SCHURQUANT: GROUPWISE DISCRETE OPTIMIZATION FOR LAYER-WISE LLM QUANTIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Weight-only post-training quantization (PTQ) enables the deployment of large language models under tight memory budgets, but accuracy often collapses at 2–3 bits. Existing backpropagation-free PTQ optimizers have two limitations: group decisions ignore the correction that the remaining continuous suffix can absorb, and discrete refinements typically keep the affine quantization grid fixed. We introduce SCHUROPT, which analytically eliminates the suffix's optimal continuous response, yielding an exact groupwise quadratic with Schur-complement curvature. It then alternates closed-form row-wise scale/zero-point refitting with coordinate descent over integer codes. With the GPTQ objective fixed, SCHUROPT improves mean zero-shot accuracy on 2-bit Qwen3-4B by 11.88 percentage points (pp). At higher precision, however, tighter reconstruction does not consistently improve end-model metrics. SCHURQUANT therefore combines SCHUROPT with quantized-prefix teacher reconstruction, reference-weight regularization, residual-add targets, and teacher-decision token weighting. Across eight Llama and Qwen models, SCHURQUANT achieves the highest mean zero-shot accuracy among the evaluated backpropagation-free PTQ baselines, outperforming the strongest baseline by 9.65 pp at 2 bits.

1 INTRODUCTION

Deploying large language models (LLMs) is increasingly constrained by memory capacity and bandwidth rather than arithmetic throughput (Kimi Team et al., 2026), and these constraints are most severe on edge devices. Weight-only post-training quantization (PTQ) addresses this by converting pretrained weights to low precision using only a small calibration set without retraining. Yet conventional weight-only PTQ collapses sharply in perplexity and downstream accuracy in 2–3-bit quantization.

Weight-only PTQ can be decomposed into two largely independent components. The *objective* determines which error is minimized, classically the output-reconstruction loss $\|WX - W^{\mathrm{ref}}X\|_F^2$ of a linear layer, where W^{ref} is the teacher weight, X contains calibration activations, and W is the quantized weight. These activations determine the curvature $XX^\top$ and thus which weight-error directions matter (Appendix A). Recent methods enrich this objective with activation errors accumulated in preceding layers, asymmetric calibration, joint correction of past error and future propagation, residual-path reconstruction, or combinations of relaxed block objectives (Arai & Ichikawa, 2025; Li et al., 2025; Zhang et al., 2026; Lin & Wan, 2026; Ichikawa et al., 2025).

The *optimizer* determines how that objective is solved on an integer grid. GPTQ visits weights along the input-column axis, rounds each to a nearby level, and propagates the induced error to the remaining full-precision weights through an inverse Hessian (Frantar et al., 2023). This update is equivalent to Babai's nearest-plane algorithm on the lattice defined by $XX^\top$ (Chen et al., 2026). More recent methods use cyclic or greedy coordinate descent to refine discrete weights or groups (Behdin et al., 2023; Nair & Suggala, 2024; Ichikawa et al., 2025). GuidedQuant additionally injects end-loss gradients into the layer-wise objective and derives a monotone scalar quantizer (Kim et al., 2025).

The missing step is to condition a group decision on variables that remain adjustable. Prior coordinate descent evaluates the group under the raw block G_{cc} of a symmetric curvature matrix G, effectively

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.

holding the continuous suffix fixed, although that suffix can absorb part of the error. The exact curvature remaining after its best continuous response is the Schur complement $S = G_{cc} - G_{cr}G_{rr}^{-1}G_{rc}$, which can differ substantially from G_{cc} under strong coupling.

Based on this observation, we introduce SCHUROPT, the Schur-conditioned discrete optimizer, and refer to its pairing with our final augmented surrogate as SCHURQUANT. We partition the input dimension into chunks of one quantization group. While quantizing a current chunk W_c, the already quantized prefix W_F is held fixed and the suffix W_r remains continuous. Analytically minimizing over W_r produces S and a corresponding conditional linear term. SCHUROPT then optimizes the group against this reduced quadratic: it jointly refits each row's scale and zero-point for fixed codes and updates codes by coordinate descent for fixed grid parameters.

