HERO-Q: A GENERAL FRAMEWORK FOR STABLE LOW-BIT QUANTIZATION VIA HESSIAN CONDITIONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Low-bit Post-Training Quantization (PTQ) can produce substantial model degradation despite small Euclidean error. Under the standard local second-order approximation, we show that quantization-induced loss depends jointly on residual magnitude, effective curvature, and alignment with sensitive directions. We propose Hessian Robust Quantization (HeRo-Q) to control these factors through complementary transformations. A Hessian-derived non-orthogonal conditioner reduces the dominant eigenvalue of the effective Hessian, tightening the curvature-dependent loss bound. An optimized orthogonal rotation then preserves the conditioned spectrum while changing the finite-bit residual and its alignment. Under normalized-proxy assumptions, we prove strict spectral compression in the effective proxy Hessian and connect the theoretical quantities to the practical objective. Experiments across dense language, mixture-of-experts, vision, and multimodal models validate the mechanism and demonstrate robust low-bit performance. We also report calibration costs and inference throughput under the evaluated configurations.

1 INTRODUCTION

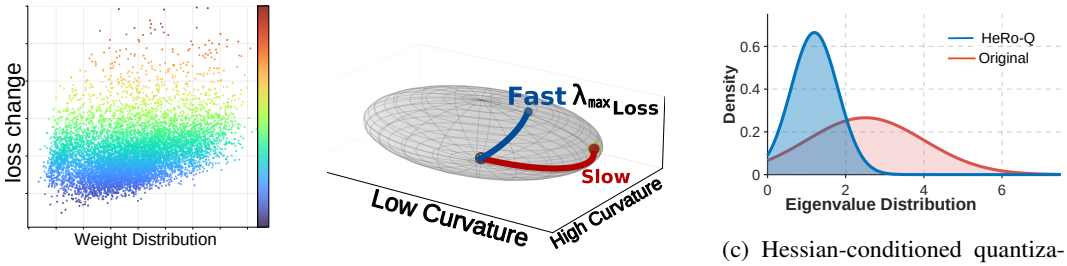
Post-training quantization (PTQ) reduces the storage and serving cost of large language models (LLMs) without full retraining Lin et al. (2024); Frantar et al. (2022); Xiao et al. (2023); Liu et al. (2025). Existing methods improve low-bit inference through rounding optimization, error compensation, outlier smoothing, and coordinate rotation. Yet under aggressive quantization, a model can exhibit only a small parameter discrepancy while its loss and downstream performance degrade substantially. This mismatch suggests that Euclidean quantization error alone does not fully characterize sensitivity to low-bit perturbations.

Figure 1a illustrates this "low-error, high-loss" behavior Zhang et al. (2025; 2024); Tang et al. (2025). Two quantized solutions may have similar residual norms yet affect the model very differently. Figure 1b explains why: the local loss geometry is highly anisotropic, and even for perturbations with the same norm, one aligned with a high-curvature direction incurs a larger local loss than one aligned with a low-curvature direction Yuan et al. (2023); He et al. (2024); An et al. (2025). Quantization sensitivity is thus determined jointly by residual magnitude, effective curvature, and their directional alignment.

Hessian-aware methods such as GPTQ and QuIP already use second-order information to guide rounding or error compensation Frantar et al. (2022); Chee et al. (2023). Transformation-based methods instead use scaling or rotation to obtain more quantization-friendly representations. These studies further raise a central question: can the quantization coordinate system itself be treated as an optimization variable, so that a unified coordinate space jointly controls the curvature amplification experienced by quantization error and the residual structure actually produced by a finite-bit quantizer?

To address this question, we propose Hessian Robust Quantization (HeRo-Q), as illustrated in Figure 1c. Rather than quantizing directly in the original parameter space, HeRo-Q learns a coordinate space better suited to quantization and performs finite-bit quantization within that space. Specifically, HeRo-Q first uses a Hessian proxy to rescale the coordinates, making the local loss geometry encountered by quantization error smoother. It then learns a rotation with the quantizer in the optimization loop to further adjust the error produced during quantization and its direction. In this

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.



(a) Low-error, high-loss (b) Anisotropic local loss geometry (c) Hessian-conditioned quantization

Figure 1: Motivation and overview of HeRo-Q. **(a)** A small Euclidean quantization error does not necessarily imply a small local loss, showing that residual magnitude alone is insufficient to characterize quantization robustness. **(b)** In an anisotropic local loss landscape, an equal-norm perturbation along a high-curvature direction incurs a larger local cost than one along a low-curvature direction. **(c)** HeRo-Q constructs curvature-aware quantization coordinates and performs finite-bit quantization in the resulting optimized space.

way, HeRo-Q jointly controls how strongly quantization error is amplified and the structure of the residual actually produced by the quantizer.

> The non-orthogonal conditioner targets curvature amplification in the transformed-coordinate loss bound, while the orthogonal rotation adjusts the realized finite-bit residual and its alignment without changing the conditioned spectrum.

This division of labor turns coordinate selection into a targeted conditioning problem rather than a generic preprocessing step. The resulting transformation requires no change to the model architecture or activation functions and remains compatible with existing PTQ pipelines. Theoretically, we derive a transformed-coordinate loss bound and show that the Hessian-derived non-orthogonal factor strictly compresses the dominant effective proxy curvature under stated normalized-proxy assumptions.

Our contributions are threefold: (1) We derive a local second-order quantization-loss bound linking residual magnitude, effective curvature, and residual–curvature alignment. (2) We propose HeRo-Q, combining tunable Hessian-proxy conditioning with subsequent rotation optimization, and prove strict compression of the largest effective proxy eigenvalue under stated normalized-proxy assumptions. (3) We test the mechanism through spectral, residual, and ablation analyses, and evaluate low-bit performance across dense language, MoE, vision, and multimodal models.

2 Related Work

Representative PTQ methods improve low-bit accuracy by optimizing rounding, reconstruction, or coding distortion. AdaRound (Nagel et al., 2020) learns rounding decisions; GPTQ (Frantar et al., 2022) uses second-order information for layer-wise reconstruction; AWQ (Lin et al., 2024) protects salient weights through activation-aware scaling; and OmniQuant (Shao et al., 2024) jointly optimizes clipping and equivalent transformations.

Transformation-based PTQ changes weight or activation coordinates before quantization. SmoothQuant (Xiao et al., 2023) uses channel-wise scaling to mitigate activation outliers, while QuaRot (Ashkboos et al., 2024) and SpinQuant (Liu et al., 2025) use fixed or learned orthogonal rotations to obtain more quantization-friendly representations. FlatQuant (Sun et al., 2025) and OSTQuant (Hu et al., 2025) learn more general transforms for reconstruction, distribution flattening, or quantization-space utilization.

Mixed-precision and sensitivity-aware methods determine where numerical precision should be spent. AutoQ (Lou et al., 2019) and HAQ (Wang et al., 2019) formulate bit allocation as a search problem, whereas other approaches use KL divergence (Cai et al., 2020; Xu et al., 2024), Hessian curvature (Dong et al., 2020; Yao et al., 2021), Gauss–Newton approximations (Chen et al., 2021), or learned importance (Tang et al., 2022). HeRo-Q instead improves the quantization coordinates under a fixed bit-width.

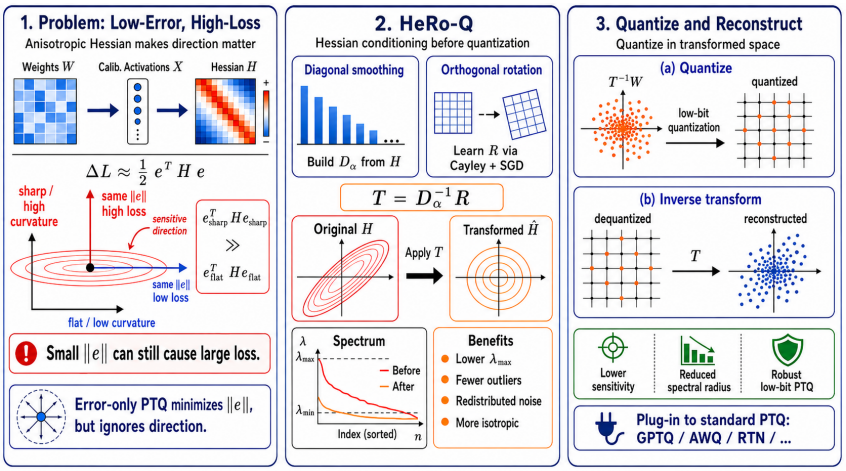


Figure 2: Robust quantization via Hessian-conditioned transformation. HeRo-Q estimates a group-wise curvature proxy, searches the conditioning strength α, and optimizes or selects a deployment-compatible orthogonal rotation $\mathbf{R}$. Quantization is performed in the coordinates defined by $\mathbf{T} = \mathbf{D}_\alpha^{-1}\mathbf{R}$. Deployment uses the same selected transforms, with final weight quantization performed after compatible parameter fusions.

QuIP (Chee et al., 2023), QuIP# (Tseng et al., 2024a), and QTIP (Tseng et al., 2024b) improve low-bit weight quantization through incoherence processing and rounding or coding design. HeRo-Q instead targets effective-curvature amplification in a local loss upper bound when designing its conditioner: it constructs a non-orthogonal conditioning matrix with tunable strength from the diagonal statistics of a Hessian proxy, and then optimizes an orthogonal rotation with the conditioning matrix fixed. The rotation preserves the spectrum of the conditioned effective proxy Hessian while adjusting the magnitude and direction of the realized finite-bit quantization residual, aiming to reduce its concentration along high-curvature eigendirections. HeRo-Q thus combines effective-curvature conditioning with quantization-residual optimization to reduce the quantization-induced loss arising from curvature amplification and residual alignment.

3 Methodology

3.1 Notation

- $\mathbf{H}$, $\mathbf{H}_{\text{proxy}} = \mathbf{X}^\top\mathbf{X}$: True/proxy Hessian.
- $\mathbf{H}_q = \mathbf{T}^\top\mathbf{H}\mathbf{T}$: Effective Hessian.
- $\mathbf{H}_{q,\text{proxy}} = \mathbf{T}^\top\mathbf{H}_{\text{proxy}}\mathbf{T}$: Effective proxy Hessian.
- $\mathbf{T} = \mathbf{D}_\alpha^{-1}\mathbf{R}$: Coordinate map; $\mathbf{D}_\alpha$ diagonal, $\mathbf{R}$ orthogonal.
- $\lambda_{\max}(\cdot)$, α: Largest eigenvalue and smoothing strength.
- $\mathbf{w}$, $\mathbf{z} = \mathbf{T}^{-1}\mathbf{w}$: Original/quantization-coordinate weights.
- $Q_b(\mathbf{z})$, $\mathbf{w}_q = \mathbf{T}Q_b(\mathbf{z})$: b-bit quantized/mapped-back weights.
- $\boldsymbol{\varepsilon} = Q_b(\mathbf{z}) - \mathbf{z}$: Quantization-coordinate residual.
- $\mathbf{e} = \mathbf{w}_q - \mathbf{w} = \mathbf{T}\boldsymbol{\varepsilon}$: Original-coordinate perturbation.
- $\Delta\mathcal{L}$: Second-order loss degradation.

3.2 Motivation

We use $\Delta\mathcal{L}$ for the local second-order PTQ loss surrogate under the convention in Appendix A. The following theorem relates it to the transformed residual and the effective Hessian; Appendix A.2 provides the proof.

As in layer-wise PTQ reconstruction, we estimate curvature with an activation-induced Hessian proxy. The smoothing coefficient α controls the strength of diagonal preconditioning and is selected empirically to balance effective-curvature compression against quantization distortion.

Theorem 1 (Spectral Loss Bound)**.** *Under the standard second-order PTQ approximation and* $\mathbf{H} \succeq \mathbf{0}$*,*

$$\Delta\mathcal{L} \leq \frac{1}{2}\lambda_{\max}(\mathbf{H}_q)\,\|\boldsymbol{\varepsilon}\|_2^2. \tag{1}$$

Therefore, improving quantization robustness requires controlling either the amplification factor $\lambda_{\max}(\mathbf{H}_q)$ or the realized residual $\boldsymbol{\varepsilon}$, motivating the two-factor design of HeRo-Q. Accordingly, Theorem 1 motivates a layer-specific quantization coordinate system rather than a generic composition of preprocessing operations. Corollary A.1 in the Appendix makes the residual–curvature alignment explicit.

This distinction is important because the two factors cannot be interpreted as interchangeable heuristics. For a fixed conditioned matrix $\mathbf{A}_\alpha = \mathbf{D}_\alpha^{-1}\mathbf{H}\mathbf{D}_\alpha^{-1}$, orthogonal similarity preserves every eigenvalue of $\mathbf{A}_\alpha$; rotation alone therefore cannot compress its spectral radius. Conversely, diagonal conditioning changes the geometry but does not determine the residual produced after rounding. The practical objective must account for both effects: the Hessian-derived diagonal defines the geometry first, and rotation is then optimized with the finite-bit quantizer inside the reconstruction loop. HeRo-Q consequently controls the two terms through different mechanisms rather than attributing both improvements to a single transform.

The empirical spectra in Figure 3 are strongly anisotropic: a small set of directions dominates the effective curvature. Theorem 1 therefore identifies two quantities that a Euclidean residual alone does not resolve—its magnitude and the curvature by which it can be amplified. The alignment decomposition further explains why equal-norm residuals need not incur equal local costs. HeRo-Q is designed around this distinction.

3.3 HeRo-Q Algorithm

HeRo-Q constructs quantization coordinates in which low-bit residuals face a better-conditioned local geometry. Figure 2 summarizes the workflow, and Algorithm 1 gives the group-wise procedure.

Let $\mathbf{H}_q \triangleq \mathbf{T}^\top\mathbf{H}\mathbf{T}$ denote the effective Hessian in quantization coordinates. Guided by Theorem 1, we factor $\mathbf{T}$ into two operators with distinct roles: *diagonal conditioning*, which can change the effective spectrum, and *orthogonal rotation*, which preserves that spectrum but changes the residual realized by finite-bit quantization. We define

$$\mathbf{T} \triangleq \mathbf{D}_\alpha^{-1}\mathbf{R}, \tag{2}$$

where $\mathbf{R} \in \mathbb{R}^{d\times d}$ is an orthogonal rotation matrix ($\mathbf{R}^\top\mathbf{R} = \mathbf{I}$), and $\mathbf{D}_\alpha$ is a diagonal smoothing matrix constructed from a group-wise Hessian proxy:

$$\mathbf{D}_\alpha = \operatorname{diag}\left(|h_1|^{\alpha/2}, |h_2|^{\alpha/2}, \ldots, |h_d|^{\alpha/2}\right), \quad \mathbf{h} = \operatorname{diag}(\mathbf{H}_{\text{proxy}}), \quad \mathbf{H}_{\text{proxy}} = \mathbf{X}^\top\mathbf{X}, \tag{3}$$

where $\alpha \in [0, 1]$ controls the conditioning strength. Larger values apply more aggressive curvature normalization but may increase clipping or reconstruction error. The following result formalizes the spectral effect under a normalized diagonal proxy.

