GABR: GAMMA-AWARE BLOCK ROTATION FOR MXFP4 LLM QUANTIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Rotation-based post-training quantization reduces outliers by inserting orthogonal transforms and fusing them into adjacent weights. For the residual-stream rotation R1, however, the learned scaling parameter γ of RMSNorm does not commute with R1 in general, so existing offline methods first fold it into the weights of subsequent linear layers. This operation can amplify magnitude disparities within quantization blocks and degrade the quantized model. We propose Gamma-aware Block Rotation (GaBR). GaBR first permutes channels to group similar $|\gamma|$ values within quantization blocks. It then decomposes the scaling parameters into a block-constant G retained in RMSNorm and a residual D fused into weights, and alternately optimizes G and a block-diagonal R1. With a 32×32 block-diagonal rotation, retaining one scale per 32 channels in RMSNorm remains exactly equivalent in full precision. Extensive experiments on seven Llama and Qwen models under MXFP4 W4A4 quantization show that GaBR consistently improves upon existing rotation-based PTQ methods.

1 INTRODUCTION

Post-training quantization (PTQ) is a practical route to reducing the memory footprint, bandwidth cost, and compute of large language models (LLMs), but activation outliers make low-bit quantization particularly challenging. QuaRot enables W4A4 quantization with fixed orthogonal rotations that redistribute outliers; SpinQuant learns such rotations from calibration data; and OSTQuant jointly learns orthogonal and scaling transformations for better distribution fitting (Ashkboos et al., 2024; Liu et al., 2025; Hu et al., 2025). Together, these studies show that rotations can reduce the impact of activation outliers on low-bit quantization.

Following QuaRot, rotations are commonly divided into four types, denoted R1–R4. R1 is applied to the global residual stream, R2 couples the attention value and output projections, R3 rotates queries and keys after rotary position embedding, and R4 rotates the intermediate MLP activation before the down projection. When considering R1, the rotation can remain in the inference graph, as in the online transformations used by FlatQuant and DuQuant++, or it can be absorbed into adjacent weights before deployment, as in OSTQuant and PeRQ (Sun et al., 2025; Lin et al., 2026; Hu et al., 2025; Sanjeet et al., 2026). We target the latter setting because offline fusion adds no rotation operator or latency at inference. The RMS normalization itself is compatible with an orthogonal R1 rotation, but its channel-wise affine scale γ generally does not commute with R1. Thus, leaving γ in RMSNorm while applying the usual paired weight updates changes the full-precision function.

Existing offline rotation methods therefore first absorb γ into the downstream weights and then introduce R1. Although this is exact in floating point, each γ_j scales a different column of the following linear layer's weight matrix. Fusion can therefore enlarge the relative magnitude within a quantization block, causing the largest entries to dominate its shared scale and reducing the effective resolution of smaller entries. Our controlled experiments show that this full-fusion step alone can increase quantization error and degrade the accuracy of the quantized model.

The need for full γ fusion above arises because a dense R1 generally does not commute with the channel-wise scale. MXFP4 is a commonly used microscaling format in which every 32 consecutive values share a scale (Rouhani et al., 2023). For this format, BRQ and DuQuant++ show that aligning 32×32 block-diagonal rotations with the scaling groups can outperform a dense rotation (Shao et al.,

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.

2026; Lin et al., 2026). This block structure provides an exception: one scale shared within each rotation block commutes with the rotation. We can therefore split γ into a block-constant part G and a remaining channel-wise part D, retain G in RMSNorm, and fuse only D into the weights of the following linear layers while preserving exact full-precision equivalence.

Based on this observation, we propose Gamma-aware Block Rotation (GaBR). GaBR first sorts channels by their cross-layer mean $|\gamma|$, placing similar magnitudes in the same MXFP4 group, and then jointly coordinates the block-constant scale G with aligned 32×32 orthogonal rotation blocks. During R1 training, the gamma-aware parameterization prevents the complete RMSNorm affine scale from being forced into the weights of the following linear layers and optimizes the retained block scale together with the rotation under the deployed quantizer. This directly mitigates the adverse quantization effect introduced by RMSNorm while the learned R1 rotation is baked into the model only once.

Our contributions are exactly threefold:

1. **Gamma-aware permutation.** We find that fully fusing γ into downstream weights can degrade quantized-model performance. We permute channels so that similar $|\gamma|$ values fall into the same 32-channel MXFP4 block, reducing the adverse quantization effect of fusing the remaining gamma factors into downstream weights.
2. **Block-wise restoration to RMSNorm.** We factor the signed affine vector into a block-constant G retained at RMSNorm and a signed residual D fused into weights. By matching G to a block-diagonal rotation, $GR = RG$, which preserves exact high-precision equivalence while giving the block quantizer a better-conditioned representation.
3. **Alternating optimization.** We design an algorithm that alternately updates the retained block scale G with the orthogonal R1 rotation fixed, and then updates the rotation with G fixed. Repeating these steps lets the scale and rotation adapt to each other during training.

2 Related Work

Equivalent transformations for LLM PTQ. Post-training quantization commonly changes the representation of a pretrained network before rounding while preserving its real-valued function. SmoothQuant uses a diagonal equivalence transform to migrate activation outliers into weights (Xiao et al., 2023). OmniQuant calibrates learnable clipping and equivalent transformations (Shao et al., 2024), while AffineQuant extends the search space from diagonal scaling to invertible affine transforms (Ma et al., 2024). These methods establish that float-equivalent parameterizations can have substantially different low-precision behavior. Transformation design is complementary to adaptive rounding: Qronos sequentially corrects weight-, activation-, and preceding-layer quantization errors and can be composed with equivalent transforms (Zhang et al., 2026).

Rotation and permutation methods. QuaRot uses fixed Hadamard rotations to suppress outliers while preserving the model function (Ashkboos et al., 2024); SpinQuant learns orthogonal rotations with a quantization-aware objective (Liu et al., 2025); and OSTQuant jointly optimizes orthogonal and scaling transformations at multiple sites (Hu et al., 2025). FlatQuant instead learns structured affine transforms that remain on parts of the online activation path (Sun et al., 2025). DuQuant composes a block-diagonal rotation, a zigzag channel permutation, and a second block rotation to redistribute outliers both within and across rotation blocks (Lin et al., 2024). PeRQ gives a complementary analysis: the effectiveness of a block Hadamard rotation depends on the blockwise ℓ_1 mass distribution, and its MassDiff permutation uses calibration activations to balance that mass before rotation (Sanjeet et al., 2026). PeRQ identifies permutation-equivariant regions in which this calibrated permutation can be absorbed into model tensors, although its principal configurations retain an online block Hadamard at the MLP down-projection input.

Microscaling FP4 quantization. MXFP4 partitions tensors into groups of 32 values that share a power-of-two E8M0 scale (Rouhani et al., 2023). This makes the placement and distribution of values inside each group part of the quantization problem. MR-GPTQ combines block-wise Hadamard transforms with format-specific weight reconstruction for FP4 (Egiazarian et al., 2026), while BRQ diagnoses the mismatch between global rotation and MXFP4 scaling and advocates rotations aligned

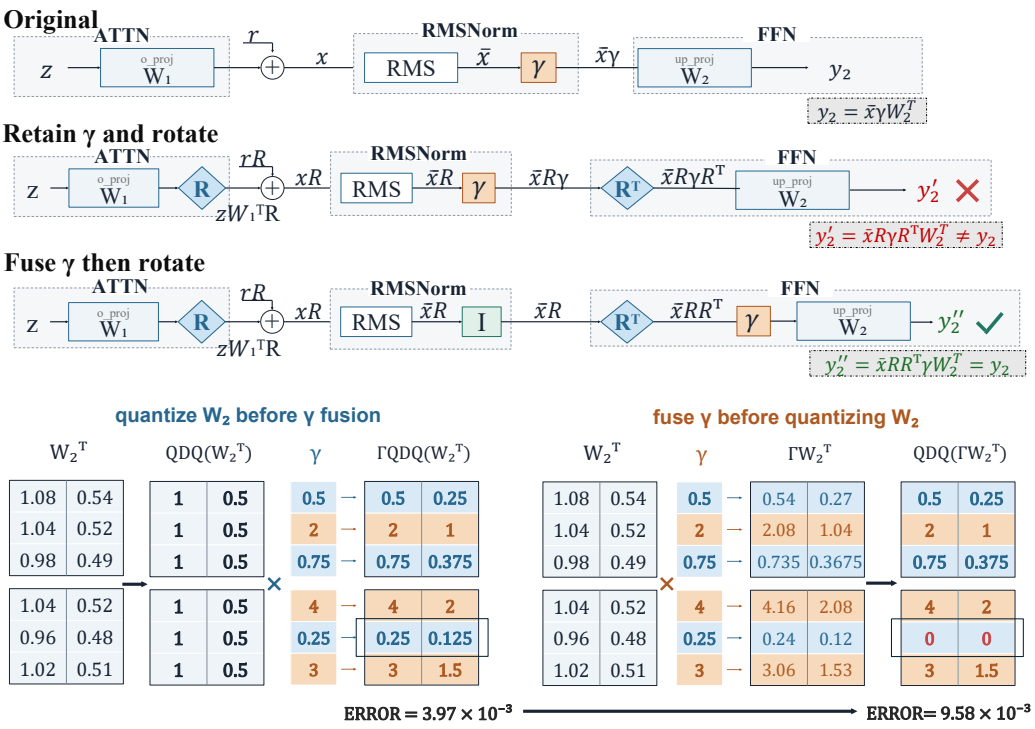


Figure 1: Why conventional R1 rotation first fuses γ into the following linear layer. Top: under the standard paired R1 updates, leaving γ in RMSNorm generally breaks full-precision equivalence, whereas fusing it into the following linear layer restores equivalence. Bottom: this fusion can worsen quantization by increasing weight-block reconstruction error, as shown by the two illustrative quantize–dequantize paths.

