BMUON: BUDGETING MUON'S SPECTRAL GEOMETRY FOR STRUCTURED COMPRESSIBILITY

Anonymous authors
Paper under double-blind review

ABSTRACT

Muon has emerged as a strong optimizer for neural network training and admits an interpretation as steepest descent under a spectral-norm geometry. Yet we find that Muon-trained Transformers can degrade sharply when their MLP hidden channels are pruned after training. To reconcile Muon's optimization performance with structured compressibility, we introduce Budget Muon (BMuon), which modifies Muon's spectral geometry so that hidden channels across blocks compete for a shared allocation budget. The resulting oracle is exactly normalized steepest descent under a family of norms that interpolate between Muon's geometry and structured sparse geometries. Our principal sparsity control is an interpretable global channel-allocation budget, rather than a setting-dependent sparsity-penalty coefficient. For linear factorization models, we show that Muon admits minimum-cost exact representations with redundant hidden width, whereas under the BMuon oracle geometry and a sufficiently scarce budget, every minimum-cost representation is width-minimal. For practical training, we derive a minorization-maximization surrogate for the exact allocation problem. On ImageNet-1K with ViT-S/16, BMuon improves over Muon under gradual structured MLP pruning, and its learned allocations provide a more effective pruning criterion than weight magnitude. On a 1B-parameter LLaMA-style model trained on FineWeb, BMuon improves over Muon under post-training MLP hidden-channel pruning.

1 INTRODUCTION

Modern Transformer models incur substantial memory and computational costs at inference time, motivating compression for more efficient deployment (Kwon et al., 2022). Pruning is a natural approach to this problem, but practical gains depend on the sparsity structure: unstructured sparsity often requires specialized kernels or hardware, whereas structured pruning removes entire computational units and translates directly into efficiency gains (Wen et al., 2016; An et al., 2024). Feed-forward modules can account for roughly two thirds of Transformer parameters (Geva et al., 2021), making their hidden channels a natural target for structured compression. Global pruning ranks channels across layers jointly, allowing different layers to retain different numbers of channels (Lee et al., 2021; An et al., 2024). Post-pruning fine-tuning adds computational cost, especially for large models (Kwon et al., 2022; Sun et al., 2024), motivating compressibility to emerge during training so that pruning requires little or no recovery.

Muon has demonstrated strong training efficiency across model scales, from small vision and language-model benchmarks to billion-parameter language models (Jordan et al., 2024; Liu et al., 2025; Shah et al., 2025). Its matrix update can be viewed as a linear minimization oracle over a spectral-norm ball, giving Muon a steepest-descent interpretation in spectral-norm geometry (Pethick et al., 2025). Prior work has shown that implicit bias can depend on optimization geometry (Gunasekar et al., 2018). For Muon, recent theory identifies a spectral-norm margin bias in linear and smooth homogeneous models (Fan et al., 2025; Gronich & Vardi, 2026). Yet we find that Muon-trained Transformers degrade rapidly under structured MLP pruning, suggesting that Muon's spectral geometry does not by itself encourage structured compressibility. As shown in Figure 1, a Muon-trained ViT-S/16 on ImageNet-1K collapses after only 30% global MLP channel pruning, with Top-1 accuracy falling from 75.23% to 3.17%. In contrast, our proposed *Budget Muon* (BMuon) algorithm retains 74.08% Top-1 accuracy after 70% pruning, nearly matching its 74.17%

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.

accuracy before pruning, with no post-pruning recovery. This raises the central question: can Muon's spectral geometry be modified to encourage structured compressibility?

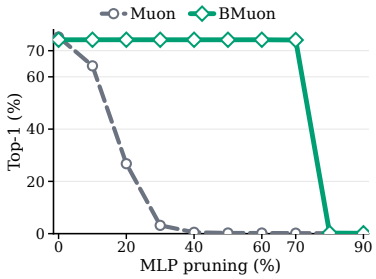


Figure 1: Post-training global MLP pruning on ImageNet-1K with ViT-S/16. Mean ± std over 3 seeds. See Appendix E.1.

To address this question, we introduce BMuon, which reweights Muon's spectral geometry under a single global budget K shared across all hidden channels of all MLP blocks. For hidden channel j in block ℓ, BMuon assigns an allocation $\mu_j^\ell \in [0, 1]$ to the corresponding row of the up-projection matrix and column of the down-projection matrix, with $\sum_{\ell,j} \mu_j^\ell = K$. For a fixed allocation, BMuon introduces a diagonal reweighting inside Muon's spectral-norm constraint, with the components of the update variable corresponding to channel j in block ℓ weighted according to its allocation μ_j^ℓ. BMuon chooses the allocation globally, intuitively asking where the update budget should be spent to obtain the greatest first-order decrease in the loss.

NuMuon (Dolatabadi et al., 2026) is the closest prior compression-aware variant of Muon. It imposes a nuclear-norm constraint on Muon's update direction to improve low-rank compressibility. BMuon instead targets structured compressibility over hidden channels by introducing a global channel budget into Muon's spectral-norm geometry. Prior work has also incorporated compressibility into training. Lu et al. (2022) train networks for unstructured pruning with Frank–Wolfe under fixed layerwise constraints on the weights, while Zimmer et al. (2025) use Frank–Wolfe under fixed norm constraints for convolutional filter pruning and low-rank compression. BMuon instead leaves the weights unconstrained and modifies Muon's spectral update geometry for global hidden-channel compression. Resource-constrained structure learning and global structured pruning have been studied across CNNs and Transformers (Gordon et al., 2018; Yu et al., 2022; Yang et al., 2023; An et al., 2024; Wang et al., 2026a). GRASPrune (Wang et al., 2026b) learns gates post hoc on a frozen LLM to select MLP channels and attention KV-head groups under a single global parameter budget, while DABP (Laroui et al., 2026) jointly trains CNN weights and continuous channel gates under a global resource budget. BMuon uses no forward gates; its budget instead controls the channel-wise reweighting of Muon's spectral update constraint during training.

Theoretically, we show that the BMuon oracle is exactly normalized steepest descent under an induced norm $\Omega_{\alpha,K}$, where α controls the strength of channel reweighting and K is the global budget over the N hidden channels. This family recovers Muon when $\alpha = 0$ or $K = N$. At $\alpha = 1$, its unit ball is the convex hull of operator-norm-bounded updates supported on at most K coupled hidden channels. This is related to group k-support norms through the use of structured support (Argyriou et al., 2012; Rao et al., 2017), but its atoms are bounded in operator norm rather than Euclidean norm. It is also related to spectral k-support norms (McDonald et al., 2014; 2016), but differs in the notion of sparsity: spectral k-support uses sparsity in singular values, whereas BMuon uses MLP-channel sparsity. At $\alpha = K = 1$, the norm reduces to an ℓ_1 aggregation over coupled channel groups, yielding a group-lasso-like geometry (Yuan & Lin, 2006). We then characterize the representation cost induced by BMuon. Following Dai et al. (2021), this is the minimum parameter-space cost among exact two-matrix factorizations that realize the same linear predictor. We prove that, under an explicit budget-scarcity condition, every minimum-cost BMuon factorization is width-minimal, whereas Muon's minimum cost can also be attained by factorizations with redundant hidden width.

Solving the exact allocation problem at every training step is expensive. We therefore derive a minorization–maximization surrogate that yields a tractable allocation update, followed by the corresponding weighted Muon update. On CIFAR-100 and ImageNet-1K, BMuon outperforms Muon under both gradual and post-training global MLP-channel pruning, even when both use the same magnitude-based pruning criterion. Under gradual pruning, using the learned allocations themselves to rank channels further improves performance, showing that they also provide a useful measure of channel importance. The non-uniform architectures identified by BMuon in the gradual-pruning experiments also outperform matched uniform-width architectures when retrained from scratch, providing evidence that the learned allocation captures useful differences in how MLP capacity should be distributed across layers. Finally, on a 1B-parameter LLaMA-style model trained on FineWeb, BMuon improves robustness to post-training global MLP-channel pruning, showing that the effect extends to language modeling at substantially larger scale.

Contributions. To summarize, our main contributions are:

- We introduce Budget Muon (BMuon), which incorporates a single global hidden-channel budget into Muon's spectral update geometry, while coupling the matrix slices that define each MLP channel.
- We characterize the induced norm family $\Omega_{\alpha,K}$ and its relation to Muon and structured sparse geometries in Section 2. In the linear factorization setting of Section 3, we show that for a sufficiently scarce budget, every minimum-cost BMuon factorization is width-minimal, whereas Muon's minimum cost is also attained by redundant-width factorizations.
- We derive a practical minorization–maximization surrogate for the global allocation problem, yielding a tractable allocation update in Section 4.
- In Section 5, we show across CIFAR-100, ImageNet-1K, and a 1B-parameter LLaMA-style model that BMuon improves structured MLP-channel compressibility relative to Muon. In the vision experiments, the learned allocations also provide effective pruning scores and identify non-uniform MLP architectures that outperform matched uniform-width architectures when retrained from scratch.

2 Budget Muon: Oracle and Geometry

2.1 Setup and Muon Preliminaries

MLP channels and global pruning. We consider L single-hidden-layer MLP blocks, as in Transformer feed-forward modules, with block ℓ given by

$$x \mapsto W_2^\ell \, \phi\big(W_1^\ell x + b^\ell\big), \qquad W_1^\ell \in \mathbb{R}^{m_\ell \times d_\ell}, \quad W_2^\ell \in \mathbb{R}^{p_\ell \times m_\ell}, \tag{1}$$

where m_ℓ is the hidden width. Hidden channel j is associated with its incoming weights $W_{1,j:}^\ell$, bias b_j^ℓ, and outgoing weights $W_{2,:,j}^\ell$. Pruning the channel removes these components jointly. The matrix parameters W_1^ℓ, W_2^ℓ are optimized by Muon, while biases are optimized separately by AdamW. We write $W := \big((W_1^\ell, W_2^\ell)\big)_{\ell=1}^L$ for the collection of MLP weight matrices and $N := \sum_{\ell=1}^L m_\ell$ for the total number of hidden channels across all blocks. Given channel scores s_j^ℓ, global pruning ranks all N channels jointly and removes the lowest-scoring ones until the target global sparsity is reached. The resulting hidden width in each block is therefore determined by this global ranking.

Muon as spectral-norm steepest descent. For an MLP matrix W_i^ℓ, let M_i^ℓ denote the momentum matrix used by Muon, following Appendix A. For any matrix A with compact singular value decomposition (SVD) $A = U\Sigma V^\top$, define its polar factor by $\mathrm{Polar}(A) := UV^\top$. The exact Muon direction is $\Delta_{i,\mathrm{Muon}}^\ell = \mathrm{Polar}(M_i^\ell)$, and W_i^ℓ is updated in the opposite direction. In practice, Muon approximates this direction using a finite Newton–Schulz iteration (Jordan et al., 2024); throughout the theoretical analysis, we use the exact polar factor. By spectral–nuclear norm duality, the exact Muon direction satisfies

$$\Delta_{i,\mathrm{Muon}}^\ell = \mathrm{Polar}(M_i^\ell) \in \operatorname*{arg\,max}_{\|\Delta_i^\ell\|_{\mathrm{op}} \le 1} \langle M_i^\ell, \Delta_i^\ell \rangle, \qquad \max_{\|\Delta_i^\ell\|_{\mathrm{op}} \le 1} \langle M_i^\ell, \Delta_i^\ell \rangle = \|M_i^\ell\|_*.$$

Here $\langle A, B \rangle = \mathrm{tr}(A^\top B)$ is the Frobenius inner product, $\|A\|_{\mathrm{op}} = \sigma_{\max}(A)$ is the operator norm, and $\|A\|_* = \sum_j \sigma_j(A)$ is the nuclear norm. Thus exact-polar Muon is normalized steepest descent under the spectral norm (Bernstein & Newhouse, 2024; Pethick et al., 2025). For all MLP matrices, write $M := \big((M_1^\ell, M_2^\ell)\big)_{\ell=1}^L$ and $\Delta := \big((\Delta_1^\ell, \Delta_2^\ell)\big)_{\ell=1}^L$. We equip this product space with the inner-product $\langle M, \Delta \rangle := \sum_{\ell=1}^L \big(\langle M_1^\ell, \Delta_1^\ell \rangle + \langle M_2^\ell, \Delta_2^\ell \rangle\big)$ and define Muon's product-space unit ball $\mathcal{A}_{\mathrm{Muon}} := \{\Delta : \|\Delta_1^\ell\|_{\mathrm{op}} \le 1,\ \|\Delta_2^\ell\|_{\mathrm{op}} \le 1,\ \forall \ell\}$. Equivalently, $\mathcal{A}_{\mathrm{Muon}} = \{\Delta : \|\Delta\|_{\mathrm{Muon}} \le 1\}$, where $\|\Delta\|_{\mathrm{Muon}} := \max_\ell \max\{\|\Delta_1^\ell\|_{\mathrm{op}}, \|\Delta_2^\ell\|_{\mathrm{op}}\}$. Muon's update direction over all MLP matrices is then

$$\Delta_{\mathrm{Muon}} \in \operatorname*{arg\,max}_{\Delta \in \mathcal{A}_{\mathrm{Muon}}} \langle M, \Delta \rangle. \tag{2}$$

Since the objective and constraints separate across matrices, this is equivalent to applying the single-matrix Muon direction independently to each W_i^ℓ.

2.2 THE BMUON ORACLE

Global channel budget. For each block ℓ of width m_ℓ, let $\mu^\ell \in [0,1]^{m_\ell}$ assign an allocation to each hidden channel, and let $D_{\mu^\ell} := \mathrm{diag}(\mu^\ell)$. Writing $\mu := (\mu^1, \dots, \mu^L) \in [0,1]^N$, for a budget $K \in (0, N]$, define $\mathcal{K}_K := \{\mu \in [0,1]^N : \sum_{\ell,j} \mu_j^\ell = K\}$. All hidden channels share the budget K, so channels across blocks compete directly for allocation.

Channel-weighted Muon directions. Fix $\alpha \in (0,1]$. For a fixed allocation $\mu \in \mathcal{K}_K$, BMuon reweights the spectral constraints in $\mathcal{A}_{\mathrm{Muon}}$ channel-wise, yielding

$$\mathcal{A}_\alpha(\mu) := \Big\{\Delta : \exists \{P_1^\ell, P_2^\ell\}_{\ell=1}^L,\ \Delta_1^\ell = D_{\mu^\ell}^\alpha P_1^\ell,\ \Delta_2^\ell = P_2^\ell D_{\mu^\ell}^\alpha,\ \|P_1^\ell\|_{\mathrm{op}} \le 1,\ \|P_2^\ell\|_{\mathrm{op}} \le 1,\ \forall\, \ell\Big\}.$$

The same factor $(\mu_j^\ell)^\alpha$ scales row j of Δ_1^ℓ and column j of Δ_2^ℓ, coupling the two components of hidden channel j. When every $\mu_j^\ell > 0$, this is equivalently $\|D_{\mu^\ell}^{-\alpha} \Delta_1^\ell\|_{\mathrm{op}} \le 1$ and $\|\Delta_2^\ell D_{\mu^\ell}^{-\alpha}\|_{\mathrm{op}} \le 1$ for all ℓ. Thus BMuon applies Muon's spectral-norm constraint after diagonally reweighting the hidden channels. If $\mu_j^\ell = 0$, the corresponding row of Δ_1^ℓ and column of Δ_2^ℓ are forced to be zero, making the connection to channel-sparse updates explicit.

BMuon oracle. Muon maximizes the alignment $\langle M, \Delta\rangle$ over a fixed constraint set, $\|\Delta\|_{\mathrm{Muon}} \le 1$. In BMuon, the admissible set for a fixed allocation is $\mathcal{A}_\alpha(\mu)$, and the oracle chooses the allocation and direction jointly:

$$\max_{\mu \in \mathcal{K}_K} \max_{\Delta \in \mathcal{A}_\alpha(\mu)} \langle M, \Delta\rangle. \tag{3}$$

The budget is allocated to maximize the attainable alignment with the momentum; when M is the gradient rather than the momentum, the oracle jointly chooses μ and Δ to maximize the first-order decrease in the loss. By spectral–nuclear norm duality, the oracle reduces to the allocation problem

$$\max_{\mu \in \mathcal{K}_K} \Phi_M(\mu), \qquad \Phi_M(\mu) := \sum_{\ell=1}^{L} \left[\|D_{\mu^\ell}^\alpha M_1^\ell\|_* + \|M_2^\ell D_{\mu^\ell}^\alpha\|_*\right], \tag{4}$$

whose maximum is attained since $\mathcal{K}_K$ is compact and Φ_M is continuous. Given a maximizer $\mu^\star$, an optimal direction is, for every ℓ,

$$\Delta_1^{\ell,\star} = D_{\mu^{\ell,\star}}^\alpha \,\mathrm{Polar}\Big(D_{\mu^{\ell,\star}}^\alpha M_1^\ell\Big), \qquad \Delta_2^{\ell,\star} = \mathrm{Polar}\Big(M_2^\ell D_{\mu^{\ell,\star}}^\alpha\Big)\, D_{\mu^{\ell,\star}}^\alpha. \tag{5}$$

The derivation is given in Appendix B.1. The parameters are then updated as $W_i^\ell \leftarrow W_i^\ell - \eta \Delta_i^{\ell,\star}$. We call this an oracle because it solves the allocation problem in Eq. (4) exactly at every step; the practical approximation is introduced in Section 4.

Allocation curvature. The exponent α controls the curvature of the allocation objective in Eq. (4).

Theorem 1 (Curvature transition)**.** *For any momentum matrices, Φ_M is concave in μ for $0 < \alpha \le \frac{1}{2}$, convex in μ for $\alpha = 1$, and in general neither convex nor concave for $\frac{1}{2} < \alpha < 1$.*

For $0 < \alpha \le \frac{1}{2}$, Eq. (4) maximizes a concave function over the convex set $\mathcal{K}_K$, and is therefore a convex program; every local maximizer is global, and optimal allocations may lie in the relative interior of $\mathcal{K}_K$, hence may be fractional. By contrast, at $\alpha = 1$ and integer K, a convex function is maximized over the polytope $\mathcal{K}_K$, so a maximizer can be chosen at an extreme point of $\mathcal{K}_K$, i.e., as a binary allocation selecting exactly K channels. The proof and details are given in Appendix B.3.

2.3 INDUCED OPTIMIZATION GEOMETRY

Joint maximization over μ and Δ in Eq. (3) is equivalent to maximizing $\langle M, \Delta\rangle$ over $\Delta \in \mathcal{A}_{\alpha,K}$, where $\mathcal{A}_{\alpha,K} := \bigcup_{\mu \in \mathcal{K}_K} \mathcal{A}_\alpha(\mu)$. Since a linear functional has the same maximum over a set and its convex hull, the BMuon oracle maximizes $\langle M, \Delta\rangle$ over $\Delta \in \mathcal{B}_{\alpha,K}$, where $\mathcal{B}_{\alpha,K} := \mathrm{conv}(\mathcal{A}_{\alpha,K})$.

As shown in Appendix B.4, $\mathcal{B}_{\alpha,K}$ is compact, convex, centrally symmetric, and has the origin in its interior. Its Minkowski functional $\Omega_{\alpha,K}(\Delta) := \inf\{t > 0 : \Delta \in t\,\mathcal{B}_{\alpha,K}\}$ is therefore a norm whose unit ball is $\mathcal{B}_{\alpha,K}$. Thus, for fixed α, K, BMuon is exactly normalized steepest descent under $\Omega_{\alpha,K}$:

$$\Delta^\star \in \operatorname*{arg\,max}_{\Omega_{\alpha,K}(\Delta) \le 1} \langle M, \Delta\rangle. \tag{6}$$

In comparison with Eq. (2), BMuon uses a modified norm constraint, hence a different geometry.

2.4 From Muon to Sparse Geometries

We prove that the family $\Omega_{\alpha,K}$ interpolates between Muon's spectral geometry and channel-sparse geometries. Proofs of these special cases are given in Appendix B.5.

Recovering Muon. If $K = N$, the constraints $\mu_j^\ell \le 1$ and $\sum_{\ell,j} \mu_j^\ell = N$ force $\mu_j^\ell = 1$ for every channel, so $\mathcal{B}_{\alpha,N} = \mathcal{A}_{\mathrm{Muon}}$ and $\Omega_{\alpha,N} = \|\cdot\|_{\mathrm{Muon}}$. Muon is also recovered for any K by extending the family to $\alpha = 0$ with $D_\mu^0 = I$.

A structured K-support geometry. Assume K is an integer and $\alpha = 1$.
Define $\mathrm{supp}_{\mathrm{ch}}(\Delta) := \{(\ell, j) : \Delta_{1,j:}^\ell \neq 0 \text{ or } \Delta_{2,:,j}^\ell \neq 0\}$. The induced unit ball is

$$\mathcal{B}_{1,K} = \mathrm{conv}\left\{\Delta : |\,\mathrm{supp}_{\mathrm{ch}}(\Delta)| \le K,\ \|\Delta_1^\ell\|_{\mathrm{op}} \le 1,\ \|\Delta_2^\ell\|_{\mathrm{op}} \le 1,\ \forall \ell\right\}.$$

Thus $\Omega_{1,K}$ is closely related to k-support and group k-support norms (Argyriou et al., 2012; Rao et al., 2017), but BMuon uses coupled hidden-channel support and operator-norm constraints rather than Euclidean ones. It is also related to spectral k-support norms (McDonald et al., 2014; 2016), but sparsity acts on hidden channels rather than singular directions.

A coupled group-ℓ_1 geometry. At $\alpha = K = 1$, $\Omega_{1,1}(\Delta) = \sum_{\ell,j} \max\{\|\Delta_{1,j:}^\ell\|_2, \|\Delta_{2,:,j}^\ell\|_2\}$. This is an ℓ_1 aggregation over coupled hidden channels and therefore promotes channel sparsity.

3 Representation Cost and Intrinsic Width

Section 2 showed that the BMuon oracle performs steepest descent under $\Omega_{\alpha,K}$. We now ask whether minimizing $\Omega_{\alpha,K}$ among parameterizations that realize the same predictor favors representations with fewer active hidden channels. This question is motivated by prior work showing that the norm underlying steepest descent can directly shape its implicit bias (Gunasekar et al., 2018; Tsilivis et al., 2025; Fan et al., 2025; Jacobs et al., 2026). Rather than establishing an implicit-bias result for BMuon, we characterize the exact parameterizations of a fixed predictor that minimize $\Omega_{\alpha,K}$, following the representation-cost view of Dai et al. (2021); Jagadeesan et al. (2022).

We fix predictors $X = (X_\ell)_{\ell=1}^L$, where each $X_\ell \in \mathbb{R}^{p_\ell \times d_\ell}$ is nonzero, and denote $r_\ell := \mathrm{rank}(X_\ell)$. We consider linear two-factor blocks $W = ((W_1^\ell, W_2^\ell))_{\ell=1}^L$, with $W_1^\ell \in \mathbb{R}^{m_\ell \times d_\ell}$, $W_2^\ell \in \mathbb{R}^{p_\ell \times m_\ell}$, and $m_\ell \ge r_\ell$. Each block realizes the matrix factorization $W_2^\ell W_1^\ell$, corresponding to the setup in Eq. (1) with the identity activation and no bias. For any parameter norm $\|\cdot\|$, define its representation cost

$$\mathcal{R}_{\|\cdot\|}(X) := \inf\left\{\|W\|^2 : W_2^\ell W_1^\ell = X_\ell,\ \forall \ell\right\}.$$

We write $\mathcal{R}_{\alpha,K} := \mathcal{R}_{\Omega_{\alpha,K}}$ for BMuon and $\mathcal{R}_{\mathrm{Muon}} := \mathcal{R}_{\|\cdot\|_{\mathrm{Muon}}}$ for Muon. For a factorization W, let $a_\ell(W)$ denote the active width of block ℓ, i.e., the number of hidden channels j for which $W_{1,j:}^\ell \neq 0$ or $W_{2,:,j}^\ell \neq 0$. Every exact factorization of X satisfies $a_\ell(W) \ge r_\ell$ for all ℓ, so r_ℓ is the minimum possible active width of block ℓ.