Theorem 2 (Strict Spectral Compression)**.** *Under a normalized diagonal Hessian proxy with* $1 = h_{\min} < h_{\max}$*, let* $\mathbf{T}_\alpha = \mathbf{D}_\alpha^{-1}\mathbf{R}$ *and* $\mathbf{H}_{q,\text{proxy}}(\alpha) = \mathbf{T}_\alpha^\top\mathbf{H}_{\text{proxy}}\mathbf{T}_\alpha$*. For any* $\alpha \in (0, 1]$*,*

$$\lambda_{\max}(\mathbf{H}_{q,\text{proxy}}(\alpha)) < \lambda_{\max}(\mathbf{H}_{q,\text{proxy}}(0)). \tag{4}$$

Thus, under the theorem's assumptions, diagonal conditioning strictly reduces the largest effective proxy curvature. In practice, HeRo-Q selects α^* by reconstruction error and learns an orthogonal rotation $\mathbf{R}^*$ in the conditioned coordinates. Because the rotation changes the quantizer input, it can change residual magnitude, direction, and alignment; the optimization does not assume that an arbitrary rotation is beneficial. Appendix C.3 reports the search and optimization settings.

The procedure has three steps. First, calibration activations define $\mathbf{H}_{\text{proxy}} = \mathbf{X}^\top\mathbf{X}$ and the channel-wise statistics used to construct $\mathbf{D}_\alpha$. This proxy is computed once per transformation group and does not require labels or end-to-end fine-tuning. We search a small grid $\mathcal{A}$ rather than fixing the conditioning strength globally because layers differ in both curvature anisotropy and quantization range.

Algorithm 1 HeRo-Q: Hessian-Robust Quantization

Input: groups $\{\mathcal{C}_g, m_g, \mathcal{R}_g\}$, weights $\{\mathbf{W}_j\}$, calibration inputs $\{\mathbf{X}_g\}$, grid $\mathcal{A}$, bits b
Output: low-bit factors $\{Z_{q,j}\}$ and deployment plans $\{T_g^*, m_g\}$
for each deployable transformation group g **do**
  $\mathbf{H}_p \leftarrow \mathbf{X}_g^\top \mathbf{X}_g, \quad \mathbf{h} \leftarrow \mathrm{diag}(\mathbf{H}_p), \quad \mathcal{L}^* \leftarrow \infty$
  for $\alpha \in \mathcal{A}$ **do**
    $\mathbf{D}_\alpha \leftarrow \mathrm{diag}(|\mathbf{h}|^{\alpha/2})$
    Optimize or select $R_{g,\alpha} \in \mathcal{R}_g$; online rotations remain in $\mathcal{R}_{\mathrm{fast}}$
    $T_{g,\alpha} \leftarrow \mathbf{D}_\alpha^{-1} R_{g,\alpha}, \quad \mathbf{H}_{q,\alpha} \leftarrow T_{g,\alpha}^\top \mathbf{H}_p T_{g,\alpha}$
    $Z_{q,j,\alpha} \leftarrow \mathrm{Quant}(T_{g,\alpha}^{-1}\mathbf{W}_j, \mathbf{H}_{q,\alpha}, b)$ for all $j \in \mathcal{C}_g$
    $\mathcal{L}_\alpha \leftarrow \sum_{j\in\mathcal{C}_g} \|\mathbf{X}_g\mathbf{W}_j - (\mathbf{X}_g T_{g,\alpha}) Z_{q,j,\alpha}\|_F^2$
    if $\mathcal{L}_\alpha < \mathcal{L}^*$ **then**
      $\mathcal{L}^* \leftarrow \mathcal{L}_\alpha, \quad T_g^* \leftarrow T_{g,\alpha}, \quad Z_{q,j} \leftarrow Z_{q,j,\alpha}$ for all $j \in \mathcal{C}_g$
    end if
  end for
end for
Compose compatible fusion updates; refresh quantizer statistics and regenerate final low-bit factors

Second, for each $\alpha \in \mathcal{A}$, HeRo-Q optimizes $\mathbf{R}$ in the coordinates already defined by $\mathbf{D}_\alpha$. For trainable dense rotations admitted by the deployment graph, we parameterize $\mathbf{R}$ with a Cayley transform so that every update remains orthogonal; for online rotations, optimization or selection is restricted from the outset to the structured family used at deployment. We minimize the sum of $\|\mathbf{XW} - \mathbf{XW}_{\mathrm{sim}}\|_F^2$ over the projections in each group with simulated finite-bit weight quantization in the loop. Both the search over α and the optimization of $\mathbf{R}$ use this weight-quantization reconstruction objective; the W4A4/W4A8 evaluation procedure is specified in Appendix C.2. The objective is deliberately practical: it selects a residual produced by the actual quantizer rather than optimizing the worst-case spectral bound in isolation. The bound explains which geometric quantities matter, whereas reconstruction supplies a tractable group-wise criterion for choosing among realizable transforms.

Third, the pair $(\alpha^*, \mathbf{R}^*)$ with the lowest reconstruction error defines $\mathbf{T}^* = \mathbf{D}_{\alpha^*}^{-1}\mathbf{R}^*$. We deploy the selected transforms through compatible fusion or the prescribed online implementation. When fusion updates a weight at both interfaces, its low-bit factors are regenerated from the final composed weight, rather than retained from an intermediate candidate. This coordinate construction is agnostic to the base quantizer: GPTQ is the default backend in the main experiments, while Appendix B.2 tests AWQ and RTN and Appendix D.4 evaluates compatibility with QuIP-style quantization.

Each group $\mathcal{C}_g$ shares an input and a transform, such as Q/K/V or MLP up/gate projections. The deployment graph fixes shared coordinates, parameter-sharing constraints, and each local rotation family before this search; a local family may contain only the identity or a fixed transform. Section 3.4 and Appendix C.4 specify the deployment interpretation of m_g and $\mathcal{R}_g$.

3.4 Inference Efficiency

HeRo-Q fixes each local rotation family $\mathcal{R}_g$, its sharing constraints, and deployment mode m_g (fused or online) before optimization. At compatible interfaces, its additional diagonal conditioners are absorbed into normalization parameters or adjacent weights; any remaining rotations retain their prescribed structured online implementation, following SpinQuant-style deployment (Liu et al., 2025). This avoids standalone conditioning operators on those paths, not all online rotation cost. Appendix C.4 gives the interface-specific fusion rules, transform order, and scope of this statement.

4 Experiments

Our evaluation addresses three questions: (i) does HeRo-Q improve predictive performance across architectures and bit-widths; (ii) do spectrum, alignment, and component measurements support the proposed mechanism; and (iii) what calibration and inference costs accompany these gains? We test W4A4, W4A8, W4A16, and W3A16 settings; implementation details and code availability are provided in Appendix C.1.

Method	Llama-3.1-8B (W4A16)				Qwen2.5-7B (W4A16)				Llama-3.1-8B (W3A16)				Llama-3.2-1B (W3A16)			
	W2↓	C4↓	MMLU↑	GSM8K↑	W2↓	C4↓	MMLU↑	GSM8K↑	W2↓	C4↓	MMLU↑	GSM8K↑	W2↓	C4↓	MMLU↑	GSM8K↑
FP16	6.14	12.28	67.31	77.17	6.66	13.26	71.48	87.96	6.14	12.28	67.31	77.17	11.41	22.48	46.38	36.44
GPTQ	7.02	12.55	65.80	74.27	6.95	13.85	69.80	85.23	20.13	28.94	25.12	26.36	41.30	65.80	14.20	15.19
AWQ	6.60	12.56	66.14	74.52	7.14	14.21	69.68	82.33	15.87	24.15	39.50	46.21	18.23	43.97	40.23	26.44
OmniQuant	6.48	12.35	66.80	75.83	6.75	13.40	70.90	86.82	9.45	15.80	59.80	63.59	17.50	32.10	36.50	21.48
SpinQuant	6.46	12.32	66.85	76.17	6.72	13.31	71.10	87.14	8.85	14.95	60.50	66.22	16.20	30.05	39.10	22.80
FlatQuant	6.46	12.32	66.93	76.32	6.71	13.35	71.17	87.22	8.77	14.55	62.56	68.38	16.11	27.47	40.18	24.97
QuIP	6.55	12.41	66.87	76.25	6.87	13.33	71.20	87.30	8.50	14.40	63.20	69.00	15.50	27.00	41.50	26.00
HeRo-Q	**6.33**	**12.30**	**67.10**	**76.85**	**6.68**	**13.28**	**71.35**	**87.50**	**8.02**	**13.98**	**63.80**	**70.15**	**14.05**	**25.80**	**42.50**	**29.10**

Method	Llama-3.1-8B (W4A4)				Llama-3.2-3B (W4A4)				Llama-3.1-8B (W4A8)				Qwen2.5-7B (W4A8)			
	W2↓	C4↓	MMLU↑	GSM8K↑	W2↓	C4↓	MMLU↑	GSM8K↑	W2↓	C4↓	MMLU↑	GSM8K↑	W2↓	C4↓	MMLU↑	GSM8K↑
FP16	6.14	12.28	67.31	77.17	9.53	17.24	61.44	68.86	6.14	12.28	67.31	77.17	6.66	13.26	71.48	87.96
AWQ	346	642.4	36.45	30.76	1000+	1000+	22.97	16.88	8.12	13.56	64.41	72.94	7.14	14.21	69.68	82.33
GPTQ	1000+	1000+	34.55	28.79	1000+	1000+	23.12	13.17	8.81	14.12	60.54	71.89	7.32	14.53	69.35	81.67
SmoothQuant	429	745	40.18	37.81	1000+	1000+	34.74	20.76	7.32	13.64	64.55	72.55	7.09	14.06	69.79	83.47
SpinQuant	12.14	17.46	43.76	39.44	15.31	25.41	36.98	32.91	7.48	13.39	64.98	73.56	6.97	13.89	70.14	84.97
FlatQuant	11.17	15.94	44.39	41.01	14.65	23.33	38.25	34.86	7.31	13.11	65.32	74.19	6.89	13.59	70.28	85.37
QuIP	10.80	15.50	44.80	41.50	14.00	22.50	39.50	36.00	7.15	13.05	65.80	74.80	6.85	13.50	70.70	85.80
HeRo-Q	**10.29**	**14.84**	**45.31**	**42.33**	**13.18**	**20.14**	**41.10**	**37.15**	**6.96**	**12.94**	**66.31**	**75.26**	**6.78**	**13.43**	**71.02**	**86.12**

Table 1: Compact comparison across bit-width regimes. W2 and C4 denote perplexity on WikiText-2 and C4; MMLU and GSM8K denote accuracy scores. Lower perplexity is better, while higher accuracy is better.

4.1 Experimental Setup

Models and tasks. We evaluate dense LLMs (Llama-3.1-8B (Team, 2024), Llama-3.2-1B/3B (Meta AI, 2024), and Qwen2.5-3B/7B (Yang et al., 2024)), vision transformers (Dosovitskiy et al., 2021; Touvron et al., 2021), LLaVA-OneVision-7B (Li et al., 2025a), and two MoE models (Dai et al., 2024; Bai et al., 2023). All evaluated LLMs use instruction-tuned checkpoints. For language models, we report WikiText-2 and C4 perplexity at sequence length 2048 (Merity et al., 2017; Raffel et al., 2020), together with MMLU, GSM8K, and HellaSwag accuracy (Hendrycks et al., 2021; Cobbe et al., 2021; Zellers et al., 2019).

Baselines and settings. We compare against RTN, GPTQ, AWQ, SmoothQuant, SpinQuant, OmniQuant, FlatQuant, and QuIP (Dettmers et al., 2022; Frantar et al., 2022; Lin et al., 2024; Xiao et al., 2023; Liu et al., 2025; Shao et al., 2024; Sun et al., 2025; Chee et al., 2023). Cross-domain experiments additionally include established vision-, multimodal-, and MoE-specific baselines (Lin et al., 2022; Yuan et al., 2022; Ding et al., 2022; Li et al., 2023; 2025b; Chen et al., 2025). Unless noted otherwise, methods share the calibration data, bit-width, and evaluation protocol, and we use their official settings.

Implementation details. HeRo-Q selects α group-wise and uses GPTQ as the default backend unless stated otherwise. Calibration and evaluation sets are disjoint. We use the standard lm-evaluation-harness protocols: 5-shot MMLU, 8-shot GSM8K, and 10-shot HellaSwag (Gao et al., 2024).

4.2 Downstream Task Evaluation

Results on dense LLMs. Table 1 compares dense LLMs across W4A16, W4A8, W4A4, and W3A16. HeRo-Q is consistently strongest or competitive across perplexity and reasoning metrics for both Llama and Qwen. Under W4A16 on Llama-3.1-8B, HeRo-Q reduces WikiText-2 perplexity from GPTQ's 7.02 to 6.33, close to the FP16 value of 6.14, while retaining 76.85 GSM8K accuracy. On Qwen2.5-7B, it reaches 71.35 MMLU and 87.50 GSM8K, exceeding the listed PTQ baselines. Under W3A16, it also improves over the matched QuIP baseline on both evaluated Llama models, with the largest margins on GSM8K.

The W4A8 results show that the gains are not confined to the most aggressive regime. HeRo-Q obtains 6.96 WikiText-2 perplexity and 75.26 GSM8K on Llama-3.1-8B, and 6.78 and 86.12, respectively, on Qwen2.5-7B. These results complement the W3A16 stress test with a deployment-oriented setting in which most baselines remain viable.