Holding the objective exactly equal to GPTQ's isolates the optimizer's contribution. On Qwen3-4B with 2-bit codes, SCHUROPT reduces WikiText-2 perplexity from 258.96 to 81.95 and improves mean zero-shot accuracy by 11.88 pp. At 3 and 4 bits, however, the same optimizer worsens perplexity while changing mean zero-shot accuracy only marginally. This reveals a second bottleneck. Layer-wise reconstruction is a loose surrogate for the final language-model loss. Motivated by this mismatch, our final configuration combines quantized-prefix teacher reconstruction with a reference-weight anchor, a residual-add target, and teacher-decision token weighting. The last term upweights the calibration positions whose clean top-1 prediction has already been changed by the quantized prefix.

Our contributions are as follows:

- We derive the exact discrete group problem conditioned on a fixed prefix and the optimal continuous suffix response, then optimize its (S, T) with a joint closed-form scale/zero-point refit and row-parallel coordinate descent. This conditioning, rather than coordinate descent itself, is the optimizer contribution.
- We integrate prior reference-weight regularization and residual-add reconstruction into the sufficient statistics and introduce teacher-decision token weighting, changing the surrogate without changing SCHUROPT.
- We separate optimizer and objective effects experimentally. Across eight Llama and Qwen models under 2–4-bit quantization, SCHURQUANT achieves the highest mean zero-shot accuracy among the evaluated backpropagation-free PTQ baselines, with its largest advantage at 2 bits.

2 BACKGROUND AND MOTIVATION

2.1 UNIFORM AFFINE QUANTIZATION

A uniform affine quantizer maps a real weight w to an integer code z as

$$z = \mathrm{clip}\Big(\Big\lfloor \frac{w}{a} \Big\rceil + o, q_{\min}, q_{\max}\Big), \qquad \widehat{w} = a(z - o), \tag{1}$$

where $a > 0$ is the scale, o the zero-point, and $\lfloor \cdot \rceil$ nearest-integer rounding. With unsigned b-bit codes, $q_{\min} = 0$ and $q_{\max} = 2^b - 1$. The symmetric case fixes $o = 0$ and uses signed codes $q_{\min} = -2^{b-1}$ and $q_{\max} = 2^{b-1} - 1$. The representable set is $\mathcal{Q}(a, o) = \{a\,(\ell - o) : \ell \in \{q_{\min}, \ldots, q_{\max}\}\}$.

Per-row group quantization assigns a quantizer to each group of g input columns in a row. We call the $d_{\text{out}} \times g$ submatrix processed at one step a *chunk*. An asymmetric quantizer stores an FP16 scale and a b-bit zero-point for each group. With FP16 metadata, this gives an effective weight precision of approximately $b + 16/g$ bits. All experiments use $g = 128$, the standard weight-only PTQ setting introduced with GPTQ, which limits metadata overhead to approximately 0.125 bits per weight.

2.2 GPTQ, PROPAGATED-ERROR OBJECTIVES, AND DISCRETE REFINEMENT

GPTQ minimizes the output-reconstruction objective $\min_{W \in \mathcal{Q}^{d_{\text{out}} \times d_{\text{in}}}} \|W^{\text{ref}}X - WX\|_F^2$ of each linear layer (Frantar et al., 2023). It initializes a uniform grid for each row–group, visits columns sequentially, and rounds the current weights. To reduce the output effect of the rounding error, it updates the remaining full-precision weights using the inverse input Hessian $H = XX^\top$

$$\delta_{>q} = -\frac{w_q - \mathrm{quant}(w_q)}{[H^{-1}]_{qq}} [H^{-1}]_{q,>q}. \tag{2}$$

Here $> q$ denotes later input columns, and all output rows are processed in parallel. The discrete decision remains nearest-level rounding; curvature enters through the subsequent correction, as its equivalence to Babai's nearest-plane algorithm makes explicit (Chen et al., 2026).