Let $\sigma_{\ell,1} \ge \cdots \ge \sigma_{\ell,r_\ell} > 0$ denote the nonzero singular values of X_ℓ, let $\sigma_\star := \max_{\ell,i} \sigma_{\ell,i}$, and define the α-effective spectral dimension $d_{\mathrm{eff}}^{(\alpha)}(X) := \sum_{\ell=1}^L \sum_{i=1}^{r_\ell} (\sigma_{\ell,i}/\sigma_\star)^{1/(2\alpha)}$.

Theorem 2 (Representation cost and intrinsic-width recovery)**.** *Let $0 < \alpha \le \frac{1}{2}$ and $0 < K \le N$. Both representation-cost infima are attained, and*

$$\mathcal{R}_{\mathrm{Muon}}(X) = \sigma_\star, \qquad \mathcal{R}_{\alpha,K}(X) = \sigma_\star \max\left\{1,\ \left(d_{\mathrm{eff}}^{(\alpha)}(X)/K\right)^{2\alpha}\right\}.$$

Moreover, Muon admits a minimizer $W^\star$ with $a_\ell(W^\star) = m_\ell$ for every ℓ, whereas if $K \le d_{\mathrm{eff}}^{(\alpha)}(X)$, every BMuon minimizer $W^\star$ satisfies $a_\ell(W^\star) = r_\ell$ for all ℓ.

Proof is given in Appendix C.1.

Interpretation. Muon's geometric cost is insensitive to hidden-channel redundancy, since its minimum can be attained at full hidden width. In contrast, BMuon is sensitive to redundant width: when $K \le d_{\mathrm{eff}}^{(\alpha)}(X)$, every minimum-cost factorization is width-minimal, so any redundancy incurs additional geometric cost. The budget raises the representation cost only when $K < d_{\mathrm{eff}}^{(\alpha)}(X)$, in which

case it is $\sigma_\star\big(d_{\text{eff}}^{(\alpha)}(X)/K\big)^{2\alpha}$; otherwise, it is $\sigma_\star$. Since $d_{\text{eff}}^{(\alpha)}(X)$ is nondecreasing in α, increasing α can only enlarge the budget regime in which width-minimality is guaranteed.

Effective spectral dimension. The α-effective spectral dimension recovers familiar spectral quantities. Let $X_{\text{bd}} := \text{diag}(X_1, \dots, X_L)$. For $\alpha = \frac{1}{4}$, $d_{\text{eff}}^{(\alpha)}(X) = \|X_{\text{bd}}\|_F^2/\|X_{\text{bd}}\|_{\text{op}}^2$, the stable rank of X_{bd}, while for $\alpha = \frac{1}{2}$, $d_{\text{eff}}^{(\alpha)}(X) = \|X_{\text{bd}}\|_*/\|X_{\text{bd}}\|_{\text{op}}$.

4 Practical BMuon

Exactly solving the allocation problem in Eq. (4) at every optimizer step is expensive. We instead perform a minorize–maximize (MM) update (Hunter & Lange, 2004), replacing the exact allocation problem by the maximization of a surrogate. For fixed momenta M and exponent α, given an allocation $\bar{\mu}$, let $\bar{P}_1^\ell = \text{Polar}(D_{\bar{\mu}^\ell}^\alpha M_1^\ell)$ and $\bar{P}_2^\ell = \text{Polar}(M_2^\ell D_{\bar{\mu}^\ell}^\alpha)$. As shown in Appendix D.1, we have the tight minorization

$$\Phi_M(\mu) \geq Q(\mu \mid \bar{\mu}) := \sum_{\ell,j} a_j^\ell (\mu_j^\ell)^\alpha, \qquad a_j^\ell = \langle M_{1,j:}^\ell, \bar{P}_{1,j:}^\ell \rangle + \langle M_{2,:,j}^\ell, \bar{P}_{2,:,j}^\ell \rangle, \tag{7}$$

with equality at $\mu = \bar{\mu}$. At the boundary $\bar{\mu}_j^\ell = 0$, row j of $D_{\bar{\mu}^\ell}^\alpha M_1^\ell$ and column j of $M_2^\ell D_{\bar{\mu}^\ell}^\alpha$ vanish, giving $a_j^\ell = 0$, so the surrogate no longer depends on μ_j^ℓ. We therefore impose a small floor $\mu_j^\ell \geq \varepsilon$ in practice and maximize the surrogate over $\mathcal{K}_{K,\varepsilon} := \{\mu \in [\varepsilon, 1]^N : \sum_{\ell,j} \mu_j^\ell = K\}$, where $0 < \varepsilon < K/N$. Proposition 10 bounds the gap between the floored and original oracle objectives. The scores a_j^ℓ are nonnegative in general and, if for every (ℓ, j), $M_{1,j:}^\ell \neq 0$ or $M_{2,:,j}^\ell \neq 0$, they are strictly positive for every $\bar{\mu} \in \mathcal{K}_{K,\varepsilon}$. Hence, for $0 < \alpha < 1$, the surrogate is strictly concave and has a unique maximizer. The KKT conditions, derived in Appendix D.2, give

$$(\mu_j^\ell)^+ = \text{clip}_{[\varepsilon,1]}\left[\left(\frac{\alpha a_j^\ell}{\lambda_\mu}\right)^{1/(1-\alpha)}\right], \tag{8}$$

where $\text{clip}_{[\varepsilon,1]}(x) := \min\{1, \max\{\varepsilon, x\}\}$, and $\lambda_\mu > 0$ is found by bisection from $\sum_{\ell,j}(\mu_j^\ell)^+ = K$. Thus, the MM allocation step introduces no additional step-size hyperparameter.

For fixed momenta, each MM step is monotone since $\Phi_M(\mu^+) \geq Q(\mu^+ \mid \bar{\mu}) \geq Q(\bar{\mu} \mid \bar{\mu}) = \Phi_M(\bar{\mu})$. Moreover, for $0 < \alpha \leq \frac{1}{2}$, repeated MM steps converge in objective value to the global optimum over $\mathcal{K}_{K,\varepsilon}$, and every accumulation point of the allocation sequence is globally optimal. The proof is given in Appendix D.3.

Algorithm 1 summarizes the resulting update. When μ is updated, the first polar evaluation constructs the MM surrogate at the current allocation. After the MM step, the polar factors are recomputed using the new allocation to form the parameter update.

Algorithm 1 Budget Muon

Require: $K \in (0, N)$, $\alpha \in (0, 1)$, $0 < \varepsilon < K/N$, allocation period ΔT_μ, learning rate η_t

$\mu_j^\ell \leftarrow K/N$ ▷ $\forall\,(\ell, j)$
for $t = 0, 1, \dots$ **do**
  Update momenta $\{M_1^\ell, M_2^\ell\}_{\ell=1}^L$ as in Muon
  if $t \bmod \Delta T_\mu = 0$ **then**
    $\widetilde{P}_1^\ell \leftarrow \text{Polar}(D_{\mu^\ell}^\alpha M_1^\ell), \quad \widetilde{P}_2^\ell \leftarrow \text{Polar}(M_2^\ell D_{\mu^\ell}^\alpha)$ ▷ $\forall\,\ell$
    $a_j^\ell \leftarrow \langle M_{1,j:}^\ell, \widetilde{P}_{1,j:}^\ell \rangle + \langle M_{2,:,j}^\ell, \widetilde{P}_{2,:,j}^\ell \rangle$ ▷ $\forall\,(\ell, j)$
    Set $\mu_j^\ell \leftarrow \text{clip}_{[\varepsilon,1]}\left[\left(\alpha a_j^\ell/\lambda_\mu\right)^{1/(1-\alpha)}\right]$ and choose λ_μ so that $\sum_{\ell,j} \mu_j^\ell = K$ ▷ $\forall\,(\ell, j)$
  end if
  $\Delta_1^\ell \leftarrow D_{\mu^\ell}^\alpha \text{Polar}(D_{\mu^\ell}^\alpha M_1^\ell), \quad \Delta_2^\ell \leftarrow \text{Polar}(M_2^\ell D_{\mu^\ell}^\alpha) D_{\mu^\ell}^\alpha$ ▷ $\forall\,\ell$
  $W_1^\ell \leftarrow W_1^\ell - \eta_t \Delta_1^\ell, \quad W_2^\ell \leftarrow W_2^\ell - \eta_t \Delta_2^\ell$ ▷ $\forall\,\ell$
end for

The budget K can be scheduled during training for example by annealing it from N to a target value. Parameters outside the MLP blocks are updated as in the corresponding Muon baseline.

The special cases in Section 2.4 place Muon at the $\alpha = 0$ endpoint and channel-sparse geometries at $\alpha = 1$, suggesting that larger α corresponds to more sparsity-favoring geometries. Moreover, away from the clipping boundaries, Eq. (8) implies $(\mu_i)^+/(\mu_j)^+ = (a_i/a_j)^{1/(1-\alpha)}$, so larger α amplifies differences between channel scores and favors more concentrated allocations. At the same time, $\alpha = 1/2$ is the largest exponent for which the allocation objective is guaranteed to remain concave by Theorem 1 and for which repeated MM steps are guaranteed to converge in objective value by Theorem 9. We therefore use $\alpha = 1/2$ throughout our experiments.

In practice, we approximate each polar factor by a finite Newton–Schulz iteration, as in standard Muon, and update the allocation by a single MM step every ΔT_μ optimizer steps, keeping it fixed between these updates. We assess these practical approximations empirically in Appendix D.5, using $\Delta T_\mu = 5$ and a certified reference allocation obtained by repeatedly applying the exact-polar MM update until its relative objective loss from the exact optimum is certified to be below a small threshold. In these diagnostics, the relative loss in the allocation objective incurred by the practical one-step Newton–Schulz-based update is certified to remain on the order of 10^{-4} to 10^{-3}, while keeping the allocation fixed for the four intervening optimizer steps keeps the corresponding certified deficit on the order of 10^{-3}. In the reported comparison, replacing the practical allocation update every five optimizer steps by the certified reference allocation leaves test accuracy and post-training compressibility unchanged within the observed seed-to-seed variability.

Relative to Muon, BMuon adds only channelwise scalings on the budgeted MLP matrices and, every ΔT_μ steps, one extra polar evaluation per such matrix; all other parameters are updated as in Muon. In our ImageNet-1K and 1B-parameter LLaMA experiments in Sections 5.2 and 5.3, respectively, we observe no measurable increase in overall training time relative to Muon (Appendix D.6).

5 Experiments

5.1 Post-Training Compression

Setup. We first test whether BMuon produces models that can be compressed directly after training without post-compression fine-tuning. We train a small Vision Transformer on CIFAR-100 for 200 epochs over three seeds. We compare Muon, Muon with a proximal Group-Lasso penalty on coupled MLP channels (Muon+GL) (Yuan & Lin, 2006; Wen et al., 2016), NuMuon (Dolatabadi et al., 2026), and BMuon. NuMuon is particularly relevant as a direct comparison: it modifies the Muon update to improve compressibility, but targets low-rank structure through the singular spectrum, whereas BMuon targets structured MLP-channel compression. BMuon uses $\alpha = 1/2$ and a budget annealed to 20% of the total MLP-channel mass. Full experimental details in Appendix E.2.

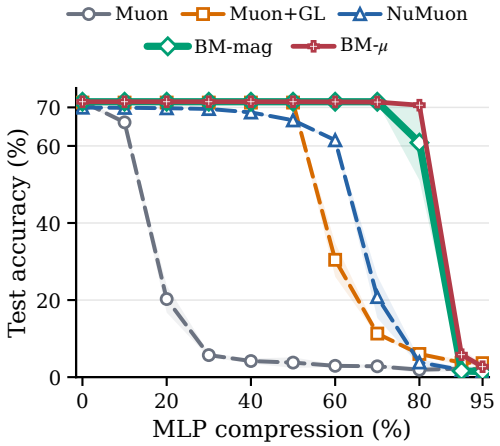


Figure 2: **CIFAR-100 ViT post-training MLP compression.** Mean test accuracy ± sample std. over three seeds after compression without recovery. Muon, Muon+GL, and BM-mag use global MLP-channel pruning ranked by magnitude; BM-μ uses μ for ranking, while NuMuon uses parameter-matched truncated-SVD compression of the MLP weights.

Post-training compression. We restrict the compression to the ViT MLP blocks. Muon, Muon+GL, and BMuon are compressed by globally pruning hidden channels across all blocks. For magnitude-based pruning, channel j in block ℓ is ranked by $s_j^\ell = \sqrt{\|W_{1,j:}^\ell\|_2^2 + \|W_{2,:,j}^\ell\|_2^2}$. BMuon is pruned using both this score (BM-mag) and its final learned allocation μ (BM-μ). NuMuon instead compresses each MLP weight matrix independently using truncated SVD, with ranks chosen to approximately match the parameter count of the corresponding channel-pruned model.

Results. Figure 2 shows the post-training fragility that motivates this work: Muon falls to 5.72% test accuracy after only 30% MLP-channel pruning. Muon+GL remains robust through 50% pruning, while NuMuon degrades more gradually under low-rank compression. Under magnitude-based pruning, BMuon is substantially more compressible than the structured baselines while matching Muon's dense accuracy: BMuon reaches 71.46%, versus 71.49% for Muon, 71.25% for Muon+GL, and 69.95% for NuMuon. Using μ for ranking provides a further gain: at 80% pruning, BM-μ retains 70.57% accuracy, compared with 60.84% for BM-mag, while Muon, Muon+GL, and NuMuon have fallen to 1.93%, 6.01%, and 3.84%, respectively. The advantage of BM-mag under the shared magnitude criterion isolates the compressibility benefit of BMuon training, while the further gain from BM-μ indicates that the learned allocation also carries useful channel-importance information.

5.2 Gradual Structured Pruning

We next ask whether BMuon's compressibility advantage persists when channel pruning is interleaved with training and recovery (Aketi et al., 2020), and whether its learned allocation identifies useful architectures when retrained from scratch.

Setup. We evaluate gradual MLP-channel pruning during training from scratch on a ViT for CIFAR-100 and a ViT-S/16 for ImageNet-1K, targeting final pruning levels of 90% and 95%. All results are reported over three seeds. Each run starts dense and alternates between training and pruning. After each training phase, we rank all surviving MLP channels globally, permanently remove the lowest-ranked ones to reach the next sparsity target, and resume training from the surviving weights. We repeat this train–prune cycle until the final pruning level is reached, and then continue training for recovery. We compare three methods. *Muon-mag* trains with Muon and ranks channels by the coupled weight-magnitude score introduced in Section 5.1. *BMuon-mag* trains the MLPs with BMuon ($\alpha = 1/2$) and uses the same magnitude ranking. *BMuon-μ* uses the same BMuon training but ranks channels by the learned allocation μ. During each BMuon training phase, the global budget is annealed to the next target channel count over the first half of the phase and held fixed for the second half.

As an additional control, we test whether the architecture selected by BMuon-μ is useful on its own. We keep only its final per-block widths, initialize a new model with those widths, and train it from scratch with standard Muon for the same training horizon. As a baseline, we also train a uniform-width model from scratch with nearly the same total number of MLP channels. Full details are given in Appendix E.4.

Table 1: **Gradual global MLP-channel pruning.** Final Top-1 accuracy (%), reported as mean $\pm$ sample std. over three seeds. The upper block reports the final pruned models after recovery; the lower block trains the architectures selected by BMuon-μ from scratch with Muon and compares their accuracy with uniform-width models having nearly the same total number of MLP channels. Details in Appendix E.4.

	CIFAR-100		**ImageNet-1K**	
Method	**90%**	**95%**	**90%**	**95%**
Gradual pruning				
Muon-mag	63.00 ± 0.28	59.70 ± 0.71	65.26 ± 0.19	61.24 ± 0.47
BMuon-mag	64.78 ± 0.89	62.88 ± 0.22	68.71 ± 0.11	65.77 ± 0.24
BMuon-μ	$\mathbf{67.24 \pm 0.22}$	$\mathbf{63.91 \pm 0.07}$	$\mathbf{69.06 \pm 0.19}$	$\mathbf{67.04 \pm 0.02}$
Architectures trained from scratch				
Uniform	64.85 ± 0.54	63.62 ± 0.36	69.18 ± 0.04	66.73 ± 0.26
BMuon-μ architecture	$\mathbf{67.00 \pm 0.54}$	$\mathbf{64.71 \pm 0.24}$	$\mathbf{69.52 \pm 0.17}$	$\mathbf{67.25 \pm 0.14}$

Gradual pruning performance. Table 1 shows that BMuon consistently improves over Muon under gradual pruning. Using the same magnitude-based pruning criterion, BMuon-mag exceeds Muon-mag by 1.78 and 3.18 percentage points on CIFAR-100 at 90% and 95% pruning, and by 3.45 and 4.53 points on ImageNet-1K. Using μ as the pruning score then provides an additional gain of 2.46 and 1.03 points on CIFAR-100, and 0.34 and 1.28 points on ImageNet-1K.

Architecture discovery. When trained from scratch, the architectures selected by BMuon-μ outperform the nearly channel-matched uniform models in all four settings. Because both the discovered and uniform architectures are initialized from scratch and trained with standard Muon, this comparison isolates the effect of the architecture's width profile. The results therefore provide evidence that BMuon-μ identifies useful non-uniform architectures.

5.3 Structured Compressibility at Language-Model Scale

The BMuon construction on MLP hidden channels extends naturally to gated MLPs such as SwiGLU, as used in LLaMA-style models. The same allocation acts on the rows of the gate and up projections and on the columns of the down projection; see Appendix B.2 for details.

To test whether BMuon's structured-compressibility advantage extends to large-scale language modeling, we train a 1B-parameter LLaMA-style model on 13.1B FineWeb tokens over three seeds. BMuon uses $\alpha = 1/2$ and $K = 0.2N$. After training, we globally prune SwiGLU hidden channels without recovery, using weight magnitude and an activation-aware score combining root-mean-square activation and output-weight norm; for BMuon, we also rank channels by μ. The pruning curves are shown in Figure 3; full details are given in Appendix E.5.

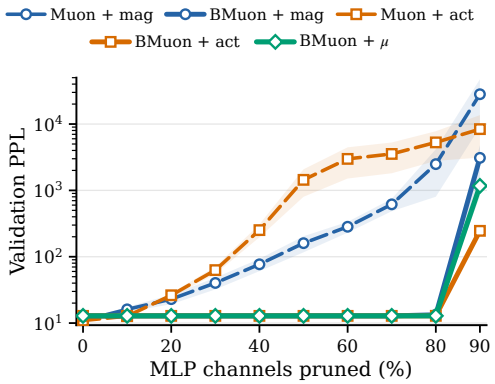


Figure 3: **Post-training MLP-channel pruning at language-model scale.** Validation perplexity after global structured pruning of a 1B-parameter LLaMA-style model trained on FineWeb. Mean $\pm$ std. over three seeds.

On the unpruned models, Muon achieves lower perplexity (10.92 versus 12.84 for BMuon), but its performance degrades rapidly under pruning. In contrast, BMuon preserves its unpruned performance up to 80% pruning, reaching 12.86 with activation ranking and 12.85 with μ ranking. BMuon substantially outperforms Muon when both use the same magnitude or activation criterion at moderate and high pruning levels, while μ again provides an effective pruning score. Overall, BMuon trades some unpruned quality for substantially greater robustness to structured channel pruning.

6 Conclusion

Muon-trained Transformers can degrade sharply under post-training MLP-channel pruning, motivating the question of whether Muon's spectral geometry can be modified to favor structured compressibility. BMuon addresses this question by reweighting hidden channels within Muon's spectral-norm constraints and making them compete for a global budget. We show that the resulting norm family recovers Muon and connects it to channel-sparse geometries. In linear two-factor models, Muon admits minimum-cost factorizations with redundant hidden width, whereas under a sufficiently scarce BMuon budget every minimum-cost factorization is width-minimal. For practical training, we derive a tractable minorization–maximization approximation to the global allocation oracle. Across vision and language models, BMuon is substantially more robust than Muon to magnitude-based MLP-channel pruning. The learned allocation also provides an effective pruning score and identifies useful non-uniform MLP architectures when retrained from scratch.

Several questions remain open. Our theoretical results characterize representation cost in linear two-factor models; studying the implicit bias of BMuon training, particularly beyond the linear setting, is an important direction for future work. Another direction is to design a practical BMuon algorithm with provable approximation guarantees to the oracle. Our experiments focus on MLP hidden channels, and extending the global budget to other structured components of Transformers remains to be explored. Finally, BMuon can trade dense-model quality for greater pruning robustness, as observed in our language-model experiments; characterizing this tradeoff at larger model scales is a natural direction for future work.

AI USE STATEMENT

Generative AI tools were used to assist with creating and editing scientific figures, writing and editing software code, identifying and summarizing relevant literature, proposing the paper title and keywords, and improving the clarity and readability of the manuscript. They were not used to develop theoretical or conceptual results, formulate mathematical claims, construct proofs, propose hypotheses, design experiments or research methodology, implement the proposed method, interpret experimental results, generate or process datasets, or perform translation. All AI-assisted content was reviewed by the authors, who take full responsibility for the final manuscript.

REPRODUCIBILITY STATEMENT

The main text and appendix provide all definitions, assumptions, derivations, and proofs for the theoretical results. For each experiment, the appendix provides the model and data configuration, training and optimization hyperparameters, pruning and compression protocols, baseline implementation details, evaluation procedures, and random seeds. We also report the exact numerical results underlying the figures in the main text. The supplementary material provides the code and configurations for reproducing the experiments reported in the paper. Finally, the appendix documents the implementation provenance of the large-scale experiments and specifies the exact adaptations made to the underlying implementations.

REFERENCES

Sai Aparna Aketi, Sourjya Roy, Anand Raghunathan, and Kaushik Roy. Gradual channel pruning while training using feature relevance scores for convolutional neural networks. *IEEE Access*, 8: 171924–171932, 2020. doi: 10.1109/ACCESS.2020.3024992.

Yongqi An, Xu Zhao, Tao Yu, Ming Tang, and Jinqiao Wang. Fluctuation-based adaptive structured pruning for large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 10865–10873, 2024. doi: 10.1609/aaai.v38i10.28960.

Andreas Argyriou, Rina Foygel, and Nathan Srebro. Sparse prediction with the k-support norm. In *Advances in Neural Information Processing Systems*, volume 25, 2012.

Jeremy Bernstein and Laker Newhouse. Old optimizer, new norm: An anthology, 2024. URL https://arxiv.org/abs/2409.20325.

Lucas Beyer, Xiaohua Zhai, and Alexander Kolesnikov. Better plain ViT baselines for ImageNet-1k. *arXiv preprint arXiv:2205.01580*, 2022.

Rajendra Bhatia. *Matrix Analysis*, volume 169 of *Graduate Texts in Mathematics*. Springer, New York, 1997.

Zhen Dai, Mina Karzand, and Nathan Srebro. Representation costs of linear neural networks: Analysis and design. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, 2021.

Hadi Mohaghegh Dolatabadi, Thalaiyasingam Ajanthan, Sameera Ramasinghe, Chamin P. Hewa Koneputugodage, Shamane Siriwardhana, Violetta Shevchenko, Karol Pajak, James Snewin, Gil Avraham, and Alexander Long. NuMuon: Nuclear-norm-constrained Muon for compressible LLM training, 2026.

Chen Fan, Mark Schmidt, and Christos Thrampoulidis. Implicit bias of spectral descent and muon on multiclass separable data. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, 2025.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 5484–5495. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.emnlp-main.446.

Ariel Gordon, Elad Eban, Ofir Nachum, Bo Chen, Hao Wu, Tien-Ju Yang, and Edward Choi. MorphNet: Fast & simple resource-constrained structure learning of deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1586–1595, 2018.

Eitan Gronich and Gal Vardi. The implicit bias of adam and muon on smooth homogeneous neural networks, 2026. URL https://arxiv.org/abs/2602.16340.

Suriya Gunasekar, Jason Lee, Daniel Soudry, and Nathan Srebro. Characterizing implicit bias in terms of optimization geometry. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1832–1841, 2018.

David R. Hunter and Kenneth Lange. A tutorial on mm algorithms. *The American Statistician*, 58 (1):30–37, 2004. doi: 10.1198/0003130042836.

Tom Jacobs, Chao Zhou, and Rebekka Burkholz. Never saddle for reparameterized steepest descent as mirror flow. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=YgudIlQ9nC.

Meena Jagadeesan, Ilya Razenshteyn, and Suriya Gunasekar. Inductive bias of multi-channel linear convolutional networks with bounded weight norm. In *Proceedings of the 35th Conference on Learning Theory (COLT)*, volume 178 of *Proceedings of Machine Learning Research*, 2022.