with the 32-value group (Shao et al., 2026). DuQuant++ similarly aligns an outlier-aware block-diagonal rotation with the MXFP4 group and replaces the original DuQuant R_1–P–R_2 sequence with a single online 32×32 rotation (Lin et al., 2026).

3 Background and Motivation

3.1 RMSNorm and rotation equivalence

Figure 1 illustrates how RMSNorm must be handled to preserve full-precision equivalence when introducing an offline R1 rotation. Let W_1 and W_2 denote the projections before and after RMSNorm. In the *Original* path, the vector entering RMSNorm and the output are

$$x = zW_1^\top + r, \qquad y = \mathrm{RMSNorm}_\Gamma(x)W_2^\top = \bar{x}\Gamma W_2^\top, \tag{1}$$

where r is the incoming residual, $\Gamma = \mathrm{diag}(\gamma)$, and

$$\bar{x} = \frac{x}{\sqrt{d^{-1}\|x\|_2^2 + \epsilon}}, \qquad x \in \mathbb{R}^d. \tag{2}$$

In the *Retain γ and rotate* path, we attempt to absorb an orthogonal R1 rotation R into the adjacent weights offline, so that no explicit rotation operator is added at inference. The paired updates are

$$W_1' = R^\top W_1, \qquad W_2' = W_2 R. \tag{3}$$

With the residual represented in the same rotated coordinates, the first update gives $x' = z(W_1')^\top + rR = xR$. Since $\|xR\|_2 = \|x\|_2$, the RMS normalization itself satisfies $\overline{xR} = \bar{x}R$. However, retaining the affine scale Γ in RMSNorm gives the crossed-out result in Figure 1:

$$\begin{aligned} y' &= \mathrm{RMSNorm}_\Gamma(xR)(W_2R)^\top \\ &= \bar{x}R\Gamma R^\top W_2^\top \\ &\neq \bar{x}\Gamma W_2^\top = y. \end{aligned} \tag{4}$$

Table 1: A Qwen3-8B weight-block example: decoder layer 6, up-projection row 1737, input channels 2272–2303 (zero-based). Weight entries and reconstructions are shown as magnitudes; displayed values are rounded. Block MSE uses all 32 entries, with each condition's unquantized weight block as its reference.

Quantity	Before fusion (W)	After fusion ($W\Gamma$)
$\lvert w_{2276}\rvert$ ($\gamma_{2276} = 8.25$)	0.1128	0.9297
$\lvert w_{2291}\rvert$ ($\gamma_{2291} = 0.34375$)	0.04297	0.01477
Shared MXFP4 scale	$2^{-6} = 0.015625$	$2^{-3} = 0.125$
Reconstructed magnitude, channel 2291	0.046875	**0**
Weight-block MSE↓	1.65×10^{-5}	1.03×10^{-3}

The channel-wise scale Γ lies between R and $R^\top$, preventing their cancellation unless $R\Gamma = \Gamma R$.

In the *Fuse γ then rotate* path, conventional offline rotation methods first fuse Γ into the downstream linear layer and set the RMSNorm affine scale to the identity before introducing the rotation (Ashkboos et al., 2024; Liu et al., 2025; Hu et al., 2025):

$$\widetilde{W}_2 = W_2\Gamma, \qquad y = \bar{x}\widetilde{W}_2^\top. \tag{5}$$

The rotation is then incorporated into the two linear layers as $W_1' = R^\top W_1$ and $\widetilde{W}_2' = \widetilde{W}_2 R$. Because RMSNorm no longer contains the intervening Γ, this path gives the check-marked result in Figure 1:

$$\begin{aligned} y' &= (\bar{x}R)(\widetilde{W}_2 R)^\top \\ &= \bar{x}RR^\top \widetilde{W}_2^\top \\ &= \bar{x}\Gamma W_2^\top = y. \end{aligned} \tag{6}$$

Thus, fusing γ before introducing the rotation restores mathematical equivalence under the standard paired orthogonal updates.

3.2 Quantization error after gamma fusion

While this fusion restores rotation equivalence, it also changes the weight distribution seen by the quantizer: each downstream weight column is scaled by its corresponding RMSNorm coefficient, $W_{:,j} \leftarrow \gamma_j W_{:,j}$. The lower paths of Figure 1 compare quantizing W_2 before and after this fusion against the same full-precision fused weight. Unequal channel scaling can increase within-block magnitude disparities and reconstruction error. In the illustrative example, MSE rises from 3.97×10^{-3} to 9.58×10^{-3}.

We further verify this effect using an actual 32-entry Qwen3-8B weight block. Table 1 compares its MXFP4 quantization before and after γ fusion. Fusion enlarges one channel while shrinking another,

Table 2: Residual RMSNorm γ-fusion ablation under Direct W4A4 MXFP4 RTN; the LM head remains in BF16. Logit errors compare each quantized graph with its own quantization-disabled counterpart on two fixed, non-overlapping 256-token WikiText-2 test windows. NMSE is $\|z_q - z_{\text{fp}}\|_2^2 / \|z_{\text{fp}}\|_2^2$, reported as a percentage. MC5 averages zero-shot PIQA, ARC-Easy, ARC-Challenge, HellaSwag, and WinoGrande. Bold marks degradation after fusion.

Model	RMSNorm γ	MXFP4 PPL↓	Logit MSE↓	Logit NMSE (%)↓	MC5↑
Llama-2-7B	retained	7.116	2.170	16.661	63.93
	fused	**7.125**	**2.432**	**18.673**	64.09
Llama-3-8B	retained	8.271	1.202	10.618	67.33
	fused	**8.360**	**1.258**	**11.113**	**67.23**
Qwen3-8B	retained	11.835	2.198	13.430	66.17
	fused	**12.366**	**2.570**	**15.700**	**65.19**

raises the shared scale eightfold, and causes the smaller entry to quantize to zero. The measured block MSE increases from 1.65×10^{-5} to 1.03×10^{-3}.

Together, the local block example in Table 1 and the end-to-end comparison in Table 2 show that full γ fusion can increase quantization error and degrade model performance. In the latter, we change only whether residual-stream γ is retained in RMSNorm or fused into downstream weights under the same Direct W4A4 MXFP4 RTN setup. Fusion increases logit MSE and PPL on all three tested models.

4 GaBR: Gamma-aware Block Rotation

We propose Gamma-aware Block Rotation (GaBR) with two components: a gamma-aware permutation that groups similarly scaled channels within quantization blocks, and block-scale restitution that retains one scale per rotation block in RMSNorm.

4.1 Gamma-aware channel permutation

Before initializing the block scale G or optimizing the rotation R, we collect the affine vectors from all residual-stream RMSNorm layers, $\{\gamma_\ell\}_{\ell=1}^{L}$, and compute a global channel score, where L is the number of these layers:

$$s_j = \frac{1}{L} \sum_{\ell=1}^{L} |\gamma_{\ell,j}|. \tag{7}$$

We set $\pi = \mathrm{argsort}_{\downarrow}(s)$, so that $s_{\pi(1)} \geq s_{\pi(2)} \geq \cdots \geq s_{\pi(d)}$, and define $P = [e_{\pi(1)}, \ldots, e_{\pi(d)}]$. Thus xP lists residual channels from the largest to the smallest mean $|\gamma|$. This ranking does not depend on a block size. We keep the same global P fixed throughout subsequent scale and rotation optimization. The permutation is applied consistently to all affected linear weights and normalization parameters to preserve the full-precision function. $|\gamma|$ is used only to score the permutation.

The permutation groups channels with similar cross-layer mean $|\gamma|$ within the same quantization block. This limits variation in the gamma factors absorbed by neighboring weight entries. Figure 2 illustrates how the grouping can mitigate the reconstruction error caused by γ fusion.