GPTQ is operation-local, not accounting for activation error produced by already quantized operations. Let X_ℓ^{ref} and X_ℓ denote the inputs to operation ℓ produced by the clean and already quantized prefixes, respectively. QEP (Arai & Ichikawa, 2025) optimizes the current quantized weight on X_ℓ to reproduce the clean teacher output $W_\ell^{\mathrm{ref}} X_\ell^{\mathrm{ref}}$:

$$\min_{W_\ell \in \mathcal{Q}^{d_{\mathrm{out}} \times d_{\mathrm{in}}}} \left\| W_\ell^{\mathrm{ref}} X_\ell^{\mathrm{ref}} - W_\ell X_\ell \right\|_F^2, \qquad \delta_\ell := X_\ell^{\mathrm{ref}} - X_\ell. \tag{3}$$

The optimum of its continuous relaxation is a corrected teacher weight, after which QEP uses the same GPTQ-style discrete optimizer. FOEM retains the first-order term arising away from a stationary pretrained weight (Zheng et al., 2026). LoaQ further extends the reconstruction scope from linear-layer outputs to sub-layer outputs by explicitly matching the residual-added hidden state, thereby accounting for error accumulated along the residual stream while retaining the layer-wise PTQ formulation (Lin & Wan, 2026). These methods, and work on asymmetric calibration or error propagation (Li et al., 2025; Zhang et al., 2026), primarily change the objective or reconstruction scope rather than how a group is conditioned on its continuous suffix.

QuantEase uses cyclic coordinate descent with a non-increasing reconstruction objective (Behdin et al., 2023); CDQuant uses greedy coordinate descent and extends it to group quantization (Nair & Suggala, 2024). To our knowledge, neither first eliminates the continuous suffix's optimal response. SCHUROPT applies parameter refitting and code updates to the resulting Schur-reduced quadratic.

2.3 SCHUR COMPLEMENT

The Schur complement (Schur, 1918) gives the curvature remaining after an optimal response. Partition a variable into current and adjustable blocks and its symmetric curvature as

$$x = \begin{bmatrix} x_c \\ x_r \end{bmatrix}, \qquad G = \begin{bmatrix} G_{cc} & G_{cr} \\ G_{rc} & G_{rr} \end{bmatrix}, \qquad G_{rc} = G_{cr}^\top. \tag{4}$$

For the quadratic form induced by G, if G_{rr} is invertible then minimizing over x_r gives $x_r^\star = -G_{rr}^{-1} G_{rc} x_c$ and

$$\min_{x_r} f(x_c, x_r) = \tfrac{1}{2} x_c^\top \underbrace{\left(G_{cc} - G_{cr} G_{rr}^{-1} G_{rc}\right)}_{G/G_{rr}} x_c. \tag{5}$$

Thus $G/G_{rr} \preceq G_{cc}$ whenever $G \succeq 0$. In quantization, x_c represents the current group and x_r the continuous suffix.

2.4 RELATED QUANTIZATION APPROACHES

Quantization-aware optimization. Quantization-aware training (QAT) uses fake-quantized weights and straight-through gradients to update weights or quantizer parameters. EfficientQAT trains weights within each Transformer block and then optimizes quantization parameters end to end (Chen et al., 2025). Although accurate, such training requires repeated forward and backward passes and is substantially more expensive than gradient-free layer-wise PTQ.

Calibration-optimized PTQ. OmniQuant learns clipping parameters and equivalent transformations through blockwise reconstruction on calibration data (Shao et al., 2024), whereas AWQ identifies salient weight channels from activation statistics and protects them through per-channel scaling (Lin et al., 2024). These are more naturally classified as PTQ: they adapt the objective or representation from calibration data rather than retraining the model end to end. SCHURQUANT is complementary, as it changes the discrete optimizer applied after the layer-wise objective has been specified.

A second line changes the representation rather than the search. Rotation-based methods apply function-preserving basis changes to spread outliers across channels, using randomized Hadamard rotations (Ashkboos et al., 2024) or the rotations learned against a calibration loss (Liu et al., 2025). Vector-quantized methods map groups of weights to codewords, combining incoherence processing

with lattice codebooks (Chee et al., 2023; Tseng et al., 2024) or learning additive codebooks (Egiazarian et al., 2024). Both families change the inference path, requiring online rotations or packed-index decoding and lookup. In contrast, SCHURQUANT retains the same scalar per-row group format as GPTQ; only the offline optimizer changes.