Keller Jordan, Yuchen Jin, Vlado Boza, Jiacheng You, Franz Cesista, Laker Newhouse, and Jeremy Bernstein. Muon: An optimizer for hidden layers in neural networks, 2024. URL https://kellerjordan.github.io/posts/muon/.

Woosuk Kwon, Sehoon Kim, Michael W. Mahoney, Joseph Hassoun, Kurt Keutzer, and Amir Gholami. A fast post-training pruning framework for transformers. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Soumia Laroui, Hamida Seba, Karima Amrouche, and Yassine Ait Ali Yahia. DABP: Dual-augmented budgeted channel pruning via sensitivity-aware gating. *Neurocomputing*, 704:134843, 2026. doi: 10.1016/j.neucom.2026.134843.

Jaeho Lee, Sejun Park, Sangwoo Mo, Sungsoo Ahn, and Jinwoo Shin. Layer-adaptive sparsity for the magnitude-based pruning. In *International Conference on Learning Representations*, 2021.

Jingyuan Liu, Jianlin Su, Xingcheng Yao, Zhejun Jiang, Guokun Lai, Yulun Du, Yidao Qin, Weixin Xu, Enzhe Lu, Junjie Yan, Yanru Chen, Huabin Zheng, Yibo Liu, Shaowei Liu, Bohong Yin, Weiran He, Han Zhu, Yuzhi Wang, Jianzhou Wang, Mengnan Dong, Zheng Zhang, Yongsheng Kang, Hao Zhang, Xinran Xu, Yutao Zhang, Yuxin Wu, Xinyu Zhou, and Zhilin Yang. Muon is scalable for LLM training. *arXiv preprint arXiv:2502.16982*, 2025.

Miao Lu, Xiaolong Luo, Tianlong Chen, Wuyang Chen, Dong Liu, and Zhangyang Wang. Learning pruning-friendly networks via frank–wolfe: One-shot, any-sparsity, and no retraining. In *International Conference on Learning Representations*, 2022.

Andrew M. McDonald, Massimiliano Pontil, and Dimitris Stamos. Spectral k-support norm regularization. In *Advances in Neural Information Processing Systems*, volume 27, 2014.

Andrew M. McDonald, Massimiliano Pontil, and Dimitris Stamos. Fitting spectral decay with the k-support norm. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2016.

Thomas Pethick, Wanyun Xie, Kimon Antonakopoulos, Zhenyu Zhu, Antonio Silveti-Falls, and Volkan Cevher. Training deep learning models with norm-constrained LMOs. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 49069–49104. PMLR, 2025.

Nikhil Rao, Miroslav Dudík, and Zaïd Harchaoui. The group k-support norm for learning with structured sparsity. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2402–2406, 2017. doi: 10.1109/ICASSP.2017.7952587.

Ishaan Shah, Anthony M. Polloreno, Karl Stratos, Philip Monk, Adarsh Chaluvaraju, Andrew Hojel, Andrew Ma, Anil Thomas, Ashish Tanwer, Darsh J. Shah, Khoi Nguyen, Kurt Smith, Michael Callahan, Michael Pust, Mohit Parmar, Peter Rushton, Platon Mazarakis, Ritvik Kapila, Saurabh Srivastava, Somanshu Singla, Tim Romanski, Yash Vanjani, and Ashish Vaswani. Practical efficiency of muon for pretraining, 2025.

Mingjie Sun, Zhuang Liu, Anna Bair, and J. Zico Kolter. A simple and effective pruning approach for large language models. In *International Conference on Learning Representations*, 2024.

Nikolaos Tsilivis, Gal Vardi, and Julia Kempe. Flavors of margin: Implicit bias of steepest descent in homogeneous neural networks. In *International Conference on Learning Representations (ICLR)*, 2025.

Ziyan Wang, Enmao Diao, Qi Le, Pu Wang, Minwoo Lee, Shu-ping Yeh, Evgeny Stupachenko, Hao Feng, and Li Yang. From local to global: Revisiting structured pruning paradigms for large language models. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 35720–35739. Association for Computational Linguistics, 2026a. doi: 10.18653/v1/2026.acl-long.1653. URL https://aclanthology.org/2026.acl-long.1653/.

Ziyang Wang, Jiangfeng Xiao, Chuan Xiao, Ruoxiang Li, Rui Mao, and Jianbin Qin. GRASPrune: Global gating for budgeted structured pruning of large language models. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10719–10736. Association for Computational Linguistics, 2026b. doi: 10.18653/v1/2026.acl-long.491. URL https://aclanthology.org/2026.acl-long.491/.

Wei Wen, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. Learning structured sparsity in deep neural networks. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

Huanrui Yang, Hongxu Yin, Maying Shen, Pavlo Molchanov, Hai Li, and Jan Kautz. Global vision transformer pruning with hessian-aware saliency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18547–18557, 2023.

Shixing Yu, Tianlong Chen, Jiayi Shen, Huan Yuan, Jianchao Tan, Sen Yang, Ji Liu, and Zhangyang Wang. Unified visual transformer compression. In *International Conference on Learning Representations*, 2022.

Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006. doi: 10.1111/j.1467-9868.2005.00532.x.

Ruijie Zhang, Yequan Zhao, Ziyue Liu, Zhengyang Wang, and Zheng Zhang. Muon+: Towards better muon via one additional normalization step. *arXiv preprint arXiv:2602.21545*, 2026.

Wenzhi Zhong, Edward Milsom, and Michael Murray. Sharpness-aware minimization and Muon: Robustness under the spectral norm. *arXiv preprint arXiv:2607.26001*, 2026.

Max Zimmer, Christoph Spiegel, and Sebastian Pokutta. Compression-aware training of neural networks using frank–wolfe. In *Mathematical Optimization for Machine Learning*, pp. 137–168. De Gruyter, 2025. doi: 10.1515/9783111376776-010.

BMuon: Budgeting Muon's Spectral Geometry for Structured Compressibility
— Appendix —

CONTENTS

A Muon Algorithm **14**

B BMuon Oracle and Geometry **15**

B.1 Derivation of the BMuon Oracle 15

B.2 Extension to Gated MLPs 16

B.3 Allocation Curvature 16

B.4 Norm Geometry of BMuon 17

B.5 Proofs for the Geometric Special Cases 20

C Compact-Representation Theory **22**

C.1 Intrinsic-Width Recovery 22

C.2 BMuon with Weight-Decay 26

D MM Allocation Update **28**

D.1 Minorization and allocation scores 28

D.2 Closed-form surrogate maximization 29

D.3 Convergence of repeated MM updates 30

D.4 Effect of the allocation floor 31

D.5 Approximation of the allocation oracle 32

D.6 Computational Overhead 34

E Experimental Details **37**

E.1 ImageNet-1K Post-Training Structured Pruning 37

E.2 CIFAR-100 Post-Training Compression 37

E.3 Effect of the Global Budget 39

E.4 Gradual Structured Pruning Experiment 40

E.5 LLaMA/FineWeb Experimental Details 41

A MUON ALGORITHM

Muon applies momentum followed by approximate matrix orthogonalization to eligible matrix parameters (Jordan et al., 2024). Algorithm 2 gives the update for one matrix parameter using the Nesterov-style momentum variant.

Algorithm 2 Muon

Require: Learning rates $\{\eta_t\}$, momentum β
1: Initialize $B_0 \leftarrow 0$
2: **for** $t = 1, 2, \ldots$ **do**
3: $\quad G_t \leftarrow \nabla_W \mathcal{L}_t(W_{t-1})$
4: $\quad B_t \leftarrow \beta B_{t-1} + G_t$
5: $\quad M_t \leftarrow G_t + \beta B_t$
6: $\quad O_t \leftarrow \mathrm{NewtonSchulz5}(M_t)$
7: $\quad W_t \leftarrow W_{t-1} - \eta_t O_t$
8: **end for**

The non-Nesterov variant instead uses $M_t = B_t$. Throughout the paper, *momentum* refers to M_t, the matrix that Muon orthogonalizes; our geometric results are stated for arbitrary M_t and therefore do not depend on the momentum variant used to produce it. We specify the variant used in each experimental setup.

Practical Muon implementations may additionally rescale each matrix update. The public Muon implementation uses the aspect-ratio correction $\sqrt{\max\{1, m/n\}}$ for $W \in \mathbb{R}^{m\times n}$ (Jordan et al., 2024), while the large-scale RMS-matching convention of Liu et al. (2025) uses $0.2\sqrt{\max\{m, n\}}$. Whenever such a rescaling is used, we state it in the corresponding experimental setup; otherwise, no rescaling is applied. We omit it from the geometric analysis.

Parameters not optimized by Muon, including biases, normalization parameters, embeddings, and classifier or output heads, are optimized separately with AdamW. With decoupled weight decay, the final update becomes

$$W_t = (1 - \eta_t\lambda)W_{t-1} - \eta_t\gamma O_t.$$

Newton–Schulz orthogonalization. For $M \in \mathbb{R}^{m\times n}$, let $\widetilde{M} = M$ if $m \le n$ and $\widetilde{M} = M^\top$ otherwise, and set $X_0 = \widetilde{M}/(\|\widetilde{M}\|_F + 10^{-7})$. Muon then applies five iterations

$$X_{k+1} = aX_k + b(X_kX_k^\top)X_k + c(X_kX_k^\top)^2X_k, \qquad k = 0, \ldots, 4,$$

with $(a, b, c) = (3.4445, -4.7750, 2.0315)$ (Jordan et al., 2024). We define $\mathrm{NewtonSchulz5}(M) = X_5$ if $m \le n$, and $\mathrm{NewtonSchulz5}(M) = X_5^\top$ otherwise. This finite iteration approximately orthogonalizes M; the main text uses the exact polar direction for the theoretical analysis.

B BMuon Oracle and Geometry

This appendix provides the derivations and proofs underlying Section 2. Appendix B.1 derives the BMuon allocation problem and update in Eqs. (4)–(5) from the joint oracle in Eq. (3). Appendix B.2 extends the construction to gated MLPs such as SwiGLU. Appendix B.3 proves the curvature transition of Theorem 1. Appendix B.4 establishes the induced norm geometry $\Omega_{\alpha,K}$ and its characterization for $0 < \alpha \le \frac{1}{2}$. Finally, Appendix B.5 proves the geometric special cases stated in Section 2.4.

Notation and setup. We consider L coupled MLP blocks with $W_1^\ell \in \mathbb{R}^{m_\ell \times d_\ell}$ and $W_2^\ell \in \mathbb{R}^{p_\ell \times m_\ell}$, and write $N := \sum_{\ell=1}^L m_\ell$ for the total number of hidden channels. Their Muon momentum matrices are denoted M_1^ℓ and M_2^ℓ, with corresponding update directions Δ_1^ℓ and Δ_2^ℓ. We write $M := ((M_1^\ell, M_2^\ell))_{\ell=1}^L$ and $\Delta := ((\Delta_1^\ell, \Delta_2^\ell))_{\ell=1}^L$, with $\langle M, \Delta\rangle := \sum_{\ell=1}^L \left(\langle M_1^\ell, \Delta_1^\ell\rangle + \langle M_2^\ell, \Delta_2^\ell\rangle\right)$. For a global budget $K \in (0, N]$, BMuon uses allocations $\mu = (\mu^1, \dots, \mu^L)$ in $\mathcal{K}_K := \left\{\mu \in [0,1]^N : \sum_{\ell,j} \mu_j^\ell = K\right\}$. For $\mu \in \mathcal{K}_K$, let $D_{\mu^\ell} := \mathrm{diag}(\mu^\ell)$. For fixed $\alpha \in (0, 1]$, the admissible BMuon directions are

$$\mathcal{A}_\alpha(\mu) := \left\{\Delta : \exists \{P_1^\ell, P_2^\ell\}_{\ell=1}^L,\ \Delta_1^\ell = D_{\mu^\ell}^\alpha P_1^\ell,\ \Delta_2^\ell = P_2^\ell D_{\mu^\ell}^\alpha,\ \|P_1^\ell\|_{\mathrm{op}} \le 1,\ \|P_2^\ell\|_{\mathrm{op}} \le 1,\ \forall \ell\right\}.$$

The BMuon oracle solves

$$\max_{\mu \in \mathcal{K}_K} \max_{\Delta \in \mathcal{A}_\alpha(\mu)} \langle M, \Delta\rangle,$$

and its allocation objective is

$$\Phi_M(\mu) = \sum_{\ell=1}^L \left[\|D_{\mu^\ell}^\alpha M_1^\ell\|_* + \|M_2^\ell D_{\mu^\ell}^\alpha\|_*\right].$$

B.1 Derivation of the BMuon Oracle

We derive Eqs. (4)–(5) from Eq. (3). Fix $\mu \in \mathcal{K}_K$. By definition of $\mathcal{A}_\alpha(\mu)$,

$$\begin{aligned}\max_{\Delta \in \mathcal{A}_\alpha(\mu)} \langle M, \Delta\rangle &= \max_{\substack{\|P_1^\ell\|_{\mathrm{op}} \le 1 \\ \|P_2^\ell\|_{\mathrm{op}} \le 1,\ \forall \ell}} \sum_{\ell=1}^L \left[\langle M_1^\ell, D_{\mu^\ell}^\alpha P_1^\ell\rangle + \langle M_2^\ell, P_2^\ell D_{\mu^\ell}^\alpha\rangle\right] \\ &= \sum_{\ell=1}^L \left[\max_{\|P_1^\ell\|_{\mathrm{op}} \le 1} \langle M_1^\ell, D_{\mu^\ell}^\alpha P_1^\ell\rangle + \max_{\|P_2^\ell\|_{\mathrm{op}} \le 1} \langle M_2^\ell, P_2^\ell D_{\mu^\ell}^\alpha\rangle\right],\end{aligned}$$

where the second equality follows because the objective is a sum over matrices and the constraints are independent across matrices.

Since $D_{\mu^\ell}^\alpha$ is diagonal, $\langle M_1^\ell, D_{\mu^\ell}^\alpha P_1^\ell\rangle = \langle D_{\mu^\ell}^\alpha M_1^\ell, P_1^\ell\rangle$ and $\langle M_2^\ell, P_2^\ell D_{\mu^\ell}^\alpha\rangle = \langle M_2^\ell D_{\mu^\ell}^\alpha, P_2^\ell\rangle$. Spectral–nuclear norm duality therefore gives

$$\max_{\Delta \in \mathcal{A}_\alpha(\mu)} \langle M, \Delta\rangle = \sum_{\ell=1}^L \left[\|D_{\mu^\ell}^\alpha M_1^\ell\|_* + \|M_2^\ell D_{\mu^\ell}^\alpha\|_*\right] = \Phi_M(\mu).$$

Maximizing this value over $\mu \in \mathcal{K}_K$ yields Eq. (4).

Moreover, Φ_M is continuous on $\mathcal{K}_K$: for $\alpha > 0$, the map $\mu \mapsto D_\mu^\alpha$ is continuous on $[0,1]^N$, and the nuclear norm is continuous. Since $\mathcal{K}_K$ is compact, a maximizer $\mu^\star$ exists.

Fix such a maximizer. By spectral–nuclear duality, $\mathrm{Polar}(A) \in \arg\max_{\|P\|_{\mathrm{op}} \le 1} \langle A, P\rangle$ for any matrix A. Therefore, for every ℓ, we can choose

$$P_1^{\ell,\star} = \mathrm{Polar}\left(D_{\mu^{\ell,\star}}^\alpha M_1^\ell\right), \qquad P_2^{\ell,\star} = \mathrm{Polar}\left(M_2^\ell D_{\mu^{\ell,\star}}^\alpha\right).$$

Returning to the update variables through the parameterization $\Delta_1^\ell = D_{\mu^\ell}^\alpha P_1^\ell$ and $\Delta_2^\ell = P_2^\ell D_{\mu^\ell}^\alpha$, substituting $\mu = \mu^\star$ and the maximizing $P_1^{\ell,\star}, P_2^{\ell,\star}$ gives, for every ℓ,

$$\Delta_1^{\ell,\star} = D_{\mu^{\ell,\star}}^\alpha \operatorname{Polar}\Big(D_{\mu^{\ell,\star}}^\alpha M_1^\ell\Big), \qquad \Delta_2^{\ell,\star} = \operatorname{Polar}\Big(M_2^\ell D_{\mu^{\ell,\star}}^\alpha\Big) D_{\mu^{\ell,\star}}^\alpha,$$

which is Eq. (5). This $\Delta^\star$ attains $\max_{\Delta\in\mathcal{A}_\alpha(\mu^\star)}\langle M,\Delta\rangle = \Phi_M(\mu^\star)$. Since $\mu^\star$ maximizes Φ_M over $\mathcal{K}_K$, the pair $(\mu^\star,\Delta^\star)$ solves the joint oracle in Eq. (3).

B.2 Extension to Gated MLPs

A SwiGLU block, omitting biases, has the form

$$x \mapsto W_{\text{down}}^\ell\Big(\operatorname{SiLU}(W_{\text{gate}}^\ell x)\odot W_{\text{up}}^\ell x\Big),$$

where $\odot$ denotes elementwise multiplication. Hidden channel j therefore couples row j of W_{gate}^ℓ, row j of W_{up}^ℓ, and column j of W_{down}^ℓ. BMuon assigns the same allocation μ_j^ℓ to these three components. Let M_{gate}^ℓ, M_{up}^ℓ, and M_{down}^ℓ denote their momentum matrices. For a fixed allocation μ, the admissible directions are parameterized as

$$\Delta_{\text{gate}}^\ell = D_{\mu^\ell}^\alpha P_{\text{gate}}^\ell, \qquad \Delta_{\text{up}}^\ell = D_{\mu^\ell}^\alpha P_{\text{up}}^\ell, \qquad \Delta_{\text{down}}^\ell = P_{\text{down}}^\ell D_{\mu^\ell}^\alpha,$$

with $\|P_{\text{gate}}^\ell\|_{\text{op}}, \|P_{\text{up}}^\ell\|_{\text{op}}, \|P_{\text{down}}^\ell\|_{\text{op}} \le 1$.

The same spectral–nuclear duality argument then gives the block contribution

$$\|D_{\mu^\ell}^\alpha M_{\text{gate}}^\ell\|_* + \|D_{\mu^\ell}^\alpha M_{\text{up}}^\ell\|_* + \|M_{\text{down}}^\ell D_{\mu^\ell}^\alpha\|_*.$$

Summing these terms over blocks gives the allocation objective under the same global constraint $\mu\in\mathcal{K}_K$. For an optimal allocation $\mu^\star$, an optimal direction is

$$\begin{aligned}
\Delta_{\text{gate}}^{\ell,\star} &= D_{\mu^{\ell,\star}}^\alpha \operatorname{Polar}\Big(D_{\mu^{\ell,\star}}^\alpha M_{\text{gate}}^\ell\Big),\\
\Delta_{\text{up}}^{\ell,\star} &= D_{\mu^{\ell,\star}}^\alpha \operatorname{Polar}\Big(D_{\mu^{\ell,\star}}^\alpha M_{\text{up}}^\ell\Big),\\
\Delta_{\text{down}}^{\ell,\star} &= \operatorname{Polar}\Big(M_{\text{down}}^\ell D_{\mu^{\ell,\star}}^\alpha\Big) D_{\mu^{\ell,\star}}^\alpha.
\end{aligned}$$

Thus the only change from the two-matrix case is that each hidden-channel allocation is shared by three coupled matrix components.

B.3 Allocation Curvature

We prove Theorem 1 by treating the three regimes separately.

Concavity for $0<\alpha\le\frac{1}{2}$. It suffices to consider one left-scaled term $F(\mu) := \|D_\mu^\alpha M\|_*$. Using $\|A\|_* = \operatorname{tr}((A^\top A)^{1/2})$,

$$F(\mu) = \operatorname{tr}\Big[\big(M^\top D_\mu^{2\alpha} M\big)^{1/2}\Big].$$

Define

$$G(x) := \operatorname{tr}\Big[\big(M^\top D_x M\big)^{1/2}\Big], \qquad x\in\mathbb{R}_+^m.$$

Then $F(\mu) = G(\mu^{2\alpha})$, where the power is applied coordinatewise.

The matrix square-root map is operator concave and operator monotone on the positive semidefinite cone (Bhatia, 1997). Since $x\mapsto M^\top D_x M$ is linear, G is concave. Moreover, G is coordinatewise nondecreasing. If $x\le y$, then $D_x \preceq D_y$, hence $M^\top D_x M \preceq M^\top D_y M$. Operator monotonicity of the square root and monotonicity of the trace therefore give $G(x)\le G(y)$.

For any $\mu,\nu\in\mathbb{R}_+^m$ and $\theta\in[0,1]$, since $0<2\alpha\le 1$, concavity of $t\mapsto t^{2\alpha}$ gives

$$(\theta\mu+(1-\theta)\nu)^{2\alpha} \ge \theta\mu^{2\alpha} + (1-\theta)\nu^{2\alpha}$$

coordinatewise. Hence, using first the monotonicity and then the concavity of G,

$$\begin{aligned} F(\theta\mu + (1-\theta)\nu) &= G\big((\theta\mu + (1-\theta)\nu)^{2\alpha}\big) \\ &\geq G\big(\theta\mu^{2\alpha} + (1-\theta)\nu^{2\alpha}\big) \\ &\geq \theta G(\mu^{2\alpha}) + (1-\theta)G(\nu^{2\alpha}) \\ &= \theta F(\mu) + (1-\theta)F(\nu). \end{aligned}$$

Thus $\mu \mapsto \|D_\mu^\alpha M\|_*$ is concave for $0 < \alpha \leq \frac{1}{2}$. Since $\|MD_\mu^\alpha\|_* = \|D_\mu^\alpha M^\top\|_*$, the same argument applies to right-scaled terms. Summing over all blocks and both terms in Eq. (4) proves that Φ_M is concave in μ.

Convexity at $\alpha = 1$. At $\alpha = 1$, both maps $\mu \mapsto D_\mu M$ and $\mu \mapsto MD_\mu$ are linear. Since the nuclear norm is convex, $\mu \mapsto \|D_\mu M\|_*$ and $\mu \mapsto \|MD_\mu\|_*$ are convex. Summing these terms over all blocks therefore shows that Φ_M is convex.

Intermediate regime $\frac{1}{2} < \alpha < 1$. Take the special case $L = 1$, $N = 2$, $K = 1$, and $M_2 = 0$. For $M_1 = I_2$, $\Phi_M(\mu) = \mu_1^\alpha + \mu_2^\alpha$. Hence, for every $\alpha < 1$,

$$\Phi_M\big(\tfrac{1}{2}, \tfrac{1}{2}\big) = 2^{1-\alpha} > \frac{\Phi_M(1,0) + \Phi_M(0,1)}{2} = 1,$$

so Φ_M is not convex.

Now let $M_1 = \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix}$. Then $\Phi_M(\mu) = (\mu_1^{2\alpha} + \mu_2^{2\alpha})^{1/2}$, and for every $\alpha > \frac{1}{2}$,

$$\Phi_M\big(\tfrac{1}{2}, \tfrac{1}{2}\big) = 2^{1/2-\alpha} < \frac{\Phi_M(1,0) + \Phi_M(0,1)}{2} = 1,$$

so Φ_M is not concave.

Thus, for $\frac{1}{2} < \alpha < 1$, Φ_M is in general neither convex nor concave. These examples show that the thresholds in Theorem 1 are sharp: for $0 < \alpha \leq 1$, Φ_M is concave for all momenta if and only if $\alpha \leq \frac{1}{2}$, and convex for all momenta if and only if $\alpha = 1$.

Extreme-point structure at $\alpha = 1$. Assume K integer. The extreme points of $\mathcal{K}_K$ are exactly the binary vectors with K ones. Indeed, every such vector is extreme because all its coordinates lie at the endpoints of $[0, 1]$. Conversely, any non-binary $\mu \in \mathcal{K}_K$ has at least two fractional coordinates: it cannot have exactly one since $\sum_i \mu_i = K$ is an integer. For two fractional coordinates $i \neq j$, sufficiently small $\varepsilon > 0$ gives $\mu^\pm = \mu \pm \varepsilon(e_i - e_j) \in \mathcal{K}_K$ with $\mu = \frac{1}{2}(\mu^+ + \mu^-)$, so μ is not extreme.