Differences widen as quantization becomes more aggressive. On Llama-3.1-8B W3A16, GPTQ reaches 20.13 WikiText-2 perplexity and 26.36 GSM8K, whereas HeRo-Q obtains 8.02 and 70.15. Under W4A4, several scalar baselines become unstable; GPTQ exceeds 1000 perplexity on this

Method	ViT/DeiT (W4A4)			ViT/DeiT (W6A6)			Method	LLaVA-OneVision-7B (W3A16)			
	ViT-B↑	DeiT-S↑	DeiT-B↑	ViT-B↑	DeiT-S↑	DeiT-B↑		MMMU↑	VizWiz↑	SQA↑	TextVQA↑
FP16	84.54	79.85	81.80	84.54	79.85	81.80	FP16	46.09	60.42	85.42	76.16
FQ-ViT	0.10	0.10	0.10	0.10	45.51	64.63	RTN	34.71	59.26	82.28	60.97
PTQ4ViT	30.69	34.08	64.39	81.65	76.28	80.25	GPTQ	41.74	56.43	82.49	71.33
APQ-ViT	41.41	43.55	67.48	82.21	77.76	80.42	AWQ	36.68	58.57	83.26	73.04
RepQ-ViT	68.48	69.03	75.61	83.62	78.90	81.27	MBQ	42.03	60.42	85.02	73.37
HeRo-Q	**70.12**	**70.01**	**77.33**	**84.10**	**79.31**	**81.66**	**HeRo-Q**	**43.12**	**60.42**	**85.31**	**74.31**

Table 2: Generalization of HeRo-Q on vision and multimodal models.

Method	DeepSeek-MoE-16B-chat								Qwen1.5-MoE-A2.7B-Chat							
	W2↓	C4↓	MMLU↑	HE↑	GSM8K↑	HS↑	OBQA↑	MathQA↑	W2↓	C4↓	MMLU↑	HE↑	GSM8K↑	HS↑	OBQA↑	MathQA↑
FP16	7.35	9.96	48.90	24.39	54.28	60.69	33.40	34.27	8.07	9.74	59.00	21.34	30.71	59.33	31.00	34.91
RTN	8.63	11.06	41.40	10.41	28.88	57.59	31.40	29.04	12.81	14.03	43.00	7.32	9.70	51.41	24.40	28.81
AWQ	7.72	10.49	46.33	18.90	39.88	58.97	**33.80**	32.86	9.97	11.90	52.06	12.20	17.74	55.37	30.40	31.46
GPTQ	7.55	10.24	46.60	13.41	47.08	59.64	33.20	32.76	8.38	10.78	57.30	15.24	26.08	58.72	**31.40**	34.17
MoEQuant+	7.85	10.23	46.40	18.90	45.41	59.03	33.60	33.14	10.12	11.55	55.34	13.60	20.87	56.64	30.60	32.50
HeRo-Q	**7.42**	**10.03**	**47.88**	**21.12**	**50.05**	**60.48**	33.40	**33.87**	**8.21**	**10.17**	**58.44**	**16.98**	**28.46**	**59.12**	31.00	**34.88**

Table 3: MoE quantization results on DeepSeek-MoE-16B-chat and Qwen1.5-MoE-A2.7B-Chat. All quantized rows use W4A16. HeRo-Q uses GPTQ, and MoE routers and the KV cache remain in FP16. W2 and C4 denote WikiText-2 and C4 perplexity; HE, HS, and OBQA denote HumanEval, HellaSwag, and OpenBookQA, respectively.

model, while HeRo-Q remains at 10.29 with 42.33 GSM8K. These results show that the advantage is most pronounced when the quantization perturbation is largest.

Two patterns are consistent across the table. First, the gain is not restricted to a single metric: reductions in WikiText-2 and C4 perplexity are accompanied by higher MMLU and GSM8K accuracy. This matters because a method tuned only to local reconstruction could otherwise improve perplexity without preserving reasoning behavior. Second, the relative advantage grows when either weights or activations are quantized more aggressively. That trend is consistent with the intended role of conditioning—limiting the amplification of the residual that remains after rounding—although the table alone does not isolate the mechanism. We therefore complement it with spectrum, alignment, and component analyses below and in Appendix D.1.

Generalization to vision and multimodal models. Table 2 tests transfer beyond dense LLMs. Under W4A4, HeRo-Q reaches 70.12 on ViT-B, 70.01 on DeiT-S, and 77.33 on DeiT-B, improving over RepQ-ViT in all three cases; under W6A6, it approaches FP16 accuracy. On LLaVA-OneVision-7B W3A16, HeRo-Q obtains 43.12 MMMU and 74.31 TextVQA, the strongest values among the listed quantized methods. Although these tasks use different architectures and metrics, the gains remain consistent with the same coordinate-conditioning principle.

Generalization to MoE models. Table 3 evaluates whether the method remains effective when activation and curvature statistics vary across experts. On DeepSeek-MoE-16B-chat, HeRo-Q obtains 7.42 WikiText-2 and 10.03 C4 perplexity, together with 47.88 MMLU and 50.05 GSM8K. On Qwen1.5-MoE-A2.7B-Chat, it reaches 58.44 MMLU, 28.46 GSM8K, and 59.12 HellaSwag, improving over the listed quantized baselines on most metrics. These results extend the empirical evidence to sparse expert architectures; Appendix B reports additional settings.

The MoE setting also tests robustness across heterogeneous experts. HeRo-Q follows the group-wise procedure in Algorithm 1: projections sharing an activation boundary and coordinate transform use the same proxy statistics and selected transform, rather than independent transforms for each linear layer. The reported gains do not establish that every expert benefits equally, but they show that the aggregate improvement persists when routing activates only a subset of parameters. Appendix B.3 extends this check to Mixtral-8×7B and reports both calibration cost and downstream performance.

4.3 Spectral Analysis Experiment

We next test the geometric effect targeted by the non-orthogonal factor. Figure 3 compares the effective Hessian spectrum of Llama-3.2-3B before and after conditioning, using both the aggregate eigenvalue

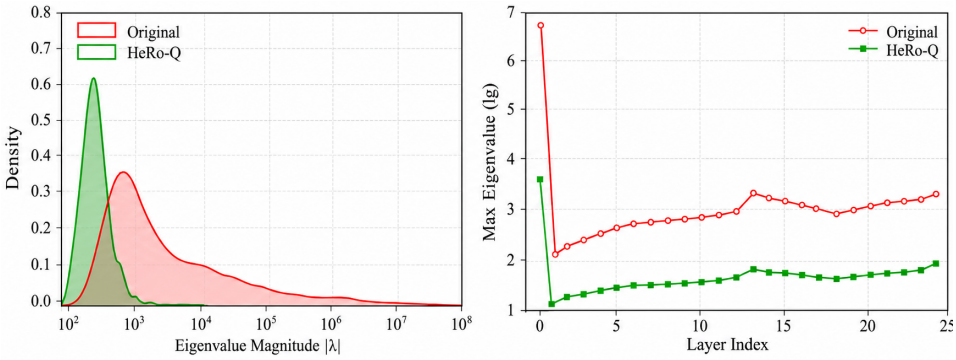


Figure 3: Effective proxy-curvature measurements on Llama-3.2-3B. (a) The conditioned coordinates exhibit a shorter-tailed eigenvalue distribution. (b) Layer-wise maximum eigenvalues decrease on a log scale. These plots verify the intended spectral effect; downstream robustness is evaluated separately.

distribution and layer-wise maxima. This measurement isolates spectrum from downstream accuracy and directly evaluates the quantity controlled by Theorem 2.

The baseline in Figure 3(a) has a long-tailed effective spectrum, so residual components along its extreme eigendirections can contribute disproportionately to the local quadratic surrogate. After conditioning, the distribution is more concentrated and its upper tail is reduced, directly verifying the proxy spectral effect targeted by HeRo-Q.

Figure 3(b) resolves the same effect across depth: the conditioned model reduces the largest effective eigenvalue at the observed layers and attenuates sharp peaks. The non-orthogonal factor is responsible for this spectral change. The orthogonal factor is isospectral after conditioning and instead changes the finite-bit residual, as analyzed separately in Appendix D.1.

Spectrum compression is necessary for the specific worst-case factor targeted by HeRo-Q, but it is not sufficient to characterize the realized quadratic cost. A residual may still concentrate along the leading eigendirections, or aggressive scaling may increase clipping error. The alignment experiment in Table 19 therefore measures ordinary residual norm, Hessian-weighted error, and normalized alignment on Llama-3.1-8B under W4A16 quantization. Relative to the direct SmoothQuant+SpinQuant control, HeRo-Q changes the ordinary error more modestly than the curvature-sensitive quantities, providing a more discriminating check of the proposed mechanism.

4.4 Computational Performance Evaluation

Table 4(a) reports W4A16 decoding throughput on an NVIDIA A800 with batch size 4, 128 input tokens, and 2048 generated tokens per sequence. HeRo-Q reaches 303.1, 210.0, and 113.1 tokens/s on the three evaluated Llama models, closely matching SpinQuant and remaining within 3% of GPTQ. These measurements exclude prefill but include all remaining online transformations during decoding; they do not attribute runtime savings to individual fused operators or guarantee performance for other bit-widths and backends. The backend and timing protocol are specified in Appendix C.5. Appendix C.4 provides deployment details and additional calibration-cost measurements.

4.5 Ablation Experiments

4.5.1 Hyperparameter Sensitivity Study

Table 4(b) varies $\alpha \in [0.1, 0.8]$ to probe the trade-off introduced by diagonal conditioning. Perplexity follows a U-shaped trend: moderate conditioning improves the effective geometry, whereas overly aggressive scaling increases quantization distortion.

For $\alpha < 0.3$, conditioning is weak and more of the proxy spectral tail remains. Theorem 1 identifies this as a larger worst-case amplification factor for a fixed transformed residual.

(a) Inference Throughput

Method	Llama-3.2-1B	Llama-3.2-3B	Llama-3.1-8B
FP16	205.2	118.4	48.5
GPTQ	**310.5**	**215.6**	**115.2**
SpinQuant	302.8	209.5	112.8
HeRo-Q	303.1	210.0	113.1

(b) α Sensitivity

Metric	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	GW Opt.
Llama-3.1-8B W2↓	6.58	6.52	**6.47**	6.49	6.62	6.80	7.15	7.50	**6.33**
Llama-3.1-8B C4↓	12.45	12.39	**12.34**	12.36	12.48	12.60	12.95	13.20	**12.30**
Qwen2.5-7B W2↓	6.85	6.74	**6.70**	6.75	6.88	7.10	7.45	7.95	**6.68**
Qwen2.5-7B C4↓	13.45	13.37	**13.32**	13.35	13.42	13.60	13.85	14.10	**13.28**

Table 4: Efficiency and hyperparameter sensitivity analyses. (a) Decoding throughput on NVIDIA A800 under W4A16 quantization, measured in tokens/s with batch size 4, 128 input tokens, and 2048 generated tokens per sequence; prefill is excluded. (b) Perplexity sensitivity to the smoothing factor α; fixed $\alpha = 0.3$ is robust(W4A16).

At larger α, stronger conditioning no longer improves perplexity. The accompanying rescaling can make the finite-bit dynamic range less favorable and increase clipping or reconstruction error. Group-wise selection captures this trade-off and gives the best aggregate results in the table; the experiment motivates selection rather than assuming maximal conditioning is optimal.

The fixed value $\alpha = 0.3$ is competitive across both evaluated model families, which makes it a practical default when search cost is constrained. Group-wise selection nevertheless improves every reported perplexity value, indicating that the optimum is not purely model-wide.

4.5.2 Component-wise Analysis

Model	FP16	GPTQ	Smoothing	Rotation	SmoothQuant + SpinQuant	HeRo-Q
Llama-3.2-1B	11.41	13.30	12.95	12.71	12.62	**11.74**
Llama-3.1-8B	6.14	7.02	6.94	6.89	6.75	**6.33**

Table 5: Component and composition ablation on WikiText-2 perplexity. All quantized variants of both models use W4A16; FP16 is the reference baseline. Smoothing uses $\mathbf{D}_\alpha^{-1}$, Rotation uses $\mathbf{R}$, and HeRo-Q combines both as $\mathbf{D}_\alpha^{-1}\mathbf{R}$ under a Hessian-conditioned objective. SmoothQuant+SpinQuant denotes a direct composition of existing transformations with GPTQ as the base quantizer.

Table 5 compares the individual and combined factors of $\mathbf{T}$ on Llama-3.2-1B and Llama-3.1-8B and includes a direct SmoothQuant+SpinQuant composition as a control. Relative to GPTQ, conditioning alone reduces perplexity from 13.30 to 12.95 on Llama-3.2-1B and from 7.02 to 6.94 on Llama-3.1-8B. Rotation alone reaches 12.71 and 6.89, while HeRo-Q improves further to 11.74 and 6.33. The full construction therefore outperforms either standalone variant on both models, though the benefit of conditioning alone is modest on 8B.

SmoothQuant+SpinQuant also improves on GPTQ, reaching 12.62 and 6.75, but remains behind HeRo-Q on both models. This gap supports the value of HeRo-Q's coordinated construction beyond combining scaling and rotation. Appendix D.1 compares the construction rules and provides complementary mechanism analyses.

5 Conclusion

This work shows that low-bit PTQ should control not only the magnitude of the quantization residual, but also the effective curvature that amplifies it and its alignment with sensitive directions. HeRo-Q addresses these factors through two distinct mechanisms: Hessian-derived non-orthogonal conditioning reduces the dominant effective curvature, while a subsequent optimized orthogonal rotation changes the finite-bit residual and its alignment in the conditioned coordinates. A transformed-coordinate loss bound and a strict spectral-compression result characterize this local mechanism, while spectrum, curvature-weighted error, alignment, and component analyses test it directly. Results across language, vision, multimodal, and mixture-of-experts models further demonstrate consistent gains under diverse low-bit settings. The reported throughput is competitive with the compared baselines under the evaluated configurations; deployment costs remain specific to the model, quantization setting, and backend.

AI USE STATEMENT

We used GPT-6 solely for language polishing. All AI-assisted edits were reviewed by the authors, who take full responsibility for the final manuscript.

REFERENCES

Yongqi An, Xu Zhao, Tao Yu, Ming Tang, and Jinqiao Wang. Systematic outliers in large language models. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=rLX7Vyyzus.

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. *Advances in Neural Information Processing Systems*, 37:100213–100240, 2024.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

Yaohui Cai, Zhewei Yao, Zhen Dong, Amir Gholami, Michael W Mahoney, and Kurt Keutzer. Zeroq: A novel zero shot quantization framework. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 13169–13178, 2020.

Jerry Chee, Yaohui Cai, Volodymyr Kuleshov, and Christopher De Sa. QuIP: 2-bit quantization of large language models with guarantees. In *Advances in Neural Information Processing Systems 36 (NeurIPS)*, 2023.