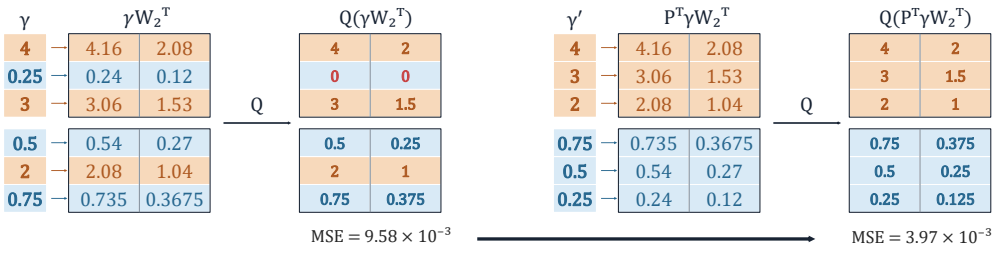


Figure 2: Gamma-guided channel permutation. The same γ-fused weights of the following linear layer are quantized in the original order (left) and after a consistent channel permutation (right). Grouping similarly scaled channels can reduce MXFP4 reconstruction error.

4.2 Block-scale restitution by commutativity

A common design for MXFP4 is to use block-diagonal rotations aligned with its 32-value quantization groups (Shao et al., 2026; Lin et al., 2026). We follow this design and partition the permuted channels into consecutive blocks of $B = 32$. In permuted coordinates, the affine matrix is

$$\Gamma_P = P^\top \Gamma P. \tag{8}$$

For each RMSNorm site, we factor it as

$$\Gamma_P = GD, \qquad G = \mathrm{Diag}(g_1 \mathbf{1}_B, \ldots, g_K \mathbf{1}_B), \qquad D = G^{-1}\Gamma_P, \tag{9}$$

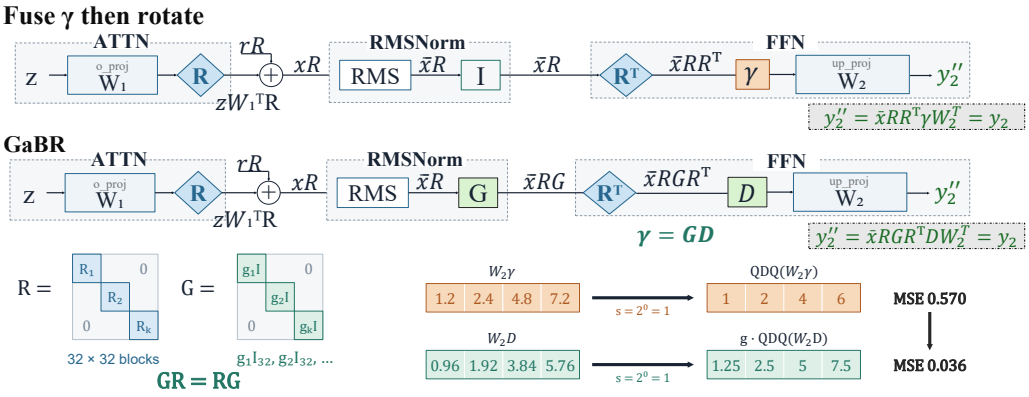


Figure 3: Block-wise restitution of RMSNorm γ. Top: both full fusion and $\Gamma = GD$ preserve the full-precision output; because the aligned block-diagonal rotation commutes with block-constant G, GaBR retains G in RMSNorm and fuses only D into the following linear layer. Bottom: the same illustrative block is quantized directly or after rescaling by $g = 1.25$, which is restored after dequantization. Both paths select the power-of-two MXFP4 scale $s = 1$ (block maxima 7.2 and 5.76), while reconstruction MSE falls from 0.570 to 0.036.

where $K = d/B$, $g_b > 0$, and D retains every residual sign and within-block ratio. We parameterize the residual rotation as

$$R = \text{blockdiag}(R_1, \ldots, R_K), \qquad R_b \in \mathbb{R}^{32\times 32}, \quad R_b^{\top} R_b = I \quad (b = 1, \ldots, K). \tag{10}$$

Because G is scalar within each aligned block,

$$GR = RG. \tag{11}$$

As shown by the block matrices in Figure 3, this commutation lets a scalar remain in RMSNorm for every 32 channels instead of fusing the entire Γ into the following weights.

To preserve equivalence, we keep G in the transformed RMSNorm output,

$$\widetilde{h} = \bar{x}PGR, \tag{12}$$

and update the weight matrix of each following linear layer as

$$\widetilde{W} = WPDR. \tag{13}$$

Then

$$\widetilde{h}\widetilde{W}^{\top} = \bar{x}PGRR^{\top}DP^{\top}W^{\top} \tag{14}$$

$$= \bar{x}PGDP^{\top}W^{\top} = \bar{x}\Gamma W^{\top} = y. \tag{15}$$

Thus the full-precision function is unchanged while only D, rather than the complete Γ, is fused into the weights of the following linear layers. Figure 3 contrasts the two equivalent constructions and illustrates how retaining G can reduce MXFP4 quantization error.

4.3 Alternating block-scale and rotation optimization

Jointly optimizing scales and an orthogonal matrix is poorly conditioned: G changes the amplitude seen by the block quantizer, whereas R changes its direction. We therefore hold P fixed and alternate two phases.

Phase G: block-scale search. We initialize each $g_{\ell,b}$ to the ε-stabilized geometric mean of $|\gamma_{\ell,\pi(j)}|$ over its 32 permuted channels. With $R^{(t)}$ fixed, we search $g_{\ell,b}$ in the log domain around its current value and minimize a hard-MXFP4 two-sided reconstruction loss,

$$\mathcal{L}_G = \frac{\|\mathcal{Q}_A(A(G,R))\mathcal{Q}_W(W(G,R))^{\top} - AW^{\top}\|_F^2}{\|AW^{\top}\|_F^2 + \epsilon}. \tag{16}$$

The activation and weight quantizers use the deployed E2M1, block-32, power-of-two-scale rule. We update D inversely after selecting G.

Phase R: quantization-aware orthogonal update. With $G^{(t+1)}$ fixed, we update the orthogonal blocks R_b using the quantized language-model objective

$$\mathcal{L}_R = \mathcal{L}_{\mathrm{CE}} + \lambda_{\mathrm{KL}} \, \mathrm{KL}(p_{\mathrm{fp}} \| p_{\mathrm{q}}), \tag{17}$$

where the forward path applies MXFP4 fake quantization to both weights and activations. Cayley/Stiefel updates or orthogonal projection maintain $R_b^\top R_b \approx I$ for each block. After the alternating G/R updates, we retain G in RMSNorm, fuse the remaining transforms and compensations into adjacent weights, and then quantize the model.

5 Experiments

5.1 Experimental Setup

Models and quantization. We evaluate Llama-2-7B (Touvron et al., 2023), Llama-3-8B (Grattafiori et al., 2024), Llama-3.2-1B/3B (Meta AI, 2024), and Qwen3-1.7B/4B/8B (Yang et al., 2025). All quantized runs use OCP-style MXFP4 E2M1 weights and activations (W4A4), contiguous groups of 32 values, power-of-two floor scales, and 16-bit KV cache. Calibration uses WikiText-2 (Merity et al., 2017) sequences of length 2048.

Evaluation. We report full-window WikiText-2 perplexity. PIQA (Bisk et al., 2020), ARC-Easy and ARC-Challenge (Clark et al., 2018), HellaSwag (Zellers et al., 2019), WinoGrande (Sakaguchi et al., 2020), and BoolQ (Clark et al., 2019) are evaluated zero-shot; MMLU (Hendrycks et al., 2021) is evaluated 5-shot. We use normalized accuracy for PIQA, ARC, and HellaSwag, and accuracy for the remaining tasks. MC5 averages PIQA, ARC-Easy, ARC-Challenge, HellaSwag, and WinoGrande. Mean7 additionally includes BoolQ and MMLU.

Baselines. The main comparison includes a BF16 reference, direct RTN, PeRQ* and PeRQ† (Sanjeet et al., 2026), DartQuant (Shao et al., 2025), and OSTQuant (Hu et al., 2025), together with our GaBR-PeRQ and GaBR-OST methods. GaBR-PeRQ replaces PeRQ's residual-stream R1 with the GaBR R1 construction, while GaBR-OST makes the corresponding R1 change to OSTQuant. Each variant retains its parent method's other transforms (R2–R4, where applicable) and their settings. For DartQuant and OSTQuant, we distinguish the RTN recipes from variants using the same learned transforms with Qronos weight reconstruction. DuQuant++ (Lin et al., 2026) is reported separately in Appendix D. All quantized results follow the same MXFP4 evaluation protocol. Methods in the main comparison can retain fixed structured rotations at non-fusible sites; for example, PeRQ retains a block Hadamard at the MLP down-projection input. The grouping does not imply zero online computation or equal inference latency.

