Depth scaling and Muon enable balanced expert usage in MoE training

author names withheld

Under Review for the Workshop on High-dimensional Learning Dynamics, 2026

Abstract

Mixture of Experts (MoE) Transformers rely on a router to distribute tokens across experts, and a severely unbalanced router wastes model capacity. We study load balance at *random initialization*, before any auxiliary loss has had an effect, and show that it is fundamentally a question about the geometry of the hidden states entering the router. When hidden states across tokens are diverse, a random linear router produces diverse logits and top-k selection naturally spreads tokens across experts; when hidden states collapse, the router collapses with them. We connect this observation to representation collapse in deep pre-norm Transformers and argue that $1/\sqrt{L}$ depth scaling, beyond its known benefits for training stability, also improves routing balance at initialization. We additionally observe Muon better preserves this balance during training by producing orthogonalized updates to the router and expert matrices, and verify our claims empirically.

1. Introduction

Depth scaling enables hyperparameter transfer across depth [1, 2, 9], ensures uniform contribution from all layers, prevents rank collapse of hidden states [4, 5, 7] and the training of very deep network.

In this paper, we demonstrate an additional advantage of depth scaling in MoE pre-training: It provides more balanced expert usage. The idea is based on the recent findings from Wang et al. [8], where the authors notice that the similarity in expert usage arises from similarity in hidden states. Therefore, if we manage to alleviate hidden states rank collapse at initialization, we can also improve router balance. This is crucial for a balanced expert usage throughout training as the gradient magnitude for experts' weights and their corresponding rows in the router depends on the number of tokens routed to each expert.

Our empirical results verify this at initialization and on a toy MoE pre-training setting. Additionally, we observe that using Muon as the optimizer further improves the balance, as its orthogonalized updates stabilize the routers under imbalanced routing..

1.1. Mixture of experts

For a decoder only LLM with L layers, let $\boldsymbol{X}^\ell = \{\boldsymbol{x}_t^\ell\}_{t=1}^T \in \mathbb{R}^{T\times D}$ denote the input to layer ℓ, where T is the sequence length and D is the residual-stream dimension. A MoE transformer block first applies self-attention and then an MoE sublayer, each wrapped with layer normalization and a residual connection:

$$\boldsymbol{H}^\ell = \boldsymbol{X}^\ell + \text{SelfAttention}(\text{LN}(\boldsymbol{X}^\ell)), \qquad \boldsymbol{Y}^\ell = \boldsymbol{H}^\ell + \text{MoE}(\text{LN}(\boldsymbol{H}^\ell)), \tag{1}$$

where we assume a pre-norm formulation, $\text{LN}(\cdot)$ denotes layer norm, $\boldsymbol{H}^\ell$ is the intermediate hidden representation after attention and $\boldsymbol{Y}^\ell$ is the output of layer ℓ.

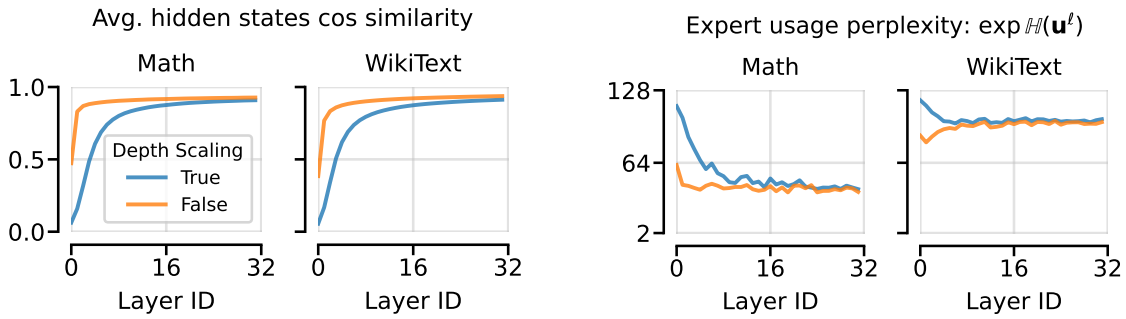


Figure 1: At initialization, depth scaling (blue lines, $\alpha = 0.2/\sqrt{L}$) reduces hidden states' increased similarity across tokens (left), providing more balanced expert usage. Additionally, datasets with more diverse contents (WikiText v.s. math corpus from [6]) also enable more balanced expert usage.

Token-choice sparse MoE The MoE module is composed of a router and a set of small FFNs (experts). Concretely, for a token representation $\boldsymbol{h}_t^\ell \in \mathbb{R}^D$, i.e. a row from $\mathrm{LN}(\boldsymbol{H}^\ell)$, the MoE router computes routing logits

$$\boldsymbol{g}_t^\ell = P^\ell \boldsymbol{h}_t^\ell \in \mathbb{R}^E, \tag{2}$$

where $P^\ell \in \mathbb{R}^{E\times D}$ is the router weight matrix and E is the number of experts. Based on these logits, the router selects a small subset $\mathcal{S}_t^\ell \subseteq \{1,\dots,E\}$ of size k, typically the top-k experts with the largest logit value. Let $s_{t,e}^\ell = \sigma(\boldsymbol{g}_t^\ell)_e$ denote the routing weight assigned to expert e, usually obtained by setting $\sigma(\cdot)$ to be a softmax or sigmoid-and-normalize. The MoE output is then computed as

$$\mathrm{MoE}(\boldsymbol{h}_t^\ell) = \Sigma_{e\in\mathcal{S}_t^\ell} s_{t,e}^\ell \,\mathrm{FFN}_e^\ell(\boldsymbol{h}_t^\ell), \tag{3}$$

where $\mathrm{FFN}_e^\ell(\cdot)$ is the feed-forward network corresponding to expert e. Thus, different tokens may be processed by different fixed-size subsets of experts, enabling the model to increase parameter count while keeping per-token computation relatively small.