Weihan Chen, Peisong Wang, and Jian Cheng. Towards mixed-precision quantization of neural networks via constrained optimization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5350–5359, 2021.

Zhixuan Chen, Xing Hu, Dawei Yang, Zukang Xu, Chen Xu, Zhihang Yuan, Sifan Zhou, and Jiangyong Yu. Moequant: Enhancing quantization for mixture-of-experts large language models via expert-balanced sampling and affinity guidance. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, Proceedings of Machine Learning Research. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/chen25aa.html.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168, 2021. URL https://arxiv.org/abs/2110.14168.

Damai Dai, Chengqi Deng, Chenggang Zhao, R. X. Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y. Wu, Zhenda Xie, Y. K. Li, Panpan Huang, Fuli Luo, Chong Ruan, Zhifang Sui, and Wenfeng Liang. Deepseekmoe: Towards ultimate expert specialization in mixture-of-experts language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pp. 1280–1297. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.ACL-LONG.70. URL https://doi.org/10.18653/v1/2024.acl-long.70.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. Llm.int8(): 8-bit matrix multiplication for transformers at scale. *CoRR*, abs/2208.07339, 2022. doi: 10.48550/ARXIV.2208.07339. URL https://doi.org/10.48550/arXiv.2208.07339.

Yifu Ding, Haotong Qin, Qinghua Yan, Zhenhua Chai, Junjie Liu, Xiaolin Wei, and Xianglong Liu. Towards accurate post-training quantization for vision transformer. In João Magalhães, Alberto Del Bimbo, Shin'ichi Satoh, Nicu Sebe, Xavier Alameda-Pineda, Qin Jin, Vincent Oria, and Laura Toni (eds.), *MM '22: The 30th ACM International Conference on Multimedia, Lisboa, Portugal, October 10 - 14, 2022*, pp. 5380–5388. ACM, 2022. doi: 10.1145/3503161.3547826. URL https://doi.org/10.1145/3503161.3547826.

Zhen Dong, Zhewei Yao, Daiyaan Arfeen, Amir Gholami, Michael W Mahoney, and Kurt Keutzer. Hawq-v2: Hessian aware trace-weighted quantization of neural networks. *Advances in neural information processing systems*, 33:18518–18529, 2020.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. GPTQ: Accurate post-training quantization for generative pre-trained transformers. *arXiv preprint arXiv:2210.17323*, 2022.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. The language model evaluation harness, 07 2024. URL https://zenodo.org/records/12608602.

Bobby He, Lorenzo Noci, Daniele Paliotta, Imanol Schlag, and Thomas Hofmann. Understanding and minimising outlier features in transformer training. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/986292a930c3692168b177a770025ab3-Abstract-Conference.html.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

Xing Hu, Yuan Cheng, Dawei Yang, Zhixuan Chen, Zukang Xu, Jiangyong Yu, Chen Xu, Zhihang Yuan, Zhe Jiang, and Sifan Zhou. Ostquant: Refining large language model quantization with orthogonal and scaling transformations for better distribution fitting. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=rAcgDBdKnP.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *Trans. Mach. Learn. Res.*, 2025, 2025a. URL https://openreview.net/forum?id=zKv8qULV6n.

Shiyao Li, Yingchun Hu, Xuefei Ning, Xihui Liu, Ke Hong, Xiaotao Jia, Xiuhong Li, Yaqi Yan, Pei Ran, Guohao Dai, Shengen Yan, Huazhong Yang, and Yu Wang. MBQ: modality-balanced quantization for large vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2025, Nashville, TN, USA, June 11-15, 2025*, pp. 4167–4177. Computer Vision Foundation / IEEE, 2025b. doi: 10.1109/CVPR52734.2025.00394. URL https://openaccess.thecvf.com/content/CVPR2025/html/Li_MBQ_Modality-Balanced_Quantization_for_Large_Vision-Language_Models_CVPR_2025_paper.html.

Zhikai Li, Junrui Xiao, Lianwei Yang, and Qingyi Gu. Repq-vit: Scale reparameterization for post-training quantization of vision transformers. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pp. 17181–17190. IEEE, 2023. doi: 10.1109/ICCV51070.2023.01580. URL https://doi.org/10.1109/ICCV51070.2023.01580.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. AWQ: Activation-aware weight quantization for on-device LLM compression and acceleration. In *Proceedings of the Seventh Annual Conference on Machine Learning and Systems (MLSys)*, 2024.

Yang Lin, Tianyu Zhang, Peiqin Sun, Zheng Li, and Shuchang Zhou. Fq-vit: Post-training quantization for fully quantized vision transformer. In Luc De Raedt (ed.), *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022*, pp. 1173–1179. ijcai.org, 2022. doi: 10.24963/IJCAI.2022/164. URL https://doi.org/10.24963/ijcai.2022/164.

Yifei Liu, Jicheng Wen, Yang Wang, Shengyu Ye, Li Lyna Zhang, Ting Cao, Cheng Li, and Mao Yang. Vptq: Extreme low-bit vector post-training quantization for large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 8181–8196, 2024.

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 *Proceedings of the Thirteenth International Conference on Learning Representations (ICLR)*, 2025.

Qian Lou, Feng Guo, Lantao Liu, Minje Kim, and Lei Jiang. Autoq: Automated kernel-wise neural network quantization. *arXiv preprint arXiv:1902.05690*, 2019.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL https://openreview.net/forum?id=Byj72udxe.

Meta AI. Llama 3.2: Model cards and prompt formats. https://www.llama.com/docs/model-cards-and-prompt-formats/llama3_2/, 2024. Accessed: 2026-05-07.

Markus Nagel, Rana Ali Amjad, Mart van Baalen, Christos Louizos, and Tijmen Blankevoort. Up or down? Adaptive rounding for post-training quantization. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, pp. 7197–7206, 2020.

Seungcheol Park, Jeongin Bae, Beomseok Kwon, Minjun Kim, Byeongwook Kim, Se Jung Kwon, U Kang, and Dongsoo Lee. Unifying uniform and binary-coding quantization for accurate compression of large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 28468–28488, 2025.

Qwen Team. Qwen1.5-moe: Matching 7b model performance with 1/3 activated parameters, March 2024. URL https://qwenlm.github.io/blog/qwen-moe/.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67, 2020. URL https://jmlr.org/papers/v21/20-074.html.

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 *Proceedings of the Twelfth International Conference on Learning Representations (ICLR)*, 2024.

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 Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, Proceedings of Machine Learning Research. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/sun25l.html.

Chen Tang, Kai Ouyang, Zhi Wang, Yifei Zhu, Wen Ji, Yaowei Wang, and Wenwu Zhu. Mixed-precision neural network quantization via learned layer-wise importance. In *European conference on computer vision*, pp. 259–275. Springer, 2022.

Qian-Yuan Tang, Yufei Gu, Yunfeng Cai, Mingming Sun, Ping Li, Zhou Xun, and Zeke Xie. Investigating the overlooked hessian structure: From cnns to llms. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, Proceedings of Machine Learning Research. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/tang25d.html.

Llama Team. The llama 3 herd of models. *CoRR*, abs/2407.21783, 2024. doi: 10.48550/ARXIV.2407.21783. URL https://doi.org/10.48550/arXiv.2407.21783.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, Proceedings of Machine Learning Research, pp. 10347–10357. PMLR, 2021. URL http://proceedings.mlr.press/v139/touvron21a.html.

Albert Tseng, Jerry Chee, Qingyao Sun, Volodymyr Kuleshov, and Christopher De Sa. QuIP#: Even better LLM quantization with hadamard incoherence and lattice codebooks. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, pp. 48630–48656, 2024a.

Albert Tseng, Qingyao Sun, David Hou, and Christopher De Sa. QTIP: Quantization with trellises and incoherence processing. In *Advances in Neural Information Processing Systems 37 (NeurIPS)*, 2024b.

Kuan Wang, Zhijian Liu, Yujun Lin, Ji Lin, and Song Han. Haq: Hardware-aware automated quantization with mixed precision. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8612–8620, 2019.

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 *Proceedings of the 40th International Conference on Machine Learning (ICML)*, pp. 38087–38099, 2023.

Ke Xu, Zhongcheng Li, Shanshan Wang, and Xingyi Zhang. Ptmq: Post-training multi-bit quantization of neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 16193–16201, 2024.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report. *CoRR*, abs/2412.15115, 2024. doi: 10.48550/ARXIV.2412.15115. URL https://doi.org/10.48550/arXiv.2412.15115.

Zhewei Yao, Zhen Dong, Zhangcheng Zheng, Amir Gholami, Jiali Yu, Eric Tan, Leyuan Wang, Qijing Huang, Yida Wang, Michael Mahoney, et al. Hawq-v3: Dyadic neural network quantization. In *International Conference on Machine Learning*, pp. 11875–11886. PMLR, 2021.

Zhihang Yuan, Chenhao Xue, Yiqi Chen, Qiang Wu, and Guangyu Sun. Ptq4vit: Post-training quantization for vision transformers with twin uniform quantization. In Shai Avidan, Gabriel J. Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner (eds.), *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part XII*, Lecture Notes in Computer Science, pp. 191–207. Springer, 2022. doi: 10.1007/978-3-031-19775-8_12. URL https://doi.org/10.1007/978-3-031-19775-8_12.

Zhihang Yuan, Lin Niu, Jiawei Liu, Wenyu Liu, Xinggang Wang, Yuzhang Shang, Guangyu Sun, Qiang Wu, Jiaxiang Wu, and Bingzhe Wu. RPTQ: reorder-based post-training quantization for large language models. *CoRR*, abs/2304.01089, 2023. doi: 10.48550/ARXIV.2304.01089. URL https://doi.org/10.48550/arXiv.2304.01089.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? In Anna Korhonen, David R. Traum, and Lluís Màrquez (eds.), *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pp. 4791–4800. Association for Computational Linguistics, 2019. doi: 10.18653/V1/P19-1472. URL https://doi.org/10.18653/v1/p19-1472.

Boyang Zhang, Daning Cheng, Yunquan Zhang, Meiqi Tu, Fangmin Liu, and Jiake Tian. A general error-theoretical analysis framework for constructing compression strategies. *CoRR*, abs/2502.15802, 2025. doi: 10.48550/ARXIV.2502.15802. URL https://doi.org/10.48550/arXiv.2502.15802.

Yushun Zhang, Congliang Chen, Tian Ding, Ziniu Li, Ruoyu Sun, and Zhi-Quan Luo. Why transformers need adam: A hessian perspective. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/ee0e45ff4de76cbfdf07015a7839f339-Abstract-Conference.html.

A Theoretical Analysis and Proofs

This appendix provides the complete proofs of Theorems 1 and 2. We first fix the coordinate convention and the standard local second-order approximation used throughout the analysis.

A.1 Coordinate Identities and Second-Order Convention

The transformation $\mathbf{T}$ maps quantization coordinates to the original weight coordinates. Therefore,

$$\mathbf{z} = \mathbf{T}^{-1}\mathbf{w}, \qquad \mathbf{w}_q = \mathbf{T}Q_b(\mathbf{z}), \qquad \boldsymbol{\varepsilon} = Q_b(\mathbf{z}) - \mathbf{z}. \tag{5}$$

The perturbation in the original coordinates is exactly

$$\begin{aligned} \mathbf{e} &= \mathbf{w}_q - \mathbf{w} \\ &= \mathbf{T}Q_b(\mathbf{z}) - \mathbf{T}\mathbf{z} \\ &= \mathbf{T}\boldsymbol{\varepsilon}. \end{aligned} \tag{6}$$

Let $\mathbf{g} = \nabla\mathcal{L}(\mathbf{w})$ and $\mathbf{H} = \nabla^2\mathcal{L}(\mathbf{w})$. The local Taylor expansion is

$$\Delta\mathcal{L} = \mathbf{g}^\top\mathbf{e} + \frac{1}{2}\mathbf{e}^\top\mathbf{H}\mathbf{e} + \mathcal{O}(\|\mathbf{e}\|_2^3). \tag{7}$$

Following the standard PTQ approximation, the pre-trained weights are assumed to lie near a stationary point, so $\mathbf{g} \approx \mathbf{0}$, and the quantization perturbation is sufficiently small for the higher-order term to be neglected. Accordingly, $\Delta\mathcal{L}$ denotes the resulting second-order approximation in the remainder of this appendix.

A.2 Proof of Theorem 1

Proof. Using Eq. equation 6, the second-order loss change can be written in the quantization coordinates as

$$\begin{aligned} \Delta\mathcal{L} &= \frac{1}{2}\mathbf{e}^\top\mathbf{H}\mathbf{e} \\ &= \frac{1}{2}\boldsymbol{\varepsilon}^\top\mathbf{T}^\top\mathbf{H}\mathbf{T}\boldsymbol{\varepsilon} \\ &= \frac{1}{2}\boldsymbol{\varepsilon}^\top\mathbf{H}_q\boldsymbol{\varepsilon}, \end{aligned} \tag{8}$$

where $\mathbf{H}_q = \mathbf{T}^\top\mathbf{H}\mathbf{T}$ is the effective Hessian in the quantization coordinates.

We next verify that the Rayleigh-quotient argument applies to $\mathbf{H}_q$. Because $\mathbf{H} = \mathbf{H}^\top \succeq \mathbf{0}$,

$$\mathbf{H}_q^\top = (\mathbf{T}^\top\mathbf{H}\mathbf{T})^\top = \mathbf{T}^\top\mathbf{H}\mathbf{T} = \mathbf{H}_q, \tag{9}$$

and, for every $\mathbf{x} \in \mathbb{R}^d$,

$$\mathbf{x}^\top\mathbf{H}_q\mathbf{x} = (\mathbf{T}\mathbf{x})^\top\mathbf{H}(\mathbf{T}\mathbf{x}) \geq 0. \tag{10}$$

Thus $\mathbf{H}_q$ is real, symmetric, and positive semidefinite.