5.2 Main Results

5.2.1 Comparison with Baselines

Table 3 compares the two GaBR instantiations, GaBR-PeRQ and GaBR-OST, with existing quantization baselines across seven models. The upper and lower panels report WikiText-2 PPL and Mean7, respectively. The dashed boxes highlight the direct R1 comparisons within each parent method: PeRQ* versus GaBR-PeRQ, and OSTQuant + Qronos versus GaBR-OST. Compared with PeRQ*, GaBR-PeRQ improves PPL on six of seven models and Mean7 on all seven; compared with OSTQuant + Qronos, GaBR-OST improves PPL on all seven and Mean7 on five. These paired results show that the GaBR R1 construction generally improves both language-modeling and downstream performance without changing the other transforms of each parent method. PeRQ† obtains lower PPL on several models, particularly Qwen3, but has lower Mean7 than GaBR-PeRQ on six of seven models. Thus its PPL advantage does not consistently translate into downstream accuracy, whereas GaBR yields more consistent task-level results across the model suite. Appendix A gives all seven constituent task accuracies; Appendix D reports DuQuant++ separately.

5.2.2 Cross-Layer Gamma Consistency

Because GaBR uses a global residual-stream R1 rotation and one shared channel permutation across layers, a gamma-guided order is most useful when channels with large or small $|\gamma|$ remain broadly

Table 3: Main comparison under W4A4 MXFP4. Panel (a) reports WikiText-2 PPL; panel (b) reports Avg, the mean accuracy (%) over seven tasks. BF16 is the unquantized reference. Bold marks the better value in each pair: PeRQ* versus GaBR-PeRQ, and OSTQuant + Qronos versus GaBR-OST. Full task-level results are in Appendix A.

Model	BF16	Direct RTN	PeRQ†	PeRQ*	**GaBR-PeRQ**	DartQuant + Qronos	OSTQuant + Qronos	**GaBR-OST**
(a) WikiText-2 PPL↓								
Llama-2-7B	5.474	7.116	5.748	5.849	**5.845**	6.000	5.986	**5.870**
Llama-3-8B	6.138	8.271	7.172	7.071	**7.047**	7.276	7.242	**7.177**
Llama-3.2-1B	9.757	15.790	11.937	**11.833**	11.849	12.570	12.358	**12.073**
Llama-3.2-3B	7.817	10.288	8.930	8.788	**8.775**	9.370	9.131	**8.918**
Qwen3-1.7B	16.716	29.665	13.107	18.546	**18.383**	20.410	17.751	**17.173**
Qwen3-4B	13.663	18.659	10.583	15.555	**15.272**	16.140	14.239	**13.207**
Qwen3-8B	9.726	11.835	8.582	10.977	**10.751**	11.349	10.301	**10.059**
(b) Avg↑ (%)								
Llama-2-7B	66.86	60.47	61.00	63.98	**64.03**	63.44	62.66	**63.04**
Llama-3-8B	72.88	66.20	68.59	69.19	**69.69**	68.99	**69.71**	69.62
Llama-3.2-1B	55.79	48.78	44.32	44.85	**51.09**	50.63	50.89	**52.11**
Llama-3.2-3B	66.67	59.65	58.91	58.31	**62.73**	61.25	62.86	**62.97**
Qwen3-1.7B	63.59	54.64	48.99	56.03	**57.13**	54.81	**57.55**	56.52
Qwen3-4B	70.76	65.99	51.56	60.13	**66.38**	64.79	66.82	**67.04**
Qwen3-8B	74.21	69.26	71.48	70.78	**71.26**	69.45	72.15	**72.26**

consistent across layers. We therefore examine this channel-wise consistency. Figure 4 summarizes Pre-MLP γ statistics and permutation effects on Qwen3-1.7B, Qwen3-8B, and Llama-3-8B. Within each model, we order channels once by their mean $|\gamma|$ across Pre-MLP layers and use that same order for every heatmap row. The orange high-$|\gamma|$ channels remain largely on the left and the blue low-$|\gamma|$ channels on the right, forming vertical bands across layers. Thus, channels with relatively large or small $|\gamma|$ in one Pre-MLP layer tend to retain that tendency in other layers, although their ranks are not identical. Mean layer-pair Spearman correlations are 0.599–0.612. The lower curves show that the ordering also reduces W4A4 MXFP4 MLP output MSE in most displayed non-final layers. These per-layer diagnostics use clean inputs and test the permutation alone, not the full GaBR pipeline. The six-model comparison and statistics for the shared deployment ordering constructed from both norm types are in Appendix E.

5.3 Ablation Studies

Effect of the $\Gamma = GD$ factorization. We compare GaBR-OST, which retains block-constant G in RMSNorm, with a variant that fixes $G = I$ and thus fuses the full Γ into the following linear layers. Both use the gamma-guided permutation and train the block rotations and channel scales before the same Qronos quantization. Table 4 reports PPL, Mean7, and whole-model logit MSE; Appendix B gives all seven task accuracies. Mean7 (%) averages six zero-shot tasks and 5-shot MMLU. PPL and Mean7 for GaBR-OST use the main-table checkpoint, whereas logit MSE uses a separate pair of with-/without-G models trained under the same protocol. We compare their quantized logits against the same original BF16 model on two fixed, non-overlapping 256-token WikiText-2 test windows. With G, most models show lower PPL and logit MSE, whereas Mean7 has no uniform trend.

Descending gamma permutation versus alternative orderings. We compare the descending *Gamma-sort* order in Eq. (7) with the original *Identity* order, DuQuant-style *Zigzag* (Lin et al., 2024), and *Gamma-MassDiff*, a γ-score adaptation of PeRQ's greedy MassDiff (Sanjeet et al., 2026). Under W4A4 MXFP4 with rotation training and Qronos, Table 5 shows that Gamma-sort achieves the lowest PPL and highest Mean7 on Llama-3.2-1B and the highest Mean7 on Qwen3-8B, whereas Identity gives the lowest PPL on Qwen3-8B and wins both metrics on Llama-3-8B. The benefit of the

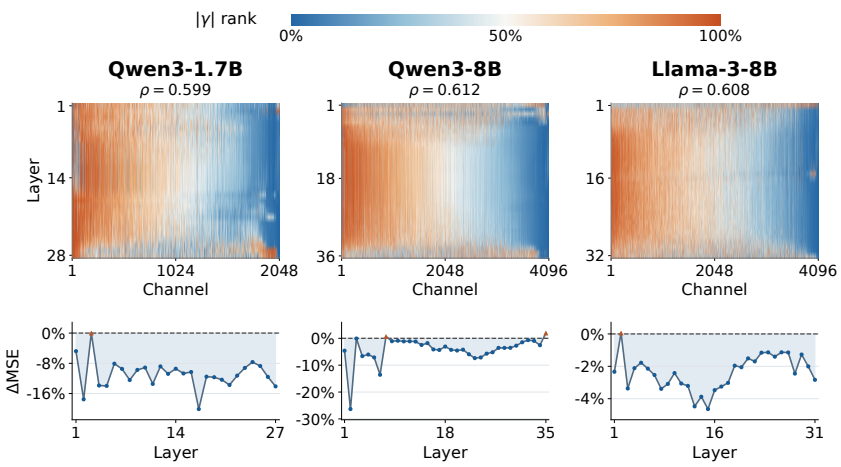


Figure 4: Pre-MLP diagnostics on three models. Channels are ordered once by cross-layer mean $|\gamma|$; each heatmap row uses this same order. Orange denotes higher and blue lower within-layer $|\gamma|$ rank, while ρ averages layer-pair Spearman correlations. Curves show signed local MLP ΔMSE (negative is better; Eq. 18). Only curves exclude the final layer, and their vertical ranges vary by model. These Pre-MLP-only diagnostics do not evaluate the shared deployment permutation.

Table 4: Effect of the $\Gamma = GD$ factorization under W4A4 MXFP4 and Qronos. Logit MSE uses the matched diagnostic runs described in the text. Bold marks the better value within each model and metric; ties at the reported precision are bold on both sides.

	GaBR-OST (G)			Without G		
Model	PPL↓	Mean7↑	Logit MSE↓	PPL↓	Mean7↑	Logit MSE↓
Llama-2-7B	**5.870**	63.04	1.073	5.916	**63.22**	**0.968**
Llama-3-8B	**7.177**	**69.62**	**0.807**	7.188	**69.62**	0.849
Llama-3.2-1B	**12.073**	**52.11**	**1.305**	12.141	50.83	1.423
Qwen3-8B	**10.059**	**72.26**	**1.487**	10.591	71.82	1.645

global ordering is therefore model-dependent. Gamma-sort uses the GaBR-OST main-table result; the alternatives use permutation-sweep runs. Appendix C gives the complete task-level results and a separate Direct-RTN control.

