datasets, aggregation procedures, and experimental configurations are described in Sections 3 and 5. We will provide the processed analysis data and plotting scripts as supplementary material to facilitate reproduction of the reported figures.

REFERENCES

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

Sangmin Bae, Yujin Kim, Reza Bayat, Sungnyun Kim, Jiyoun Ha, Tal Schuster, Adam Fisch, Hrayr Harutyunyan, Ziwei Ji, Aaron C. Courville, and Se-Young Yun. Mixture-of-recursions: Learning dynamic recursive depths for adaptive token-level computation. In *Advances in Neural Information Processing Systems*, volume 38, 2026.

Cenk Baykal, Dylan Cutler, Nishanth Dikkala, Nikhil Ghosh, Rina Panigrahy, and Xin Wang. Alternating updates for efficient transformers. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Yekun Chai, Shuo Jin, and Xinwen Hou. Highway transformer: Self-gating enhanced self-attentive networks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 6887–6900, 2020.

Damai Dai, Chengqi Deng, Chenggang Zhao, R. X. Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y. Wu, Zhenda Xie, Y. K. Li, Panpan Huang, Fuli Luo, Chong Ruan, Zhifang Sui, and Wenfeng Liang. DeepSeekMoE: Towards ultimate expert specialization in mixture-of-experts language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1280–1297, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.70. URL https://aclanthology.org/2024.acl-long.70/.

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

Nan Du, Yanping Huang, Andrew M. Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, Barret Zoph, Liam Fedus, Maarten P. Bosma, Zongwei Zhou, Tao Wang, Emma Wang, Kellie Webster, Marie Pellat, Kevin Robinson, Kathleen Meier-Hellstern, Toju Duke, Lucas Dixon, Kun Zhang, Quoc Le, Yonghui Wu, Zhifeng Chen, and Claire Cui. GLaM: Efficient scaling of language models with mixture-of-experts. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162, pp. 5547–5569, 2022.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.

Zhuocheng Gong, Ang Lv, Jian Guan, Junxi Yan, Wei Wu, Huishuai Zhang, Minlie Huang, Dongyan Zhao, and Rui Yan. Mixture-of-modules: Reinventing transformers as dynamic assemblies of modules. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 20924–20938, Miami, Florida, USA, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.1164. URL https://aclanthology.org/2024.emnlp-main.1164/.

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.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pp. 3519–3529, 2019.

Jeffrey Li, Alex Fang, Georgios Smyrnis, et al. DataComp-LM: In search of the next generation of training sets for language models. In *Advances in Neural Information Processing Systems: Datasets and Benchmarks Track*, volume 37, 2024.

Brian Mak and Jeffrey Flanigan. Residual matrix transformers: Scaling the size of the residual stream. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 42712–42729, 2025.

Gaurav Menghani, Ravi Kumar, and Sanjiv Kumar. LAuReL: Learned augmented residual layer. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 43826–43836, 2025.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=Byj72udxe.

Matteo Pagliardini, Amirkeivan Mohtashami, Francois Fleuret, and Martin Jaggi. DenseFormer: Enhancing information flow in transformers via depth weighted averaging. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Qwen Team. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

David Raposo, Sam Ritter, Blake Richards, Timothy Lillicrap, Peter Conway Humphreys, and Adam Santoro. Mixture-of-depths: Dynamically allocating compute in transformer-based language models. *arXiv preprint arXiv:2404.02258*, 2024.

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

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. RoFormer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024. doi: 10.1016/j.neucom.2023.127063.

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*, volume 30, 2017.

Zhaofeng Wu, Oliver Sieberling, Shawn Tan, Rameswar Panda, Yury Polyanskiy, and Yoon Kim. Variable-width transformers. *arXiv preprint arXiv:2606.18246*, 2026.

Shufang Xie, Huishuai Zhang, Junliang Guo, Xu Tan, Jiang Bian, Hany Hassan Awadalla, Arul Menezes, Tao Qin, and Rui Yan. ResiDual: Transformer with dual residual connections. *arXiv preprint arXiv:2304.14802*, 2023.

Zhenda Xie, Yixuan Wei, Huanqi Cao, Chenggang Zhao, Chengqi Deng, Jiashi Li, Damai Dai, Huazuo Gao, Jiang Chang, Kuai Yu, Liang Zhao, Shangyan Zhou, Zhean Xu, Zhengyan Zhang, Wangding Zeng, Shengding Hu, Yuqing Wang, Jingyang Yuan, Lean Wang, and Wenfeng Liang. mHC: Manifold-constrained Hyper-Connections. In *Proceedings of the 43rd 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*, 2025.

APPENDIX

A COMPLETE LAYERWISE STREAM-GEOMETRY TRAJECTORIES

The layer-resolved analysis in Section 5.3 reports representative trajectories for Layers 2 and 5. To provide a more complete view across network depth, Figure 5 reports the same three stream-geometry diagnostics for the remaining layers of the 8-layer model. Together with Figure 4, these results cover all eight layers and track the evolution of native mHC, mHC+DSE, and DAHC throughout training. The complete trajectories further illustrate that the effect of the proposed augmentation is strongly depth dependent. In the early and middle layers, particularly Layers 1, 3, and 4, mHC+DSE and DAHC generally exhibit lower cross-stream similarity and larger effective stream width than native mHC, although their relative ordering depends on the metric and layer. These differences are especially pronounced in several intermediate layers, where the augmented models develop substantially different geometry trajectories from the native baseline.