Since Φ_M is continuous and convex at $\alpha = 1$, a maximizer over the polytope $\mathcal{K}_K$ can be chosen at an extreme point. Thus an optimal allocation can be chosen binary, selecting exactly K channels. Writing $S_\ell := \{j : (\ell, j) \in S\}$ and $D_{S_\ell} := \mathrm{diag}(\mathbf{1}_{S_\ell})$, the allocation problem becomes

$$\max_{\mu \in \mathcal{K}_K} \Phi_M(\mu) = \max_{\substack{S \subseteq \{(\ell,j):\, 1 \leq \ell \leq L,\ 1 \leq j \leq m_\ell\} \\ |S| = K}} \sum_{\ell=1}^{L} \left[\|D_{S_\ell} M_1^\ell\|_* + \|M_2^\ell D_{S_\ell}\|_*\right].$$

Hence, at $\alpha = 1$, the continuous allocation problem admits an equivalent combinatorial formulation over subsets of K global channels.

B.4 Norm Geometry of BMuon

Using $\mathcal{K}_K$ and $\mathcal{A}_\alpha(\mu)$ as defined in the notation and setup at the beginning of Appendix B, define

$$\mathcal{A}_{\alpha,K} := \bigcup_{\mu \in \mathcal{K}_K} \mathcal{A}_\alpha(\mu), \qquad \mathcal{B}_{\alpha,K} := \mathrm{conv}(\mathcal{A}_{\alpha,K}).$$

Recall Muon's product norm and unit ball:

$$\|\Delta\|_{\mathrm{Muon}} := \max_\ell \max\left\{\|\Delta_1^\ell\|_{\mathrm{op}}, \|\Delta_2^\ell\|_{\mathrm{op}}\right\}, \qquad \mathcal{A}_{\mathrm{Muon}} := \{\Delta : \|\Delta\|_{\mathrm{Muon}} \leq 1\}.$$

Lemma 3 (BMuon norm and comparison with Muon)**.** *Let $0 < \alpha \le 1$ and $0 < K \le N$. Then $\mathcal{B}_{\alpha,K}$ is compact, convex, centrally symmetric, and has the origin in its interior. Consequently, its Minkowski functional $\Omega_{\alpha,K}$ is a norm with unit ball $\mathcal{B}_{\alpha,K}$. Moreover, for every Δ,*

$$\|\Delta\|_{\mathrm{Muon}} \le \Omega_{\alpha,K}(\Delta) \le \left(\frac{N}{K}\right)^{\alpha} \|\Delta\|_{\mathrm{Muon}}.$$

Proof. Fix $\mu \in \mathcal{K}_K$ and $\Delta \in \mathcal{A}_\alpha(\mu)$. By definition, for every ℓ, there exist P_1^ℓ, P_2^ℓ with $\|P_1^\ell\|_{\mathrm{op}}, \|P_2^\ell\|_{\mathrm{op}} \le 1$ such that $\Delta_1^\ell = D_{\mu^\ell}^\alpha P_1^\ell$ and $\Delta_2^\ell = P_2^\ell D_{\mu^\ell}^\alpha$. Since $0 \le \mu_j^\ell \le 1$, $\|D_{\mu^\ell}^\alpha\|_{\mathrm{op}} \le 1$. Therefore, by submultiplicativity of the operator norm, $\|\Delta_1^\ell\|_{\mathrm{op}} \le 1$, and $\|\Delta_2^\ell\|_{\mathrm{op}} \le 1$ for every ℓ. Hence $\mathcal{A}_\alpha(\mu) \subseteq \mathcal{A}_{\mathrm{Muon}}$ for every $\mu \in \mathcal{K}_K$, so $\mathcal{A}_{\alpha,K} \subseteq \mathcal{A}_{\mathrm{Muon}}$. Since the Muon unit ball is convex, $\mathcal{B}_{\alpha,K} \subseteq \mathcal{A}_{\mathrm{Muon}}$.

Conversely, consider the uniform allocation $\bar{\mu}_j^\ell = K/N$, which belongs to $\mathcal{K}_K$. Then $D_{\bar{\mu}^\ell}^\alpha = (K/N)^\alpha I$, and therefore $\mathcal{A}_\alpha(\bar{\mu}) = (K/N)^\alpha \mathcal{A}_{\mathrm{Muon}}$. Since $\mathcal{A}_\alpha(\bar{\mu}) \subseteq \mathcal{B}_{\alpha,K}$,

$$\left(\frac{K}{N}\right)^{\alpha} \mathcal{A}_{\mathrm{Muon}} \subseteq \mathcal{B}_{\alpha,K} \subseteq \mathcal{A}_{\mathrm{Muon}}. \tag{9}$$

It remains to verify the geometric properties of $\mathcal{B}_{\alpha,K}$. For compactness, let

$$\mathcal{X} := \mathcal{K}_K \times \prod_{\ell=1}^{L} \Big(\{P_1^\ell : \|P_1^\ell\|_{\mathrm{op}} \le 1\} \times \{P_2^\ell : \|P_2^\ell\|_{\mathrm{op}} \le 1\} \Big),$$

which is compact in finite dimension, and define $F(\mu, \{P_1^\ell, P_2^\ell\}) := ((D_{\mu^\ell}^\alpha P_1^\ell, P_2^\ell D_{\mu^\ell}^\alpha))_{\ell=1}^L$. The map F is continuous for $\alpha > 0$, and $\mathcal{A}_{\alpha,K} = F(\mathcal{X})$; hence $\mathcal{A}_{\alpha,K}$ is compact. Its convex hull $\mathcal{B}_{\alpha,K}$ is therefore compact in finite dimension and is convex by definition. Moreover, $\mathcal{A}_{\alpha,K}$ is centrally symmetric, since replacing every P_i^ℓ by $-P_i^\ell$ replaces Δ by $-\Delta$; the convex hull of a centrally symmetric set is centrally symmetric. Finally, the left inclusion in Eq. (9) shows that $\mathcal{B}_{\alpha,K}$ contains a neighborhood of the origin.

Thus the Minkowski functional of $\mathcal{B}_{\alpha,K}$ is a norm whose unit ball is $\mathcal{B}_{\alpha,K}$. Since a larger set has a smaller Minkowski functional, the two inclusions in Eq. (9) give $\|\Delta\|_{\mathrm{Muon}} \le \Omega_{\alpha,K}(\Delta) \le (N/K)^\alpha \|\Delta\|_{\mathrm{Muon}}$. □

The following Lemma gives a characterization of the norm $\Omega_{\alpha,K}$ in the regime $0 < \alpha \le \frac{1}{2}$.

Lemma 4 (Characterization of the BMuon norm)**.** *Let $0 < \alpha \le \frac{1}{2}$ and $0 < K \le N$. Then $\mathcal{A}_{\alpha,K}$ is convex, and hence $\mathcal{B}_{\alpha,K} = \mathcal{A}_{\alpha,K}$. For $\mu \in \mathcal{K}_K$, define*

$$\rho_\mu(\Delta) := \max_\ell \max \left\{ \|D_{\mu^\ell}^{-\alpha} \Delta_1^\ell\|_{\mathrm{op}}, \|\Delta_2^\ell D_{\mu^\ell}^{-\alpha}\|_{\mathrm{op}} \right\},$$

with the following convention: if $\mu_j^\ell = 0$, finiteness requires $\Delta_{1,j:}^\ell = 0$ and $\Delta_{2,:,j}^\ell = 0$; the corresponding zero rows and columns are then omitted when evaluating the two operator norms. If either is nonzero, set $\rho_\mu(\Delta) = +\infty$. Then

$$\Omega_{\alpha,K}(\Delta) = \min_{\mu \in \mathcal{K}_K} \rho_\mu(\Delta).$$

Proof. Set $\beta := 2\alpha \in (0, 1]$, and define

$$\varphi_\beta(z, s) := \begin{cases} z^2 s^{-\beta}, & s > 0, \\ 0, & (z, s) = (0, 0), \\ +\infty, & s = 0,\ z \ne 0. \end{cases}$$

For $s > 0$, the Hessian of φ_β is positive semidefinite: $\partial_{zz}^2 \varphi_\beta = 2s^{-\beta} > 0$ and

$$\det \nabla^2 \varphi_\beta(z, s) = 2\beta(1 - \beta) z^2 s^{-2\beta - 2} \ge 0,$$

since $\beta \in (0, 1]$. Hence φ_β is jointly convex on $\mathbb{R} \times (0, \infty)$.

It remains to check the boundary $s = 0$. The only boundary point at which φ_β is finite is $(0, 0)$. For $s > 0$ and $\theta \in [0, 1]$,

$$\varphi_\beta(\theta z, \theta s) = \theta^{2-\beta}\varphi_\beta(z, s) \le \theta\varphi_\beta(z, s),$$

because $2-\beta \ge 1$, which is exactly the convexity inequality between (z, s) and $(0, 0)$. If a boundary point has $s = 0$ and $z \neq 0$, its value is $+\infty$, so the convexity inequality is automatic. Thus φ_β is convex on $\mathbb{R} \times [0, \infty)$.

It is also lower semicontinuous. It is continuous for $s > 0$. At $(0, 0)$, nonnegativity gives

$$\liminf_{(z,s)\to(0,0)} \varphi_\beta(z, s) \ge 0 = \varphi_\beta(0, 0).$$

If $z_0 \neq 0$, then as $(z, s) \to (z_0, 0)$, we eventually have $|z| \ge |z_0|/2$. If $s = 0$, then $\varphi_\beta(z, s) = +\infty$; if $s > 0$, then

$$\varphi_\beta(z, s) = z^2 s^{-\beta} \ge \frac{z_0^2}{4} s^{-\beta} \longrightarrow +\infty.$$

Hence

$$\liminf_{(z,s)\to(z_0,0)} \varphi_\beta(z, s) = +\infty = \varphi_\beta(z_0, 0).$$

Therefore φ_β is lower semicontinuous on $\mathbb{R} \times [0, \infty)$.

For every block ℓ,

$$\begin{aligned}
\|D_{\mu^\ell}^{-\alpha}\Delta_1^\ell\|_{\mathrm{op}}^2 &= \sup_{\|x\|_2=1} \|D_{\mu^\ell}^{-\alpha}\Delta_1^\ell x\|_2^2 \\
&= \sup_{\|x\|_2=1} \sum_{j=1}^{m_\ell} (\mu_j^\ell)^{-2\alpha} \langle \Delta_{1,j:}^\ell, x\rangle^2 \\
&= \sup_{\|x\|_2=1} \sum_{j=1}^{m_\ell} \varphi_\beta(\langle \Delta_{1,j:}^\ell, x\rangle, \mu_j^\ell).
\end{aligned}$$

This interpretation agrees with the zero-allocation convention in the definition of ρ_μ. Indeed, if $\mu_j^\ell = 0$ and $\Delta_{1,j:}^\ell = 0$, then $\varphi_\beta(0, 0) = 0$, which amounts to omitting that row from the operator norm. If instead $\mu_j^\ell = 0$ and $\Delta_{1,j:}^\ell \neq 0$, there exists x such that $\langle \Delta_{1,j:}^\ell, x\rangle \neq 0$, and hence the supremum is $+\infty$, matching the zero-allocation convention for ρ_μ.

Similarly,

$$\|\Delta_2^\ell D_{\mu^\ell}^{-\alpha}\|_{\mathrm{op}}^2 = \sup_{\|y\|_2=1} \sum_{j=1}^{m_\ell} \varphi_\beta(\langle \Delta_{2,:,j}^\ell, y\rangle, \mu_j^\ell).$$

For fixed ℓ, j, and x, the map $(\Delta, \mu) \mapsto \varphi_\beta(\langle \Delta_{1,j:}^\ell, x\rangle, \mu_j^\ell)$ is jointly convex and lower semicontinuous, since φ_β has these properties and its arguments depend linearly on (Δ, μ). Finite sums preserve both properties, and so does taking the pointwise supremum over x. Hence $(\Delta, \mu) \mapsto \|D_{\mu^\ell}^{-\alpha}\Delta_1^\ell\|_{\mathrm{op}}^2$ is jointly convex and lower semicontinuous. The same argument applies to $(\Delta, \mu) \mapsto \|\Delta_2^\ell D_{\mu^\ell}^{-\alpha}\|_{\mathrm{op}}^2$. Finally, $g(\Delta, \mu) := \rho_\mu(\Delta)^2$ is the finite maximum of these functions over the blocks ℓ, and is therefore jointly convex and lower semicontinuous.

We now show that

$$\mathcal{A}_\alpha(\mu) = \{\Delta : \rho_\mu(\Delta) \le 1\}. \tag{10}$$

If $\Delta \in \mathcal{A}_\alpha(\mu)$, then there exist $\{P_1^\ell, P_2^\ell\}_{\ell=1}^L$ such that, for every ℓ, $\Delta_1^\ell = D_{\mu^\ell}^\alpha P_1^\ell$ and $\Delta_2^\ell = P_2^\ell D_{\mu^\ell}^\alpha$, with $\|P_1^\ell\|_{\mathrm{op}}, \|P_2^\ell\|_{\mathrm{op}} \le 1$. Let $S_\ell := \{j : \mu_j^\ell > 0\}$. If $j \notin S_\ell$, then $\Delta_{1,j:}^\ell = \Delta_{2,:,j}^\ell = 0$. In particular, if $S_\ell = \varnothing$, then $\Delta_1^\ell = \Delta_2^\ell = 0$, so both operator norms appearing in ρ_μ for this block are zero under the zero-allocation convention.

If $S_\ell \neq \varnothing$, then

$$D_{\mu_{S_\ell}^\ell}^{-\alpha}\Delta_{1,S_\ell:}^\ell = P_{1,S_\ell:}^\ell, \qquad \Delta_{2,:,S_\ell}^\ell D_{\mu_{S_\ell}^\ell}^{-\alpha} = P_{2,:,S_\ell}^\ell.$$

Removing rows or columns cannot increase the operator norm, so $\|P_{1,S_\ell:}^\ell\|_{\mathrm{op}} \le \|P_1^\ell\|_{\mathrm{op}} \le 1$ and $\|P_{2,:,S_\ell}^\ell\|_{\mathrm{op}} \le \|P_2^\ell\|_{\mathrm{op}} \le 1$. By the zero-allocation convention defining ρ_μ, the rows and

columns with $\mu_j^\ell = 0$, which are zero in Δ, are omitted from the operator-norm computation. Hence $\|D_{\mu^\ell}^{-\alpha}\Delta_1^\ell\|_{\mathrm{op}} = \|D_{\mu_{S_\ell}^\ell}^{-\alpha}\Delta_{1,S_\ell:}^\ell\|_{\mathrm{op}} = \|P_{1,S_\ell:}^\ell\|_{\mathrm{op}} \le 1$, and similarly $\|\Delta_2^\ell D_{\mu^\ell}^{-\alpha}\|_{\mathrm{op}} \le 1$. Since this holds for every ℓ, we conclude $\rho_\mu(\Delta) \le 1$.

Conversely, if $\rho_\mu(\Delta) \le 1$, finiteness forces $\Delta_{1,j:}^\ell = \Delta_{2,:,j}^\ell = 0$ whenever $j \notin S_\ell$. Define $P_{1,S_\ell:}^\ell = D_{\mu_{S_\ell}^\ell}^{-\alpha}\Delta_{1,S_\ell:}^\ell$ and $P_{2,:,S_\ell}^\ell = \Delta_{2,:,S_\ell}^\ell D_{\mu_{S_\ell}^\ell}^{-\alpha}$, and set the remaining rows of P_1^ℓ and columns of P_2^ℓ to zero. By definition of ρ_μ, $\|P_1^\ell\|_{\mathrm{op}} \le 1$ and $\|P_2^\ell\|_{\mathrm{op}} \le 1$, while $\Delta_1^\ell = D_{\mu^\ell}^{\alpha} P_1^\ell$ and $\Delta_2^\ell = P_2^\ell D_{\mu^\ell}^{\alpha}$. Thus $\Delta \in \mathcal{A}_\alpha(\mu)$.

Therefore, Eq. (10) is proved.

Now take $\Delta, \Delta' \in \mathcal{A}_{\alpha,K}$. By definition of $\mathcal{A}_{\alpha,K}$, there exist $\mu, \mu' \in \mathcal{K}_K$ such that $\Delta \in \mathcal{A}_\alpha(\mu)$ and $\Delta' \in \mathcal{A}_\alpha(\mu')$. For $\theta \in [0,1]$, set $\mu_\theta = \theta\mu + (1-\theta)\mu' \in \mathcal{K}_K$. Joint convexity of $g(\Delta, \mu) = \rho_\mu(\Delta)^2$ gives

$$g(\theta\Delta + (1-\theta)\Delta', \mu_\theta) \le \theta g(\Delta, \mu) + (1-\theta) g(\Delta', \mu') \le 1.$$

Hence $\theta\Delta + (1-\theta)\Delta' \in \mathcal{A}_\alpha(\mu_\theta) \subseteq \mathcal{A}_{\alpha,K}$. Therefore $\mathcal{A}_{\alpha,K}$ is convex, and thus $\mathcal{B}_{\alpha,K} = \mathcal{A}_{\alpha,K}$.

Using the definition of the Minkowski functional and the positive homogeneity of ρ_μ,

$$\begin{aligned}\Omega_{\alpha,K}(\Delta) &= \inf\{t > 0 : \Delta/t \in \mathcal{A}_{\alpha,K}\} \\ &= \inf\{t > 0 : \exists\, \mu \in \mathcal{K}_K,\ \rho_\mu(\Delta) \le t\} \\ &= \inf_{\mu \in \mathcal{K}_K} \rho_\mu(\Delta).\end{aligned}$$

Finally, for fixed Δ, $g(\Delta, \mu)$ is lower semicontinuous in μ. Since $\mathcal{K}_K$ is compact, it attains its minimum, and the same allocation minimizes $\rho_\mu(\Delta) = \sqrt{g(\Delta, \mu)}$. Thus the last infimum is a minimum. □

B.5 Proofs for the Geometric Special Cases

We prove the special cases stated in Section 2.4.

Recovery of Muon. If $K = N$, then $0 \le \mu_j^\ell \le 1$ and $\sum_{\ell,j} \mu_j^\ell = N$ imply $\mu_j^\ell = 1$ for every channel. Hence $D_{\mu^\ell}^{\alpha} = I$ for every ℓ, so $\mathcal{A}_{\alpha,N} = \mathcal{A}_{\mathrm{Muon}}$. Since $\mathcal{A}_{\mathrm{Muon}}$ is convex, $\mathcal{B}_{\alpha,N} = \mathcal{A}_{\mathrm{Muon}}$, and therefore $\Omega_{\alpha,N} = \|\cdot\|_{\mathrm{Muon}}$.

Likewise, under the extension $D_\mu^0 := I$, $\mathcal{A}_0(\mu) = \mathcal{A}_{\mathrm{Muon}}$ for every $\mu \in \mathcal{K}_K$. Thus $\mathcal{B}_{0,K} = \mathcal{A}_{\mathrm{Muon}}$ and $\Omega_{0,K} = \|\cdot\|_{\mathrm{Muon}}$.

Structured K-support geometry at $\alpha = 1$. Assume K is an integer and let $\mathcal{V}_K := \{\mathbf{1}_S : |S| = K\}$, where $S \subseteq \{(\ell, j) : 1 \le \ell \le L,\ 1 \le j \le m_\ell\}$ and $\mathbf{1}_S^\ell$ denotes the block-ℓ component of $\mathbf{1}_S$, with $(\mathbf{1}_S^\ell)_j = \mathbf{1}\{(\ell, j) \in S\}$. As shown in Appendix B.3, the extreme points of $\mathcal{K}_K$ are exactly $\mathcal{V}_K$. Since $\mathcal{K}_K$ is a polytope, $\mathcal{K}_K = \mathrm{conv}(\mathcal{V}_K)$.

Let $\mu \in \mathcal{K}_K$, it can be written as $\mu = \sum_r \lambda_r \mathbf{1}_{S_r}$, where $\lambda_r \ge 0$, $\sum_r \lambda_r = 1$, and $|S_r| = K$. Now let $\Delta \in \mathcal{A}_1(\mu)$. Then, for some P_1^ℓ, P_2^ℓ satisfying $\|P_1^\ell\|_{\mathrm{op}}, \|P_2^\ell\|_{\mathrm{op}} \le 1$, $\Delta = \big(D_{\mu^\ell} P_1^\ell,\ P_2^\ell D_{\mu^\ell}\big)_{\ell=1}^L$. Since D_{μ^ℓ} is linear in μ^ℓ, $\Delta = \sum_r \lambda_r \big(D_{\mathbf{1}_{S_r}^\ell} P_1^\ell,\ P_2^\ell D_{\mathbf{1}_{S_r}^\ell}\big)_{\ell=1}^L$, and each term belongs to $\mathcal{A}_1(\mathbf{1}_{S_r})$. Therefore

$$\bigcup_{\mu \in \mathcal{K}_K} \mathcal{A}_1(\mu) \subseteq \mathrm{conv}\left(\bigcup_{|S|=K} \mathcal{A}_1(\mathbf{1}_S)\right).$$

Conversely, $|S| = K$ implies $\mathbf{1}_S \in \mathcal{K}_K$. Hence, after taking convex hulls,

$$\mathcal{B}_{1,K} = \mathrm{conv}\left(\bigcup_{\mu \in \mathcal{K}_K} \mathcal{A}_1(\mu)\right) = \mathrm{conv}\left(\bigcup_{|S|=K} \mathcal{A}_1(\mathbf{1}_S)\right).$$

For a binary allocation $\mathbf{1}_S$,

$$\begin{aligned}\mathcal{A}_1(\mathbf{1}_S) &= \Big\{\Delta:\ \Delta_1^\ell = D_{\mathbf{1}_S^\ell} P_1^\ell,\ \Delta_2^\ell = P_2^\ell D_{\mathbf{1}_S^\ell},\ \|P_1^\ell\|_{\mathrm{op}} \le 1,\ \|P_2^\ell\|_{\mathrm{op}} \le 1,\ \forall \ell\Big\} \\ &= \Big\{\Delta:\ \Delta_{1,j:}^\ell = 0,\ \Delta_{2,:,j}^\ell = 0 \text{ for all } (\ell, j) \notin S,\ \|\Delta_1^\ell\|_{\mathrm{op}} \le 1,\ \|\Delta_2^\ell\|_{\mathrm{op}} \le 1,\ \forall \ell\Big\} \\ &= \Big\{\Delta:\ \mathrm{supp}_{\mathrm{ch}}(\Delta) \subseteq S,\ \|\Delta_1^\ell\|_{\mathrm{op}} \le 1,\ \|\Delta_2^\ell\|_{\mathrm{op}} \le 1,\ \forall \ell\Big\}.\end{aligned}$$

The second equality uses that multiplication by $D_{\mathbf{1}_S^\ell}$ zeros exactly the channels outside S and does not increase the operator norm; conversely, one may take $P_i^\ell = \Delta_i^\ell$.

Consequently,

$$\begin{aligned}\mathcal{B}_{1,K} &= \mathrm{conv}\left(\bigcup_{|S|=K} \left\{\Delta : \mathrm{supp}_{\mathrm{ch}}(\Delta) \subseteq S,\ \|\Delta_1^\ell\|_{\mathrm{op}} \le 1,\ \|\Delta_2^\ell\|_{\mathrm{op}} \le 1,\ \forall \ell\right\}\right) \\ &= \mathrm{conv}\Big\{\Delta : |\mathrm{supp}_{\mathrm{ch}}(\Delta)| \le K,\ \|\Delta_1^\ell\|_{\mathrm{op}} \le 1,\ \|\Delta_2^\ell\|_{\mathrm{op}} \le 1,\ \forall \ell\Big\}.\end{aligned}$$

Explicit norm for $\alpha = K = 1$. Since $\mathcal{B}_{\alpha,K}$ is the unit ball of $\Omega_{\alpha,K}$,

$$\Omega_{\alpha,K}^*(M) = \max_{\Delta \in \mathcal{B}_{\alpha,K}} \langle M, \Delta \rangle = \max_{\mu \in \mathcal{K}_K} \Phi_M(\mu).$$

As shown in Appendix B.3, at $\alpha = 1$ a maximizer can be chosen at an extreme point of $\mathcal{K}_K$. For $K = 1$, these extreme points are exactly the binary vectors with a single nonzero entry. Therefore

$$\begin{aligned}\Omega_{1,1}^*(M) &= \max_{\mu \in \mathcal{K}_1} \sum_{r=1}^{L} \left[\|D_{\mu^r} M_1^r\|_* + \|M_2^r D_{\mu^r}\|_*\right] \\ &= \max_{\ell,j} \sum_{r=1}^{L} \left[\|D_{\mathbf{1}^r_{\{(\ell,j)\}}} M_1^r\|_* + \|M_2^r D_{\mathbf{1}^r_{\{(\ell,j)\}}}\|_*\right] \\ &= \max_{\ell,j} \left[\|D_{\mathbf{1}_{\{j\}}} M_1^\ell\|_* + \|M_2^\ell D_{\mathbf{1}_{\{j\}}}\|_*\right] \\ &= \max_{\ell,j} \left[\|M_{1,j:}^\ell\|_2 + \|M_{2,:,j}^\ell\|_2\right].\end{aligned}$$

The last equality holds because the two matrices in the third equality have respectively a single nonzero row and a single nonzero column, and the nuclear norm of each equals the Euclidean norm of its nonzero row or column.