1.2. Depth scaling prevents representation collapse (a simplified analysis)

We now track the residual update token-wise,

$$\boldsymbol{h}_i^{\ell+1} \;=\; \boldsymbol{h}_i^\ell + \alpha\,\boldsymbol{\Delta}_i^\ell, \tag{4}$$

where i indexes tokens and $\boldsymbol{\Delta}_i^\ell$ is updates from the attention and MoE block output at layer ℓ. The cross-token inner product evolves as

$$\langle \boldsymbol{h}_i^{\ell+1}, \boldsymbol{h}_j^{\ell+1}\rangle \;=\; \langle \boldsymbol{h}_i^\ell, \boldsymbol{h}_j^\ell\rangle + \alpha\big[\langle \boldsymbol{h}_i^\ell, \boldsymbol{\Delta}_j^\ell\rangle + \langle \boldsymbol{\Delta}_i^\ell, \boldsymbol{h}_j^\ell\rangle\big] + \alpha^2\langle \boldsymbol{\Delta}_i^\ell, \boldsymbol{\Delta}_j^\ell\rangle. \tag{5}$$

At random initialization, $\boldsymbol{\Delta}_j^\ell$ is a function of the layer-ℓ weight matrices with zero-mean and independent of $\boldsymbol{h}_i^\ell$ (whose randomness comes from earlier layers' weights). Hence $\mathbb{E}\left[\langle \boldsymbol{h}_i^\ell, \boldsymbol{\Delta}_j^\ell\rangle\right] = 0$, and the cross terms in (5) vanish in expectation. Taking expectation,

$$\mathbb{E}\left[\langle \boldsymbol{h}_i^{\ell+1}, \boldsymbol{h}_j^{\ell+1}\rangle\right] \;=\; \mathbb{E}\left[\langle \boldsymbol{h}_i^\ell, \boldsymbol{h}_j^\ell\rangle\right] + \alpha^2\,\mathbb{E}\left[\langle \boldsymbol{\Delta}_i^\ell, \boldsymbol{\Delta}_j^\ell\rangle\right], \tag{6}$$

$$\mathbb{E}\left[\left\|\boldsymbol{h}_i^{\ell+1}\right\|^2\right] \;=\; \mathbb{E}\left[\left\|\boldsymbol{h}_i^\ell\right\|^2\right] + \alpha^2\,\mathbb{E}\left[\left\|\boldsymbol{\Delta}_i^\ell\right\|^2\right], \tag{7}$$

where off-diagonal recursion accumulates contributions purely at second order in α. For $i \neq j$, $\mathbb{E}\left[\langle \boldsymbol{\Delta}_i^\ell, \boldsymbol{\Delta}_j^\ell \rangle\right]$ need not to be zero. There are two structural sources of cross-token correlation in the block output:

- *Attention value sharing.* Self-attention writes $\boldsymbol{\Delta}_i^\ell$ as a weighted sum over the same set of value vectors $\{\boldsymbol{V}\boldsymbol{h}_k^\ell\}_k$ as $\boldsymbol{\Delta}_j^\ell$. Both updates are convex combinations of a common pool, so $\langle \boldsymbol{\Delta}_i^\ell, \boldsymbol{\Delta}_j^\ell \rangle$ inherits a contribution from every value vector both tokens attend to.
- *Shared projection weights.* If $\boldsymbol{\Delta}_i = \boldsymbol{W}\phi(\boldsymbol{h}_i)$ and $\boldsymbol{\Delta}_j = \boldsymbol{W}\phi(\boldsymbol{h}_j)$ with $\boldsymbol{W}$ shared across tokens (true for the value, output, and MLP projections), then

$$\mathop{\mathbb{E}}_{\boldsymbol{W}}\left[\langle \boldsymbol{\Delta}_i, \boldsymbol{\Delta}_j \rangle\right] \;=\; \phi(\boldsymbol{h}_i)^\top \mathbb{E}\left[\boldsymbol{W}^\top \boldsymbol{W}\right] \phi(\boldsymbol{h}_j),$$

  which is non-zero whenever $\phi(\boldsymbol{h}_i)$ and $\phi(\boldsymbol{h}_j)$ have non-trivial inner product.