Table 5: Gamma-permutation comparison under W4A4 MXFP4 and Qronos. Each entry is PPL↓ / Mean7↑ (%). Bold marks the best value for each model and metric.

Model	Identity	Zigzag	Gamma-MassDiff	Gamma-sort
Llama-3-8B	**7.175 / 69.74**	7.231 / 69.42	7.226 / 69.38	7.177 / 69.62
Llama-3.2-1B	12.099 / 50.73	12.203 / 50.89	12.166 / 51.48	**12.073 / 52.11**
Qwen3-8B	**9.829** / 71.38	10.071 / 72.01	10.094 / 72.23	10.059 / **72.26**

6 Conclusion

We identified RMSNorm affine fusion as a hidden coordinate choice in block-scaled LLM quantization. A gamma-aware permutation aligns channels with MXFP4 groups; a block-constant G and a matched block-diagonal R commute, allowing one scale per 32 channels to remain at RMSNorm while preserving exact high-precision equivalence; and alternating scale/rotation updates adapt this factorization to the deployed quantizer. The resulting method consistently improves perplexity over its full-dimensional OSTQuant+Qronos counterpart across seven models, while the mixed task metrics delineate the remaining optimization problem.

AI USE STATEMENT

Generative AI tools were used to assist with language editing, code inspection, and experiment bookkeeping. The authors defined the method, verified the algebra, executed the experiments, and remain responsible for all claims and reported results.

ETHICS STATEMENT

This work uses publicly available pretrained models and benchmark datasets. It does not involve a human-subject study or the collection of private data. Quantization can reduce the memory and computational costs of deploying LLMs, but it does not remove undesirable behaviors inherited from the source models and may change how those behaviors appear. Quantized models should therefore undergo application-specific safety and robustness evaluation before deployment.

REPRODUCIBILITY STATEMENT

The main text specifies the gamma-aware permutation, the $\Gamma = GD$ factorization, alternating block-scale and rotation optimization, calibration data, MXFP4 settings, baselines, and evaluation protocol. The appendix provides implementation details, ablations, and individual downstream task accuracies. The accompanying source package includes GaBR-OST and GaBR-PeRQ implementations, baseline code, configurations, and evaluation scripts. The pretrained models and benchmark datasets are publicly available.

REFERENCES

Saleh Ashkboos, Amirkeivan Mohtashami, Maximilian L. Croci, Bo Li, Pashmina Cameron, Martin Jaggi, Dan Alistarh, Torsten Hoefler, and James Hensman. QuaRot: Outlier-free 4-bit inference in rotated LLMs. In *Advances in Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2404.00456.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. PIQA: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020. doi: 10.1609/aaai.v34i05.6239. URL https://ojs.aaai.org/index.php/AAAI/article/view/6239.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, 2019. URL https://aclanthology.org/N19-1300/.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try ARC, the AI2 reasoning challenge. 2018. URL https://arxiv.org/abs/1803.05457.

Vage Egiazarian, Roberto Castro, Denis Kuznedelev, Andrei Panferov, Eldar Kurtic, Shubhra Pandit, Alexandre Marques, Mark Kurtz, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. Bridging the gap between promise and performance for microscaling FP4 quantization. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/b87bb4f6346d727b265088235e5bc389-Abstract-Conference.html.

Aaron Grattafiori et al. The Llama 3 herd of models. 2024. URL https://arxiv.org/abs/2407.21783.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2009.03300.

Xing Hu, Yuan Cheng, Zhixuan Chen, Dawei Yang, Jiangyong Yu, XUCHEN, Zhihang Yuan, Zhe Jiang, and Sifan Zhou. OSTQuant: Refining large language model quantization with orthogonal and scaling transformations for better distribution fitting. In *International Conference on Learning Representations*, 2025.

URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/5cebc89b113920dbff7c79854ba765a3-Abstract-Conference.html.

Haokun Lin, Haobo Xu, Yichen Wu, Jingzhi Cui, Yingtao Zhang, Linzhan Mou, Linqi Song, Zhenan Sun, and Ying Wei. DuQuant: Distributing outliers via dual transformation makes stronger quantized LLMs. In *Advances in Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2406.01721.

Haokun Lin, Xinle Jia, Haobo Xu, Bingchen Yao, Xianglong Guo, Yichen Wu, Zhichao Lu, Ying Wei, Qingfu Zhang, and Zhenan Sun. DuQuant++: Fine-grained rotation enhances microscaling FP4 quantization. *arXiv preprint arXiv:2604.17789*, 2026. URL https://arxiv.org/abs/2604.17789.

Zechun Liu, Changsheng Zhao, Igor Fedorov, Bilge Soran, Dhruv Choudhary, Raghuraman Krishnamoorthi, Vikas Chandra, Yuandong Tian, and Tijmen Blankevoort. SpinQuant: LLM quantization with learned rotations. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=ogO6DGE6FZ.

Yuexiao Ma, Huixia Li, Xiawu Zheng, Feng Ling, Xuefeng Xiao, Rui Wang, Shilei Wen, Fei Chao, and Rongrong Ji. AffineQuant: Affine transformation quantization for large language models. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2403.12544.

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

Meta AI. Llama 3.2 model card, 2024. URL https://github.com/meta-llama/llama-models/blob/main/models/llama3_2/MODEL_CARD.md.

Bita Darvish Rouhani, Nitin Garegrat, Tom Savell, Ankit More, et al. OCP microscaling formats (MX) specification, version 1.0. Technical report, Open Compute Project, 2023. URL https://www.opencompute.org/documents/ocp-microscaling-formats-mx-v1-0-spec-final-pdf.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. WinoGrande: An adversarial Winograd schema challenge at scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020. URL https://ojs.aaai.org/index.php/AAAI/article/view/6399.

Sai Sanjeet, Ian Colbert, Pablo Monteagudo-Lago, Giuseppe Franco, Yaman Umuroglu, and Nicholas J. Fraser. Pushing the limits of block rotations in post-training quantization. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=nvehxSdMqg.

Wenqi Shao, Mengzhao Chen, Zhaoyang Zhang, Peng Xu, Lirui Zhao, Zhiqian Li, Kaipeng Zhang, Peng Gao, Yu Qiao, and Ping Luo. OmniQuant: Omnidirectionally calibrated quantization for large language models. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2308.13137.

Yuantian Shao, Yuanteng Chen, Peisong Wang, Jianlin Yu, Jing Lin, Yiwu Yao, Zhihui Wei, and Jian Cheng. DartQuant: Efficient rotational distribution calibration for LLM quantization. In *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2511.04063.

Yuantian Shao, Peisong Wang, Yuanteng Chen, Chang Xu, Zhihui Wei, and Jian Cheng. Block rotation is all you need for MXFP4 quantization. In *Forty-third International Conference on Machine Learning*, 2026. URL https://icml.cc/virtual/2026/poster/62818.

Yuxuan Sun, Ruikang Liu, Haoli Bai, Han Bao, Kang Zhao, Yuening Li, Jiaxin Hu, Xianzhi Yu, Lu Hou, Chun Yuan, Xin Jiang, Wulong Liu, and Jun Yao. FlatQuant: Flatness matters for LLM quantization. In *International Conference on Machine Learning*, 2025. URL https://arxiv.org/abs/2410.09426.

Hugo Touvron et al. Llama 2: Open foundation and fine-tuned chat models. 2023. URL https://arxiv.org/abs/2307.09288.

Guangxuan Xiao, Ji Lin, Mickael Seznec, Hao Wu, Julien Demouth, and Song Han. SmoothQuant: Accurate and efficient post-training quantization for large language models. In *International Conference on Machine Learning*, 2023. URL https://arxiv.org/abs/2211.10438.

An Yang et al. Qwen3 technical report. 2025. URL https://arxiv.org/abs/2505.09388.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019. URL https://aclanthology.org/P19-1472/.

Shihao Zhang, Haoyu Zhang, Ian Colbert, and Rayan Saab. Qronos: Correcting the past by shaping the future... in post-training quantization. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/8d22a09f18d9fc85dcbe6f482e13e1d7-Abstract-Conference.html.

A DATASET-LEVEL DOWNSTREAM RESULTS

Table 6: Dataset-level results under W4A4 MXFP4. PPL is WikiText-2 perplexity; Avg is Mean7 (%). The six tasks before MMLU are zero-shot, and MMLU is 5-shot. Task metrics follow Section 5.1.

