# Harmonized Hyper-Connections:
# Stabilizing Residual Transport Through Feedback on Geometry

J. W. Miller
j.w.miller@curv.institute
Founder & Research Director
CURV Institute

**Abstract**

Hyper-Connections (HC) expand the residual stream and introduce learnable residual mixing, increasing expressivity but often compromising training stability due to unbounded signal amplification across depth. Manifold-Constrained Hyper-Connections (mHC) address this instability by projecting residual transport onto a doubly stochastic manifold, restoring identity-like propagation and bounding composite gain. However, hard manifold projection can overconstrain residual transport and degrade performance on routing-heavy tasks.

We propose a minimal alternative: *Harmonized Hyper-Connections*, which enforce stability through feedback acting on the geometry of residual transport by directly regulating the *applied* composite gain. We evaluate unconstrained HC, mHC, and Harmonized Hyper-Connections on a key–value retrieval task across multiple random seeds and extended training horizons. Unconstrained HC exhibits large gain variance across seeds and monotonic gain drift over time, while mHC enforces perfect stability at the cost of routing capability. In contrast, Harmonized Hyper-Connections maintain bounded applied gain with low variance while recovering task performance comparable to unconstrained HC. These results demonstrate that global feedback control can stabilize residual transport without hard manifold projection, provided control is applied to effective transport geometry rather than raw parameters.

## 1 Introduction

Residual connections owe their effectiveness to the identity mapping property, which enables stable forward and backward signal propagation through depth [1, 2]. Hyper-Connections (HC) generalize this paradigm by widening the residual stream and introducing learnable residual mixing matrices, increasing architectural expressivity without increasing the FLOPs of the core computation [3]. Residual streams are a common architectural motif across deep learning models, including CNNs, MLPs, diffusion models, and Transformers. Empirically, this additional flexibility can yield meaningful performance gains.

However, when extended across many layers, unconstrained residual transport compromises identity propagation. Signals can be amplified or attenuated multiplicatively through depth, leading to instability, gradient explosion, or late-stage training failure.

Manifold-Constrained Hyper-Connections (mHC) made this failure mode precise by analyzing the composite residual mapping across layers and identifying unbounded amplification as the governing mechanism [4]. mHC introduced the Amax composite gain—defined via the maximum absolute row and column sums of the composed residual transport—as a diagnostic that correlates strongly with instability, and proposed restoring stability by projecting residual transport onto the manifold of doubly stochastic matrices.

This work studies a complementary question: whether stability must be enforced through hard per-layer manifold constraints, or whether it can instead be achieved through feedback acting on the geometry of residual transport itself.

## 2 Background and Composite Gain

In HC-style architectures, the residual stream update takes the form

$$x_{\ell+1} = H_\ell^{\mathrm{res}} x_\ell + H_\ell^{\mathrm{post}\top} F(H_\ell^{\mathrm{pre}} x_\ell, W_\ell),$$

where $H_\ell^{\mathrm{res}} \in \mathbb{R}^{n\times n}$ mixes $n$ parallel residual streams.

Across depth, signal propagation is governed by the composite residual mapping

$$Y = \prod_{\ell=1}^{L} H_\ell^{\mathrm{res}}.$$

Following mHC, instability is quantified using the *Amax composite gain magnitude*, defined as the maximum absolute row sum (forward gain) and column sum (backward gain) of $Y$ [4]. Large composite gain indicates amplification under depth composition and is strongly associated with training instability.

## 3 Method: Harmonized Hyper-Connections

Harmonized Hyper-Connections enforce stability through feedback acting on the geometry of residual transport, rather than through hard per-layer constraints. Residual transport at each layer is parameterized as a deviation from identity,

$$H_\ell^{\mathrm{raw}} = I + \epsilon\Theta_\ell, \qquad H_\ell^{\mathrm{res}}(s) = I + s\,(H_\ell^{\mathrm{raw}} - I),$$

where $\Theta_\ell$ are unconstrained learnable parameters, $\epsilon$ is a small constant, and $s \in [s_{\min}, 1]$ is a global transport scale.

At each training step, the controller computes two quantities: the composite gain of the raw transport and the composite gain of the *applied* transport formed from $H_\ell^{\mathrm{res}}(s)$. The transport scale $s$ is updated using feedback on the applied composite gain to maintain a target gain budget.

Crucially, control is applied to the transport actually used by the network rather than to raw parameter magnitudes. This decouples parameter drift from functional transport, allowing raw parameters to evolve freely while effective residual transport remains bounded and stable.