Solving the recursion. By symmetry over tokens, write $D^\ell := \mathbb{E}\left[\left\|\boldsymbol{h}_i^\ell\right\|^2\right]$ and $C^\ell := \mathbb{E}\left[\langle \boldsymbol{h}_i^\ell, \boldsymbol{h}_j^\ell \rangle\right]$ for any $i \neq j$. The cross-token cosine similarity at depth L is

$$\rho^L \;:=\; \frac{\mathbb{E}\left[\langle \boldsymbol{h}_i^L, \boldsymbol{h}_j^L \rangle\right]}{\mathbb{E}\left[\left\|\boldsymbol{h}_i^L\right\|^2\right]} = \frac{C^L}{D^L} \tag{8}$$

Define the per-layer *correlation injection rate* as $\kappa^\ell \;:=\; \frac{\mathbb{E}\left[\langle \boldsymbol{\Delta}_i^\ell, \boldsymbol{\Delta}_j^\ell \rangle\right]}{\mathbb{E}\left[\left\|\boldsymbol{\Delta}_i^\ell\right\|^2\right]}, \kappa^\ell \in [0, 1]$, which captures how much of the block's update lies along directions shared across tokens. Under the structural sources above, κ^ℓ is bounded away from zero at random initialization. Note that κ^ℓ is not constant with depth; it has a self-amplification effect, which can be modeled as

$$\kappa^\ell \;=\; \kappa_0 + \beta\, \rho^\ell, \tag{9}$$

where $\kappa_0 \geq 0$ is the *exogenous* correlation injection (independent of the current ρ^ℓ, controlled by attention pattern entropy and FFN nonlinearity bias) and $\beta \geq 0$ is the *self-amplification* coefficient.

Assume for simplicity that $\mathbb{E}\left[\left\|\boldsymbol{\Delta}_i^\ell\right\|^2\right] = \sigma^2$ (true under pre-LN) Then (6) and (7) solve to

$$C^{\ell+1} \;=\; C^\ell + \alpha^2(\kappa_0 + \beta\rho^\ell)\,\sigma^2, \tag{10}$$

$$D^{\ell+1} \;=\; D^\ell + \alpha^2\sigma^2. \tag{11}$$

Dividing (10) by $D^{\ell+1}$ and using $C^\ell = \rho^\ell D^\ell = \rho^\ell(D^{\ell+1} - \alpha^2\sigma^2)$,

$$\rho^{\ell+1} \;=\; \frac{C^\ell + \alpha^2(\kappa_0 + \beta\rho^\ell)\sigma^2}{D^\ell + \alpha^2\sigma^2} \;=\; \rho^\ell + \frac{\alpha^2\sigma^2}{D^{\ell+1}}\left[\kappa_0 + (\beta - 1)\rho^\ell\right].$$

For large enough ℓ that $D^\ell \approx \ell\alpha^2\sigma^2$ (Eq. (11) starting from $D^0 = \sigma^2 = d$), this is well-approximated by the continuous-depth ODE

$$\frac{d\rho}{d\ell} \;=\; \frac{\alpha^2}{1 + \ell\alpha^2}\Big[\kappa_0 - (1 - \beta)\rho\Big], \tag{12}$$

whose solution with $\rho^0 = 0$ is

$$\boxed{\rho^\ell \;=\; \frac{\kappa_0}{1 - \beta}\left[1 - (1 + \ell\alpha^2)^{-(1-\beta)}\right]} \qquad (\beta \neq 1). \tag{13}$$

Vanilla pre-norm ($\alpha = 1$). The factor $1 + \ell\alpha^2 = \ell + 1$ grows linearly. In the subcritical regime ($\beta < 1$), $\rho^L \to \kappa_0/(1-\beta)$ as $L \to \infty$; in the critical regime (($\beta = 1$)) it grows as $\log L$; in the supercritical regime ($\beta > 1$) it saturates at $\rho = 1$ within $O(1)$ layers. The $\beta > 1$ behavior matches the doubly-exponential rank collapse documented empirically in attention-only stacks [3].

Depth-scaled pre-norm ($\alpha^2 = 1/L$). The factor $1 + \ell\alpha^2$ ranges from 1 to 2 across the network. Substituting into (13), the cosine at the top layer is

$$\rho^L \;=\; \frac{\kappa_0}{1-\beta}\big[1 - 2^{-(1-\beta)}\big], \tag{14}$$

independent of L, in all three regimes including supercritical. Depth scaling does not just slow the supercritical blowup—it caps it at a constant determined only by κ_0 and β.

2. Depth scaling improves expert balance at initialization

At initialization, the router performs a random linear projection, with weights independent of the hidden states; Therefore the similarity in hidden states space translates to router logits and the set of activated experts. Formalizing this intuition, we can show that