Model	Method	PPL↓	PIQA	ARC-E	ARC-C	Hella	Wino	BoolQ	MMLU	Avg↑
Llama-2 7B	BF16	5.474	78.94	74.45	46.08	75.99	69.06	77.80	45.66	66.86
	Direct RTN	7.116	76.93	67.38	39.68	70.61	65.04	71.56	32.08	60.47
	PeRQ*	5.849	76.99	71.97	41.04	72.40	67.88	75.90	41.68	63.98
	PeRQ†	5.748	76.55	65.40	39.51	71.95	63.46	73.09	37.07	61.00
	GaBR-PeRQ	5.845	76.82	72.73	42.92	72.24	68.35	74.40	40.72	64.03
	DartQuant + Qronos	6.000	76.33	70.75	42.49	71.36	67.48	75.17	40.53	63.44
	OSTQuant + Qronos	5.986	76.39	69.23	42.24	70.88	65.43	73.88	40.55	62.66
	GaBR-OST	5.870	76.55	71.00	42.06	71.78	65.35	74.74	39.77	63.04
Llama-3 8B	BF16	6.138	81.01	77.78	53.07	79.14	72.77	81.04	65.34	72.88
	Direct RTN	8.271	77.64	70.50	46.67	73.82	68.03	73.64	53.07	66.20
	PeRQ*	7.071	78.18	74.79	47.27	74.64	70.40	79.14	59.87	69.19
	PeRQ†	7.172	77.80	76.26	49.49	76.34	68.43	74.50	57.28	68.59
	GaBR-PeRQ	7.047	78.62	76.18	49.66	74.67	69.30	80.00	59.43	69.69
	DartQuant + Qronos	7.276	76.99	75.00	47.70	74.20	70.48	79.72	58.87	68.99
	OSTQuant + Qronos	7.242	78.07	78.16	50.34	74.59	68.35	79.60	58.84	69.71
	GaBR-OST	7.177	78.67	76.14	48.21	74.49	72.22	79.76	57.84	69.62
Llama-3.2 1B	BF16	9.757	74.37	60.52	36.26	63.65	60.93	63.91	30.89	55.79
	Direct RTN	15.790	68.61	52.15	29.61	53.43	55.64	56.54	25.51	48.78
	PeRQ*	11.833	64.58	42.34	31.66	48.51	57.22	44.01	25.61	44.85
	PeRQ†	11.937	65.07	41.92	30.38	49.46	56.35	41.50	25.59	44.32
	GaBR-PeRQ	11.849	69.70	56.02	33.36	55.80	55.56	62.29	24.90	51.09
	DartQuant + Qronos	12.570	69.26	55.01	32.08	53.10	56.35	61.80	26.78	50.63
	OSTQuant + Qronos	12.358	70.67	54.71	31.23	54.26	56.75	62.72	25.88	50.89
	GaBR-OST	12.073	70.89	57.83	33.62	55.29	57.38	62.42	27.37	52.11
Llama-3.2 3B	BF16	7.817	77.53	71.72	45.73	73.71	69.06	72.75	56.15	66.67
	Direct RTN	10.288	73.83	64.77	39.51	66.65	63.54	63.39	45.88	59.65
	PeRQ*	8.788	73.45	60.44	39.76	67.15	64.88	55.08	47.42	58.31
	PeRQ†	8.930	72.91	59.81	40.10	68.12	63.38	62.60	45.46	58.91
	GaBR-PeRQ	8.775	74.97	68.06	41.47	68.20	66.93	71.10	48.36	62.73
	DartQuant + Qronos	9.370	73.67	64.39	38.99	66.97	63.38	73.30	48.07	61.25
	OSTQuant + Qronos	9.131	74.32	69.65	42.24	67.47	67.40	71.01	47.90	62.86
	GaBR-OST	8.918	74.97	70.58	42.41	68.29	64.33	72.51	47.67	62.97
Qwen3 1.7B	BF16	16.716	72.25	70.16	43.09	60.39	61.48	77.65	60.12	63.59
	Direct RTN	29.665	64.69	55.72	35.15	53.57	56.83	68.62	47.89	54.64
	PeRQ*	18.546	66.49	54.34	38.74	47.46	58.41	75.20	51.57	56.03
	PeRQ†	13.107	62.24	39.90	33.45	39.68	56.91	63.12	47.64	48.99
	GaBR-PeRQ	18.383	68.28	56.19	35.75	52.94	60.46	74.25	52.04	57.13
	DartQuant + Qronos	20.410	65.18	54.84	32.76	50.73	54.85	74.71	50.60	54.81
	OSTQuant + Qronos	17.751	68.28	62.58	38.48	52.68	60.54	70.55	49.77	57.55
	GaBR-OST	17.173	67.19	57.32	34.64	51.61	57.70	76.21	50.99	56.52
Qwen3 4B	BF16	13.663	74.97	78.32	53.50	68.44	65.11	84.95	70.03	70.76
	Direct RTN	18.659	72.20	70.62	47.44	63.11	61.80	83.39	63.39	65.99
	PeRQ*	15.555	68.39	53.11	41.98	51.52	63.77	78.87	63.25	60.13
	PeRQ†	10.583	64.25	39.94	34.30	42.02	61.33	66.45	52.59	51.56
	GaBR-PeRQ	15.272	72.20	72.85	47.95	61.48	61.96	84.13	64.09	66.38
	DartQuant + Qronos	16.140	71.65	69.99	43.52	60.60	61.80	83.70	62.27	64.79
	OSTQuant + Qronos	14.239	72.80	73.11	47.27	62.84	61.96	84.80	64.96	66.82
	GaBR-OST	13.207	72.42	73.99	47.70	62.62	63.93	84.31	64.30	67.04
Qwen3 8B	BF16	9.726	77.64	80.72	56.74	74.93	67.88	86.61	74.93	74.21
	Direct RTN	11.835	74.76	73.36	50.17	69.45	63.14	85.23	68.70	69.26
	PeRQ*	10.977	75.35	77.69	52.73	67.98	66.46	85.66	69.57	70.78
	PeRQ†	8.582	76.71	76.89	53.75	72.90	67.88	83.85	68.34	71.48
	GaBR-PeRQ	10.751	75.73	78.75	53.16	67.87	67.64	86.51	69.16	71.26
	DartQuant + Qronos	11.349	73.29	76.09	50.94	66.92	64.88	85.14	68.90	69.45
	OSTQuant + Qronos	10.301	76.66	79.64	55.59	70.83	67.40	85.05	69.90	72.15
	GaBR-OST	10.059	75.95	79.34	55.20	70.41	68.90	85.75	70.28	72.26

B Dataset-level results for the $\Gamma = GD$ factorization

Table 7: Full task-level comparison for the $\Gamma = GD$ factorization. PPL is WikiText-2 perplexity. PIQA, ARC-Easy (ARC-E), ARC-Challenge (ARC-C), and HellaSwag (Hella) use normalized accuracy; WinoGrande (Wino), BoolQ, and MMLU use accuracy. The first six tasks are zero-shot, MMLU is 5-shot, and all accuracies and Mean7 are percentages. Bold marks the better value within each model and metric; ties at the reported precision are bold on both sides.

Model	Condition	PPL↓	PIQA	ARC-E	ARC-C	Hella	Wino	BoolQ	MMLU	Mean7↑
Llama-2-7B	GaBR-OST (G)	**5.870**	76.55	71.00	**42.06**	**71.78**	65.35	**74.74**	39.77	63.04
	Without G	5.916	**76.88**	**72.26**	40.96	71.56	**66.61**	73.30	**40.96**	**63.22**
Llama-3-8B	GaBR-OST (G)	**7.177**	**78.67**	76.14	48.21	74.49	**72.22**	**79.76**	**57.84**	**69.62**
	Without G	7.188	78.13	**77.48**	**50.09**	**74.75**	70.96	78.23	57.71	**69.62**
Llama-3.2-1B	GaBR-OST (G)	**12.073**	**70.89**	**57.83**	**33.62**	55.29	**57.38**	**62.42**	**27.37**	**52.11**
	Without G	12.141	70.73	53.20	32.94	**55.74**	55.01	61.93	26.24	50.83
Qwen3-8B	GaBR-OST (G)	**10.059**	**75.95**	**79.34**	**55.20**	**70.41**	**68.90**	**85.75**	70.28	**72.26**
	Without G	10.591	75.84	78.66	54.69	69.80	67.56	85.35	**70.84**	71.82

Table 7 expands Table 4. The no-G runs use gamma sorting, three rounds of independent block-rotation training (30 steps per round), and a final 30-step channel-scale stage, while holding every block scale at one. MMLU covers 57 subjects and 14,042 questions at 5-shot. The with-G rows are the main-table GaBR-OST results.

C Gamma-permutation ablation details

Table 8: Dataset-level gamma-permutation results. PPL is WikiText-2 perplexity; all task accuracies and Mean7 are percentages. PIQA, ARC-Easy (ARC-E), ARC-Challenge (ARC-C), and HellaSwag (Hella) use normalized accuracy; WinoGrande (Wino), BoolQ, and MMLU use accuracy.