If $\boldsymbol{\varepsilon} = \mathbf{0}$, the desired inequality is immediate. Otherwise, define the unit vector $\mathbf{u} = \boldsymbol{\varepsilon}/\|\boldsymbol{\varepsilon}\|_2$. By the spectral theorem,

$$\mathbf{H}_q = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^\top, \qquad \boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_d), \tag{11}$$

where $\mathbf{U}$ is orthogonal and $0 \leq \lambda_i \leq \lambda_{\max}(\mathbf{H}_q)$. Let $\mathbf{c} = \mathbf{U}^\top\mathbf{u}$. Orthogonality gives $\|\mathbf{c}\|_2^2 = \|\mathbf{u}\|_2^2 = 1$. Hence the Rayleigh quotient satisfies

$$\begin{aligned} \mathbf{u}^\top\mathbf{H}_q\mathbf{u} &= \mathbf{c}^\top\boldsymbol{\Lambda}\mathbf{c} \\ &= \sum_{i=1}^{d}\lambda_i c_i^2 \\ &\leq \lambda_{\max}(\mathbf{H}_q)\sum_{i=1}^{d} c_i^2 \\ &= \lambda_{\max}(\mathbf{H}_q). \end{aligned} \tag{12}$$

Multiplying by $\|\varepsilon\|_2^2$ yields

$$\varepsilon^\top \mathbf{H}_q \varepsilon \le \lambda_{\max}(\mathbf{H}_q)\|\varepsilon\|_2^2. \tag{13}$$

Combining Eqs. equation 8 and equation 13 gives

$$\Delta\mathcal{L} \le \frac{1}{2}\lambda_{\max}(\mathbf{H}_q)\|\varepsilon\|_2^2, \tag{14}$$

which proves Theorem 1. Equality holds precisely when a nonzero ε lies entirely in the eigenspace associated with $\lambda_{\max}(\mathbf{H}_q)$. □

A.3 Curvature–Residual–Alignment Decomposition

Corollary A.1 (Curvature–Residual–Alignment Decomposition)**.** *Assume* $\mathbf{H}_q \succeq \mathbf{0}$*. For* $\varepsilon \neq \mathbf{0}$ *and* $\lambda_{\max}(\mathbf{H}_q) > 0$*, define*

$$a(\varepsilon, \mathbf{H}_q) = \frac{\varepsilon^\top \mathbf{H}_q \varepsilon}{\lambda_{\max}(\mathbf{H}_q)\|\varepsilon\|_2^2}. \tag{15}$$

Then $a(\varepsilon, \mathbf{H}_q) \in [0, 1]$ *and*

$$\Delta\mathcal{L} = \frac{1}{2}\|\varepsilon\|_2^2 \lambda_{\max}(\mathbf{H}_q) a(\varepsilon, \mathbf{H}_q). \tag{16}$$

If $\varepsilon = \mathbf{0}$ *or* $\lambda_{\max}(\mathbf{H}_q) = 0$*, we set* $a = 0$*.*

Proof. Let $\mathbf{H}_q = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$ and $\mathbf{r} = \mathbf{U}^\top\varepsilon$. Then

$$\varepsilon^\top \mathbf{H}_q \varepsilon = \sum_{i=1}^{d} \lambda_i r_i^2, \qquad \|\mathbf{r}\|_2^2 = \|\varepsilon\|_2^2. \tag{17}$$

Since $0 \le \lambda_i \le \lambda_{\max}(\mathbf{H}_q)$,

$$0 \le \sum_{i=1}^{d} \lambda_i r_i^2 \le \lambda_{\max}(\mathbf{H}_q) \sum_{i=1}^{d} r_i^2. \tag{18}$$

Dividing by $\lambda_{\max}(\mathbf{H}_q)\|\varepsilon\|_2^2$ gives $0 \le a \le 1$. Substituting Eq. equation 15 into Eq. equation 8 gives Eq. equation 16. □

Corollary A.1 separates the second-order loss into the transformed-space residual magnitude, the largest effective curvature, and the alignment of the residual with the curvature eigenspace. This decomposition supports the distinct roles of smoothing and rotation in HeRo-Q.

A.4 Proof of Theorem 2

Proof. Under the normalized diagonal proxy assumption,

$$\mathbf{H}_{\text{proxy}} = \operatorname{diag}(h_1, \dots, h_d), \qquad 1 = h_{\min} \le h_i \le h_{\max}, \qquad h_{\max} > 1. \tag{19}$$

Because $h_i > 0$, the absolute values in the smoothing definition can be dropped:

$$\mathbf{D}_\alpha = \operatorname{diag}\left(h_1^{\alpha/2}, \dots, h_d^{\alpha/2}\right). \tag{20}$$

For $\mathbf{T}_\alpha = \mathbf{D}_\alpha^{-1}\mathbf{R}$,

$$\begin{aligned} \mathbf{H}_{q,\text{proxy}}(\alpha) &= \mathbf{T}_\alpha^\top \mathbf{H}_{\text{proxy}} \mathbf{T}_\alpha \\ &= \mathbf{R}^\top \mathbf{D}_\alpha^{-1} \mathbf{H}_{\text{proxy}} \mathbf{D}_\alpha^{-1} \mathbf{R}. \end{aligned} \tag{21}$$

The middle matrix is diagonal, with

$$\begin{aligned} \mathbf{D}_\alpha^{-1}\mathbf{H}_{\text{proxy}}\mathbf{D}_\alpha^{-1} &= \operatorname{diag}\left(h_i^{-\alpha/2} h_i h_i^{-\alpha/2}\right)_{i=1}^{d} \\ &= \operatorname{diag}\left(h_1^{1-\alpha}, \dots, h_d^{1-\alpha}\right). \end{aligned} \tag{22}$$

Since $\mathbf{R}$ is orthogonal, Eq. equation 21 is an orthogonal similarity transformation and therefore preserves every eigenvalue. Consequently,

$$\lambda_i(\mathbf{H}_{q,\text{proxy}}(\alpha)) = h_i^{1-\alpha}, \qquad \lambda_{\max}(\mathbf{H}_{q,\text{proxy}}(\alpha)) = h_{\max}^{1-\alpha}. \tag{23}$$

At $\alpha = 0$,

$$\lambda_{\max}(\mathbf{H}_{q,\text{proxy}}(0)) = h_{\max}. \tag{24}$$

For every $\alpha \in (0, 1]$, we have $1 - \alpha < 1$ and $h_{\max} > 1$. Therefore,

$$h_{\max}^{1-\alpha} < h_{\max}, \tag{25}$$

which proves

$$\lambda_{\max}(\mathbf{H}_{q,\text{proxy}}(\alpha)) < \lambda_{\max}(\mathbf{H}_{q,\text{proxy}}(0)). \tag{26}$$

This is exactly the claim of Theorem 2. Moreover, because the minimum eigenvalue remains one,

$$\kappa(\mathbf{H}_{q,\text{proxy}}(\alpha)) = \kappa(\mathbf{H}_{\text{proxy}})^{1-\alpha}. \tag{27}$$

□

Extension to a non-diagonal proxy. The exact expression in Eq. equation 23 uses the diagonal proxy assumption of Theorem 2. For a general positive-semidefinite proxy, define

$$\mathbf{A}_\alpha = \mathbf{D}_\alpha^{-1}\mathbf{H}_{\text{proxy}}\mathbf{D}_\alpha^{-1}. \tag{28}$$

Under the same normalization, $\mathbf{D}_\alpha \succeq \mathbf{I}$. Hence, for every unit vector $\mathbf{x}$,

$$\begin{aligned} \mathbf{x}^\top \mathbf{A}_\alpha \mathbf{x} &= (\mathbf{D}_\alpha^{-1}\mathbf{x})^\top \mathbf{H}_{\text{proxy}}(\mathbf{D}_\alpha^{-1}\mathbf{x}) \\ &\le \lambda_{\max}(\mathbf{H}_{\text{proxy}})\|\mathbf{D}_\alpha^{-1}\mathbf{x}\|_2^2 \\ &\le \lambda_{\max}(\mathbf{H}_{\text{proxy}}). \end{aligned} \tag{29}$$

Taking the maximum over unit vectors and using the orthogonal invariance induced by $\mathbf{R}$ gives

$$\lambda_{\max}(\mathbf{H}_{q,\text{proxy}}(\alpha)) = \lambda_{\max}(\mathbf{A}_\alpha) \le \lambda_{\max}(\mathbf{H}_{\text{proxy}}). \tag{30}$$

Thus smoothing is non-expansive for a general positive-semidefinite proxy. The inequality is strict whenever the leading eigenspace of $\mathbf{H}_{\text{proxy}}$ contains no nonzero vector supported entirely on the unscaled coordinates $\{i : h_i = h_{\min}\}$. This explicit condition replaces the informal claim that a long-tailed diagonal alone always implies strict contraction in the non-diagonal case.

A.5 Connection Between True and Proxy Curvature

Proxy-curvature discrepancy. Define the true and proxy second-order losses for the same perturbation by

$$\Delta\mathcal{L}_{\mathbf{H}} = \frac{1}{2}\mathbf{e}^\top \mathbf{H}\mathbf{e}, \qquad \Delta\mathcal{L}_{\text{proxy}} = \frac{1}{2}\mathbf{e}^\top \mathbf{H}_{\text{proxy}}\mathbf{e}. \tag{31}$$

Their difference satisfies

$$\begin{aligned} |\Delta\mathcal{L}_{\mathbf{H}} - \Delta\mathcal{L}_{\text{proxy}}| &= \frac{1}{2}\left|\mathbf{e}^\top(\mathbf{H} - \mathbf{H}_{\text{proxy}})\mathbf{e}\right| \\ &\le \frac{1}{2}\|\mathbf{H} - \mathbf{H}_{\text{proxy}}\|_2\|\mathbf{e}\|_2^2 \\ &= \frac{1}{2}\|\mathbf{H} - \mathbf{H}_{\text{proxy}}\|_2\|\mathbf{T}\boldsymbol{\varepsilon}\|_2^2. \end{aligned} \tag{32}$$

This bound explicitly separates the exact Hessian used in Theorem 1 from the activation-induced proxy used by the algorithm and Theorem 2.

Equivalence to the practical reconstruction objective. For a weight matrix, let $\mathbf{E} = \mathbf{W}_q - \mathbf{W}$. With $\mathbf{H}_{\text{proxy}} = \mathbf{X}^\top\mathbf{X}$, the layer-wise reconstruction objective used by HeRo-Q obeys

$$\begin{aligned} \|\mathbf{X}\mathbf{W} - \mathbf{X}\mathbf{W}_q\|_F^2 &= \|\mathbf{X}\mathbf{E}\|_F^2 \\ &= \mathrm{Tr}\left(\mathbf{E}^\top\mathbf{X}^\top\mathbf{X}\mathbf{E}\right) \\ &= \mathrm{Tr}\left(\mathbf{E}^\top\mathbf{H}_{\text{proxy}}\mathbf{E}\right). \end{aligned} \tag{33}$$

Thus the practical search over $(\alpha, \mathbf{R})$ minimizes a proxy-curvature-weighted perturbation, rather than an unweighted Euclidean weight error.

Distinct roles of smoothing and rotation. Let $\mathbf{A}_\alpha = \mathbf{D}_\alpha^{-1}\mathbf{H}_{\text{proxy}}\mathbf{D}_\alpha^{-1}$. Then

$$\mathbf{H}_{q,\text{proxy}} = \mathbf{R}^\top \mathbf{A}_\alpha \mathbf{R}, \qquad \lambda_i(\mathbf{H}_{q,\text{proxy}}) = \lambda_i(\mathbf{A}_\alpha). \tag{34}$$

Hence the orthogonal rotation does not change the effective eigenvalues; the spectral compression is produced by $\mathbf{D}_\alpha$. However, $\mathbf{T}_\alpha^{-1} = \mathbf{R}^\top \mathbf{D}_\alpha$, so the transformed-space residual is

$$\boldsymbol{\varepsilon}(\mathbf{R}) = Q_b\big(\mathbf{R}^\top \mathbf{D}_\alpha \mathbf{w}\big) - \mathbf{R}^\top \mathbf{D}_\alpha \mathbf{w}. \tag{35}$$

Therefore $\mathbf{R}$ changes both the residual magnitude and its alignment with the effective-curvature eigenspace. HeRo-Q optimizes $\mathbf{R}$ through the reconstruction objective in Eq. equation 33; an arbitrary rotation is not claimed to reduce the loss.

B Supplementary Experiments

B.1 Additional Results Across Llama and Qwen Families

Table 6 extends the W4A8 evaluation to Llama-3.2-1B, Llama-3.2-3B, and Qwen2.5-3B, covering two model families and multiple scales.

Across the three models, HeRo-Q improves over the listed GPTQ, SmoothQuant, and SpinQuant baselines. On Llama-3.2-1B, it changes GPTQ's C4 perplexity from 24.07 to 22.84 and GSM8K accuracy from 24.24 to 31.43. The direction of the gain is consistent on Llama-3.2-3B and Qwen2.5-3B across all reported metrics.

Method	Llama-3.2-1B					Llama-3.2-3B					Qwen2.5-3B				
	C4↓	W2↓	GSM8K↑	HS↑	MMLU↑	C4↓	W2↓	GSM8K↑	HS↑	MMLU↑	C4↓	W2↓	GSM8K↑	HS↑	MMLU↑
FP16	22.48	11.41	36.44	59.94	46.38	17.24	9.53	68.86	70.05	61.44	14.68	7.95	81.13	72.09	65.80
GPTQ	24.07	16.18	24.24	57.04	40.58	18.99	9.82	55.75	68.35	57.16	15.65	8.87	76.44	70.74	63.25
SmoothQuant	24.01	14.10	25.87	56.89	41.32	18.87	9.80	56.33	69.34	58.73	15.54	8.34	77.36	71.34	63.96
SpinQuant	23.56	13.89	26.13	57.68	42.64	18.77	9.75	57.93	69.83	58.31	15.32	8.14	78.65	71.94	65.08
HeRo-Q	**22.84**	**12.67**	**31.43**	**57.98**	**43.77**	**17.87**	**9.73**	**61.81**	**69.97**	**59.87**	**14.98**	**8.01**	**79.68**	**72.01**	**65.49**

Table 6: Additional W4A8 results across Llama and Qwen models. C4 and W2 (WikiText2) report perplexity, where lower is better. GSM8K, HS (HellaSwag), and MMLU report accuracy, where higher is better. HeRo-Q achieves the best overall performance among quantized methods across the reported models and metrics.

B.2 Compatibility with Different Base Quantizers

We next test whether the coordinate construction transfers across PTQ backends by pairing it with GPTQ, AWQ, and RTN. Table 7 uses matched model, calibration, and bit-width settings.

On Llama-3.1-8B, conditioning improves GPTQ from 8.81 to 6.96 on WikiText-2 and from 60.54 to 66.31 on MMLU; it also improves AWQ and RTN. The same pattern holds for all three backends on Qwen2.5-3B. These results support backend compatibility across second-order reconstruction, activation-aware scaling, and direct rounding, without implying that the integrations are algorithmically identical.