Dualizing gives the result

$$\Omega_{1,1}(\Delta) = \sum_{\ell=1}^{L} \sum_{j=1}^{m_\ell} \max\left\{\|\Delta_{1,j:}^\ell\|_2,\ \|\Delta_{2,:,j}^\ell\|_2\right\}.$$

C Compact-Representation Theory

Appendix C.1 proves Theorem 2, showing that a sufficiently scarce BMuon budget forces every minimum-cost linear factorization to have minimal hidden width, whereas Muon admits minimum-cost representations at full width. Appendix C.2 gives a complementary weight-decay result: channel norms are controlled by their allocation histories, and channels whose allocations vanish are driven to zero.

C.1 Intrinsic-Width Recovery

Let $X = (X_\ell)_{\ell=1}^L$, where $X_\ell \in \mathbb{R}^{p_\ell \times d_\ell}$ and $X_\ell \neq 0$ for every ℓ. Let $r_\ell := \operatorname{rank}(X_\ell) \geq 1$, and let $\sigma_{\ell,1} \geq \cdots \geq \sigma_{\ell,r_\ell} > 0$ denote the nonzero singular values of X_ℓ. Set $\sigma_\star := \max_{\ell,i} \sigma_{\ell,i}$.

Each block ℓ consists of $W_1^\ell \in \mathbb{R}^{m_\ell \times d_\ell}$ and $W_2^\ell \in \mathbb{R}^{p_\ell \times m_\ell}$. Assume $m_\ell \geq r_\ell$ for every ℓ. We consider the linear, bias-free setting, so block ℓ represents the map $W_2^\ell W_1^\ell$. Let $N := \sum_{\ell=1}^L m_\ell$ denote the total number of hidden channels across blocks. An exact factorization W satisfies $W_2^\ell W_1^\ell = X_\ell$ for every ℓ.

Recall that, for $W = \left((W_1^\ell, W_2^\ell)\right)_{\ell=1}^L$, $\|W\|_{\mathrm{Muon}} := \max_{1\leq \ell \leq L} \max\left\{\|W_1^\ell\|_{\mathrm{op}}, \|W_2^\ell\|_{\mathrm{op}}\right\}$, and define

$$\mathcal{R}_{\mathrm{Muon}}(X) := \inf_{\substack{W_2^\ell W_1^\ell = X_\ell \\ \forall \ell}} \|W\|_{\mathrm{Muon}}^2, \qquad \mathcal{R}_{\alpha,K}(X) := \inf_{\substack{W_2^\ell W_1^\ell = X_\ell \\ \forall \ell}} \Omega_{\alpha,K}(W)^2.$$

For an exact factorization W, let $a_\ell(W)$ denote the number of channels j such that $W_{1,j:}^\ell \neq 0$ or $W_{2,:,j}^\ell \neq 0$. Finally, recall $d_{\mathrm{eff}}^{(\alpha)}(X) := \sum_{\ell=1}^L \sum_{i=1}^{r_\ell} (\sigma_{\ell,i}/\sigma_\star)^{1/(2\alpha)}$.

Theorem 2 (restated). Let $0 < \alpha \leq \frac{1}{2}$ and $0 < K \leq N$. Both representation-cost infima are attained, and

$$\mathcal{R}_{\mathrm{Muon}}(X) = \sigma_\star, \qquad \mathcal{R}_{\alpha,K}(X) = \sigma_\star \max\left\{1, \left(\frac{d_{\mathrm{eff}}^{(\alpha)}(X)}{K}\right)^{2\alpha}\right\}.$$

Moreover, Muon admits a minimizer $W^\star$ satisfying $a_\ell(W^\star) = m_\ell$ for every ℓ, whereas if $K \leq d_{\mathrm{eff}}^{(\alpha)}(X)$, every BMuon minimizer $W^\star$ satisfies $a_\ell(W^\star) = r_\ell$ for every ℓ.

For $\mu \in \mathcal{K}_K$, recall the fixed-allocation radius

$$\rho_\mu(W) := \max_{1\leq \ell \leq L} \max\left\{\|D_{\mu^\ell}^{-\alpha} W_1^\ell\|_{\mathrm{op}}, \|W_2^\ell D_{\mu^\ell}^{-\alpha}\|_{\mathrm{op}}\right\},$$

with the zero-allocation convention of Lemma 4. By that lemma, $\Omega_{\alpha,K}(W) = \min_{\mu \in \mathcal{K}_K} \rho_\mu(W)$.

We first consider the following single-block case.

Lemma 5 (Fixed-allocation factorization)**.** *Let $\alpha > 0$. Consider a single block ($L = 1$). Let $X \in \mathbb{R}^{p\times d}$ have rank $r \geq 1$ and nonzero singular values $\sigma_1 \geq \cdots \geq \sigma_r > 0$. For $\mu \in [0,1]^m$, $m \geq r$, let $\mu_{(1)} \geq \cdots \geq \mu_{(m)}$ denote its ordered coordinates. For $W = (W_1, W_2)$, define*

$$\rho_\mu(W) := \max\left\{\|D_\mu^{-\alpha} W_1\|_{\mathrm{op}}, \|W_2 D_\mu^{-\alpha}\|_{\mathrm{op}}\right\},$$

with the zero-allocation convention of Lemma 4. Then

$$\inf_{W_2 W_1 = X} \rho_\mu(W)^2 = \max_{1\leq i\leq r} \frac{\sigma_i}{\mu_{(i)}^{2\alpha}}.$$

The right-hand side is $+\infty$ if $\mu_{(i)} = 0$ for some $1 \leq i \leq r$. Whenever it is finite, the infimum is attained.

Proof. If $\mu_{(r)} = 0$, then fewer than r coordinates satisfy $\mu_j > 0$. Suppose a $W = (W_1, W_2)$ is an exact factorization of X, (i.e. $W_2 W_1 = X$) having finite fixed-allocation radius $\rho_\mu(W)$. Finiteness

of the radius forces $W_{1,j:} = 0$ and $W_{2,:,j} = 0$ whenever $\mu_j = 0$ (convention of Lemma 4). Hence $X = \sum_{j:\mu_j>0} W_{2,:,j} W_{1,j:}$ is a sum of fewer than r rank-one matrices. Therefore $\mathrm{rank}(X) < r$, contradicting $\mathrm{rank}(X) = r$. Thus, when $\mu_{(r)} = 0$, every W satisfying exact factorization has $\rho_\mu(W) = +\infty$. Since the right-hand side is also $+\infty$ by convention when $\mu_{(r)} = 0$, the claimed equality holds in this case.

Assume now $\mu_{(r)} > 0$, and let $W = (W_1, W_2)$ be an exact factorization of X, i.e. $W_2 W_1 = X$, with finite fixed-allocation radius $\rho_\mu(W) < \infty$. Let $S := \{j : \mu_j > 0\}$. Finite radius implies $W_{1,S^c:} = 0$ and $W_{2,:,S^c} = 0$ (convention of Lemma 4), hence $X = \big(W_{2,:,S} D_{\mu_S}^{-\alpha}\big) D_{\mu_S}^{2\alpha} \big(D_{\mu_S}^{-\alpha} W_{1,S:}\big)$. Using $\sigma_i(ABC) \le \|A\|_{\mathrm{op}} \sigma_i(B) \|C\|_{\mathrm{op}}$, we obtain, for $i \le r$,

$$\sigma_i \le \max\left\{\|D_\mu^{-\alpha} W_1\|_{\mathrm{op}}^2,\, \|W_2 D_\mu^{-\alpha}\|_{\mathrm{op}}^2\right\} \mu_{(i)}^{2\alpha}.$$

The weighted operator norms are interpreted according to the zero-allocation convention in the definition of ρ_μ in Lemma 4.

So, for any exact factorization with finite fixed-allocation radius, for every $i \le r$, $\rho_\mu(W)^2 \ge \frac{\sigma_i}{\mu_{(i)}^{2\alpha}}$, hence

$$\rho_\mu(W)^2 \ge \max_{1\le i\le r} \frac{\sigma_i}{\mu_{(i)}^{2\alpha}}.$$

The same bound is trivial when $\rho_\mu(W) = +\infty$. Taking the infimum over all exact factorizations W gives

$$\inf_{W_2 W_1 = X} \rho_\mu(W)^2 \ge \max_{1\le i\le r} \frac{\sigma_i}{\mu_{(i)}^{2\alpha}}.$$

For the reverse inequality, permutation of the hidden coordinates leaves both $W_2 W_1$ and $\|D_\mu^{-\alpha} W_1\|_{\mathrm{op}}, \|W_2 D_\mu^{-\alpha}\|_{\mathrm{op}}$ unchanged. So assume without loss of generality that $\mu_1 \ge \cdots \ge \mu_m$. Let $X = U \,\mathrm{diag}(\sigma_1, \ldots, \sigma_r) V^\top$ be a compact SVD. Set $W_{1,i:} = \sqrt{\sigma_i}\, v_i^\top$ and $W_{2,:,i} = \sqrt{\sigma_i}\, u_i$ for $i \le r$, with all remaining rows and columns equal to zero. Then $W_2 W_1 = X$, and since U and V have orthonormal columns, the nonzero singular values of both $D_\mu^{-\alpha} W_1$ and $W_2 D_\mu^{-\alpha}$ are $\{\sqrt{\sigma_i}/\mu_i^\alpha\}_{i=1}^r$. Hence

$$\|D_\mu^{-\alpha} W_1\|_{\mathrm{op}}^2 = \|W_2 D_\mu^{-\alpha}\|_{\mathrm{op}}^2 = \max_{1\le i\le r} \frac{\sigma_i}{\mu_i^{2\alpha}},$$

which gives the reverse inequality and completes the proof. □

We now prove Theorem 2.

Proof. We first consider Muon. Recall that an exact factorization $W = (W^\ell)_{\ell=1}^L$, with $W^\ell = (W_1^\ell, W_2^\ell)$, satisfies $W_2^\ell W_1^\ell = X_\ell$ for every $\ell \in \{1, \ldots, L\}$. For any such factorization W,

$$\sigma_{\ell,1} = \|X_\ell\|_{\mathrm{op}} \le \|W_2^\ell\|_{\mathrm{op}} \|W_1^\ell\|_{\mathrm{op}} \le \|W\|_{\mathrm{Muon}}^2,$$

hence $\|W\|_{\mathrm{Muon}}^2 \ge \sigma_\star$.

To attain this bound at full width, let $X_\ell = U_\ell \Sigma_\ell V_\ell^\top$ be a compact SVD and choose $A_\ell \in \mathbb{R}^{m_\ell \times r_\ell}$ such that $A_\ell^\top A_\ell = I$ and every row of A_ℓ is nonzero. For example, take its first column to be $m_\ell^{-1/2}\mathbf{1}$ and complete it to an orthonormal family. Define $W_1^\ell = A_\ell \Sigma_\ell^{1/2} V_\ell^\top$ and $W_2^\ell = U_\ell \Sigma_\ell^{1/2} A_\ell^\top$, then $W_2^\ell W_1^\ell = X_\ell$ and $\|W_1^\ell\|_{\mathrm{op}}^2 = \|W_2^\ell\|_{\mathrm{op}}^2 = \sigma_{\ell,1}$. Thus $\mathcal{R}_{\mathrm{Muon}}(X) = \sigma_\star$. Since every row of A_ℓ is nonzero and Σ_ℓ has strictly positive diagonal entries, every row of W_1^ℓ and every column of W_2^ℓ is nonzero. Hence this minimizer satisfies $a_\ell(W) = m_\ell$ for every ℓ, which concludes the Muon part of the proof.

We now turn to BMuon. Lemma 4 implies:

$$\mathcal{R}_{\alpha,K}(X) = \inf_{\mu\in\mathcal{K}_K} \inf_{\substack{W_2^\ell W_1^\ell = X_\ell \\ \forall \ell}} \rho_\mu(W)^2. \tag{11}$$

For fixed μ, denote

$$c_\ell := \inf_{W_2^\ell W_1^\ell = X_\ell} \max\left\{\|D_{\mu^\ell}^{-\alpha} W_1^\ell\|_{\mathrm{op}}^2,\ \|W_2^\ell D_{\mu^\ell}^{-\alpha}\|_{\mathrm{op}}^2\right\}.$$

We want to prove that, for this fixed allocation μ,

$$\inf_{\substack{W_2^\ell W_1^\ell = X_\ell \\ \forall \ell}} \rho_\mu(W)^2 = \max_{1\le \ell \le L} c_\ell. \tag{12}$$

For every exact factorization W and every $\ell \in \{1, \dots, L\}$,

$$\max\left\{\|D_{\mu^\ell}^{-\alpha} W_1^\ell\|_{\mathrm{op}}^2,\ \|W_2^\ell D_{\mu^\ell}^{-\alpha}\|_{\mathrm{op}}^2\right\} \ge c_\ell.$$

Hence $\rho_\mu(W)^2 \ge \max_{1\le \ell \le L} c_\ell$, and thus

$$\inf_{\substack{W_2^\ell W_1^\ell = X_\ell \\ \forall \ell}} \rho_\mu(W)^2 \ge \max_{1\le \ell \le L} c_\ell.$$

Conversely, suppose first that every $c_\ell < \infty$. Lemma 5 shows that, for every ℓ, the infimum defining c_ℓ is attained by some exact factorization $W^{\ell,\star}$. Let $W^\star := \left(W^{\ell,\star}\right)_{\ell=1}^L$, we have $\rho_\mu(W^\star)^2 = \max_{1\le \ell \le L} c_\ell$. Therefore

$$\inf_{\substack{W_2^\ell W_1^\ell = X_\ell \\ \forall \ell}} \rho_\mu(W)^2 \le \max_{1\le \ell \le L} c_\ell.$$

The same inequality holds if for some ℓ, $c_\ell = +\infty$. Hence, in all cases, Eq. (12) holds.

For this fixed μ, let $\mu_{(1)}^\ell \ge \cdots \ge \mu_{(m_\ell)}^\ell$ denote the ordered coordinates of μ^ℓ. By Lemma 5, $c_\ell = \max_{1\le i\le r_\ell} \frac{\sigma_{\ell,i}}{(\mu_{(i)}^\ell)^{2\alpha}}$.

Therefore, combining Eqs. (11) and (12),

$$\mathcal{R}_{\alpha,K}(X) = \inf_{\mu\in\mathcal{K}_K} \max_{1\le\ell\le L} \max_{1\le i\le r_\ell} \frac{\sigma_{\ell,i}}{(\mu_{(i)}^\ell)^{2\alpha}}.$$

Set $q := 1/(2\alpha)$. For any $\mu \in \mathcal{K}_K$, denote $M(\mu) := \max_{1\le\ell\le L} \max_{1\le i\le r_\ell} \frac{\sigma_{\ell,i}}{(\mu_{(i)}^\ell)^{2\alpha}}$.

Assume first that $M(\mu) < \infty$. Since $\mu_{(i)}^\ell \le 1$, we have $M(\mu) \ge \sigma_{\ell,i}$ for every ℓ and $i \le r_\ell$. Hence $M(\mu) \ge \sigma_\star$. Moreover, for every ℓ and $i \le r_\ell$, $M(\mu) \ge \sigma_{\ell,i}/(\mu_{(i)}^\ell)^{2\alpha}$ implies $\mu_{(i)}^\ell \ge \left(\frac{\sigma_{\ell,i}}{M(\mu)}\right)^q$. Therefore

$$K = \sum_{\ell=1}^L \sum_{j=1}^{m_\ell} \mu_j^\ell \ge \sum_{\ell=1}^L \sum_{i=1}^{r_\ell} \mu_{(i)}^\ell \ge \frac{1}{M(\mu)^q} \sum_{\ell=1}^L \sum_{i=1}^{r_\ell} \sigma_{\ell,i}^q,$$

and thus

$$M(\mu) \ge \left(\frac{\sum_{\ell,i} \sigma_{\ell,i}^q}{K}\right)^{1/q}.$$

Hence $M(\mu) \ge \max\left\{\sigma_\star,\ \left(\frac{\sum_{\ell,i} \sigma_{\ell,i}^q}{K}\right)^{1/q}\right\}$, which also holds directly when $M(\mu) = +\infty$.

Since this holds for every $\mu \in \mathcal{K}_K$, and $\mathcal{R}_{\alpha,K}(X) = \inf_{\mu\in\mathcal{K}_K} M(\mu)$, we obtain

$$\mathcal{R}_{\alpha,K}(X) \ge \max\left\{\sigma_\star,\ \left(\frac{\sum_{\ell,i} \sigma_{\ell,i}^q}{K}\right)^{1/q}\right\}. \tag{13}$$

We now prove the reverse inequality. Set $T := \max\left\{\sigma_\star,\ \left(\frac{\sum_{\ell,i} \sigma_{\ell,i}^q}{K}\right)^{1/q}\right\}$. For each block ℓ, set $\mu_i^\ell = (\sigma_{\ell,i}/T)^q$ for $1 \le i \le r_\ell$, and set the remaining coordinates to zero. Since $T \ge \sigma_\star$, all these

coordinates lie in $[0,1]$. Moreover,

$$\sum_{\ell=1}^{L}\sum_{i=1}^{r_\ell}\mu_i^\ell = \frac{\sum_{\ell,i}\sigma_{\ell,i}^q}{T^q} \le K.$$

Let $S := \sum_{\ell=1}^{L}\sum_{i=1}^{r_\ell}\mu_i^\ell$. The current allocation has total mass S, so the amount still to be added to obtain an allocation in $\mathcal{K}_K$ is $K-S$. The total mass that can be added without violating the constraint $\mu_j^\ell \le 1$ is $\sum_{\ell=1}^{L}\sum_{j=1}^{m_\ell}(1-\mu_j^\ell) = N-S$. Since $K \le N$, we have $K-S \le N-S$. Hence we can increase the coordinates, without making any of them exceed 1, until the total mass becomes K. Denote the resulting allocation by $\widetilde{\mu}$. Then $\widetilde{\mu} \in \mathcal{K}_K$, and since $\widetilde{\mu}_j^\ell \ge \mu_j^\ell$ for every ℓ, j,

$$\widetilde{\mu}_{(i)}^\ell \ge \mu_{(i)}^\ell = \left(\frac{\sigma_{\ell,i}}{T}\right)^q, \qquad 1 \le i \le r_\ell.$$

We have therefore constructed an allocation $\widetilde{\mu} \in \mathcal{K}_K$ such that

$$M(\widetilde{\mu}) = \max_{1\le\ell\le L}\max_{1\le i\le r_\ell}\frac{\sigma_{\ell,i}}{(\widetilde{\mu}_{(i)}^\ell)^{2\alpha}} \le T.$$

Hence

$$\mathcal{R}_{\alpha,K}(X) \le T. \tag{14}$$

By Eq. (13) and Eq. (14),

$$\mathcal{R}_{\alpha,K}(X) = \max\left\{\sigma_\star,\ \left(\frac{\sum_{\ell,i}\sigma_{\ell,i}^q}{K}\right)^{1/q}\right\}.$$

Since $\sum_{\ell,i}\sigma_{\ell,i}^q = \sigma_\star^q d_{\text{eff}}^{(\alpha)}(X)$ and $1/q = 2\alpha$, we obtain

$$\mathcal{R}_{\alpha,K}(X) = \sigma_\star \max\left\{1,\ \left(\frac{d_{\text{eff}}^{(\alpha)}(X)}{K}\right)^{2\alpha}\right\}.$$

We now prove attainment. Let $\widetilde{\mu} \in \mathcal{K}_K$ be the allocation constructed above. By Lemma 5, for every block ℓ, there exists an exact factorization $W^{\ell,\star}$ attaining $\inf_{W_2^\ell W_1^\ell = X_\ell}\max\left\{\|D_{\widetilde{\mu}^\ell}^{-\alpha}W_1^\ell\|_{\text{op}}^2,\ \|W_2^\ell D_{\widetilde{\mu}^\ell}^{-\alpha}\|_{\text{op}}^2\right\}$. Let $W^\star := \left(W^{\ell,\star}\right)_{\ell=1}^{L}$, then $W^\star$ is an exact factorization and

$$\rho_{\widetilde{\mu}}(W^\star)^2 = \max_{1\le\ell\le L}\max_{1\le i\le r_\ell}\frac{\sigma_{\ell,i}}{(\widetilde{\mu}_{(i)}^\ell)^{2\alpha}} \le T = \mathcal{R}_{\alpha,K}(X).$$

Since $\Omega_{\alpha,K}(W^\star) = \min_{\nu\in\mathcal{K}_K}\rho_\nu(W^\star)$ (Lemma 4),

$$\mathcal{R}_{\alpha,K}(X) \le \Omega_{\alpha,K}(W^\star)^2 \le \rho_{\widetilde{\mu}}(W^\star)^2 \le \mathcal{R}_{\alpha,K}(X).$$

Hence all these terms are equal, and $W^\star$ attains the infimum defining $\mathcal{R}_{\alpha,K}(X)$.

We now prove intrinsic-width recovery. Assume $K \le d_{\text{eff}}^{(\alpha)}(X)$. Then $\mathcal{R}_{\alpha,K}(X) = \left(\frac{\sum_{\ell,i}\sigma_{\ell,i}^q}{K}\right)^{1/q}$.

Let $W^\star$ be any minimizer of $\mathcal{R}_{\alpha,K}(X)$. By Lemma 4, there exists $\mu^\star \in \mathcal{K}_K$ such that $\Omega_{\alpha,K}(W^\star) = \rho_{\mu^\star}(W^\star)$. Since $W^\star$ is a minimizer, $\rho_{\mu^\star}(W^\star)^2 = \mathcal{R}_{\alpha,K}(X)$. By Lemma 5 applied to each block ℓ,

$$\max_{1\le i\le r_\ell}\frac{\sigma_{\ell,i}}{(\mu_{(i)}^{\ell,\star})^{2\alpha}} \le \mathcal{R}_{\alpha,K}(X).$$

Thus, for every $i \le r_\ell$,

$$\mu_{(i)}^{\ell,\star} \ge \left(\frac{\sigma_{\ell,i}}{\mathcal{R}_{\alpha,K}(X)}\right)^q.$$

Therefore

$$K = \sum_{\ell,i}\left(\frac{\sigma_{\ell,i}}{\mathcal{R}_{\alpha,K}(X)}\right)^q \le \sum_{\ell=1}^{L}\sum_{i=1}^{r_\ell}\mu_{(i)}^{\ell,\star} \le \sum_{\ell=1}^{L}\sum_{j=1}^{m_\ell}\mu_j^{\ell,\star} = K.$$

Hence all the terms are equal and $\mu^{\ell,\star}_{(j)} = 0$ for every $j > r_\ell$. Since $\rho_{\mu^\star}(W^\star) < \infty$, every channel with zero allocation has both its corresponding row of $W_1^{\ell,\star}$ and column of $W_2^{\ell,\star}$ equal to zero (convention of Lemma 4). Therefore

$$a_\ell(W^\star) \le r_\ell.$$

On the other hand, $X_\ell = \sum_{j:\text{ channel } j \text{ active}} W^{\ell,\star}_{2,:,j} W^{\ell,\star}_{1,j:}$ is a sum of at most $a_\ell(W^\star)$ rank-one matrices. Hence $r_\ell = \operatorname{rank}(X_\ell) \le a_\ell(W^\star)$. Thus

$$a_\ell(W^\star) = r_\ell \qquad \text{for every } \ell,$$

which completes the proof. □

C.2 BMuon with Weight-Decay

We show in this section that, when BMuon is combined with decoupled weight decay, its channel allocations have a direct effect on the learned weights. Consider the BMuon oracle with fixed budget K and decoupled weight decay,

$$W^\ell_{i,t+1} = (1 - \eta_t\lambda) W^\ell_{i,t} - \eta_t \Delta^{\ell,\star}_{i,t},$$

where $\lambda > 0$, $\eta_t \ge 0$, and $0 \le \lambda\eta_t < 1$. At each step t, let $\mu_t^\star \in \mathcal{K}_K$ denote a solution of Eq. (4). Then, by Eq. (5),

$$\Delta^{\ell,\star}_{1,t} = D^\alpha_{\mu^{\ell,\star}_t} \operatorname{Polar}\Big(D^\alpha_{\mu^{\ell,\star}_t} M^\ell_{1,t}\Big), \qquad \Delta^{\ell,\star}_{2,t} = \operatorname{Polar}\Big(M^\ell_{2,t} D^\alpha_{\mu^{\ell,\star}_t}\Big) D^\alpha_{\mu^{\ell,\star}_t}.$$

For a channel $c = (\ell, j)$, write $\mu_{c,t} := \mu^{\ell,\star}_{j,t}$ and $r_{c,t} := (\|W^\ell_{1,t,j:}\|_2^2 + \|W^\ell_{2,t,:,j}\|_2^2)^{1/2}$. For $0 \le s \le T$, define $D_{s,T} := \prod_{u=s}^{T-1}(1 - \lambda\eta_u)$, with $D_{T,T} := 1$.