Model	Order	PPL↓	PIQA	ARC-E	ARC-C	Hella	Wino	BoolQ	MMLU	Mean7↑
Llama-3-8B	Identity	7.175	78.35	76.77	49.40	74.68	70.48	79.45	59.07	69.74
	Zigzag	7.231	78.13	75.38	48.46	74.66	70.80	79.63	58.86	69.42
	Gamma-MassDiff	7.226	78.40	76.05	49.06	74.96	69.30	79.91	57.98	69.38
	Gamma-sort (main table)	7.177	78.67	76.14	48.21	74.49	72.22	79.76	57.84	69.62
Llama-3.2-1B	Identity	12.099	71.33	55.05	31.31	54.77	56.83	59.57	26.25	50.73
	Zigzag	12.203	70.40	55.98	32.17	56.11	55.09	60.89	25.62	50.89
	Gamma-MassDiff	12.166	71.00	56.19	33.02	56.11	56.91	61.38	25.74	51.48
	Gamma-sort (main table)	12.073	70.89	57.83	33.62	55.29	57.38	62.42	27.37	52.11
Qwen3-8B	Identity	9.829	76.93	77.86	52.56	69.55	67.56	85.20	69.99	71.38
	Zigzag	10.071	76.28	79.42	54.61	69.89	66.69	86.48	70.73	72.01
	Gamma-MassDiff	10.094	76.28	80.35	55.80	69.49	67.88	85.17	70.66	72.23
	Gamma-sort (main table)	10.059	75.95	79.34	55.20	70.41	68.90	85.75	70.28	72.26

Table 8 expands the summary in Table 5 into full task-level results. Identity, Zigzag, and Gamma-MassDiff are from the permutation-sweep runs. Gamma-sort is the corresponding GaBR-OST main-table result. All runs use W4A4 MXFP4 with Qronos; the first six tasks are zero-shot and MMLU is 5-shot.

For context, Table 9 reports the earlier Direct-RTN channel-ordering control before rotation training or Qronos. It uses MC5 rather than Mean7 and should not be read as the end-to-end comparison in Table 8.

Table 9: Direct-RTN permutation control under W4A4 MXFP4, without rotation or Qronos. Each entry is PPL↓ / MC5↑ (%); MC5 averages PIQA, ARC-Easy, ARC-Challenge, HellaSwag, and WinoGrande. Llama-3.2-1B uses block-scale search, whereas the other models use full residual-gamma fusion.

Model	Identity	Zigzag	Gamma-MassDiff	Gamma-sort
Llama-3-8B	8.36 / 67.23	8.36 / 67.15	8.40 / 67.40	8.24 / 67.40
Llama-3.2-1B	16.85 / 49.83	16.73 / 51.04	16.99 / 50.37	16.06 / 51.24
Qwen3-8B	12.37 / 65.19	12.75 / 64.96	12.63 / 65.51	12.47 / 67.13

D DuQuant++ Comparison

Table 10: DuQuant++ under the same W4A4/KV16/block-32 evaluation protocol as Table 3. Its additional online activation transforms mean that the methods are not compared at equal inference cost. Entries are PPL↓ / Mean7↑.

Model	DuQuant++
Llama-2-7B	5.752 / 64.78
Llama-3-8B	6.897 / 69.85
Llama-3.2-1B	11.453 / 53.14
Llama-3.2-3B	8.629 / 63.70
Qwen3-1.7B	17.657 / 58.96
Qwen3-4B	14.892 / 67.14
Qwen3-8B	10.218 / 72.25

E Detailed cross-layer gamma diagnostics

Measurement details. Figure 5 reports the complete Pre-MLP comparison on Qwen3-1.7B/4B/8B, Llama-3.2-1B/3B, and Llama-3-8B; Figure 4 displays three of these models. Across all six, mean layer-pair Spearman correlations range from 0.453 to 0.679, but the local MSE changes are not uniformly favorable, as illustrated by Qwen3-4B. Within each model, we sort channels once in descending order of their mean $|\gamma|$ across all Pre-MLP layers, then use this identical order in every heatmap row. Each vertical column tracks the same physical channel across layers. Colors encode within-layer magnitude-rank percentiles, with blue indicating small $|\gamma|$ and orange indicating large $|\gamma|$; no row-wise sorting or channel averaging is applied. The reported ρ is the mean Spearman correlation over all distinct Pre-MLP layer pairs, with average ranks for ties. These Pre-MLP-only orderings diagnose within-type structure; they do not replace the single deployment permutation shared by both norm types.

For the MLP-error curves, we compare the original and permuted MLPs under W4A4 MXFP4, compensating the input permutation in the gate/up weight columns. All gate/up/down weights and their input activations are quantized with block size 32; no rotation, gamma fusion, scale optimization, or reconstruction is applied. Each layer receives the same clean input collected from the original BF16 model on eight non-overlapping 512-token WikiText-2 test windows (seed 42), so quantization errors do not propagate between layers. Both arms use the same unquantized MLP output as reference, computed in FP32 from the original input and weights before residual addition, with TF32 disabled. MSE averages squared output errors over all tokens and channels. For layer ℓ, we plot the signed relative change

$$\Delta\mathrm{MSE}_\ell = 100\left(\frac{\mathrm{MSE}_{\mathrm{sorted},\ell}}{\mathrm{MSE}_{\mathrm{original},\ell}} - 1\right)\%. \tag{18}$$

Negative values indicate lower error; positive values indicate higher error. The curves display layers $1, \ldots, L-1$: the final layer is omitted from the curves only, and remains included in the heatmaps and ordering construction. These local, non-final-layer results do not establish a reduction in whole-model mean MSE or end-to-end error.

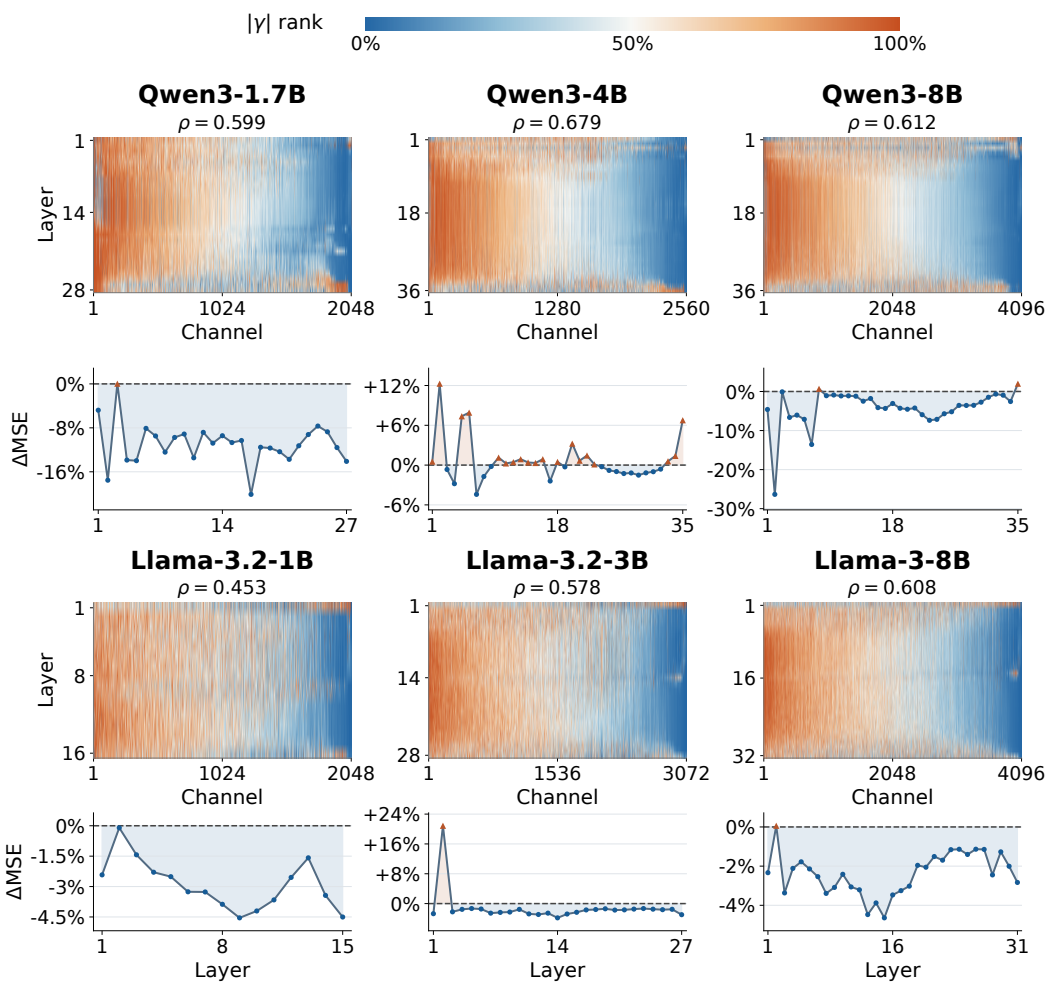


Figure 5: Complete six-model Pre-MLP diagnostics, using the same definitions as Figure 4. Heatmaps and the fixed per-model channel orders include all layers. Curves show signed local MLP MSE changes under W4A4 MXFP4: blue/negative indicates lower error and orange/positive indicates higher error. The final layer is omitted only from the curves; curve vertical ranges vary by model. These within-type diagnostics do not evaluate the shared deployment permutation.