In the deeper layers, the geometry becomes less uniformly ordered across configurations. At Layers 6–8, the trajectories of mHC, mHC+DSE, and DAHC are closer under several diagnostics, and no single configuration exhibits the strongest differentiation under every metric. This behavior is consistent with the depth-wise analysis in the main text: the stronger layer-averaged differentiation of DAHC does not result from imposing a uniform increase in stream separation at every layer. Instead, DSE and LDA alter how cross-stream differences emerge and are subsequently reorganized by deeper network computation.

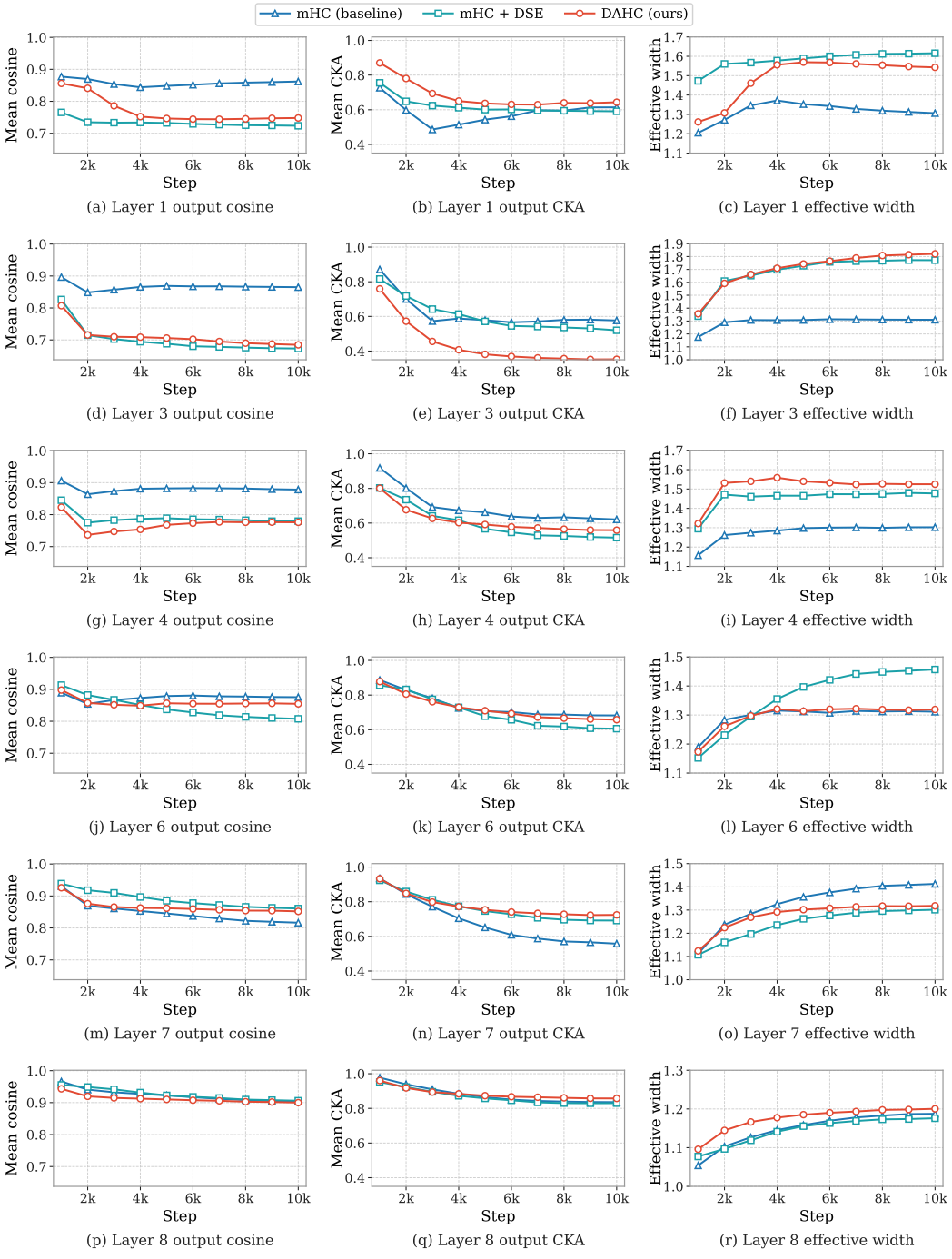


Figure 5: Complete layerwise stream-geometry trajectories during training. We report mean pairwise cosine similarity, linear CKA, and effective stream width at the outputs of Layers 1, 3, 4, 6, 7, and 8 for native mHC, mHC+DSE, and DAHC. Results for Layers 2 and 5 are shown in Figure 4. The relative geometry of the three configurations varies across depth, complementing the layer-averaged trends in Figure 1.