3 SCHUROPT AND SCHURQUANT

We define a shared quadratic interface, derive SCHUROPT in Sections 3.2–3.4, and augment the objective to obtain SCHURQUANT in Section 3.5.

3.1 PROBLEM FORMULATION

Consider one linear projection with weights $W \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$, calibration inputs $X = [x_1, \ldots, x_N] \in \mathbb{R}^{d_{\text{in}} \times N}$, and teacher targets $Y = [y_1, \ldots, y_N] \in \mathbb{R}^{d_{\text{out}} \times N}$. Let $\Omega = \operatorname{diag}(\omega_1, \ldots, \omega_N)$ contain nonnegative token weights. Here W^{ref} is the pretrained weight and W the student weight, with $W \in \mathcal{Q}^{d_{\text{out}} \times d_{\text{in}}}$ after quantization. During a group update, W_F and W_c are discrete and W_r is the *continuous suffix*. Unless otherwise stated for GPTQ-objective experiments, X comes from the quantized prefix and Y from the clean teacher. We optimize

$$\mathcal{L}(W) = \frac{1}{2}\|(WX - Y)\Omega^{1/2}\|_F^2 + \frac{\lambda}{2}\|W - W^{\text{ref}}\|_F^2, \tag{6}$$

where the dimensionless λ anchors the quantized solution to the pretrained weights. Expanding Eq. 6 gives

$$G = X\Omega X^\top + \lambda I, \qquad C = Y\Omega X^\top + \lambda W^{\text{ref}}, \tag{7}$$

$$\mathcal{L}(W) = \tfrac{1}{2}\operatorname{tr}(WGW^\top) - \operatorname{tr}(CW^\top) + \text{const}. \tag{8}$$

Thus (G, C) are sufficient statistics: the optimizer need not retain X, Y, or Ω. If $\lambda > 0$, G is positive definite even when $X\Omega X^\top$ is rank deficient.

Eq. 8 is also the interface between SCHUROPT and a layer-wise objective. Setting $\Omega = I$, $\lambda = 0$, and $Y = W^{\text{ref}}X$ recovers GPTQ. Using quantized-prefix inputs with clean teacher outputs recovers the QEP-style asymmetric teacher-reconstruction objective. Residual-add targets and token weights only change how (G, C) are collected; the optimizer below is unchanged.

3.2 THE SCHUR-REDUCED GROUP OBJECTIVE

For any candidate W_c, the suffix can still react before it is quantized. We therefore eliminate its optimal continuous response,

$$W_c^{\star,\text{Schur}} = \arg\min_{W_c \in \mathcal{Q}} \mathcal{L}(W_c, W_r^\star(W_c)), \qquad W_r^\star(W_c) = \arg\min_{W_r} \mathcal{L}(W_c, W_r), \tag{9}$$

so that the selected group minimizes the loss remaining after the best possible continuous suffix correction, not merely the loss of the group in isolation.

Proposition 1 (Schur reduction)**.** *Fix W_F and assume that G_{rr} is invertible. Eliminating W_r yields the equivalent local objective*

$$\mathcal{L}_{\text{local}}(W_c) = \frac{1}{2}\operatorname{tr}(W_c S W_c^\top) - \operatorname{tr}(TW_c^\top) + \text{const}, \tag{10}$$

$$S = G_{cc} - G_{cr}G_{rr}^{-1}G_{rc}, \qquad T = C_c^{\text{eff}} - C_r^{\text{eff}}G_{rr}^{-1}G_{rc}, \tag{11}$$

$$C_c^{\text{eff}} = C_c - W_F G_{Fc}, \qquad C_r^{\text{eff}} = C_r - W_F G_{Fr}. \tag{12}$$

Appendix B.1 gives the proof. The suffix absorbs curvature $G_{cr}G_{rr}^{-1}G_{rc}$, so $S \preceq G_{cc}$; T accounts for prefix error and the suffix's response.