Model	Base Method	C4 ↓	W2 ↓	MMLU ↑	HellaSwag ↑
Llama-3.1-8B	GPTQ	14.12 → **12.94**	8.81 → **6.96**	60.54 → **66.31**	76.91 → **77.28**
Llama-3.1-8B	AWQ	13.56 → **12.77**	8.12 → **6.88**	64.41 → **66.87**	76.78 → **77.26**
Llama-3.1-8B	RTN	15.98 → **13.87**	9.11 → **7.59**	57.12 → **64.12**	75.11 → **77.23**
Qwen2.5-3B	GPTQ	15.65 → **14.98**	8.87 → **8.01**	63.25 → **65.49**	70.74 → **72.01**
Qwen2.5-3B	AWQ	15.46 → **14.72**	8.46 → **8.09**	63.98 → **65.77**	71.18 → **72.04**
Qwen2.5-3B	RTN	18.12 → **16.32**	10.23 → **8.67**	62.17 → **64.11**	61.25 → **63.25**

Table 7: Compatibility of HeRo-Q with different base quantizers. Each cell reports the performance before and after applying HeRo-Q. Lower perplexity is better, and higher accuracy is better.

B.3 Additional Results on MoE model

We additionally evaluate Mixtral-8×7B to test a larger sparse architecture, where experts may exhibit heterogeneous activation and curvature statistics.

Table 8 reports gains in both tested settings. Under W3A16g128, HeRo-Q obtains 4.12 WikiText-2 and 7.86 C4 perplexity, compared with GPTQ's 4.93 and 8.52. Under W6A6, it improves SmoothQuant from 4.28 to 3.96 on WikiText-2 and from 7.89 to 7.67 on C4.

Configuration	Method	WikiText2 ↓	C4 ↓
FP16	FP16	3.84	7.40
W3A16g128	AWQ	4.73	8.29
W3A16g128	GPTQ	4.93	8.52
W3A16g128	**HeRo-Q**	**4.12**	**7.86**
W6A6	SmoothQuant	4.28	7.89
W6A6	**HeRo-Q**	**3.96**	**7.67**

Table 8: Perplexity results on Mixtral-8x7B under different quantization configurations.

Table 9 reports a separate W4A16 evaluation on reasoning, code, and commonsense tasks, alongside perplexity. Relative to GPTQ, HeRo-Q changes C4 perplexity from 7.67 to 7.11 and MMLU from 68.50 to 69.32; it also exceeds the listed MoEQuant+ values on every reported metric. Together with the smaller MoE models in the main text, this provides empirical evidence across three sparse expert architectures.

Method	W2 ↓	C4 ↓	MMLU ↑	HumanEval ↑	GSM8K ↑	HellaSwag ↑	OpenBookQA ↑	MathQA ↑
FP16	3.84	7.40	70.50	32.93	65.88	64.88	35.80	42.41
RTN	5.41	8.13	62.20	28.05	27.90	61.73	32.20	37.35
AWQ	5.01	7.98	62.75	25.00	38.67	62.11	33.60	38.43
GPTQ	4.03	7.67	68.50	27.60	57.92	64.08	30.60	41.07
MoEQuant+	5.15	7.84	64.66	25.45	50.66	62.73	34.00	39.77
HeRo-Q	**3.91**	**7.11**	**69.32**	**28.94**	**59.44**	**64.21**	**34.60**	**41.68**

Table 9: Downstream evaluation on Mixtral-8x7B under W4A16 quantization. All quantized rows use 4-bit weights with group size 128 and FP16 activations; FP16 is the unquantized reference. HeRo-Q uses GPTQ, and MoE routers and the KV cache remain in FP16. Full quantization and calibration settings are given in Appendix C.2. HeRo-Q achieves the best or near-best performance among quantized methods across most evaluated metrics.

B.4 Comparison with Vector and Non-Uniform Quantization Methods

Table 10 compares HeRo-Q with the vector-quantized VPTQ (Liu et al., 2024) and non-uniform UniQuanF (Park et al., 2025) on Llama-3.1-8B under matched bit-widths.

At 4 and 3 bits, HeRo-Q improves the listed VPTQ results in both perplexity and HellaSwag. Relative to UniQuanF, it reduces 4-bit WikiText-2 perplexity from 7.01 to 6.54 and improves MMLU and GSM8K by 0.98 and 1.18 points; at 3 bits, the corresponding MMLU and GSM8K gains are 4.22 and 4.86 points. The comparison shows that the observed benefit is not limited to uniform scalar PTQ baselines.

Method	Llama-3.1-8B			Method	Llama-3.1-8B			
	Bit-width	WikiText2↓	HellaSwag↑		Bit-width	WikiText2↓	MMLU↑	GSM8K↑
VPTQ	4-bit	7.42	59.30	UniQuanF	4-bit	7.01	61.43	72.38
HeRo-Q	**4-bit**	**7.11**	**60.23**	**HeRo-Q**	**4-bit**	**6.54**	**62.41**	**73.56**
VPTQ	3-bit	7.97	58.40	UniQuanF	3-bit	8.75	53.46	58.73
HeRo-Q	**3-bit**	**7.39**	**59.44**	**HeRo-Q**	**3-bit**	**7.94**	**57.68**	**63.59**

Table 10: Additional comparison with VPTQ and UniQuanF on Llama-3.1-8B. WikiText2 reports perplexity, where lower is better; HellaSwag, MMLU, and GSM8K report accuracy, where higher is better. Within each panel, HeRo-Q and the baseline use the same bit-width and underlying quantizer; the two panels use different quantizers, so HeRo-Q results are compared only within each panel.

B.5 Calibration Sensitivity

We vary both calibration source (C4 versus WikiText-2) and budget (64 versus 128 samples) to quantify dependence on calibration choices.

Across these four configurations, the reported metrics vary within narrow ranges. For Llama-3.1-8B W4A8, WikiText-2 perplexity ranges from 6.87 to 7.01 and MMLU from 66.2 to 66.3; under W3A16, GSM8K ranges from 70.0 to 70.2. Llama-3.2-1B shows similarly small variation in the evaluated settings.

Increasing the budget to 128 samples gives small, generally favorable changes; for example, Llama-3.1-8B W3A16 perplexity decreases from 8.09 to 8.02 with C4 and from 7.87 to 7.69 with WikiText-2. This experiment supports robustness to the tested calibration choices, while not claiming invariance to arbitrary domain shift.

(a) Llama-3.1-8B

Calibration Set	Samples	W4A8 C4 ↓	W4A8 W2 ↓	W4A8 MMLU ↑	W4A8 GSM ↑	W3A16 C4 ↓	W3A16 W2 ↓	W3A16 MMLU ↑	W3A16 GSM ↑
C4	64	12.94	7.01	66.23	75.33	13.99	8.09	63.72	70.12
C4	128	12.94	6.96	66.31	75.26	13.98	8.02	63.80	70.15
WikiText-2	64	13.07	6.91	66.19	74.97	14.07	7.87	63.80	70.03
WikiText-2	128	13.01	6.87	66.31	75.11	14.02	7.69	63.91	70.11

(b) Llama-3.2-1B

Calibration Set	Samples	W4A8 C4 ↓	W4A8 W2 ↓	W4A8 MMLU ↑	W4A8 GSM ↑	W3A16 C4 ↓	W3A16 W2 ↓	W3A16 MMLU ↑	W3A16 GSM ↑
C4	64	22.89	12.72	43.79	31.41	25.89	14.12	42.50	29.04
C4	128	22.84	12.67	43.77	31.43	25.80	14.05	42.50	29.10
WikiText-2	64	22.98	12.61	43.79	31.33	25.99	14.39	42.61	29.12
WikiText-2	128	22.94	12.57	43.88	31.47	25.84	14.31	42.69	29.17

Table 11: Calibration sensitivity on Llama-3.1-8B and Llama-3.2-1B. Across both models, HeRo-Q remains stable with respect to the calibration dataset and sample size.

B.6 Control Experiment with Downstream Calibration

We test calibration-domain mismatch as an alternative explanation by replacing C4 with disjoint GSM8K-training samples while holding the token budget, quantization setting, and evaluation pipeline fixed. Table 12 shows that target-domain calibration can improve GSM8K, particularly for AWQ, but worsens its perplexity and does not prevent severe W4A4 degradation. HeRo-Q retains its advantage under both calibration sources. Thus, mismatch contributes to performance but does not fully explain the observed gap in this controlled setting. This is an alternative-hypothesis check, not causal evidence that curvature alone determines downstream accuracy.

Setting	Method	W2 / C4 ↓		MMLU / GSM8K ↑	
		C4 calib.	GSM8K calib.	C4 calib.	GSM8K calib.
W4A8	AWQ	8.12 / 13.56	9.20 / 14.90	64.41 / 72.94	65.50 / 74.20
W4A8	HeRo-Q	**6.96 / 12.94**	**7.09 / 12.95**	**66.31 / 75.26**	**66.40 / 75.30**
W4A4	AWQ	346.0 / 642.4	> 500 / > 800	36.45 / 30.76	38.50 / 39.20
W4A4	HeRo-Q	**10.29 / 14.84**	**10.32 / 14.88**	**45.31 / 42.33**	**47.50 / 47.54**
W3A16	AWQ	15.87 / 24.15	18.50 / 28.30	39.50 / 46.21	51.20 / 57.50
W3A16	HeRo-Q	**8.02 / 13.98**	**8.05 / 14.02**	**63.80 / 70.15**	**64.95 / 71.25**

Table 12: Control experiment with downstream calibration on Llama-3.1-8B. The C4 calibration set is replaced by disjoint GSM8K-training samples under the same token budget. Each entry reports the indicated pair of metrics.

C Implementation and Efficiency

C.1 Code

https://anonymous.4open.science/r/HeRo-Q-3775

C.2 Quantization Configurations

HeRo-Q directly optimizes only the weight-side quantization coordinates. No HeRo-Q-specific modification, calibration procedure, or optimization is introduced for activation quantization. In the W4A8 and W4A4 experiments, activations are quantized using the same standard dynamic per-token uniform quantizer as the corresponding baselines. These experiments evaluate whether HeRo-Q's weight-side curvature conditioning remains beneficial when activation quantization error is also present; they do not imply that the current HeRo-Q formulation directly optimizes activation quantization error.

Optimization versus evaluation. The search over α and the optimization of $\mathbf{R}$ simulate only weight quantization, including the reconstruction-error comparison used to select the transformation. The group-wise selection objective is

$$\mathcal{L}_g = \sum_{j \in \mathcal{C}_g} \left\| \mathbf{X}_g \mathbf{W}_j - (\mathbf{X}_g \mathbf{T}_g) Q_W(\mathbf{T}_g^{-1} \mathbf{W}_j) \right\|_F^2 . \tag{36}$$

For W4A4 and W4A8, activation quantization is enabled only for evaluation and inference after the transformation has been selected, following Eq. equation 37.

Table 13 summarizes the configurations with GPTQ as the default weight quantization backend.

Setting	Weight quantization	Activation quantization	Activation granularity	Integer range
W4A16	4-bit GPTQ, group size 128	FP16	—	W: $[0, 15]$
W3A16	3-bit GPTQ, group size 128	FP16	—	W: $[0, 7]$
W4A8	4-bit GPTQ, group size 128	8-bit dynamic per-token uniform	One scale and zero point per token over the complete hidden dimension	W: $[0, 15]$; A: $[0, 255]$
W4A4	4-bit GPTQ, group size 128	4-bit dynamic per-token uniform	One scale and zero point per token over the complete hidden dimension	W: $[0, 15]$; A: $[0, 15]$

Table 13: Quantization configurations for the default GPTQ-backed HeRo-Q experiments. W and A denote weights and activations, respectively.

Weight quantization. Each weight matrix is quantized per output channel, with groups of 128 consecutive elements formed along the input dimension. We use asymmetric uniform quantization with one scale and zero point per group. The GPTQ Hessian damping ratio and block size are set to 0.01 and 128, respectively. The activation-ordering and static-group heuristics are disabled, and no additional weight clipping is applied.

Activation quantization. For both W4A4 and W4A8, activation quantization is applied after the HeRo-Q coordinate transformation and immediately before each quantized linear GEMM. Specifically, the transformed activation is $\widetilde{\mathbf{X}} = \mathbf{X}\mathbf{T}$, and the quantized linear output is

$$\mathbf{Y}_q = Q_A(\mathbf{X}\mathbf{T})\, Q_W(\mathbf{T}^{-1}\mathbf{W}), \tag{37}$$

where Q_A and Q_W denote activation and weight quantization with reconstruction to real-valued tensors, respectively. Thus, Q_A acts on $\mathbf{X}\mathbf{T}$, not on the original activation $\mathbf{X}$ before the transformation.

This logical order is preserved whether the transformation is fused into adjacent operations or implemented online.

For W4A8 and W4A4, activations are quantized using dynamic per-token asymmetric min–max uniform quantization with round-to-nearest. The scale and zero point are recomputed at runtime for each token over the complete hidden dimension. The activation group size is set to -1, the clipping ratio is 1.0, and no additional activation clipping or calibration-based activation optimization is used.

Precision exceptions and matched settings. We quantize the weights of all linear projections in attention (query, key, value, and output), dense MLPs, and MoE experts, including the up, gate (where present), and down projections. MoE routers are excluded from quantization and remain in FP16; this exception does not apply to the gate projections within MLPs or experts. The KV cache, embeddings, normalization layers, softmax, and LM head also remain in FP16. In W4A16 and W3A16, all activations also remain in FP16. Within each matched comparison, methods use the same bit widths, group size, activation quantizer, calibration data, and evaluation protocol. Calibration uses 128 C4 sequences with a sequence length of 2048. HeRo-Q modifies the weight-side transformation while leaving the activation quantizer unchanged.

C.3 HeRo-Q Hyperparameters List

Table 14 lists the smoothing-search, rotation-optimization, and calibration settings used in the reported experiments. Optimization was run on a single NVIDIA A800 GPU. We search α over

Category	Hyperparameter	Value / Setting
Smoothing Search	Search Grid $\mathcal{A}$	$\{0, 0.1, 0.2, \ldots, 0.8\}$
	Search Metric	MSE of layer output
Rotation Optimization	Optimizer	Cayley SGD
	Learning Rate (η)	0.01
	Optimization Steps (T_{steps})	200
	Momentum	0.9
	Batch Size	32
Calibration Data	Dataset	C4
	Number of Samples	128
	Sequence Length	2048

Table 14: List of hyperparameters used in HeRo-Q optimization and quantization.