Proposition 1 (Balanced routing under bounded concentration (Proof in App. A)) *Given hidden states $\boldsymbol{h}_1, \ldots, \boldsymbol{h}_N \in \mathbb{R}^d$ entering a MoE router, define the normalized representations $\tilde{\boldsymbol{h}}_i = \boldsymbol{h}_i / \|\boldsymbol{h}_i\|_2$ and $M := \frac{1}{N}\sum_{i=1}^N \tilde{\boldsymbol{h}}_i \tilde{\boldsymbol{h}}_i^\top$. Let the router matrix $P \in \mathbb{R}^{E\times d}$ have i.i.d. Gaussian entries with variance $1/d$, independent of the hidden states. Let u_e denote the empirical top-k usage frequency of expert e over the N tokens. Then $\mathbb{E}_P[u_e] = \frac{1}{E}$ for every expert e, and*

$$\mathop{\mathbb{E}}_{P}\left[\left(u_e - \tfrac{1}{E}\right)^2\right] \;\le\; \frac{k(E-k)}{k^2E^2}\left(\frac{1}{N} + \sqrt{\|M\|_{op}}\right). \tag{15}$$

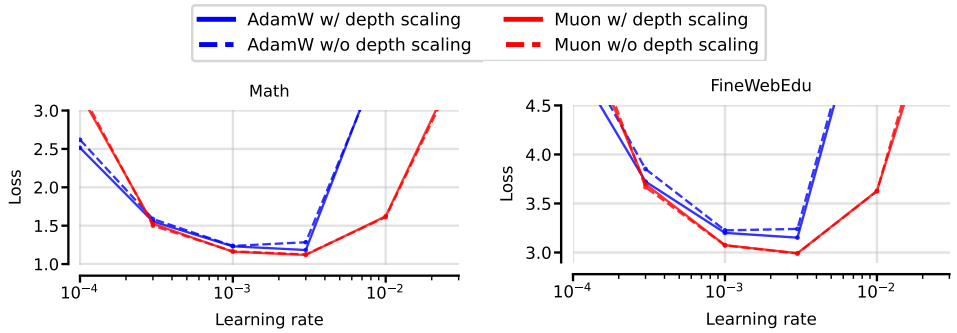
Here $\|M\|_{op}$ satisfies $\frac{1}{\min\{N,d\}} \le \|M\|_{op} \le 1$, with larger values corresponding to more concentrated or correlated hidden states.

Therefore, any mechanisms boosting hidden state diversity (reducing $\|M\|_{\text{op}}$) can improve the router balance at initialization, and setting depth scaling $\alpha \propto \mathcal{O}(1/\sqrt{L})$ will also do so. Of course, other ingredients boosting the diversity can also help, from the data perspective such as batch size and more diverse data mixture, or from the model side (values of κ_0 and β), determined by, e.g. the dimension of the hidden states, and initialization of attention weights.

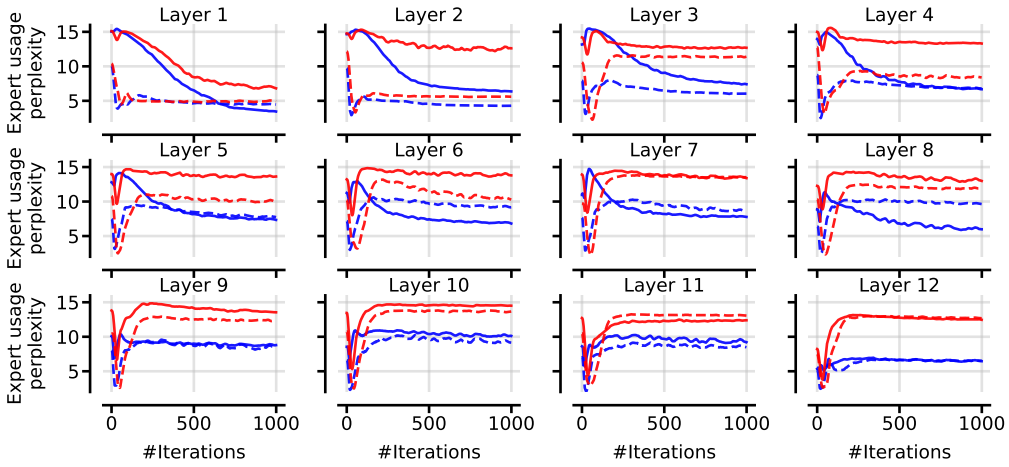
Empirical verification Fig. 1 checked hidden states correlation and expert usage for a 40-layer MoE ($L = 40$) at random initialization, under $\alpha = 1$ v.s. $\alpha = 0.2/\sqrt{L}$. We consider sequences from two pre-training datasets: Math-only one and a more generic WikiText. Broadly, we confirmed that scaling the residual stream with depth alleviates the increase in hidden-state correlation, providing more balanced expert usage. Additionally, datasets with more diverse topics naturally give more balanced expert usage despite having similar hidden states similarity.

3. Pre-training experiments

Now we consider pre-training a 12-layer, 500M MoE. We adopt an extreme setting, where we do *not* utilize load balancing mechanisms (no auxiliary loss or learnable bias) to better understand the



(a) Training loss vs. learning rate on Nemotron-CC-*Math*-v1 [6] and *FineWebEdu*, using a batch size of 0.7M tokens over 2000 iterations ($\sim$ 3 TPP), with 100 linear-warmup steps and linear decay to zero for leanring rate scheduling. Results are based on a single seed.



(b) Per-layer router balance during Nemotron Math pre-training at a learning rate of 10^{-3}, measured as the perplexity of the expert-frequency vector (range $(2, 16)$, where higher values indicate better balance).