Theorem 6 (Allocation controls channel norms)**.** *Let $0 < \alpha \le 1$. For every channel c and every $T \ge 1$,*

$$r_{c,T} \le D_{0,T} r_{c,0} + \sqrt{2} \sum_{t=0}^{T-1} \eta_t D_{t+1,T} \mu^\alpha_{c,t}. \tag{15}$$

Moreover, writing $r_T := (r_{c,T})_{c=1}^N$,

$$\|r_T\|_{1/\alpha} \le D_{0,T}\|r_0\|_{1/\alpha} + \frac{\sqrt{2}K^\alpha}{\lambda}(1 - D_{0,T}). \tag{16}$$

If $\sum_{t=0}^\infty \eta_t = \infty$, then

$$\limsup_{T\to\infty} \|r_T\|_{1/\alpha} \le \frac{\sqrt{2}K^\alpha}{\lambda}. \tag{17}$$

Proof. For $c = (\ell, j)$, let $P^\ell_{1,t} = \operatorname{Polar}(D^\alpha_{\mu^{\ell,\star}_t} M^\ell_{1,t})$ and $P^\ell_{2,t} = \operatorname{Polar}(M^\ell_{2,t} D^\alpha_{\mu^{\ell,\star}_t})$. Since $\|P^\ell_{1,t}\|_{\mathrm{op}} \le 1$ and $\|P^\ell_{2,t}\|_{\mathrm{op}} \le 1$,

$$\|\Delta^{\ell,\star}_{1,t,j:}\|_2 \le \mu^\alpha_{c,t}, \qquad \|\Delta^{\ell,\star}_{2,t,:,j}\|_2 \le \mu^\alpha_{c,t}.$$

Using the weight-decay update and the triangle inequality,

$$\begin{aligned} r_{c,t+1} &= \Big(\|(1-\lambda\eta_t)W^\ell_{1,t,j:} - \eta_t\Delta^{\ell,\star}_{1,t,j:}\|_2^2 + \|(1-\lambda\eta_t)W^\ell_{2,t,:,j} - \eta_t\Delta^{\ell,\star}_{2,t,:,j}\|_2^2\Big)^{1/2} \\ &\le (1-\lambda\eta_t)\Big(\|W^\ell_{1,t,j:}\|_2^2 + \|W^\ell_{2,t,:,j}\|_2^2\Big)^{1/2} + \eta_t\Big(\|\Delta^{\ell,\star}_{1,t,j:}\|_2^2 + \|\Delta^{\ell,\star}_{2,t,:,j}\|_2^2\Big)^{1/2} \\ &\le (1-\lambda\eta_t) r_{c,t} + \sqrt{2}\,\eta_t \mu^\alpha_{c,t}. \end{aligned} \tag{18}$$

Iterating Eq. (18) gives Eq. (15).

Set $p := 1/\alpha$. Applying the ℓ_p norm to the componentwise inequality in Eq. (18) and using the triangle inequality,

$$\|r_{t+1}\|_p \le (1-\lambda\eta_t)\|r_t\|_p + \sqrt{2}\,\eta_t \|\mu_t^\alpha\|_p.$$

Since $p = 1/\alpha$ and $\sum_c \mu_{c,t} = K$,

$$\|\mu_t^\alpha\|_{1/\alpha} = \left(\sum_c \mu_{c,t}\right)^\alpha = K^\alpha.$$

Iterating this inequality and using

$$\lambda \sum_{t=0}^{T-1} \eta_t D_{t+1,T} = 1 - D_{0,T} \tag{19}$$

gives Eq. (16).

Finally, $D_{0,T} \le \exp(-\lambda \sum_{t=0}^{T-1} \eta_t)$. Hence $\sum_{t=0}^{\infty} \eta_t = \infty$ implies $D_{0,T} \to 0$, and Eq. (17) follows. □

The channelwise bound also shows that sustained under-allocation suppresses the corresponding channel.

Corollary 7 (Persistent under-allocation)**.** *Assume $\sum_{t=0}^{\infty} \eta_t = \infty$. If, for some channel c, there exist $t_0 \ge 0$ and $\bar{\mu}_c \in [0, 1]$ such that $\mu_{c,t} \le \bar{\mu}_c$ for every $t \ge t_0$, then*

$$\limsup_{T\to\infty} r_{c,T} \le \frac{\sqrt{2}}{\lambda} \bar{\mu}_c^\alpha. \tag{20}$$

In particular, if $\mu_{c,t} \to 0$, then $r_{c,t} \to 0$.

Proof. Applying Eq. (15) from t_0 gives

$$r_{c,T} \le D_{t_0,T} r_{c,t_0} + \sqrt{2} \sum_{t=t_0}^{T-1} \eta_t D_{t+1,T} \mu_{c,t}^\alpha.$$

Since $\mu_{c,t} \le \bar{\mu}_c$ for every $t \ge t_0$,

$$r_{c,T} \le D_{t_0,T} r_{c,t_0} + \sqrt{2}\, \bar{\mu}_c^\alpha \sum_{t=t_0}^{T-1} \eta_t D_{t+1,T}.$$

Using Eq. (19) from t_0,

$$r_{c,T} \le D_{t_0,T} r_{c,t_0} + \frac{\sqrt{2}\, \bar{\mu}_c^\alpha}{\lambda}(1 - D_{t_0,T}).$$

Since $D_{t_0,T} \le \exp(-\lambda \sum_{t=t_0}^{T-1} \eta_t) \to 0$, we get Eq. (20).

If $\mu_{c,t} \to 0$, then for every $\varepsilon > 0$ there exists t_0 such that $\mu_{c,t}^\alpha \le \varepsilon$ for every $t \ge t_0$. Hence $\limsup_{T\to\infty} r_{c,T} \le \sqrt{2}\,\varepsilon/\lambda$. Since this holds for every $\varepsilon > 0$ and $r_{c,T} \ge 0$, we get $r_{c,T} \to 0$. □

Thus, weight decay makes BMuon's allocation persistent in parameter space: channel norms are controlled by their discounted allocation histories, implying that channels receiving vanishing allocations are driven toward zero.

D MM ALLOCATION UPDATE

This appendix provides the derivations and proofs underlying the MM allocation update of Section 4. Appendix D.1 proves the tight minorization in Eq. (7) and establishes the sign properties of the allocation scores a_j^ℓ. Appendix D.2 solves the surrogate maximization problem and derives the closed-form update in Eq. (8), including the determination of λ_μ by bisection. Appendix D.3 proves monotonicity of repeated MM updates and, for $0 < \alpha \le \frac{1}{2}$, convergence in objective value to the global optimum over $\mathcal{K}_{K,\varepsilon}$ together with global optimality of every accumulation point. Appendix D.4 bounds the effect of replacing the original feasible set $\mathcal{K}_K$ by $\mathcal{K}_{K,\varepsilon}$. Appendix D.5 quantifies the two approximations used in practice: performing a single MM update at each allocation refresh and keeping the allocation fixed between refreshes, and compares the practical update with a certified allocation solve end-to-end. Finally, Appendix D.6 characterizes the computational overhead of the practical BMuon update and reports training-time measurements for ImageNet-1K and 1B-parameter LLaMA.

Throughout this appendix, the momenta $M = \{M_1^\ell, M_2^\ell\}_{\ell=1}^L$ are fixed and all polar factors are exact. There are N globally budgeted channels, indexed by (ℓ, j), and

$$\mathcal{K}_K := \left\{ \mu \in [0,1]^N : \sum_{\ell,j} \mu_j^\ell = K \right\}, \qquad \mathcal{K}_{K,\varepsilon} := \left\{ \mu \in [\varepsilon,1]^N : \sum_{\ell,j} \mu_j^\ell = K \right\},$$

where $0 < K < N$ and $0 < \varepsilon < K/N$. For fixed M, the oracle allocation objective of Eq. (4) is

$$\Phi_M(\mu) = \sum_{\ell=1}^{L} \left[\|D_{\mu^\ell}^\alpha M_1^\ell\|_* + \|M_2^\ell D_{\mu^\ell}^\alpha\|_* \right], \qquad 0 < \alpha < 1.$$

Given $\bar{\mu} \in \mathcal{K}_{K,\varepsilon}$, the MM surrogate is

$$Q(\mu \mid \bar{\mu}) = \sum_{\ell,j} a_j^\ell(\bar{\mu})(\mu_j^\ell)^\alpha,$$

where

$$a_j^\ell(\bar{\mu}) = \langle M_{1,j,:}^\ell, \bar{P}_{1,j:}^\ell \rangle + \langle M_{2,:,j}^\ell, \bar{P}_{2,:,j}^\ell \rangle,$$

with $\bar{P}_1^\ell = \mathrm{Polar}(D_{\bar{\mu}^\ell}^\alpha M_1^\ell)$ and $\bar{P}_2^\ell = \mathrm{Polar}(M_2^\ell D_{\bar{\mu}^\ell}^\alpha)$. One MM step maximizes $Q(\cdot \mid \bar{\mu})$ over $\mathcal{K}_{K,\varepsilon}$.

D.1 MINORIZATION AND ALLOCATION SCORES

We first prove the tight minorization in Eq. (7). Fix an allocation $\bar{\mu} \in \mathcal{K}_{K,\varepsilon}$ and, for each block, let

$$\bar{P}_1^\ell = \mathrm{Polar}(D_{\bar{\mu}^\ell}^\alpha M_1^\ell), \qquad \bar{P}_2^\ell = \mathrm{Polar}(M_2^\ell D_{\bar{\mu}^\ell}^\alpha),$$

as in Section 4. By spectral–nuclear duality, $\|A\|_* = \max_{\|P\|_{\mathrm{op}} \le 1} \langle A, P \rangle$.

We will use the following equality characterization both here and in the convergence analysis.

Lemma 8. *Let $\|P\|_{\mathrm{op}} \le 1$. If $\langle A, P \rangle = \|A\|_*$, then*

$$AP^\top = (AA^\top)^{1/2}, \qquad P^\top A = (A^\top A)^{1/2}.$$

Proof. Let $A = U\Sigma V^\top$ be a compact SVD. Then

$$\langle A, P \rangle = \sum_r \sigma_r u_r^\top P v_r, \qquad |u_r^\top P v_r| \le 1.$$

Since $\sigma_r > 0$, equality $\langle A, P \rangle = \sum_r \sigma_r$ implies $u_r^\top P v_r = 1$ for every r. Moreover, $\|Pv_r\|_2 \le 1$, so equality in Cauchy–Schwarz gives $Pv_r = u_r$. Similarly, $\|P^\top u_r\|_2 \le 1$ gives $P^\top u_r = v_r$. Hence $PV = U$ and $P^\top U = V$, and therefore

$$AP^\top = U\Sigma U^\top = (AA^\top)^{1/2}, \qquad P^\top A = V\Sigma V^\top = (A^\top A)^{1/2}.$$

□

Since $\|\bar{P}_1^\ell\|_{\mathrm{op}} \le 1$ and $\|\bar{P}_2^\ell\|_{\mathrm{op}} \le 1$, for any allocation μ,

$$\begin{aligned}\Phi_M(\mu) &= \sum_{\ell=1}^{L} \left[\|D_{\mu^\ell}^\alpha M_1^\ell\|_* + \|M_2^\ell D_{\mu^\ell}^\alpha\|_*\right] \\ &\ge \sum_{\ell=1}^{L} \left[\left\langle D_{\mu^\ell}^\alpha M_1^\ell, \bar{P}_1^\ell\right\rangle + \left\langle M_2^\ell D_{\mu^\ell}^\alpha, \bar{P}_2^\ell\right\rangle\right] \\ &= \sum_{\ell,j} \left(\langle M_{1,j:}^\ell, \bar{P}_{1,j:}^\ell\rangle + \langle M_{2,:,j}^\ell, \bar{P}_{2,:,j}^\ell\rangle\right)(\mu_j^\ell)^\alpha = Q(\mu \mid \bar{\mu}). \end{aligned} \tag{21}$$

At $\mu = \bar{\mu}$, using $\langle A, \mathrm{Polar}(A)\rangle = \|A\|_*$, $Q(\bar{\mu} \mid \bar{\mu}) = \sum_{\ell=1}^{L}\left[\|D_{\bar{\mu}^\ell}^\alpha M_1^\ell\|_* + \|M_2^\ell D_{\bar{\mu}^\ell}^\alpha\|_*\right] = \Phi_M(\bar{\mu})$. Hence the minorization is tight, proving Eq. (7).

We now establish the sign properties of the scores.

Fix (ℓ, j) and consider first the row-side term. Write $A = D_{\bar{\mu}^\ell}^\alpha M_1^\ell$, $\bar{P}_1^\ell = \mathrm{Polar}(A)$, and $S = (AA^\top)^{1/2}$. Since $\bar{P}_1^\ell$ attains the spectral–nuclear dual maximum, Lemma 8 gives $A(\bar{P}_1^\ell)^\top = S \succeq 0$,

$$S_{jj} = (\bar{\mu}_j^\ell)^\alpha \langle M_{1,j:}^\ell, \bar{P}_{1,j:}^\ell\rangle \ge 0.$$

Since $\bar{\mu} \in \mathcal{K}_{K,\varepsilon}$, we have $\bar{\mu}_j^\ell \ge \varepsilon > 0$. Applying the same argument to the column-side term gives

$$\begin{aligned}\langle M_{1,j:}^\ell, \bar{P}_{1,j:}^\ell\rangle &= \frac{\left[\left(D_{\bar{\mu}^\ell}^\alpha M_1^\ell (M_1^\ell)^\top D_{\bar{\mu}^\ell}^\alpha\right)^{1/2}\right]_{jj}}{(\bar{\mu}_j^\ell)^\alpha}, \\ \langle M_{2,:,j}^\ell, \bar{P}_{2,:,j}^\ell\rangle &= \frac{\left[\left(D_{\bar{\mu}^\ell}^\alpha (M_2^\ell)^\top M_2^\ell D_{\bar{\mu}^\ell}^\alpha\right)^{1/2}\right]_{jj}}{(\bar{\mu}_j^\ell)^\alpha}. \end{aligned} \tag{22}$$

Both numerators in Eq. (22) are diagonal entries of positive semidefinite matrices. Hence both score contributions are nonnegative, and therefore

$$a_j^\ell = \langle M_{1,j:}^\ell, \bar{P}_{1,j:}^\ell\rangle + \langle M_{2,:,j}^\ell, \bar{P}_{2,:,j}^\ell\rangle \ge 0.$$

We next establish strict positivity when all channels are nontrivial.

If $M_{1,j:}^\ell \ne 0$, then

$$(S^2)_{jj} = (AA^\top)_{jj} = (\bar{\mu}_j^\ell)^{2\alpha}\|M_{1,j:}^\ell\|_2^2 > 0.$$

Since S is symmetric positive semidefinite, $S^2 \preceq \|S\|_{\mathrm{op}} S$. Hence $(S^2)_{jj} \le \|S\|_{\mathrm{op}} S_{jj}$, so $(S^2)_{jj} > 0$ implies $S_{jj} > 0$. Since $\bar{\mu}_j^\ell > 0$, we obtain $\langle M_{1,j:}^\ell, \bar{P}_{1,j:}^\ell\rangle > 0$.

The same argument applied to the column-side term shows that $M_{2,:,j}^\ell \ne 0$ implies $\langle M_{2,:,j}^\ell, \bar{P}_{2,:,j}^\ell\rangle > 0$. Therefore, if $M_{1,j:}^\ell \ne 0$ or $M_{2,:,j}^\ell \ne 0$, then $a_j^\ell > 0$. In particular, under the nontrivial-channel condition (i.e., $M_{1,j:}^\ell \ne 0$ or $M_{2,:,j}^\ell \ne 0$ for every (ℓ, j)), all scores are strictly positive for every $\bar{\mu} \in \mathcal{K}_{K,\varepsilon}$.

D.2 Closed-form surrogate maximization

We now derive the closed-form update in Eq. (8). Under the nontrivial-channel condition, Appendix D.1 gives $a_j^\ell > 0$ for every (ℓ, j). Since $0 < \alpha < 1$, each function $\mu_j^\ell \mapsto a_j^\ell(\mu_j^\ell)^\alpha$ is strictly concave on $[\varepsilon, 1]$. Hence $Q(\mu \mid \bar{\mu})$ is strictly concave, and its maximization over the convex set $\mathcal{K}_{K,\varepsilon}$ has a unique maximizer, which we denote by μ^+.

Since $0 < \varepsilon < K/N < 1$, the uniform allocation $\mu_j^\ell = K/N$ satisfies the budget constraint and both bound constraints strictly. Hence Slater's condition holds, and the KKT conditions are necessary

and sufficient. Introduce a multiplier λ_μ for the budget constraint and multipliers $\rho_j^\ell, \sigma_j^\ell \geq 0$ for the lower and upper bounds, respectively. The stationarity and complementary-slackness conditions are

$$\alpha a_j^\ell (\mu_j^\ell)^{\alpha-1} - \lambda_\mu + \rho_j^\ell - \sigma_j^\ell = 0, \tag{23}$$

$$\rho_j^\ell(\mu_j^\ell - \varepsilon) = 0, \qquad \sigma_j^\ell(1 - \mu_j^\ell) = 0. \tag{24}$$

Because $K < N$, at least one coordinate of any feasible allocation is strictly smaller than 1. For such a coordinate $\sigma_j^\ell = 0$, and Eq. (23) implies $\lambda_\mu > 0$.

For a fixed (ℓ, j), define

$$\widehat{\mu}_j^\ell = \left(\frac{\alpha a_j^\ell}{\lambda_\mu}\right)^{1/(1-\alpha)}.$$

If $\varepsilon < \mu_j^\ell < 1$, both bound multipliers vanish and $\mu_j^\ell = \widehat{\mu}_j^\ell$. If $\mu_j^\ell = \varepsilon$, then $\sigma_j^\ell = 0$ and stationarity gives $\lambda_\mu \geq \alpha a_j^\ell \varepsilon^{\alpha-1}$, equivalently $\widehat{\mu}_j^\ell \leq \varepsilon$. Similarly, if $\mu_j^\ell = 1$, then $\rho_j^\ell = 0$ and stationarity gives $\lambda_\mu \leq \alpha a_j^\ell$, equivalently $\widehat{\mu}_j^\ell \geq 1$. Hence the unique maximizer is

$$(\mu_j^\ell)^+ = \mathrm{clip}_{[\varepsilon,1]}\left[\left(\frac{\alpha a_j^\ell}{\lambda_\mu}\right)^{1/(1-\alpha)}\right],$$

which proves Eq. (8).

It remains to determine λ_μ. Define

$$H(\lambda) := \sum_{\ell,j} \mathrm{clip}_{[\varepsilon,1]}\left[\left(\frac{\alpha a_j^\ell}{\lambda}\right)^{1/(1-\alpha)}\right], \qquad \lambda > 0.$$

Since every $a_j^\ell > 0$, H is continuous and nonincreasing, with $H(\lambda) \to N$ as $\lambda \downarrow 0$ and $H(\lambda) \to N\varepsilon$ as $\lambda \uparrow \infty$. Because $N\varepsilon < K < N$, there exists $\lambda_\mu > 0$ such that $H(\lambda_\mu) = K$, and such a value can be found by bisection. This completes the derivation of the MM allocation update.

D.3 Convergence of repeated MM updates

We now prove the convergence statement of Section 4.

Theorem 9 (Convergence of repeated MM updates)**.** *Assume the nontrivial-channel condition (i.e., $M_{1,j:}^\ell \neq 0$ or $M_{2,:,j}^\ell \neq 0$ for every (ℓ, j)), and let $0 < \alpha \leq \frac{1}{2}$. Starting from any $\mu^{(0)} \in \mathcal{K}_{K,\varepsilon}$, define*

$$\mu^{(t+1)} = \arg\max_{\mu \in \mathcal{K}_{K,\varepsilon}} Q(\mu \mid \mu^{(t)}).$$

Then

$$\Phi_M(\mu^{(t)}) \longrightarrow \Phi_\varepsilon^\star := \max_{\mu \in \mathcal{K}_{K,\varepsilon}} \Phi_M(\mu),$$

and every accumulation point of $\{\mu^{(t)}\}_{t\geq 0}$ is a global maximizer of Φ_M over $\mathcal{K}_{K,\varepsilon}$.

Proof. By Appendix D.1, the surrogate is a tight minorizer: $Q(\mu \mid \bar{\mu}) \leq \Phi_M(\mu)$ for every μ, with equality at $\mu = \bar{\mu}$. Since $\mu^{(t+1)}$ maximizes $Q(\cdot \mid \mu^{(t)})$,

$$\Phi_M(\mu^{(t+1)}) \geq Q(\mu^{(t+1)} \mid \mu^{(t)}) \geq Q(\mu^{(t)} \mid \mu^{(t)}) = \Phi_M(\mu^{(t)}).$$

Hence $(\Phi_M(\mu^{(t)}))_t$ is nondecreasing. Since $\mathcal{K}_{K,\varepsilon}$ is compact and Φ_M is continuous, $\Phi_M(\mu^{(t)})$ converges.

Define the MM map $T(\bar{\mu}) := \arg\max_{\mu \in \mathcal{K}_{K,\varepsilon}} Q(\mu \mid \bar{\mu})$. By Appendix D.2, under the nontrivial-channel condition the surrogate has a unique maximizer. Thus T is well defined.

We next show that T is continuous. By Eq. (22),

$$a_j^\ell(\bar{\mu}) = \frac{\left[\left(D_{\bar{\mu}^\ell}^\alpha M_1^\ell (M_1^\ell)^\top D_{\bar{\mu}^\ell}^\alpha\right)^{1/2}\right]_{jj}}{(\bar{\mu}_j^\ell)^\alpha} + \frac{\left[\left(D_{\bar{\mu}^\ell}^\alpha (M_2^\ell)^\top M_2^\ell D_{\bar{\mu}^\ell}^\alpha\right)^{1/2}\right]_{jj}}{(\bar{\mu}_j^\ell)^\alpha}.$$

Since $\bar{\mu} \in \mathcal{K}_{K,\varepsilon}$, we have $\bar{\mu}_j^\ell \geq \varepsilon > 0$. Moreover, $\bar{\mu} \mapsto D^\alpha_{\bar{\mu}^\ell}$ is continuous, and the matrix square root is continuous on the set of positive semidefinite matrices. Therefore $\bar{\mu} \mapsto a_j^\ell(\bar{\mu})$ is continuous for every (ℓ, j). Hence $(\mu, \bar{\mu}) \mapsto Q(\mu \mid \bar{\mu}) = \sum_{\ell,j} a_j^\ell(\bar{\mu})(\mu_j^\ell)^\alpha$ is continuous on $\mathcal{K}_{K,\varepsilon} \times \mathcal{K}_{K,\varepsilon}$. Since $\mathcal{K}_{K,\varepsilon}$ is compact and $Q(\cdot \mid \bar{\mu})$ has a unique maximizer, Berge's maximum theorem, together with uniqueness of the maximizer, implies that T is continuous.