Table 11 instead evaluates the shared permutation constructed from both pre-attention and Pre-MLP RMSNorm. We inspect all such affine vectors in Qwen3-8B, Llama-3-8B, and Llama-2-7B: respectively 72, 64, and 64 sites, each with 4096 channels. Final RMSNorm and attention-local Q/K norms are excluded, matching the implementation. We compute block homogeneity from the *actual site-specific* gamma vector, rather than its cross-layer mean:

$$\mathrm{BCV}_\ell(P) = \frac{1}{K}\sum_{b=1}^{K} \frac{\mathrm{Std}_{j\in\mathcal{B}_b}\left(|\gamma_{\ell,\pi(j)}|\right)}{\max\left(\mathrm{Mean}_{j\in\mathcal{B}_b}\left(|\gamma_{\ell,\pi(j)}|\right), \varepsilon\right)}. \tag{19}$$

Here $K = d/32$, Std uses the sample standard deviation within each 32-channel block, and $\varepsilon = 10^{-12}$. We average BCV equally across RMSNorm sites. The fixed global permutation improves every measured site (72/72, 64/64, and 64/64). As a control against self-inclusion, we recompute the ordering after excluding both RMSNorm sites of each held-out transformer layer, and evaluate BCV only at that layer. The corresponding reductions remain 37.0%, 22.4%, and 17.7%; 72/72, 63/64, and 64/64 held-out sites improve. Random global permutations provide essentially no average benefit (32 trials, seed 42).

Table 11: Whole-model channel consistency and shared-permutation block-homogeneity statistics on original models. Spearman values average all distinct layer-pair rank correlations within pre-attention and pre-MLP RMSNorm, respectively. BCV measures actual per-site gamma variation within 32-channel blocks, averaged over all sites, using one common ordering constructed from both norm types. This differs from the within-type diagnostic in Figure 4. LOO excludes both norm sites of the tested layer when constructing the ordering. Reductions are relative to the original channel order.

Model	All-layer ρ pre-attn / pre-MLP	Original BCV	Global-sort BCV	Global-sort reduction	LOO-sort reduction
Qwen3-8B	0.421 / 0.612	0.2580	0.1598	38.1%	37.0%
Llama-3-8B	0.315 / 0.608	0.1355	0.1016	25.0%	22.4%
Llama-2-7B	0.299 / 0.411	0.0777	0.0596	23.2%	17.7%

The statistics in Table 11 use only original RMSNorm parameters, without calibration tokens, quantization, or training. Figure 6 includes every measured norm site. When different norm types are included as well, the mean Spearman correlation over distinct-layer pairs is 0.314, 0.095, and 0.119 for Qwen3-8B, Llama-3-8B, and Llama-2-7B. These mixed averages are lower than the within-type correlations, illustrating why a universal "large stays large" assumption would be too strong.

For the held-out diagnostic, we exclude both sites of one transformer layer, compute the mean-absolute-gamma score from all remaining layers, and compare it with each held-out site's channel ranks. Mean Spearman correlations are 0.364, 0.183, and 0.254. We also select the top and bottom 10% of channels using only the other layers ($\lceil 0.1d \rceil = 410$ channels per group) and evaluate their percentiles in the held-out sites. The shaded regions in Figure 6 span the two norm-site values; they are not confidence intervals. Some held-out sites reverse or weaken the ordering, even though block-grouping improvements transfer.

As an additional control, 32 random global permutations give mean per-site BCV values of 0.2580, 0.1348, and 0.0776, compared with global gamma sorting's 0.1598, 0.1016, and 0.0596. Independent sorting at every norm site would give 0.0082, 0.0053, and 0.0043, respectively. This latter diagnostic uses different permutations at different sites and is not a deployable substitute for the single global permutation in our method. It illustrates the remaining gap between a shared ordering and site-specific grouping.

Separate within-type diagnostics. For norm type t, we form $s_j^{(t)} = L^{-1} \sum_{\ell=1}^{L} |\gamma_{\ell,j}^{(t)}|$, sort it once, and evaluate that order only on type t. The held-out version excludes layer ℓ from the same-type score before evaluating that layer. For pre-attention, the fixed-order BCV reductions on Qwen3-8B and Llama-3-8B are 48.9% and 38.4%, with held-out reductions of 47.9% and 33.3%. For Pre-MLP, the fixed-order BCV reductions are 58.9% and 63.9%, with held-out reductions of 56.5% and 62.5%. Figure 5 instead pairs the Pre-MLP heatmaps with local MLP output-error changes. These separately constructed orderings quantify within-type structure without imposing that the two types agree on channel grouping.

Qwen3-4B illustrates why this distinction matters. Its Pre-MLP mean layer-pair rank correlation is 0.679; sorting from Pre-MLP sites alone reduces their mean BCV from 0.14045 to 0.08731 (37.8%), or to 0.09035 with the tested layer held out (35.7%). However, the shared ordering built from both norm types increases the overall mean BCV from 0.17164 to 0.18259 (6.4%). Strong within-type consistency therefore does not ensure that the current shared-score rule improves block homogeneity over the original channel order on every model.

F Derivation with producer compensation

For a residual coordinate transform $C = PR$, any downstream linear layer receiving the transformed residual state gets the right-side transform in Eq. (13); a linear map that produces the residual state receives the inverse output-coordinate transform. For row vectors, if $z = xA^\top$ and the stored residual is changed to $\widetilde{z} = zC$, then $\widetilde{A} = C^\top A$. The paired updates to downstream and residual-producing

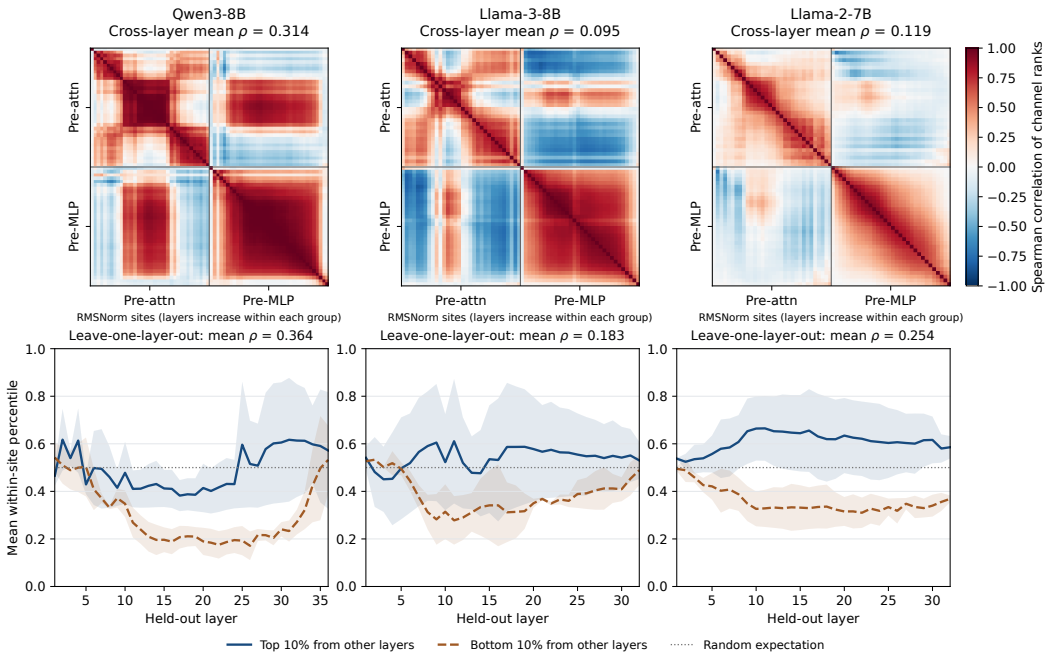


Figure 6: Complete rank-consistency diagnostics. Top: pairwise Spearman correlations over channel-wise $|\gamma|$ ranks, with sites ordered by norm type and then layer. Positive and negative correlations are both shown; diagonal self-correlations are displayed but excluded from reported cross-layer means. Bottom: held-out percentiles of channel groups selected from all other layers. Lines average the two RMSNorm sites of each held-out layer, and bands span those two values. A random channel group has expected mean percentile 0.5. These diagnostics show partial, norm-dependent consistency rather than invariant channel ranks.

linear layers cancel along every residual edge. Norm-local G is not part of the residual coordinate; it is paired with D at each RMSNorm site through Eq. (9). This separation is also why attention-internal Q/K normalization should not be treated as an R1 residual site.