Figure 2: Pre-training of a 12-layer, 500M parameter MoE (2 experts activated out of 16 experts) without using load-balancing mechanisms. Near initialization, depth scaling improves router balance in the earlier layers (consistent with Fig. 1) and improves AdamW's performance. Throughout training, Muon consistently achieves better balance and lower loss than AdamW, with or without depth scaling, likely due to its whitening operation balancing out imbalanced gradient signals.

interactions between hidden states dynamics and expert balance. We consider both AdamW and Muon, sweeping the learning rate on a grid with multiplicative factor 3, with and without depth scaling.

The results are presented in Fig. 2. Broadly, we found that depth scaling (solid lines) improves the loss as well as load balancing in earlier layers and the early phase of the training. However, as training proceeds, balance gets worse under AdamW as there are no load balancing mechanisms, whereas Muon is capable of maintaining balance throughout training, and even recovers from imbalanced routing at initialization (App. B.1 gives a mechanistic explanation.). Lastly, the performance gain from depth scaling is negligible under Muon; we hypothesize that the performance improvement would emerge as depth increases further.

References

[1] Blake Bordelon, Lorenzo Noci, Mufan Bill Li, Boris Hanin, and Cengiz Pehlevan. Depthwise hyperparameter transfer in residual networks: Dynamics and scaling limit. *arXiv preprint arXiv:2309.16620*, 2023.

[2] Nolan Dey, Bin Claire Zhang, Lorenzo Noci, Mufan Li, Blake Bordelon, Shane Bergsma, Cengiz Pehlevan, Boris Hanin, and Joel Hestness. Don't be lazy: Completep enables compute-efficient deep transformers. *arXiv preprint arXiv:2505.01618*, 2025.

[3] Yihe Dong, Jean-Baptiste Cordonnier, and Andreas Loukas. Attention is not all you need: Pure attention loses rank doubly exponentially with depth. In *International conference on machine learning*, pages 2793–2803. PMLR, 2021.

[4] Soufiane Hayou, Eugenio Clerico, Bobby He, George Deligiannidis, Arnaud Doucet, and Judith Rousseau. Stable resnet. In Arindam Banerjee and Kenji Fukumizu, editors, *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 1324–1332. PMLR, 13–15 Apr 2021. URL https://proceedings.mlr.press/v130/hayou21a.html.

[5] Lorenzo Noci, Sotiris Anagnostidis, Luca Biggio, Antonio Orvieto, Sidak Pal Singh, and Aurelien Lucchi. Signal propagation in transformers: Theoretical perspectives and the role of rank collapse, 2022. URL https://arxiv.org/abs/2206.03126.

[6] Shrimai Prabhumoye Mostofa Patwary Mohammad Shoeybi Bryan Catanzaro Rabeeh Karimi Mahabadi, Sanjeev Satheesh. Nemotron-cc-math: A 133 billion-token-scale high quality math pretraining dataset. 2025. URL https://arxiv.org/abs/2508.15096.

[7] Samuel S. Schoenholz, Justin Gilmer, Surya Ganguli, and Jascha Sohl-Dickstein. Deep information propagation, 2017.

[8] Xi Wang, Soufiane Hayou, and Eric Nalisnick. The myth of expert specialization in moes: Why routing reflects geometry, not necessarily domain expertise. *arXiv preprint arXiv:2604.09780*, 2026.

[9] Greg Yang, Dingli Yu, Chen Zhu, and Soufiane Hayou. Tensor programs VI: Feature learning in infinite depth neural networks. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=17pVDnpwwl.

Appendix A. Proof of Prop. 1

We prove the two parts in turn. Throughout, let $\boldsymbol{z}_i := P\boldsymbol{h}_i \in \mathbb{R}^E$ denote the routing logit vector for token i, write $\sigma_i^2 := \|\boldsymbol{h}_i\|_2^2 / d$ for the marginal variance of each coordinate of $\boldsymbol{z}_i$, set $X_i := \mathbf{1}[e \in T_i]$ for the indicator that expert e is selected for token i, and define $\sigma^2 := \mathrm{Var}(X_i) = (k/E)(1-k/E) = k(E-k)/E^2$.

Part 1: $\mathbb{E}[u_e] = 1/E$. The rows $\boldsymbol{r}_1, \ldots, \boldsymbol{r}_E$ of P are i.i.d. $\mathcal{N}(0, I_d/d)$, so for fixed $\boldsymbol{h}_i$, the coordinates $z_{i,e} = \langle \boldsymbol{r}_e, \boldsymbol{h}_i \rangle$ are i.i.d. $\mathcal{N}(0, \sigma_i^2)$. By exchangeability, the rank of any particular $z_{i,e}$ among $z_{i,1}, \ldots, z_{i,E}$ is uniformly distributed on $\{1, \ldots, E\}$, so $T_i = \mathrm{TopK}(\boldsymbol{z}_i)$ is uniform over the $\binom{E}{k}$ size-k subsets of $[E]$. Hence $\Pr[e \in T_i] = k/E$, and

$$\mathbb{E}[u_e] \;=\; \frac{1}{Nk} \sum_{i=1}^{N} \Pr[e \in T_i] \;=\; \frac{1}{Nk} \cdot N \cdot \frac{k}{E} \;=\; \frac{1}{E}.$$