$\{0, 0.1, \ldots, 0.8\}$. The Cayley-SGD settings above apply to trainable dense rotations permitted by the deployment graph; online rotations remain in the structured family used at inference and are not unconstrained dense Cayley updates. Unless stated otherwise, 128 C4 sequences of length 2048 form the calibration set.

C.4 Inference overhead

Shared basis and local transforms. We describe a compatible RMSNorm/SwiGLU realization of the local construction $T_g = D_g^{-1} R_g$. A single shared residual-stream rotation R_1 is established using SpinQuant-style reparameterization (Liu et al., 2025) before local search. The normalization gains are first absorbed into all consuming projections; RMS normalization without its gain satisfies $n(hR_1) = n(h)R_1$. Embedding, residual-branch output, and output-head weights are adjusted consistently. Thus R_1 defines the shared reference basis, rather than an independent rotation learned for every Q/K/V or up/gate group.

All weights below are full-precision reference weights in that basis. Collect actual local inputs $\bar{X}_g$ there and form $\bar{H}_g = \bar{X}_g^\top \bar{X}_g$. The effective local proxy is

$$H_{q,\mathrm{proxy},g} = R_g^\top D_g^{-1} \bar{H}_g D_g^{-1} R_g. \tag{38}$$

Here R_g is a local post-conditioning rotation, not the shared R_1. At Norm outputs feeding Q/K/V or up/gate, set $R_g = I$: no additional local rotation is introduced. The identity $\bar{X}_g = X_g R_1$ is not

assumed at nonlinear FFN or attention-head interfaces. These choices preserve the local factorization without interchanging D_g^{-1} and R_1.

Normalization-output conditioning. For $x_g = \mathrm{Norm}_{\gamma_g,\beta_g}(h_g)$ and positive diagonal D_g, update the row-vector affine parameters and all consuming weights:

$$\gamma_g' = \gamma_g D_g^{-1}, \qquad \beta_g' = \beta_g D_g^{-1}, \qquad W_j' = D_g W_j. \tag{39}$$

Since the local rotation is the identity,

$$\mathrm{Norm}_{\gamma_g',\beta_g'}(h_g) W_j' = (x_g D_g^{-1})(D_g W_j) = x_g W_j. \tag{40}$$

Omit the normalization bias update when absent; projection biases are unchanged. One Norm output supplies one shared conditioner to its consumers, including any FP16 router. Different consumer-specific scales cannot all be absorbed into the same affine vector. The scale acts after normalization, not on its input or the residual skip path.

Value–output interface. Within a value head, let $T_2 = D_2^{-1} R_2$ with an admissible learned orthogonal R_2. For its attention matrix A, set

$$W_v' = W_v T_2, \qquad W_o' = T_2^{-1} W_o, \qquad A(VT_2)(T_2^{-1} W_o) = AVW_o. \tag{41}$$

The value bias is also multiplied by T_2. Both factors are fused offline. Transforms act within heads, not across heads with different attention matrices. In grouped-query attention, query heads sharing a KV head must share its value transform, with inverse compensation on every corresponding output-weight block; conditioner statistics and search variables respect these ties. No general diagonal conditioner is inserted into the Q/K dot product.

SwiGLU–down interface. With $z = \mathrm{SiLU}(xW_{\mathrm{gate}}) \odot (xW_{\mathrm{up}})$, the diagonal conditioner can be absorbed into the linear up branch:

$$\begin{aligned} W_{\mathrm{up}}' &= W_{\mathrm{up}} D_4^{-1}, & z' &= z D_4^{-1}, \\ W_{\mathrm{down}}' &= R_4^\top D_4 W_{\mathrm{down}}, & (z' R_4) W_{\mathrm{down}}' &= z W_{\mathrm{down}}. \end{aligned} \tag{42}$$

Scale the up-branch bias likewise when present. This does not move a scale through SiLU and assumes no uncompensated consumers of the up output. The prescribed normalized Hadamard R_4 remains online; it is fixed during local conditioner search, not substituted for a learned dense rotation later. Here the online family $\mathcal{R}_{\mathrm{fast}}$ in Algorithm 1 contains only transforms supported by that inference kernel; a prescribed fixed R_4 gives a singleton local family.

Interface	Local rotation	Conditioner absorption	Online transform
Q/K/V Norm output	$R_g = I$; shared basis R_1	Norm affine; all consumers	0
MLP up/gate input	$R_g = I$; same basis R_1	Norm affine; all consumers	0
V–O	Head-compatible learned R_2	Paired V–O weights	0
Q/K post-RoPE	R_3 disabled in this realization	No conditioner	0
SwiGLU–down	Fixed online R_4	Up-branch weight and bias	$O(d_{\mathrm{ffn}} \log d_{\mathrm{ffn}})$

Table 15: Compatible RMSNorm/SwiGLU deployment with FP16 KV caches and post-RoPE rotation disabled. Zero denotes no standalone online transform at that interface, not zero operator latency. FWHT cost assumes supported dimensions. This is an algebraic operator inventory, not a new timing measurement.

Final quantization and scope. Compose all selected fusion updates before final quantization. For example, when both MLP interfaces are conditioned, the final up weight is $D_{\mathrm{mlp}} W_{\mathrm{up}} D_4^{-1}$, not either one-sided update alone. Refresh quantizer statistics on the composed graph and quantize its final weights; intermediate candidate low-bit factors are not reused if another fusion changes their weights. The fixed-input group objective is a search surrogate, not a guarantee for the final model after both producer and consumer weights are quantized. Activation quantization remains disabled during search and is enabled at its prescribed locations for W4A4/W4A8 evaluation and deployment.

Under the interfaces and sharing constraints above, all additional diagonal conditioners are absorbed, so they require no standalone online scaling operator. The online R_4 path remains, while post-RoPE R_3 is explicitly disabled in this FP16-KV realization; FP16 caching alone does not require that choice. These rules do not imply that arbitrary group-wise rotations can be fused, that all architectures admit the same absorption, or that wall-clock latency is identical to SpinQuant. Reported throughput is specific to the configurations and timing protocol in Appendix C.5.

C.5 Calibration Cost Analysis

HeRo-Q adds a one-time search and rotation-optimization stage. Table 16 reports its wall-clock calibration time, peak memory, and inference throughput on a single NVIDIA A800 (80GB), alongside the corresponding baselines.

Inference benchmark protocol. The inference throughput benchmarks use a PyTorch/CUDA backend with actual W4A16 kernels, batch size 4, 128 input tokens per sequence, and 2048 generated tokens per sequence. We time decoding separately from prefill and exclude prefill from the reported decoding time. We perform three untimed warm-up runs and synchronize the GPU immediately before and after the timed decoding region. The measured time includes all remaining online transformations.

(a) Offline calibration cost

Model	Parameters (Total / Active)	Calibration Time (min) GPTQ / OmniQuant / HeRo-Q	Peak Memory (GB) GPTQ / OmniQuant / HeRo-Q
Llama-3.2-1B	1B / 1B	5 / 15 / 8	6 / 12 / 10
Llama-3.2-3B	3B / 3B	12 / 35 / 18	10 / 18 / 16
Qwen2.5-3B	3B / 3B	12 / 35 / 18	10 / 18 / 16
Qwen2.5-7B	7B / 7B	22 / 65 / 33	14 / 26 / 20
Llama-3.1-8B	8B / 8B	25 / 74 / 38	16 / 30 / 24
Qwen1.5-MoE-A2.7B	14.3B / 2.7B	30 / 85 / 45	20 / 32 / 28
DeepSeek-MoE-16B	16.4B / ∼2.8B	35 / 95 / 52	22 / 35 / 30
Mixtral-8×7B	46.7B / ∼13B	100 / 310 / 150	28 / 45 / 38

(b) Inference throughput

Model	Throughput (tok/s) GPTQ / OmniQuant	Throughput (tok/s) SpinQuant	Throughput (tok/s) HeRo-Q
Llama-3.2-1B	310.5	302.8	303.1
Llama-3.2-3B	215.6	209.5	210.0
Qwen2.5-3B	215.6	209.5	210.0
Qwen2.5-7B	138.0	133.5	134.2
Llama-3.1-8B	115.2	112.8	113.1
Qwen1.5-MoE-A2.7B	195.0	187.0	188.5
DeepSeek-MoE-16B	175.0	168.0	169.5
Mixtral-8×7B	65.0	62.0	62.5

Table 16: Offline calibration cost and inference throughput across dense and MoE models. Measurements were conducted on a single NVIDIA A800 (80GB) GPU. DeepSeek-MoE-16B has approximately 16.4B total and 2.8B activated parameters, following the original report (Dai et al., 2024). Qwen1.5-MoE-A2.7B has approximately 14.3B total and 2.7B activated parameters, following the official model description (Qwen Team, 2024).

Across the eight settings, HeRo-Q uses approximately $1.51\times$ GPTQ's calibration time and 52% of OmniQuant's, with lower peak memory than OmniQuant. For Mixtral-8×7B, calibration takes 150 minutes and peaks at 38 GB. The reported inference throughput is at least 96% of the shared GPTQ/OmniQuant kernel reference in these configurations; these aggregate timings do not isolate the contribution of fusion.

D Novelty Statement and Controlled Comparisons

D.1 Conditioning Rule and Relation to Prior Methods

Specific contribution. HeRo-Q combines a tunable Hessian-proxy conditioning rule, guided by a local quantization-loss bound, with rotation optimization in the conditioned coordinates. For each candidate conditioning strength, the conditioner is fixed while the rotation is optimized within the deployment-supported family; group-wise finite-bit reconstruction then selects the configuration. The contribution is this construction and selection procedure, rather than scaling, rotation, or Hessian information individually. Spectral and alignment analyses explain and test its effect. SpinQuant-style rotation deployment is reused infrastructure, not a separate contribution (Liu et al., 2025).

Effective curvature as a design target. Under the local second-order convention in Appendix A, Theorem 1 gives

$$\Delta\mathcal{L} \le \frac{1}{2}\lambda_{\max}(\mathbf{H}_q)\|\boldsymbol{\varepsilon}\|_2^2.$$

HeRo-Q uses this bound to guide the coordinates in which quantization is performed. It does not change the original Hessian $\mathbf{H}$; it constructs an effective Hessian through

$$\begin{aligned} \mathbf{T} &= \mathbf{D}_\alpha^{-1}\mathbf{R}, \qquad & \mathbf{R}^\top\mathbf{R} &= \mathbf{I}, \\ \mathbf{H}_q &= \mathbf{R}^\top\mathbf{D}_\alpha^{-1}\mathbf{H}\mathbf{D}_\alpha^{-1}\mathbf{R}. \end{aligned} \tag{43}$$

The diagonal entries of $\mathbf{D}_\alpha$ are obtained from the diagonal statistics of $\mathbf{H}_{\text{proxy}} = \mathbf{X}^\top\mathbf{X}$, with α controlling the conditioning strength. Under the normalized diagonal-proxy assumptions of Theorem 2, this construction strictly reduces the largest effective proxy eigenvalue for $\alpha \in (0, 1]$. The practical procedure selects the strength by reconstruction error rather than minimizing the spectral factor alone.

Residual optimization after conditioning. For a fixed $\mathbf{D}_\alpha$, orthogonal similarity preserves the spectrum of $\mathbf{A}_\alpha = \mathbf{D}_\alpha^{-1}\mathbf{H}\mathbf{D}_\alpha^{-1}$. Rotation therefore does not further reduce that fixed spectral factor. It instead changes the input to the finite-bit quantizer and hence the realized residual:

$$\boldsymbol{\varepsilon}(\mathbf{R}) = Q_b\big(\mathbf{R}^\top\mathbf{D}_\alpha\mathbf{w}\big) - \mathbf{R}^\top\mathbf{D}_\alpha\mathbf{w} \tag{44}$$

Corollary A.1 expresses the quadratic surrogate as

$$\begin{aligned} \Delta\mathcal{L} &= \frac{1}{2}\boldsymbol{\varepsilon}^\top\mathbf{H}_q\boldsymbol{\varepsilon} \\ &= \frac{1}{2}\|\boldsymbol{\varepsilon}\|_2^2\lambda_{\max}(\mathbf{H}_q)a(\boldsymbol{\varepsilon}, \mathbf{H}_q), \end{aligned} \tag{45}$$

where $a(\boldsymbol{\varepsilon}, \mathbf{H}_q) \in [0, 1]$ measures normalized residual–curvature alignment. The conditioner targets effective-curvature amplification, while rotation optimization can change residual magnitude and alignment without changing the conditioned spectrum. These are design roles, not independent effects: the conditioner also changes the quantizer input. The group-wise reconstruction objective accounts for the realized residual when selecting $(\alpha, \mathbf{R})$; alignment is a diagnostic rather than a separately optimized loss term.

Comparison of construction rules. Table 17 compares the scaling rules, rotation or quantization strategies, and specific distinctions from HeRo-Q. The comparison focuses on how transforms are constructed and selected, not merely on which methods contain scaling, rotation, or Hessian information.

Boundary with QuIP-style methods. QuIP's diagonal scaling balances weight and Hessian scales (Chee et al., 2023). HeRo-Q instead uses effective-curvature amplification analysis to guide the construction of a tunable conditioning matrix. Through rotation optimization after conditioning and transform selection using group-wise finite-bit reconstruction, HeRo-Q addresses effective-curvature amplification, the magnitude of the realized quantization residual, and its alignment with high-curvature directions. QuIP# and QTIP further develop incoherence processing and lattice or trellis coding (Tseng et al., 2024a;b); HeRo-Q does not propose a new codebook or coding structure.