Since $\mathcal{K}_{K,\varepsilon}$ is compact, the sequence $\{\mu^{(t)}\}$ has at least one accumulation point. Let $\mu^\star$ be any accumulation point. There exists a subsequence $\mu^{(t_k)} \to \mu^\star$. Since $\Phi_M(\mu^{(t)})$ converges,

$$\Phi_M(\mu^{(t_k+1)}) - \Phi_M(\mu^{(t_k)}) \longrightarrow 0.$$

Using $\mu^{(t_k+1)} = T(\mu^{(t_k)})$ and continuity of T and Φ_M gives $\Phi_M(T(\mu^\star)) = \Phi_M(\mu^\star)$.

Suppose that $T(\mu^\star) \neq \mu^\star$. Since $T(\mu^\star)$ is the unique maximizer of $Q(\cdot \mid \mu^\star)$, $Q(T(\mu^\star) \mid \mu^\star) > Q(\mu^\star \mid \mu^\star)$. By minorization and tightness,

$$\Phi_M(T(\mu^\star)) \geq Q(T(\mu^\star) \mid \mu^\star) > Q(\mu^\star \mid \mu^\star) = \Phi_M(\mu^\star),$$

which is a contradiction. Therefore $T(\mu^\star) = \mu^\star$.

It remains to show that every fixed point is globally optimal. We first show first-order tangency in directional derivatives.

For a function F, write $F'(x; d) := \lim_{s \downarrow 0}[F(x + sd) - F(x)]/s$ for its one-sided directional derivative.

By spectral–nuclear duality, $\|D^\alpha_{\mu^\ell} M_1^\ell\|_* = \max_{\|P\|_{\mathrm{op}} \leq 1} \langle D^\alpha_{\mu^\ell} M_1^\ell, P \rangle$. For fixed P, the directional derivative at $\bar{\mu}$ in direction d is

$$\sum_j \alpha(\bar{\mu}_j^\ell)^{\alpha-1} d_j^\ell \langle M_{1,j:}^\ell, P_{j:} \rangle.$$

Let $A = D^\alpha_{\bar{\mu}^\ell} M_1^\ell$. By Lemma 8, every maximizing P satisfies $AP^\top = (AA^\top)^{1/2}$, so $(\bar{\mu}_j^\ell)^\alpha \langle M_{1,j:}^\ell, P_{j:} \rangle = [(AA^\top)^{1/2}]_{jj}$. Thus the directional derivative above is the same for every maximizing P. Since the operator-norm unit ball is compact and the inner objective is continuously differentiable in μ on the positive orthant, Danskin's theorem, together with the identical argument for the column-side term, gives

$$\Phi'_M(\bar{\mu}; d) = \sum_{\ell,j} \alpha a_j^\ell(\bar{\mu})(\bar{\mu}_j^\ell)^{\alpha-1} d_j^\ell = Q'(\bar{\mu} \mid \bar{\mu}; d).$$

Since $T(\mu^\star) = \mu^\star$, for every $\mu \in \mathcal{K}_{K,\varepsilon}$,

$$Q'(\mu^\star \mid \mu^\star; \mu - \mu^\star) \leq 0.$$

By first-order tangency, $\Phi'_M(\mu^\star; \mu - \mu^\star) \leq 0$. For $0 < \alpha \leq \frac{1}{2}$, Φ_M is concave by Theorem 1, and therefore

$$\Phi_M(\mu) \leq \Phi_M(\mu^\star) + \Phi'_M(\mu^\star; \mu - \mu^\star) \leq \Phi_M(\mu^\star).$$

Thus $\mu^\star$ is a global maximizer of Φ_M over $\mathcal{K}_{K,\varepsilon}$.

Since $\mu^{(t_k)} \to \mu^\star$ and Φ_M is continuous, $\Phi_M(\mu^{(t_k)}) \to \Phi_M(\mu^\star) = \Phi^\star_\varepsilon$. Because the full sequence $\Phi_M(\mu^{(t)})$ converges, its limit must also be $\Phi^\star_\varepsilon$. Hence $\Phi_M(\mu^{(t)}) \to \Phi^\star_\varepsilon$. □

D.4 Effect of the allocation floor

We finally quantify the effect of replacing the original feasible set $\mathcal{K}_K$ by $\mathcal{K}_{K,\varepsilon}$. Let

$$\Phi^\star := \max_{\mu \in \mathcal{K}_K} \Phi_M(\mu), \qquad \Phi^\star_\varepsilon := \max_{\mu \in \mathcal{K}_{K,\varepsilon}} \Phi_M(\mu).$$

Proposition 10 (Effect of the allocation floor)**.** *Let* $0 < \alpha \leq \frac{1}{2}$ *and* $0 < \varepsilon < K/N$. *Then*

$$0 \leq \Phi^\star - \Phi^\star_\varepsilon \leq \frac{\varepsilon N}{K} \left[\Phi^\star - \Phi_M\left(\frac{K}{N} \mathbf{1} \right) \right].$$

Proof. Since $\mathcal{K}_{K,\varepsilon} \subseteq \mathcal{K}_K$, we immediately have $\Phi^\star_\varepsilon \le \Phi^\star$.

Let $\mu^\star \in \mathcal{K}_K$ be a maximizer of Φ_M, and define $\delta := \varepsilon N/K$ and $u := (K/N)\mathbf{1}$. Since $0 < \varepsilon < K/N$, we have $0 < \delta < 1$. Consider $\widetilde{\mu} := (1-\delta)\mu^\star + \delta u$. For every (ℓ, j), $\widetilde{\mu}^\ell_j \ge \delta K/N = \varepsilon$, while $\sum_{\ell,j} \widetilde{\mu}^\ell_j = K$ and $\widetilde{\mu}^\ell_j \le 1$. Hence $\widetilde{\mu} \in \mathcal{K}_{K,\varepsilon}$.

By Theorem 1, Φ_M is concave for $0 < \alpha \le \frac{1}{2}$. Therefore

$$\Phi^\star_\varepsilon \ge \Phi_M(\widetilde{\mu}) \ge (1-\delta)\Phi_M(\mu^\star) + \delta\Phi_M(u) = (1-\delta)\Phi^\star + \delta\Phi_M\left(\frac{K}{N}\mathbf{1}\right).$$

Rearranging and using $\delta = \varepsilon N/K$ gives

$$\Phi^\star - \Phi^\star_\varepsilon \le \frac{\varepsilon N}{K}\left[\Phi^\star - \Phi_M\left(\frac{K}{N}\mathbf{1}\right)\right],$$

which completes the proof. □

D.5 Approximation of the allocation oracle

As in Algorithm 1, BMuon does not recompute the allocation μ at every optimizer step. Instead, every five optimizer steps it performs an *allocation refresh*: μ is recomputed and then kept fixed for the next four optimizer steps. The budget may be scheduled over training, so we write K_t for the budget at optimizer step t. At a refresh step t, the new allocation is computed under the current budget K_t.

The practical implementation therefore introduces two approximations. First, at each allocation refresh, we perform only one MM update instead of solving the allocation problem to convergence. Second, we refresh μ only once every five optimizer steps, rather than at every optimizer step. We measure these two errors separately, and then test whether replacing the practical one-step update by a certified solve at each refresh changes the training and compression results.

Two paired training runs. We use the same CIFAR-100 ViT and training setup as in Appendix E.2, and train for 100 epochs with three seeds. As in our training experiments, we use $\alpha = 1/2$. For each seed, we run two paired versions of BMuon that differ only in how the allocation μ is computed at an allocation refresh. The MLPs contain $N = 4608$ hidden channels in total. The budget K_t decreases linearly from N to $0.3N$ during the first third of training and is then kept fixed. In this experiment, K_t changes only at allocation refreshes, so it is constant during the four intervening optimizer steps. In both runs, μ is refreshed every five optimizer steps and is held fixed between two consecutive refreshes.

The two runs differ only in how μ is computed at a refresh step t. In the **1-step refresh run**, we start from the allocation carried from the previous refresh, compute the MM scores using NS_5, the five-step Newton–Schulz approximation defined in Appendix A, and perform one MM update under the current budget K_t. In the **certified refresh run**, we instead use exact polar factors and repeat the MM update until the stopping condition Eq. (25) is satisfied. In both runs, the allocation computed at step t satisfies the current constraint $\sum_i \mu_i = K_t$.

Apart from this difference, the two runs use the same BMuon weight update. For each seed, they also use the same initialization, data order, augmentation stream, and budget schedule.

Notation and certified error bound. Let M_t denote all MLP momentum matrices at optimizer step t, and define $\mathcal{C}_t = \{\mu \in [\varepsilon, 1]^N : \mathbf{1}^\top \mu = K_t\}$. For $\alpha = \frac{1}{2}$, the allocation objective is

$$\Phi_M(\mu) = \sum_{\ell=1}^{L}\left(\left\|D^{1/2}_{\mu^\ell} M^\ell_1\right\|_* + \left\|M^\ell_2 D^{1/2}_{\mu^\ell}\right\|_*\right).$$

We write $\Phi^\star_t = \max_{\mu\in\mathcal{C}_t} \Phi_{M_t}(\mu)$.

For any feasible μ, define $g_t(\mu) = \max_{\nu\in\mathcal{C}_t}\langle \nabla\Phi_{M_t}(\mu), \nu - \mu\rangle$. We call $g_t(\mu)$ the Frank–Wolfe gap. Since Φ_{M_t} is concave (Theorem 1), for every $\nu \in \mathcal{C}_t$, $\Phi_{M_t}(\nu) \le \Phi_{M_t}(\mu) + \langle \nabla\Phi_{M_t}(\mu), \nu - \mu\rangle$.

Maximizing over ν gives $\Phi_t^\star - \Phi_{M_t}(\mu) \le g_t(\mu)$, which implies in particular that $g_t(\mu) \ge 0$ for every feasible μ.

At optimizer step t, let $\mu_t^{\mathrm{ref}} \in \mathcal{C}_t$ denote the allocation obtained by repeatedly applying the exact-polar MM update over $\mathcal{C}_t$ (Theorem 9 shows that the repeated MM updates converge) until

$$\frac{g_t(\mu_t^{\mathrm{ref}})}{\Phi_{M_t}(\mu_t^{\mathrm{ref}})} \le 10^{-4}. \tag{25}$$

Define $U_t = \Phi_{M_t}(\mu_t^{\mathrm{ref}}) + g_t(\mu_t^{\mathrm{ref}})$. The previous inequality gives $U_t \ge \Phi_t^\star$.

For any feasible candidate $\widetilde{\mu}$, we therefore define $d_t(\widetilde{\mu}) = 1 - \Phi_{M_t}(\widetilde{\mu})/U_t$. Because $U_t \ge \Phi_t^\star$,

$$1 - \frac{\Phi_{M_t}(\widetilde{\mu})}{\Phi_t^\star} \le 1 - \frac{\Phi_{M_t}(\widetilde{\mu})}{U_t} = d_t(\widetilde{\mu}).$$

Thus $d_t(\widetilde{\mu})$ is a certified upper bound on the relative loss in the exact allocation objective incurred by using $\widetilde{\mu}$ instead of an optimal allocation.

In particular, for μ_t^{ref},

$$d_t(\mu_t^{\mathrm{ref}}) = \frac{g_t(\mu_t^{\mathrm{ref}})}{\Phi_{M_t}(\mu_t^{\mathrm{ref}}) + g_t(\mu_t^{\mathrm{ref}})} \le \frac{g_t(\mu_t^{\mathrm{ref}})}{\Phi_{M_t}(\mu_t^{\mathrm{ref}})} \le 10^{-4},$$

where the last inequality follows from Eq. (25).

Hence μ_t^{ref} is certified to have relative objective loss at most 10^{-4} from the exact optimum. We therefore use μ_t^{ref} as a certified approximation, in objective value, to an optimal allocation at step t, and call it the *reference allocation*. In the *certified refresh* run, at each refresh step t, we use μ_t^{ref} as the BMuon allocation and keep it fixed until the next refresh.

For each run and seed, we perform the diagnostics described below at ten refresh steps: the first refresh at which K_t/N reaches approximately $0.9, 0.8, \ldots, 0.3$, followed by refreshes near 50%, 75%, and 99% of training. We denote this set of ten refresh steps by $\mathcal{D}$.

The two approximation errors. Figure 4 measures separately the two approximations used by the practical implementation.

At each $t \in \mathcal{D}$, let $\widehat{\mu}_t$ denote the allocation obtained by applying the practical update rule once: one MM step with scores computed using NS_5, starting from the allocation carried from the previous refresh. Figure 4(a) reports $d_t(\widehat{\mu}_t)$.

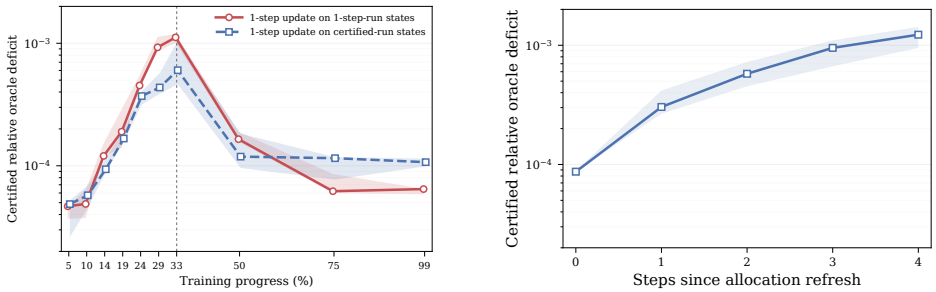
For the red curve, we apply the practical one-step allocation update to the 1-step refresh run at step t. For the blue curve, we apply the same update to the certified refresh run at step t, only for this diagnostic; the resulting allocation is not used to continue the certified refresh run.

At each sampled training stage $t \in \mathcal{D}$, the plotted value for each curve is the median over the three seeds, and the shaded region is their minimum–maximum range. The vertical dashed line marks the end of the budget decrease, after which $K_t/N = 0.3$ is fixed.

The certified deficit of one practical allocation update remains of order 10^{-4} to 10^{-3} over the sampled states. Its largest sampled value is about 1.2×10^{-3} on states from the 1-step refresh run and about 1.1×10^{-3} on states from the certified refresh run. The similar behavior on the two trajectories shows that the accuracy of the one-step rule is not specific to states produced by that same rule.

Figure 4(b) measures the second approximation: keeping μ fixed between two refreshes. We use the certified refresh run for this experiment. For a diagnostic refresh $t \in \mathcal{D}$, BMuon uses μ_t^{ref} and keeps it fixed for the following five optimizer steps. For $s \in \{0, 1, 2, 3, 4\}$, we evaluate this same allocation under the current momentum state M_{t+s} and report $d_{t+s}(\mu_t^{\mathrm{ref}})$. This quantity upper-bounds the relative objective loss incurred by continuing to use the allocation computed at step t, instead of recomputing an optimal allocation for the current momentum state M_{t+s}. Thus $s = 0$ corresponds to the refresh step, while $s = 4$ corresponds to the last optimizer step before the next refresh.

To avoid hiding a difficult sampled window by averaging over different training steps, Figure 4(b) first takes the worst case over $t \in \mathcal{D}$. For seed j and offset s, define $w_{j,s} = \max_{t \in \mathcal{D}} d_{t+s}^{(j)}(\mu_t^{\mathrm{ref},(j)})$.



(a) One allocation update. **(b)** Allocation held between refreshes.

Figure 4: **Accuracy of the practical allocation approximation. (a)** Certified relative deficit of one practical NS_5-scored MM update. The same update is evaluated on states generated by the 1-step refresh run and by the certified refresh run. At each training stage, the line is the median over three seeds and the shaded region is their minimum–maximum range. The vertical dashed line marks the beginning of the fixed-budget phase $K_t/N = 0.3$. **(b)** Certified relative deficit caused by keeping the allocation installed by the certified refresh run fixed for $s = 0, \ldots, 4$ optimizer steps. For each seed and each s, we first take the largest deficit over the ten sampled windows; the line is the median of these three seed-wise maxima and the shaded region is their minimum–maximum range.

The plotted value at offset s is the median of $w_{0,s}, w_{1,s}, w_{2,s}$, and the shaded region is their minimum–maximum range.

Even under this worst-over-sampled-windows summary, the certified deficit after four optimizer steps remains on the order of 10^{-3}. Thus, at the sampled states, keeping μ fixed for the four steps between two refreshes produces only a small additional error in the allocation objective.

End-to-end effect. The previous experiments measure the approximation error in the allocation objective. We now test whether this error changes the trained model.

Figure 5 compares the two runs. Recall that both refresh μ every five optimizer steps: the 1-step refresh run uses one practical MM update, while the certified refresh run uses μ_t^{ref}.

Figure 5(a) shows CIFAR-100 test accuracy during training. Figure 5(b) shows the accuracy of the final models after structured MLP-channel pruning, without fine-tuning. For each model, we rank all MLP channels globally by the corresponding entries of its final allocation μ and remove those with the smallest values. In both panels, the plotted value for each curve is the median over the three seeds, and the shaded region is their minimum–maximum range.

Using μ_t^{ref} at each refresh leaves test accuracy and post-training compressibility unchanged within the observed seed-to-seed variability. At the final epoch, the absolute differences between the paired runs are 0.69, 0.26, and 0.11 percentage points for seeds 0, 1, and 2, respectively. At 70% pruning, they are 0.61, 0.22, and 0.46 percentage points. For comparison, changing the seed of the 1-step refresh run gives differences of 0.09–0.80 percentage points at the final epoch and 0.05–0.87 percentage points at 70% pruning. The sign of the paired difference is not consistent across seeds.

Scope of the comparison. Both runs refresh μ every five optimizer steps. Thus, the end-to-end comparison isolates the difference between one practical MM update and the certified allocation μ_t^{ref} at each refresh; it does not compare against a run that recomputes a certified allocation at every optimizer step. Figure 4(b) separately shows that keeping μ fixed during the four intervening steps introduces only a small objective error at the sampled states.

D.6 Computational Overhead

Relative to Muon, BMuon's additional computation is confined to the budgeted matrices (the MLP weights; Algorithm 1); all other parameters are updated exactly as in Muon. On steps where the allocation is not updated, BMuon uses the same number of Newton–Schulz polar evaluations as Muon, with only additional channelwise row/column scalings by D_μ^α. On an allocation-update step,

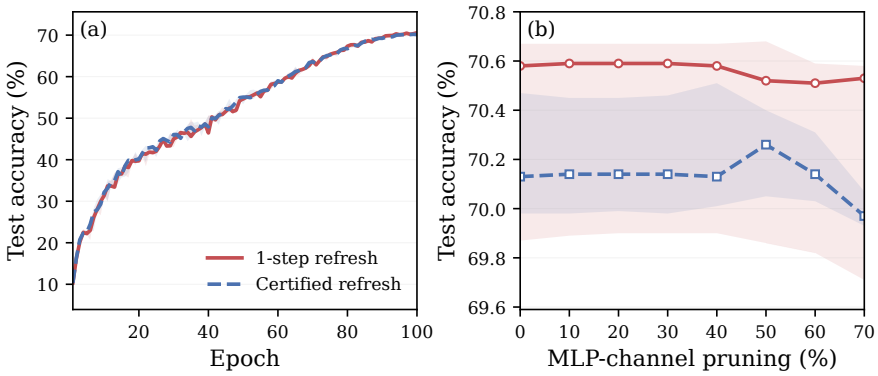


Figure 5: **End-to-end effect of the allocation approximation.** We compare the 1-step refresh run with the certified refresh run; both refresh μ every five optimizer steps. **(a)** CIFAR-100 test accuracy during 100 epochs of training. **(b)** Test accuracy after global structured MLP-channel pruning of the final trained model, using its final μ to rank channels, without fine-tuning. For each curve, the line is the median over three seeds and the shaded region is their minimum–maximum range.

Table 2: **ImageNet-1K training time.** Mean epoch time in seconds over each full 150-epoch gradual-pruning run, excluding evaluation. Differences between optimizers are comparable to seed-to-seed variation.

	90% pruning		95% pruning	
Seed	BMuon	Muon	BMuon	Muon
0	92.55	92.06	92.47	92.50
1	92.02	90.72	92.27	91.59
2	92.81	93.05	91.57	91.48

BMuon performs one additional polar evaluation per budgeted matrix to construct the MM surrogate, before recomputing the weighted polar factor used for the parameter update. Updating the allocation every ΔT_μ steps therefore raises the amortized number of polar evaluations per budgeted matrix from 1 to $1 + 1/\Delta T_\mu$ per optimizer step, i.e. 1.2 for $\Delta T_\mu = 5$. Once the surrogate polar factors are available, the channel scores require only row/column inner products, and the allocation update consists of channelwise operations and a scalar bisection over the global budget. The only persistent additional optimizer state is the allocation vector $\mu \in \mathbb{R}^N$, one scalar per budgeted channel, which is negligible next to the momentum buffers.

We verify empirically that this overhead is small in practice by comparing end-to-end training times against Muon at the allocation-update frequency used throughout our experiments, $\Delta T_\mu = 5$, on the ImageNet-1K ViT-S/16 and 1B-parameter LLaMA settings of Appendices E.4 and E.5. All measurements use eight H100 GPUs and the same Muon and BMuon implementations as the corresponding training runs.

ImageNet-1K. Table 2 reports the mean epoch time for each full 150-epoch gradual-pruning run (training only, excluding evaluation). Averaged over three seeds, BMuon and Muon require 92.5 ± 0.4 and 91.9 ± 1.2 seconds per epoch at 90% final pruning, and 92.1 ± 0.5 and 91.9 ± 0.6 seconds at 95%. The mean differences (0.6% and 0.3%) are comparable to the seed-to-seed variation, so the runs do not resolve an end-to-end slowdown of BMuon relative to Muon.

LLaMA. Table 3 reports timing measurements from the 13.1B-token LLaMA training runs. For each run, the reported time per step is the total measured training time divided by the 6,242 training steps. The training time excludes evaluation and checkpointing.

Averaged over three seeds, a step takes 4858 ± 42 ms with BMuon and 4918 ± 22 ms with Muon. The 1.2% difference is comparable to the seed-to-seed variation and can also be affected by normal

Table 3: **LLaMA training time.** Mean time per training step and total training time for each full 13.1B-token run, excluding evaluation and checkpointing. The difference between optimizers is comparable to seed-to-seed variation and does not indicate a slowdown of BMuon.

	BMuon		Muon	
Seed	ms/step	Time (h)	ms/step	Time (h)
0	4832.4	8.379	4929.3	8.547
1	4836.0	8.385	4893.5	8.485
2	4906.5	8.507	4932.0	8.551

system-level variation between separate jobs, such as job placement, per-GPU performance, and communication, so we do not attribute it to the optimizer. The runs show no end-to-end slowdown of BMuon relative to Muon.

E EXPERIMENTAL DETAILS

This section provides the experimental details and additional results for Section 5. Appendix E.1 gives the ImageNet-1K post-training pruning setup and full pruning results. Appendix E.2 gives the CIFAR-100 post-training compression setup and full pruning results. Appendix E.3 studies the effect of the global budget K. Appendix E.4 gives the gradual-pruning setup and architecture controls for CIFAR-100 and ImageNet-1K. Appendix E.5 provides the experimental details and full pruning results for the LLaMA/FineWeb experiments.

E.1 IMAGENET-1K POST-TRAINING STRUCTURED PRUNING

Architecture and training setup. We use a custom PyTorch ViT-S/16 implementation following the plain-ViT design of Beyer et al. (2022): 224×224 inputs, 16×16 patches, embedding dimension 384, 12 Transformer blocks, six attention heads, MLP width 1536, fixed two-dimensional sinusoidal positional embeddings, global average pooling over patch tokens, and a pre-logits representation layer. The model therefore contains $12 \times 1536 = 18{,}432$ MLP hidden channels. Our ImageNet-1K optimization setup follows the ViT-S/16 configuration of Zhong et al. (2026). Matrix-valued parameters within the Transformer blocks are optimized with Muon, while the remaining parameters are optimized with AdamW. We use global batch size 4096, Muon learning rate $3\sqrt{3} \times 10^{-3}$, AdamW learning rate 3×10^{-3}, weight decay 0.3, Muon momentum 0.95 with Nesterov, and dropout 0.1. Training uses sigmoid binary cross entropy with label smoothing 0.1, random resized crops, and horizontal flips, without Mixup, CutMix, RandAugment, random erasing, or stochastic depth. Muon uses five Newton–Schulz iterations following Jordan et al. (2024) and the Moonlight-style RMS-matching scale $0.2\sqrt{\max(d_{\text{out}}, d_{\text{in}})}$ (Liu et al., 2025). All models are trained for 150 epochs with 5000 warmup steps followed by cosine learning-rate decay to zero, using a half-length adaptation of the 300-epoch, 10,000-step ViT-S/16 schedule of Zhong et al. (2026).