Part 2: Variance decomposition. Expanding $\mathrm{Var}(u_e) = (Nk)^{-2} \mathrm{Var}\big(\sum_i X_i\big)$,

$$\mathrm{Var}(u_e) \;=\; \frac{1}{N^2k^2} \Big[N\sigma^2 \;+\; \sum_{i \neq j} \mathrm{Cov}(X_i, X_j) \Big]. \tag{16}$$

The first term is the diagonal contribution $\sigma^2/(Nk^2)$. We bound the off-diagonal sum by reducing it to cross-token cosine similarities $\rho_{ij} := \langle \tilde{\boldsymbol{h}}_i, \tilde{\boldsymbol{h}}_j \rangle$.

Lemma 2 (Off-diagonal covariance bound) *For any pair of tokens $i \neq j$,*

$$|\mathrm{Cov}(X_i, X_j)| \;\leq\; \sigma^2 \cdot |\rho_{ij}|.$$

Proof The pair $(\boldsymbol{z}_i, \boldsymbol{z}_j) \in \mathbb{R}^{2E}$ is jointly Gaussian. Different rows of P are independent, so coordinates across distinct experts $e \neq e'$ are independent: $\mathrm{Cov}(z_{i,e}, z_{j,e'}) = 0$ for $e \neq e'$. Within a single coordinate,

$$\mathrm{Cov}(z_{i,e}, z_{j,e}) \;=\; \mathbb{E}[\boldsymbol{r}_e^\top \boldsymbol{h}_i \, \boldsymbol{h}_j^\top \boldsymbol{r}_e] \;=\; \frac{\langle \boldsymbol{h}_i, \boldsymbol{h}_j \rangle}{d},$$

giving correlation ρ_{ij} after rescaling. Define the marginally normalized logits $\tilde{\boldsymbol{z}}_i := \boldsymbol{z}_i/\sigma_i$ and $\tilde{\boldsymbol{z}}_j := \boldsymbol{z}_j/\sigma_j$. Each is standard Gaussian in $\mathbb{R}^E$, with $\mathrm{Cov}(\tilde{z}_{i,e}, \tilde{z}_{j,e'}) = \rho_{ij}\, \delta_{ee'}$. Since the events $\{e \in T_i\}$ depend only on rank statistics of $\boldsymbol{z}_i$, which are invariant under positive scaling, we have $X_i = g(\tilde{\boldsymbol{z}}_i)$ and $X_j = g(\tilde{\boldsymbol{z}}_j)$ for the common indicator function $g(\boldsymbol{z}) := \mathbf{1}[e \in \mathrm{TopK}(\boldsymbol{z})]$.

Write $\rho := \rho_{ij}$. Express $\tilde{\boldsymbol{z}}_j$ as the noisy version of $\tilde{\boldsymbol{z}}_i$ at correlation ρ:

$$\tilde{z}_{j,e} \;=\; \rho\, \tilde{z}_{i,e} + \sqrt{1-\rho^2}\, \xi_e, \qquad \xi_e \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1) \text{ independent of } \tilde{\boldsymbol{z}}_i.$$

Then $\mathbb{E}[X_j \mid \tilde{\boldsymbol{z}}_i] = (\mathcal{P}_\rho g)(\tilde{\boldsymbol{z}}_i)$, where $\mathcal{P}_\rho$ is the Ornstein–Uhlenbeck noise operator on functions of standard Gaussian vectors in $\mathbb{R}^E$. By Mehler's formula,

$$\mathcal{P}_\rho g \;=\; \sum_{n \geq 0} \rho^n \, \pi_n g,$$

where π_n is the orthogonal projection onto the n-th Hermite chaos in $L^2(\gamma_E)$, with γ_E the standard Gaussian measure on $\mathbb{R}^E$. Therefore

$$\mathrm{Cov}(X_i, X_j) \;=\; \mathbb{E}[g(\tilde{\boldsymbol{z}}_i)\,\mathcal{P}_\rho g(\tilde{\boldsymbol{z}}_i)] - \mathbb{E}[g]^2 \;=\; \sum_{n\geq 1} \rho^n\,\alpha_n,$$

where $\alpha_n := \|\pi_n g\|^2_{L^2(\gamma_E)} \geq 0$. The constant term ($n = 0$) cancels with $\mathbb{E}[g]^2$, and Parseval gives $\sum_{n\geq 1} \alpha_n = \mathrm{Var}(g) = \sigma^2$. For $|\rho| \leq 1$,

$$|\mathrm{Cov}(X_i, X_j)| \;=\; \Big|\sum_{n\geq 1} \rho^n\,\alpha_n\Big| \;\leq\; \sum_{n\geq 1} |\rho|^n\,\alpha_n \;\leq\; |\rho| \sum_{n\geq 1} \alpha_n \;=\; |\rho|\,\sigma^2,$$

using $|\rho|^n \leq |\rho|$ for $n \geq 1$ and $|\rho| \leq 1$. ■

Aggregation via Cauchy–Schwarz. Apply Lemma 2 and bound the sum of $|\rho_{ij}|$ by its ℓ^2 counterpart:

$$\sum_{i\neq j} |\mathrm{Cov}(X_i, X_j)| \;\leq\; \sigma^2 \sum_{i\neq j} |\rho_{ij}| \;\leq\; \sigma^2 \sqrt{N(N-1)} \sqrt{\sum_{i\neq j} \rho_{ij}^2} \;\leq\; \sigma^2\, N \sqrt{\sum_{i,j} \rho_{ij}^2}.$$