## 4 Experimental Setup

We evaluate on a synthetic key–value retrieval task with sequence length 256 and 8 key–value pairs. A query token appears in the sequence and the model must output the associated value. This task emphasizes selective routing rather than memorization and is commonly used as a probe of associative recall and memory routing [5, 6].

Key–value retrieval is deliberately chosen as a routing-heavy probe, as it requires moderate amplification and suppression of residual pathways rather than uniform convex mixing.

All experiments use a 96-layer Transformer variant with $n = 16$ residual streams, embedding dimension 128, batch size 8, and identical attention, optimizer, and training settings. Each configuration is evaluated across random seeds 0, 1, and 2. Code, logs, and plots sufficient to reproduce all experiments are available at: `https://github.com/curv-institute/harmonized-hyper-connections`.

# 5 Results

## 5.1 Seed Robustness

Across seeds, unconstrained HC exhibits severe instability. Composite gain varies widely (approximately 12–33), accompanied by large variance in task accuracy, including complete failure for some initializations. These results demonstrate that unconstrained residual mixing produces non-repeatable training dynamics.

mHC enforces perfect stability across all seeds, with composite gain tightly bounded near unity. However, task accuracy collapses to near-chance levels, confirming that hard manifold constraints remove transport degrees of freedom required for routing-heavy tasks.

Harmonized Hyper-Connections enforce strong and repeatable stability across all seeds. Applied composite gain remains tightly bounded with low variance, while task accuracy recovers to levels comparable to the mean performance of unconstrained HC.

## 5.2 Long-Horizon Stability

In long-horizon training, unconstrained HC exhibits monotonic, delayed gain drift. Composite gain grows from values near unity early in training to values exceeding $10^2$–$10^3$ at later stages, despite reasonable intermediate accuracy. This confirms that HC instability is time-coupled and compositional.

In contrast, Harmonized Hyper-Connections maintain bounded applied composite gain over the full training horizon. While raw composite gain grows without bound, applied gain remains stable near the target value, and training remains well-behaved. Task accuracy is preserved without the instability observed in unconstrained HC.

# 6 Interpretation

These results establish three distinct regimes of residual transport. Unconstrained HC offers expressivity but suffers from seed-dependent and time-coupled instability. mHC achieves perfect stability through hard per-layer constraints but collapses routing capability. Harmonized Hyper-Connections occupy a third regime in which raw parameters are unconstrained, but effective transport geometry is regulated through feedback.

Stability and capability are reconciled when control is applied to the geometry of applied transport rather than to raw parameter norms. This decoupling enables stable long-horizon training while preserving routing capacity.

# 7 Practical Implications

Applied-gain feedback reduces late-stage training failures without resorting to hard architectural constraints, improving reliability and reducing wasted compute in long-horizon training. By shaping effective transport rather than suppressing it, the approach expands the viable design space for residual architectures that require selective routing or internal amplification.

Stability becomes a tunable property via an explicit gain target, allowing different operating regimes to be selected across training phases or deployment contexts. Separating raw parameter dynamics from applied transport geometry also provides clearer diagnostics for safe scaling and controlled experimentation.

## 8 Conclusion and Acknowledgment

This work builds directly on the analysis introduced in *Manifold-Constrained Hyper-Connections* by Zhenda Xie and colleagues at DeepSeek-AI, who identified composite residual gain as the governing instability mechanism [4].

In parallel with that line of research, researchers at the CURV Institute have been independently investigating the same residual transport instability and its underlying geometric structure. We present a complementary approach that stabilizes residual transport through feedback acting on applied geometry rather than strict per-layer manifold projection.

These results demonstrate that feedback acting on effective transport geometry provides a viable alternative to hard manifold constraints for stabilizing widened residual architectures while preserving routing capability.

## References

[1] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[2] K. He, X. Zhang, S. Ren, and J. Sun. Identity mappings in deep residual networks. In *ECCV*, 2016.

[3] D. Zhu et al. Hyper-Connections. arXiv preprint arXiv:2409.19606, 2024.

[4] Z. Xie et al. Manifold-Constrained Hyper-Connections. arXiv preprint arXiv:2512.24880, 2025.

[5] J. Weston, S. Chopra, and A. Bordes. Memory networks. arXiv preprint arXiv:1410.3916, 2014.

[6] A. Graves et al. Neural Turing machines. arXiv preprint arXiv:1410.5401, 2014.