BMuon training. BMuon is applied only to the MLP matrices; the remaining Muon-eligible Transformer matrices use ordinary Muon. We use $\alpha = 1/2$, one MM allocation update every five optimizer steps, and $\varepsilon = 10^{-6}$. Let $N = 18{,}432$ denote the total number of MLP hidden channels. The global allocation budget is initialized at $K(0) = N$ and linearly annealed to $K_{\text{target}} = 5{,}530 \approx 0.30N$ during the first 25% of training, after which it is held fixed for the remaining 75%.

Post-training pruning. After training, we take the final epoch-150 checkpoint and perform one-shot global structured pruning of coupled MLP hidden channels. Each pruning level is evaluated independently from the same dense checkpoint; pruning is therefore not cumulative, and no post-pruning fine-tuning or recovery is performed. For channel j in block ℓ, we use the coupled magnitude score $s_j^\ell = \sqrt{\|W_{1,j:}^\ell\|_2^2 + \|W_{2,:,j}^\ell\|_2^2}$. All $18{,}432$ channels are ranked jointly across the 12 blocks, subject to a minimum of four surviving channels per block. We evaluate global MLP-channel pruning levels $0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 95\%$. Figure 1 compares ordinary Muon and BMuon using this same magnitude-based pruning rule.

Evaluation. We run seeds $0, 1, 2$ and evaluate Top-1 accuracy on the full 50,000-image ImageNet-1K validation set. Reported curves show the mean and sample standard deviation across the three seeds. Table 4 reports the exact results corresponding to Figure 1.

E.2 CIFAR-100 POST-TRAINING COMPRESSION

Architecture and shared training. We use a six-block Vision Transformer on CIFAR-100 with 4×4 patches, embedding dimension 192, three attention heads, and MLP width 768. All methods are trained for 200 epochs with seeds $\{0, 1, 2\}$ using batch size 256. For the matrix optimizer, we use peak learning rate 0.02, momentum 0.95, weight decay 0.05, and five Newton–Schulz steps. Biases, LayerNorm parameters, the CLS token, positional embeddings, patch-embedding layer, and classifier head use AdamW with peak learning rate 3×10^{-4}, betas $(0.9, 0.999)$, and weight decay 0.05. We use five warmup epochs followed by cosine decay. Training uses standard CIFAR-100 augmentations, including random cropping, horizontal flipping, RandAugment, random erasing, Mixup,

Table 4: **ImageNet-1K post-training structured pruning.** Top-1 accuracy (%) after global MLP-channel pruning of ViT-S/16 using coupled weight-magnitude ranking. Results are mean ± sample standard deviation over three seeds. No post-pruning recovery is performed.

MLP pruning (%)	Muon + mag.	BMuon + mag.
0	75.230 ± 0.022	74.174 ± 0.220
10	64.201 ± 1.184	74.183 ± 0.236
20	26.762 ± 1.387	74.188 ± 0.230
30	3.167 ± 0.784	74.180 ± 0.228
40	0.513 ± 0.211	74.182 ± 0.225
50	0.249 ± 0.057	74.177 ± 0.230
60	0.239 ± 0.025	74.185 ± 0.227
70	0.185 ± 0.062	74.083 ± 0.324
80	0.149 ± 0.012	0.227 ± 0.057
90	0.132 ± 0.050	0.138 ± 0.047
95	0.105 ± 0.009	0.138 ± 0.031

and CutMix. All reported compression results are recomputed from the final epoch-200 checkpoint. Evaluation uses the CIFAR-100 test set with batch size 512 and no test-time augmentation. Reported uncertainties are sample standard deviations over the three seeds.

BMuon. BMuon uses $\alpha = 1/2$ and a global budget over all 4608 MLP hidden channels. The budget is linearly annealed from 4608 to 922, corresponding approximately to 20% of the total channel mass, over the first third of training and then held fixed. The allocation is updated every five optimizer steps using one MM update, with $\varepsilon = 10^{-6}$. Non-MLP Transformer matrices use ordinary Muon.

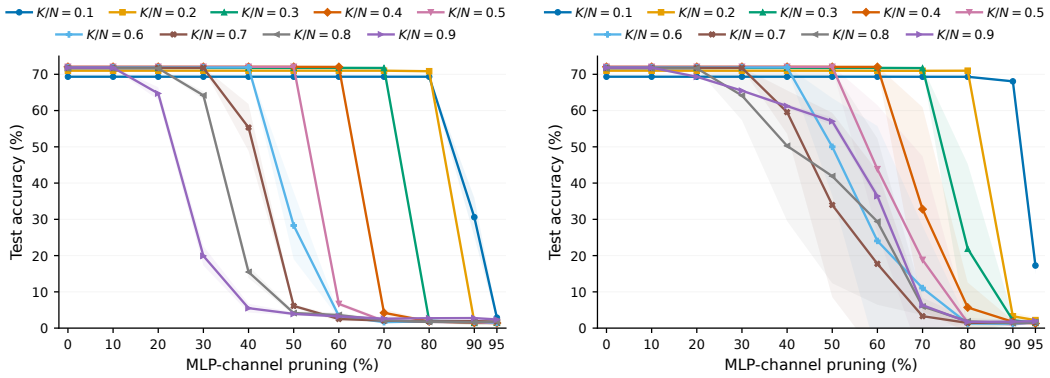
Muon+GL. Muon+GL applies an exact proximal Group-Lasso step with coefficient λ_{GL} to coupled MLP hidden-channel groups $(W^{\ell}_{1,j:}, W^{\ell}_{2,:,j})$, using the group norm $\sqrt{\|W^{\ell}_{1,j:}\|_2^2 + \|W^{\ell}_{2,:,j}\|_2^2}$. We tune λ_{GL} over $\{0.01, 0.02, 0.05, 0.1, 0.2, 0.5\}$, yielding mean dense accuracies of $\{71.50, 71.58, 71.25, 70.03, 69.23, 61.21\}\%$, respectively. Increasing λ_{GL} strengthens the sparsity-inducing penalty and can improve post-training compressibility at the expense of dense accuracy. To compare at similar pre-compression accuracy, we restrict attention here to settings within one percentage point of BMuon's 71.46%, leaving $\lambda_{\mathrm{GL}} \in \{0.01, 0.02, 0.05\}$. At 50% MLP-channel pruning, these settings retain 9.71%, 16.30%, and 71.25% test accuracy, respectively. We therefore use $\lambda_{\mathrm{GL}} = 0.05$ throughout this experiment.

NuMuon. NuMuon (Dolatabadi et al., 2026) is applied during training to all Muon-eligible Transformer weight matrices: the fused QKV projection, attention output projection, and the two MLP projections in each of the six blocks, for 24 matrices in total. Following the authors' cosine rank-scheduling strategy, we use a 10% initial full-rank period and, in our implementation, cosine-anneal the relative update rank from 1.0 to 0.25 between 10% and 80% of training, holding it at 0.25 thereafter. Since all eligible matrices have $q = \min(d_{\mathrm{in}}, d_{\mathrm{out}}) = 192$, this corresponds to a final update rank of 48. Full-rank steps use the standard Newton–Schulz Muon update, while reduced-rank steps use randomized Block Krylov with two Krylov iterations. The NuMuon paper reports an oversampling setting of 8; in our implementation, we realize this as a Krylov working dimension $b = \min(q, k + 8)$, where $k = \lceil r(t)q \rceil$ is the target update rank.

Post-training compression. Compression is restricted to the ViT MLP blocks. We evaluate compression ratios $\{0, 0.1, \ldots, 0.9, 0.95\}$ independently from the same final checkpoint, without cumulative pruning or post-compression fine-tuning. Muon, Muon+GL, and BMuon use global hidden-channel pruning over all six blocks. Magnitude pruning ranks channel j in block ℓ by $s^{\ell}_j = \sqrt{\|W^{\ell}_{1,j:}\|_2^2 + \|W^{\ell}_{2,:,j}\|_2^2}$. BMuon is evaluated with both this ranking and its final learned allocation μ. During global pruning, we skip any candidate removal that would leave an MLP block with fewer than four hidden channels.

Table 5: **CIFAR-100 post-training compression results.** Test accuracy (%) reported as mean ± sample standard deviation over three seeds. All compression levels are evaluated independently from the final epoch-200 checkpoint, without post-compression fine-tuning.

Compression	Muon	Muon+GL	NuMuon	BM-mag	BM-μ
0%	71.49 ± 0.21	71.25 ± 0.02	69.95 ± 0.13	71.46 ± 0.15	71.46 ± 0.15
10%	66.12 ± 0.86	71.25 ± 0.02	69.90 ± 0.08	71.46 ± 0.15	71.46 ± 0.15
20%	20.26 ± 3.53	71.25 ± 0.02	69.79 ± 0.17	71.46 ± 0.15	71.46 ± 0.15
30%	5.72 ± 0.55	71.25 ± 0.02	69.60 ± 0.04	71.46 ± 0.15	71.46 ± 0.16
40%	4.18 ± 0.64	71.25 ± 0.02	68.70 ± 0.14	71.46 ± 0.15	71.46 ± 0.16
50%	3.77 ± 1.38	71.25 ± 0.02	66.65 ± 0.36	71.46 ± 0.15	71.46 ± 0.14
60%	2.94 ± 0.35	30.45 ± 4.40	61.49 ± 0.56	71.46 ± 0.15	71.42 ± 0.09
70%	2.81 ± 0.49	11.31 ± 0.67	20.81 ± 5.48	71.46 ± 0.15	71.38 ± 0.05
80%	1.93 ± 0.78	6.01 ± 0.46	3.84 ± 0.83	60.84 ± 9.74	70.57 ± 0.18
90%	2.19 ± 0.60	3.76 ± 0.76	1.86 ± 0.68	1.62 ± 0.20	5.82 ± 1.46
95%	2.06 ± 0.77	3.63 ± 0.35	1.58 ± 0.39	1.44 ± 0.41	2.71 ± 0.73



(a) Magnitude ranking.

(b) μ ranking.

Figure 6: **Effect of the budget on post-training compression.** CIFAR-100 test accuracy after global MLP-channel pruning for different budget ratios K/N. Panel (a) uses coupled weight-magnitude ranking, while panel (b) uses the final learned allocation μ. Mean ± sample std. over three seeds.

NuMuon compression and matching. At evaluation, NuMuon compresses only the 12 MLP weight matrices using truncated SVD; attention weights remain dense. We distribute the retained rank budget as evenly as possible across matrices, with minimum rank 4. At each compression level, we choose the NuMuon rank so that the resulting whole-model parameter count is as close as possible to that of the corresponding structured-pruned model.

Table 5 reports the exact post-training compression results corresponding to Fig. 2.

E.3 Effect of the Global Budget

We ablate the global budget K on the CIFAR-100 post-training compression setup of Appendix E.2.

Setup. We use the same six-block ViT, 200-epoch training recipe, and optimization settings as in Appendix E.2. We train BMuon with $\alpha = 1/2$ and $K/N \in \{0.1, 0.2, \ldots, 0.9\}$, with no budget schedule and holding all other hyperparameters fixed. After training, we perform global MLP-channel pruning without recovery, ranking channels either by coupled weight magnitude, $s_j^\ell = \sqrt{\|W_{1,j:}^\ell\|_2^2 + \|W_{2,:,j}^\ell\|_2^2}$, or by the final learned allocation μ.

Figure 6 shows that the budget directly controls the compression level favored during training. With magnitude pruning, models remain robust approximately up to pruning a fraction $1 - K/N$ of the channels, and degrade sharply beyond this point. The same pattern holds with μ ranking, but the

degradation is generally more gradual, further showing that μ provides a useful channel-importance signal. For the most restrictive budgets, particularly $K/N = 0.1$ and 0.2, the accuracy of the unpruned model also decreases slightly, indicating a tradeoff between aggressive compressibility and performance before pruning.

E.4 Gradual Structured Pruning Experiment

All gradual-pruning experiments use seeds $0, 1, 2$. Only MLP hidden channels are pruned, and all surviving channels are ranked globally across blocks at each pruning event. We enforce at least four surviving channels per block. Muon-mag and BMuon-mag use the coupled magnitude score $s_j^\ell = \sqrt{\|W_{1,j:}^\ell\|_2^2 + \|W_{2,:,j}^\ell\|_2^2}$, while BMuon-$\mu$ uses $s_j^{\ell,\mu} = \mu_j^\ell$. After pruning, training resumes from the surviving weights and optimizer state; pruned channels never return.

BMuon schedule. On both CIFAR-100 and ImageNet-1K, BMuon is applied only to the MLP matrices, with $\alpha = 1/2$, one MM step every five optimizer steps, and $\varepsilon = 10^{-6}$. At the start of each training phase i, let $N_{\mathrm{start},i}$ denote the current number of surviving channels and let $N_{\mathrm{end},i}$ denote the number of channels to retain at the next pruning event. During the first half of phase i, the BMuon budget is linearly annealed from $N_{\mathrm{start},i}$ to $N_{\mathrm{end},i}$; during the second half, it is held fixed at $N_{\mathrm{end},i}$. At the end of the phase, pruning reduces the network to exactly $N_{\mathrm{end},i}$ surviving channels. After pruning, the allocation coordinates μ of all surviving channels are reset to one.

CIFAR-100. We use a six-block ViT with 32×32 inputs, 4×4 patches, embedding dimension 112, four attention heads, and MLP width 448, giving 2688 prunable channels. Training lasts 200 epochs with batch size 256. Muon uses learning rate 0.02, momentum 0.95 without Nesterov, weight decay 0.05, and five Newton–Schulz steps; auxiliary parameters use AdamW with learning rate 3×10^{-4} and weight decay 0.05. We use five epochs of linear warmup followed by cosine decay to 1% of the initial learning rate. Training uses random cropping and horizontal flipping, RandAugment $(2, 9)$, random erasing with probability 0.1, and Mixup/CutMix with parameters $0.8/1.0$ and total probability 0.5.

For the 90% experiment, cumulative pruning levels of $20, 40, 60, 75, 85$, and 90% are applied after epochs $30, 60, 90, 120, 150$, and 180, respectively, followed by 20 recovery epochs. The corresponding surviving channel counts are $2150, 1613, 1075, 672, 403$, and 269. For the 95% experiment, cumulative pruning levels of $20, 40, 60, 75, 85, 90$, and 95% are applied after epochs $26, 52, 78, 104, 130, 155$, and 180, respectively, followed by 20 recovery epochs. The corresponding surviving channel counts are $2150, 1613, 1075, 672, 403, 269$, and 134.

ImageNet-1K. We use the same ViT-S/16 architecture, ImageNet-1K data pipeline, base optimizer hyperparameters, and 150-epoch learning-rate schedule as in Appendix E.1. Unlike the post-training pruning experiment, here physical MLP-channel removal is interleaved with training according to the gradual-pruning protocol described above.

For the 90% experiment, cumulative pruning levels of $20, 40, 60, 70, 80$, and 90% are applied after epochs $22, 44, 66, 87, 108$, and 129, respectively, followed by 21 recovery epochs. The corresponding surviving channel counts are $14{,}746, 11{,}059, 7{,}373, 5{,}530, 3{,}686$, and $1{,}843$. For the 95% experiment, cumulative pruning levels of $20, 40, 60, 70, 80, 90$, and 95% are applied after epochs $19, 38, 57, 76, 95, 114$, and 132, respectively, followed by 18 recovery epochs. The corresponding surviving channel counts are $14{,}746, 11{,}059, 7{,}373, 5{,}530, 3{,}686, 1{,}843$, and 922.

Evaluation. For CIFAR-100, we report final Top-1 accuracy on the test set at epoch 200. For ImageNet-1K, we report final Top-1 accuracy on the full 50,000-image validation set at epoch 150. In both cases, evaluation is performed after the final recovery stage. Reported means and standard deviations are over seeds $0, 1, 2$.

Architecture controls. For each BMuon-μ pruning run, we record the final per-block MLP widths. To evaluate the selected architecture on its own, we instantiate a new model with these widths and train it from scratch with standard Muon, without pruning, using the same training recipe and horizon. As a baseline, we also train a uniform-width model from scratch with standard Muon,

Table 6: **BMuon-μ architectures on CIFAR-100.** Per-block MLP widths for blocks 1–6, selected at the end of gradual pruning.

Pruning	Seed 0	Seed 1	Seed 2
90%	$[119, 94, 18, 4, 9, 25]$	$[120, 95, 20, 4, 13, 17]$	$[114, 99, 18, 6, 13, 19]$
95%	$[80, 37, 4, 5, 4, 4]$	$[78, 39, 5, 4, 4, 4]$	$[79, 37, 5, 5, 4, 4]$

Table 7: **BMuon-μ architectures on ImageNet-1K.** Per-block MLP widths for blocks 1–12, selected at the end of gradual pruning.

Pruning	Seed	MLP widths, blocks 1–12
90%	0	$[250, 168, 194, 164, 159, 153, 141, 140, 106, 122, 177, 69]$
	1	$[254, 174, 204, 197, 151, 132, 141, 139, 98, 135, 153, 65]$
	2	$[245, 194, 216, 199, 158, 119, 131, 137, 104, 121, 159, 60]$
95%	0	$[181, 120, 120, 106, 100, 67, 58, 43, 40, 34, 39, 14]$
	1	$[170, 121, 133, 127, 91, 60, 62, 53, 34, 22, 43, 6]$
	2	$[156, 110, 107, 99, 100, 73, 59, 66, 37, 40, 55, 20]$

with its total MLP channel count matched as closely as possible to the corresponding architecture discovered by BMuon-μ. Seed pairing is one-to-one: the architecture discovered with seed i is retrained from scratch using seed i. Thus, for seeds $0, 1, 2$, we perform three retraining runs rather than a 3×3 crossing of discovery and retraining seeds.

The selected architectures are reported in Tables 6 and 7.

On CIFAR-100, the uniform baseline uses width 45 in each of the six blocks at 90% pruning and width 22 at 95%, giving 270 and 132 total MLP channels, respectively, compared with 269 and 134 channels in the corresponding BMuon-μ architectures. On ImageNet-1K, the uniform baseline uses width 154 in each of the 12 blocks at 90% pruning and width 77 at 95%, giving 1848 and 924 channels, compared with 1843 and 922 channels in the corresponding BMuon-μ architectures.

E.5 LLaMA/FineWeb Experimental Details

Implementation provenance. Our LLaMA/FineWeb experiments are based on the LLaMA-1B setup of Muon+ (Zhang et al., 2026)[1]. We adopt its architecture and training recipe, reducing only the number of Transformer blocks from 24 to 17 to obtain a $986{,}746{,}880$-parameter model; all other architectural hyperparameters are unchanged. The Muon+ codebase builds on Muon (Jordan et al., 2024) and the `modded-nanogpt` training framework[2]. We retain its LLaMA-2 tokenizer and FineWeb preprocessing and sharding conventions, Muon parameter routing, auxiliary AdamW-style updates for non-Muon parameters, momentum/Nesterov convention, five-step Newton–Schulz polar approximation, matrix learning-rate shape/RMS scaling, warmup–cosine schedule, and BF16 training conventions. For our experiments, we integrate BMuon into the SwiGLU MLPs.

Model and data. We use a 17-layer pre-norm LLaMA-style causal decoder with hidden dimension 2048, 32 attention and 32 key–value heads, SwiGLU width 5461, context length 4096, vocabulary size $32{,}000$, RMSNorm ($\epsilon = 10^{-5}$), and RoPE with $\theta = 10{,}000$. The token embedding and language-model head are untied. The model has $986{,}746{,}880$ parameters. Each SwiGLU layer contains gate, up, and down projections $W_g, W_u \in \mathbb{R}^{5461\times 2048}$ and $W_d \in \mathbb{R}^{2048\times 5461}$, giving $N = 17 \times 5461 = 92{,}837$ structural MLP channels in total. We train on $13{,}090{,}422{,}784$ tokens from the `sample-100BT` configuration of FineWeb, using the LLaMA-2 tokenizer.

Training. Training uses BF16 with an effective global batch of 512 sequences, corresponding to $2{,}097{,}152$ tokens per optimizer step. We use seeds $\{0, 1, 2\}$ for all methods. For Muon matrices, we

[1] `https://github.com/K1seki221/MuonPlus`

[2] `https://github.com/KellerJordan/modded-nanogpt`

Table 8: **Post-training structured pruning of the 1B-parameter LLaMA-style model.** Validation perplexity after global SwiGLU-channel pruning. Entries report mean ± sample std. over seeds $\{0, 1, 2\}$. No post-pruning recovery is performed.

MLP pruning	Muon		BMuon		
	Magnitude	Activation	Magnitude	Activation	μ
0%	10.921 ± 0.033	10.921 ± 0.033	12.843 ± 0.051	12.843 ± 0.051	12.843 ± 0.051
10%	16.100 ± 1.411	12.701 ± 0.229	12.843 ± 0.051	12.843 ± 0.051	12.843 ± 0.051
20%	22.944 ± 3.728	26.249 ± 1.974	12.843 ± 0.051	12.843 ± 0.051	12.843 ± 0.051
30%	40.241 ± 8.109	62.840 ± 7.917	12.843 ± 0.051	12.843 ± 0.051	12.843 ± 0.051
40%	76.861 ± 16.225	251.840 ± 55.803	12.843 ± 0.051	12.843 ± 0.051	12.843 ± 0.051
50%	160.049 ± 43.632	1441.609 ± 641.338	12.843 ± 0.051	12.843 ± 0.051	12.843 ± 0.051
60%	283.004 ± 52.645	2984.678 ± 1476.332	12.843 ± 0.051	12.843 ± 0.051	12.843 ± 0.051
70%	615.949 ± 131.121	3543.424 ± 1733.400	12.843 ± 0.051	12.843 ± 0.051	12.843 ± 0.051
80%	2486.429 ± 1684.264	5290.062 ± 2508.521	13.342 ± 0.178	12.857 ± 0.053	12.854 ± 0.053
90%	28140.823 ± 19185.679	8382.535 ± 5262.682	3091.740 ± 1717.202	245.636 ± 42.251	1168.909 ± 351.685

use learning rate 0.02, momentum 0.95 with Nesterov momentum, five Newton–Schulz iterations, decoupled weight decay 0.1, gradient clipping at 1.0, a 10% linear warmup, and cosine decay to zero. Embeddings, normalization parameters, and the language-model head use the auxiliary AdamW-style branch with $\beta_1 = \beta_2 = 0.95$ and $\epsilon = 10^{-8}$. In the Muon baseline, attention and MLP matrices use Muon. For BMuon, the three SwiGLU projection matrices in each block are instead updated by BMuon, while attention matrices continue to use standard Muon.

BMuon configuration. We use $\alpha = 1/2$ and a budget $K = 18{,}567 \approx 0.2N$ throughout training, with no budget schedule. The allocation is updated every five optimizer steps with one MM iteration, using a numerical floor $\epsilon = 10^{-6}$.

Post-training pruning. We prune only SwiGLU hidden channels. Pruning channel j in layer ℓ jointly removes $W_g^\ell[j,:]$, $W_u^\ell[j,:]$, and $W_d^\ell[:,j]$. For each pruning ratio $r \in \{0, 0.1, \ldots, 0.9\}$, we restore the unpruned checkpoint and prune directly to ratio r. Scores are ranked globally across all Transformer blocks, with a minimum of 10 surviving channels per layer.

For both Muon and BMuon, the magnitude score is

$$s_{\ell j}^{\mathrm{mag}} = \left(\|W_g^\ell[j,:]\|_2^2 + \|W_u^\ell[j,:]\|_2^2 + \|W_d^\ell[:,j]\|_2^2\right)^{1/2}.$$

For BMuon, we additionally use

$$s_{\ell j}^{\mu} = \mu_j^\ell.$$

For the activation-aware criterion, let $h_{\ell j}(x) = \mathrm{SiLU}\big(W_g^\ell[j,:]x\big)\,\big(W_u^\ell[j,:]x\big)$ denote the hidden activation of channel j. We score it by

$$s_{\ell j}^{\mathrm{act}} = \sqrt{\mathbb{E}_x[h_{\ell j}(x)^2]}\,\|W_d^\ell[:,j]\|_2.$$

Activation statistics are accumulated in FP32 over $1{,}048{,}576$ calibration tokens. Each pruned model is evaluated without recovery on $5{,}046{,}272$ validation tokens.