The remaining quantity is the squared Frobenius norm of the cosine Gram matrix. Let $\tilde{\boldsymbol{H}} \in \mathbb{R}^{N\times d}$ have rows $\tilde{\boldsymbol{h}}_i^\top$. The Gram matrix $G := \tilde{\boldsymbol{H}}\tilde{\boldsymbol{H}}^\top$ has $G_{ij} = \rho_{ij}$ and the same nonzero eigenvalues as $\tilde{\boldsymbol{H}}^\top\tilde{\boldsymbol{H}} = NM$. Hence

$$\sum_{i,j} \rho_{ij}^2 \;=\; \|G\|_F^2 \;=\; \mathrm{tr}\big((NM)^2\big) \;=\; N^2\,\|M\|_F^2\,, \tag{17}$$

giving $\sum_{i\neq j} |\mathrm{Cov}(X_i, X_j)| \leq \sigma^2 N^2\,\|M\|_F$.

Combining. Substitute back into (16):

$$\mathrm{Var}(u_e) \;\leq\; \frac{1}{N^2k^2}\big[N\sigma^2 + \sigma^2 N^2\,\|M\|_F\big] \;=\; \frac{\sigma^2}{k^2}\left(\frac{1}{N} + \|M\|_F\right).$$

Since $\|M\|_F^2 \leq \|M\|_{\mathrm{op}} \cdot \mathrm{tr}(M) = \tau \cdot 1 = \tau$, so $\|M\|_F \leq \sqrt{\tau}$. Substituting $\sigma^2 = k(E-k)/E^2$ yields the bound (15). □

Appendix B. Muon balances router updates

Muon is a first-order optimizer for matrix-valued parameters. For a matrix parameter $\boldsymbol{W} \in \mathbb{R}^{m\times n}$ with gradient $\boldsymbol{G}_t$ at step t, Muon maintains a momentum buffer $\boldsymbol{M}_t$ and applies the update

$$\boldsymbol{M}_t = \mu\,\boldsymbol{M}_{t-1} + \boldsymbol{G}_t, \tag{18}$$
$$\boldsymbol{O}_t = \mathrm{Ortho}(\boldsymbol{M}_t), \tag{19}$$
$$\boldsymbol{W}_{t+1} = \boldsymbol{W}_t - \eta\,\boldsymbol{O}_t, \tag{20}$$

where $\mu \in [0, 1)$ is the momentum coefficient and $\mathrm{Ortho}(\boldsymbol{M})$ returns the orthogonal factor of $\boldsymbol{M}$: if $\boldsymbol{M} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$ is the singular-value decomposition, then $\mathrm{Ortho}(\boldsymbol{M}) = \boldsymbol{U}\boldsymbol{V}^\top$. In practice this is approximated by a short Newton–Schulz iteration using only matrix multiplications.

The defining property of Muon is *spectral-norm normalization*: for any non-zero $\boldsymbol{M}$, the non-zero singular values of $\mathrm{Ortho}(\boldsymbol{M})$ are equal to 1. Consequently,

$$\|\eta \boldsymbol{O}_t\|_{\mathrm{op}} = \eta, \tag{21}$$

independent of the scale or spectrum of $\boldsymbol{G}_t$. Equivalently, for any vector $\boldsymbol{u} \in \mathbb{R}^n$,

$$\|\eta \boldsymbol{O}_t \boldsymbol{u}\|_2 \;\le\; \eta \, \|\boldsymbol{u}\|_2 \,, \tag{22}$$

with equality when $\boldsymbol{u}$ lies in the row space of $\boldsymbol{M}_t$.

B.1. Router stability under Muon

Applied to the router $P \in \mathbb{R}^{E \times d}$, Muon bounds the per-step change of every routing logit uniformly across experts and tokens. Let $\boldsymbol{h} \in \mathbb{R}^d$ be the hidden state of a token and let

$$\boldsymbol{z}_t(\boldsymbol{h}) = P_t \boldsymbol{h} \tag{23}$$

denote the vector of router logits. Under the Muon update,

$$P_{t+1} = P_t - \eta \boldsymbol{O}_t, \tag{24}$$

the change in router logits is

$$\Delta \boldsymbol{z}_t(\boldsymbol{h}) := \boldsymbol{z}_{t+1}(\boldsymbol{h}) - \boldsymbol{z}_t(\boldsymbol{h}) = -\eta \boldsymbol{O}_t \boldsymbol{h}. \tag{25}$$

Therefore, by the spectral-norm bound in Eq. (22),

$$\|\Delta \boldsymbol{z}_t(\boldsymbol{h})\|_2 \le \eta \, \|\boldsymbol{h}\|_2 \,. \tag{26}$$

In particular, since each coordinate is bounded by the vector norm, for every expert $e \in [E]$,

$$|\Delta z_{t,e}(\boldsymbol{h})| \le \eta \, \|\boldsymbol{h}\|_2 \,. \tag{27}$$

Thus, no expert logit can change by more than $\eta \, \|\boldsymbol{h}\|_2$ in a single Muon step, regardless of the raw magnitude or anisotropy of the router gradient.