Method	Scaling / conditioning rule	Rotation or quantization strategy	Specific distinction from HeRo-Q
SmoothQuant	Channel scaling from activation and weight magnitudes	Balances weight and activation quantization difficulty	HeRo-Q derives tunable conditioning from Hessian diagonal statistics and combines it with rotation optimization.
QuaRot	Orthogonal transformations as the core mechanism	Fixed structured rotations improve quantization representations	HeRo-Q adds non-orthogonal curvature conditioning and optimizes rotations where deployment permits learning.
SpinQuant	Rotation learning as the core mechanism	Optimizes orthogonal rotations for quantized-model performance	HeRo-Q optimizes rotation with each candidate conditioner fixed and selects the conditioning strength.
FlatQuant	Learns general invertible transformations	Improves quantization representations through local reconstruction	HeRo-Q uses a parameterized conditioning rule determined by Hessian diagonal statistics.
OSTQuant	Combines scaling and orthogonal transformations	Improves distribution fitting and quantization-space utilization	HeRo-Q uses effective-curvature analysis to guide conditioning and group-wise finite-bit reconstruction to select transforms.
QuIP	Non-orthogonal diagonal scaling from Hessian and weight statistics	Random orthogonal incoherence processing and Hessian-aware rounding	HeRo-Q uses tunable Hessian-diagonal conditioning, strength search, and rotation optimization after conditioning.
QuIP#	Improved incoherence-processing pipeline	Randomized Hadamard transforms and lattice codebooks	HeRo-Q focuses on conditioning construction and transform selection, rather than a new lattice codebook.
QTIP	Combines incoherence processing with coding	Trellis quantization balances accuracy and execution efficiency	HeRo-Q focuses on conditioning and residual optimization, rather than a new coding structure.
HeRo-Q (ours)	**Tunable $\mathbf{D}_\alpha$ from Hessian-proxy diagonal statistics**	Fixes each candidate $\mathbf{D}_\alpha$ before optimizing rotation; selects both through group-wise finite-bit reconstruction	**Conditioning rule coupled with rotation optimization**; spectral and alignment analyses explain and test the mechanism.

Table 17: Comparison of conditioning rules and quantization strategies. HeRo-Q is distinguished by its tunable Hessian-proxy conditioning rule and rotation optimization with each candidate conditioner fixed, rather than by the presence of scaling or rotation alone.

Hessian incoherence versus residual–curvature alignment. These quantities concern different geometric objects. QuIP calls a symmetric matrix $\mathbf{H} \in \mathbb{R}^{d\times d}$ μ-incoherent if it admits an orthogonal eigendecomposition $\mathbf{H} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^\top$ with $|Q_{ij}| \le \mu/\sqrt{d}$ (Chee et al., 2023). For a fixed eigenbasis $\mathbf{Q}$, the corresponding coefficient is

$$\mu_{\mathbf{Q}}(\mathbf{H}) = \sqrt{d}\max_{i,j}|Q_{ij}|. \tag{46}$$

It describes *Hessian eigenvectors relative to coordinate axes*, without specifying a particular quantization residual. The eigenbasis is stated explicitly because it need not be unique for repeated eigenvalues.

In contrast, the alignment in Corollary A.1 describes the *realized finite-bit residual relative to curvature eigendirections*. Let $\mathbf{H}_q = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top \succeq \mathbf{0}$, with $\lambda_1 = \lambda_{\max}(\mathbf{H}_q) > 0$, and let $\boldsymbol{\varepsilon} \neq \mathbf{0}$. Writing $\mathbf{c} = \mathbf{U}^\top\boldsymbol{\varepsilon}/\|\boldsymbol{\varepsilon}\|_2$ gives

$$a(\boldsymbol{\varepsilon}, \mathbf{H}_q) = \frac{\boldsymbol{\varepsilon}^\top\mathbf{H}_q\boldsymbol{\varepsilon}}{\lambda_1\|\boldsymbol{\varepsilon}\|_2^2} = \sum_{i=1}^{d}\frac{\lambda_i}{\lambda_1}c_i^2, \qquad \sum_{i=1}^{d}c_i^2 = 1. \tag{47}$$

As in the corollary, we set $a = 0$ if the residual or the largest eigenvalue is zero. Thus a is the residual-energy-weighted average of normalized curvature: it approaches one when the residual is concentrated in the top eigenspace, and is small when the residual lies mainly in directions with $\lambda_i \ll \lambda_1$. For fixed residual norm and largest eigenvalue, it measures the fraction of the worst-case quadratic cost that is realized; a alone does not determine the absolute loss.

The distinction is explicit even for a fixed anisotropic $\mathbf{H}_q$: its incoherence coefficient stays unchanged, whereas residuals along its top and bottom eigenvectors yield $a = 1$ and $a = \lambda_d/\lambda_1$, respectively. The two metrics are therefore not interchangeable. This does not imply that QuIP ignores curvature-weighted residuals: its rounding procedure already minimizes a quadratic proxy objective (Chee et al., 2023). Here, a diagnoses the residual produced by the selected finite-bit transform; it is not a new definition of incoherence.

D.2 Conditioning Control with Matched Rotation Optimization

Control design. To compare the conditioning recipes without changing the rotation-optimization procedure or the weight quantizer, we evaluate QuIP scaling with the same rotation-optimization procedure as HeRo-Q and use GPTQ for both methods. All three rows use Llama-3.1-8B under W3A16. HeRo-Q retains its tunable Hessian-diagonal conditioner and reconstruction-based strength selection. The original QuIP pipeline is included as a reference, with its own scaling, random rotation, and quantization procedure. Thus, the conditioning comparison is between the two GPTQ rows in Table 18, not between either row and the complete QuIP reference pipeline.

Method	Diagonal scaling	Rotation	Weight quantizer	W2↓	C4↓	MMLU↑	GSM8K↑
QuIP (reference)	QuIP scaling	QuIP random rotation	QuIP pipeline	8.50	14.40	63.20	69.00
QuIP scaling + matched rotation optimization	QuIP scaling	Same optimization procedure as HeRo-Q	GPTQ	8.40	14.33	63.02	68.29
HeRo-Q	Hessian-diagonal conditioning $\mathbf{D}_\alpha$	Rotation optimization after conditioning	GPTQ	**8.02**	**13.98**	**63.80**	**70.15**

Table 18: Conditioning control on Llama-3.1-8B under W3A16, with a shared rotation-optimization procedure and GPTQ weight quantization in the bottom two rows. QuIP is reported as a reference using its original pipeline. W2 denotes WikiText-2 perplexity; C4 is also reported as perplexity, and MMLU and GSM8K as accuracy (%).

Results. Relative to QuIP scaling with matched rotation optimization, HeRo-Q reduces WikiText-2 perplexity from 8.40 to 8.02 and C4 perplexity from 14.33 to 13.98, while increasing MMLU from 63.02 to 63.80 and GSM8K from 68.29 to 70.15. This comparison supports the benefit of HeRo-Q's conditioning recipe under the shared rotation-optimization and GPTQ pipeline. It directly addresses whether replacing HeRo-Q conditioning with QuIP scaling reproduces the reported performance.

The QuIP-scaling control improves perplexity over the original QuIP reference but not MMLU or GSM8K, so it should not be interpreted as a uniformly stronger version of QuIP.

D.3 Spectral, Residual, and Component Analyses

The controlled comparisons test the practical value of the construction. We separately examine the quantities used to motivate it: effective proxy curvature, realized residual magnitude, and residual–curvature alignment. Figure 3 reports the effective proxy spectrum on Llama-3.2-3B. The alignment analysis in Table 19 uses Llama-3.1-8B under W4A16 and therefore provides a separate diagnostic rather than a measurement on the same model.

Metric definitions. We measure quantization residuals in the transformed coordinates, using $\boldsymbol{\varepsilon}$ and the effective proxy Hessian $\mathbf{H}_{q,\mathrm{proxy}} = \mathbf{T}^\top \mathbf{H}_{\mathrm{proxy}} \mathbf{T}$, where $\mathbf{H}_{\mathrm{proxy}} = \mathbf{X}^\top \mathbf{X}$ retains the full Gram matrix. For each quantized linear layer ℓ, we compute the residual norm n_ℓ, the curvature-weighted error c_ℓ, and the alignment score a_ℓ as

$$n_\ell^2 = \sum_j \|\boldsymbol{\varepsilon}_{\ell,j}\|_2^2, \qquad c_\ell = \sum_j \boldsymbol{\varepsilon}_{\ell,j}^\top \mathbf{H}_{q,\mathrm{proxy},\ell} \boldsymbol{\varepsilon}_{\ell,j},$$

$$a_\ell = \frac{c_\ell}{\lambda_{\max}(\mathbf{H}_{q,\mathrm{proxy},\ell})\, n_\ell^2},$$

where j indexes output channels. Following the theoretical definition, $a_\ell = 0$ when the denominator is zero. The two relative metrics are obtained by normalizing each layer's n_ℓ and c_ℓ by the corresponding GPTQ values and then taking an unweighted mean across layers. Alignment is the unweighted mean of a_ℓ, without normalization to GPTQ.

Method	Rel. residual norm	Rel. curvature-weighted error	Align.	W2 PPL
GPTQ	1.00	1.00	0.36	7.02
SmoothQuant+SpinQuant	0.88	0.46	0.21	6.75
HeRo-Q	**0.81**	**0.29**	**0.13**	**6.33**

Table 19: Curvature-alignment analysis on Llama-3.1-8B under W4A16. Metrics are computed in the transformed coordinates using the full Gram proxy. Relative errors are normalized to GPTQ within each layer before equal-weight averaging; alignment is averaged without GPTQ normalization. Alignment measures residual–curvature alignment, not Hessian incoherence. Lower is better for all reported metrics.

Residual and alignment results. Relative to SmoothQuant+SpinQuant, HeRo-Q reduces the relative residual norm from 0.88 to 0.81, the relative proxy-curvature-weighted error from 0.46 to 0.29, and the alignment score from 0.21 to 0.13. Its corresponding WikiText-2 perplexity is 6.33, compared with 7.02 for GPTQ and 6.75 for SmoothQuant+SpinQuant. The larger proportional reductions in the curvature-weighted and alignment metrics are consistent with the motivation that Euclidean residual magnitude alone does not characterize quantization sensitivity. These are measured outcomes of the selected transform, not guarantees that every rotation improves alignment.

Component ablations. Table 5 evaluates conditioning and rotation separately and together. Conditioning alone improves over GPTQ on both Llama-3.2-1B and Llama-3.1-8B, although its gain is smaller on the latter (7.02 to 6.94). Rotation alone also improves both models. The combined method further reduces perplexity and outperforms the direct SmoothQuant+SpinQuant composition. These results support the full procedure in the evaluated W4A16 settings, while showing that the standalone benefit of conditioning varies across models.

D.4 Comparison and Compatibility with QuIP-Style Methods

Comparison with complete methods. We compare QuIP (Chee et al., 2023), QuIP# (Tseng et al., 2024a), QTIP (Tseng et al., 2024b), and HeRo-Q on Llama-3.2-1B under W3A16. All methods use the

same calibration data, token budget, downstream evaluation protocol, and bit-width setting. Table 20 shows that HeRo-Q achieves the lowest perplexity and highest MMLU and GSM8K accuracy among these methods. This complete-method comparison evaluates overall performance; the conditioning control in Table 18 separately compares the conditioning recipes within a shared rotation-optimization and GPTQ pipeline.

Method	Model	Setting	W2↓	C4↓	MMLU↑	GSM8K↑
QuIP	Llama-3.2-1B	W3A16	15.50	27.00	41.50	26.00
QuIP#	Llama-3.2-1B	W3A16	15.13	26.77	41.87	27.78
QTIP	Llama-3.2-1B	W3A16	14.65	26.09	42.02	28.79
HeRo-Q	**Llama-3.2-1B**	**W3A16**	**14.05**	**25.80**	**42.50**	**29.10**

Table 20: Matched comparison with QuIP-style methods. All methods use the same Llama-3.2-1B W3A16 setting and evaluation protocol.

Compatibility with QuIP. We also evaluate the insertion of the HeRo-Q transformation before the QuIP quantization pipeline under the same evaluation protocol. Table 21 shows that HeRo-Q+QuIP improves over both standalone methods in every reported setting. The results support practical compatibility and additional performance gains from combining the methods. They do not require assigning QuIP exclusively to residual optimization or HeRo-Q exclusively to curvature control: the distinction remains the conditioning rule and how the transform is selected.

Model	Setting	QuIP				HeRo-Q				HeRo-Q + QuIP			
		W2↓	C4↓	MMLU↑	GSM↑	W2↓	C4↓	MMLU↑	GSM↑	W2↓	C4↓	MMLU↑	GSM↑
Llama-3.1-8B	W3A16	8.50	14.40	63.20	69.00	8.02	13.98	63.80	70.15	**7.85**	**13.72**	**64.65**	**71.40**
Llama-3.2-1B	W3A16	15.50	27.00	41.50	26.00	14.05	25.80	42.50	29.10	**13.58**	**25.15**	**43.35**	**30.65**
Llama-3.1-8B	W4A4	10.80	15.50	44.80	41.50	10.29	14.84	45.31	42.33	**9.92**	**14.38**	**46.25**	**43.85**
Llama-3.2-3B	W4A4	14.00	22.50	39.50	36.00	13.18	20.14	41.10	37.15	**12.65**	**19.55**	**42.15**	**38.60**
Llama-3.1-8B	W4A8	7.15	13.05	65.80	74.80	6.96	12.94	66.31	75.26	**6.82**	**12.75**	**66.85**	**76.15**
Qwen2.5-7B	W4A8	6.85	13.50	70.70	85.80	6.78	13.43	71.02	86.12	**6.69**	**13.31**	**71.32**	**87.05**

Table 21: Compatibility between HeRo-Q and QuIP under the same evaluation protocol. The combined method improves perplexity and downstream accuracy over both standalone methods in the reported settings.

D.5 Scope and Limitations

Theoretical scope. The loss analysis uses a local second-order approximation, and the strict spectral-compression result applies under the normalized diagonal-proxy assumptions of Theorem 2. The spectral and alignment measurements test the practical mechanism, while downstream performance is established by experiments rather than guaranteed by the local theorem. Under very aggressive quantization, clipping, saturation, and nonlinear changes can make the quadratic approximation less representative of task loss.

Calibration dependence. HeRo-Q constructs its conditioner from the activation-induced proxy $\mathbf{H}_{\text{proxy}} = \mathbf{X}^\top \mathbf{X}$. This proxy need not match the true end-task Hessian, and its statistics depend on the calibration data. A substantial shift between calibration and deployment distributions can therefore reduce the benefit of the selected conditioning and rotation.

Calibration and deployment cost. Searching conditioning strengths and optimizing or selecting rotations adds offline time and memory relative to simpler PTQ methods. Although performed once without end-to-end model fine-tuning, this cost matters for large models or frequent re-quantization. Deployment also constrains the admissible rotation family; practical efficiency must account for any transformations that cannot be absorbed into existing operators.

Evaluation coverage. The experiments span dense language, MoE, vision, and multimodal models, but do not exhaust architectures, domains, context lengths, calibration budgets, or hardware backends.

Broader distribution-shift evaluations and repeated runs with different seeds or calibration subsets remain useful directions.