This contrasts with an unconstrained gradient update. If the router is updated by ordinary gradient descent,

$$P_{t+1} = P_t - \eta \boldsymbol{G}_t, \tag{28}$$

then the corresponding logit change satisfies only

$$\|\Delta \boldsymbol{z}_t(\boldsymbol{h})\|_2 = \eta \, \|\boldsymbol{G}_t \boldsymbol{h}\|_2 \le \eta \, \|\boldsymbol{G}_t\|_{\mathrm{op}} \, \|\boldsymbol{h}\|_2 \,, \tag{29}$$

which depends directly on the operator norm of the raw gradient. Large or highly anisotropic router gradients can therefore cause disproportionately large changes in routing logits. Muon removes this dependence by normalizing the update in spectral norm.

Balancing expert-row updates. The router matrix has one row per expert. When the number of experts is no larger than the hidden dimension, $E \leq d$, the orthogonal factor satisfies

$$\boldsymbol{O}_t \boldsymbol{O}_t^\top = \boldsymbol{I}_E \tag{30}$$

whenever $\boldsymbol{M}_t$ has full row rank. Hence the rows of the Muon update are orthonormal:

$$\|(\boldsymbol{O}_t)_{e,:}\|_2 = 1, \qquad \langle (\boldsymbol{O}_t)_{e,:}, (\boldsymbol{O}_t)_{e',:} \rangle = 0 \quad \text{for } e \neq e'. \tag{31}$$

Consequently, each expert row receives an update with the same norm,

$$\|\Delta P_{t,e,:}\|_2 = \eta, \qquad e \in [E], \tag{32}$$

and different expert rows are updated in orthogonal directions.

This gives Muon a natural balancing effect on router learning. It does not directly enforce balanced expert usage, i.e.,

$$u_e \approx \frac{1}{E}, \tag{33}$$

nor does it constrain the softmax probabilities or top-k assignments. Instead, it balances the *geometry of the router update*: expert rows receive comparable-norm updates, and their update directions are decorrelated. Thus Muon can stabilize router training by preventing a few high-gradient expert rows from dominating the update, while still allowing the routing distribution to be determined by the learned logits.

Interpretation. For an MoE router with $E \leq d$, Muon should therefore be viewed not as a row-normalization method for the router weights, but as a row-orthogonalization method for the router updates. At each step, the update has bounded spectral norm and, under the usual full-row-rank condition, equal row norms across experts. This provides a simple mechanism by which Muon can balance the per-step learning dynamics of different experts without imposing an explicit load-balancing constraint.

Why AdamW does not provide the same effect. AdamW applies an elementwise adaptive rescaling of the gradient, followed by decoupled weight decay. For a router matrix, its update has the form

$$P_{t+1} = (1 - \eta\lambda) P_t - \eta \frac{\boldsymbol{m}_t}{\sqrt{\boldsymbol{v}_t} + \epsilon}, \tag{34}$$

where $\boldsymbol{m}_t$ and $\boldsymbol{v}_t$ are the first- and second-moment estimates, and the division is applied coordinate-wise. Thus the AdamW update matrix

$$\boldsymbol{A}_t := \frac{\boldsymbol{m}_t}{\sqrt{\boldsymbol{v}_t} + \epsilon} \tag{35}$$

is not constrained in spectral norm, rank, or row geometry. In particular, AdamW does not ensure

$$\|\boldsymbol{A}_t\|_{\text{op}} = 1, \qquad \boldsymbol{A}_t \boldsymbol{A}_t^\top = \boldsymbol{I}_E, \qquad \text{or} \qquad \|(\boldsymbol{A}_t)_{e,:}\|_2 = \left\|(\boldsymbol{A}_t)_{e',:}\right\|_2 . \tag{36}$$

Consequently, the per-step logit change under AdamW satisfies only

$$\|\Delta \boldsymbol{z}_t(\boldsymbol{h})\|_2 = \eta \, \|\boldsymbol{A}_t \boldsymbol{h}\|_2 \leq \eta \, \|\boldsymbol{A}_t\|_{\text{op}} \, \|\boldsymbol{h}\|_2 , \tag{37}$$

where $\|\boldsymbol{A}_t\|_{\mathrm{op}}$ can vary across steps and can be large if the adaptive update is anisotropic. Moreover, because the normalization is coordinatewise rather than matrixwise, two expert rows can still receive highly correlated update directions or substantially different update norms. Therefore AdamW can equalize the scale of individual coordinates, but it does not orthogonalize expert-row updates, does not flatten the singular values of the router update matrix, and does not provide the row-balancing geometry induced by Muon.