Computing S by factorizing every suffix G_{rr} would be prohibitively expensive. For a chunk starting at input column t, we maintain $P = (G_{[t:,t:]})^{-1}$ partitioned as $\left[\begin{smallmatrix} P_{cc} & P_{cr} \\ P_{cr}^\top & P_{rr} \end{smallmatrix}\right]$ with $P_{cc} \in \mathbb{R}^{g \times g}$. The block-inverse identity gives

$$S = P_{cc}^{-1}, \qquad G_{rr}^{-1}G_{rc} = -P_{cr}^\top S, \qquad P_{\text{next}} = P_{rr} - P_{cr}^\top S P_{cr}. \tag{13}$$

Each chunk therefore requires one $g \times g$ inverse and an $O(m^2 g)$ update for a suffix of length m, rather than a new $O(m^3)$ factorization.

3.3 EXACT ROW-WISE QUANTIZER-PARAMETER UPDATE

For output row i, write the quantized current group as $q_i = a_i(z_i - o_i\mathbf{1}^\top)$ with integer codes $z_i \in \{q_{\min}, \dots, q_{\max}\}^g$, scale a_i, and zero-bias o_i. A symmetric quantizer is the special case $o_i = 0$. Because Eq. 10 separates across output rows,

$$\mathcal{L}_{\text{local}}(Q) = \sum_{i=1}^{d_{\text{out}}} \left(\tfrac{1}{2} q_i S q_i^\top - t_i q_i^\top\right), \tag{14}$$

all rows can be updated in parallel.

Proposition 2 (Row-wise scale and zero-point update)**.** *For fixed z_i and a candidate o, define $u_i(o) = z_i - o\mathbf{1}^\top$, $A_i(o) = u_i(o) S u_i(o)^\top$, and $d_i(o) = t_i u_i(o)^\top$. The optimal positive scale and zero-point are*

$$a_i^\star(o) = \max\left(\frac{d_i(o)}{A_i(o)}, \epsilon_a\right), \qquad o_i^\star = \arg\min_{o \in \{0,\dots,2^b-1\}} \left[\frac{1}{2} a_i^\star(o)^2 A_i(o) - a_i^\star(o) d_i(o)\right]. \tag{15}$$

Only 2^b zero-points are considered, so the enumeration is inexpensive at the target precisions. The required coefficients are obtained from one product ZS, followed by row-wise reductions and evaluation of the 2^b candidates. Appendix B.2 gives the proof.

3.4 COORDINATE DESCENT OVER INTEGER CODES

Stack the row code vectors into Z, with scales $a \in \mathbb{R}_+^{d_{\text{out}}}$ and integer zero-points $o \in \{0, \dots, 2^b - 1\}^{d_{\text{out}}}$. For fixed (a, o), the current chunk is $Q = \operatorname{diag}(a)(Z - o\mathbf{1}^\top)$. At column j, row i can choose among the 2^b candidates $v_{i,k} = a_i(k - o_i)$. Retaining only terms that depend on v yields the coordinatewise objective, up to an additive constant,

$$\ell_{i,j}(v) = \frac{1}{2} S_{jj} v^2 + \left(\underbrace{(q_i S)_j - q_{i,j} S_{jj}}_{\text{cross}_{i,j}} - t_{i,j}\right) v, \qquad z_{i,j} \leftarrow \arg\min_k \ell_{i,j}(a_i(k - o_i)). \tag{16}$$

At the start of a sweep, we initialize $\Phi \leftarrow QS$. Maintaining this state makes $\text{cross}_{i,j} = \Phi_{i,j} - q_{i,j} S_{jj}$ immediately available. When the coordinate changes, define $\Delta_i = q_{i,j}^{\text{new}} - q_{i,j}^{\text{old}}$ and update the state by the rank-one operation $\Phi_{i,:} \leftarrow \Phi_{i,:} + \Delta_i S_{j,:}$. Evaluating all 2^b candidates requires $\mathcal{O}(2^b)$ time and the state update $\mathcal{O}(g)$ per coordinate, for a total sweep complexity of $\mathcal{O}(d_{\text{out}} g(2^b + g))$.

The cross term is the key difference from independent rounding. It captures coupling with other columns in the group, while S already accounts for correction available outside the group. GPTQ reduces the curvature to a scalar but still makes a nearest-level decision. SCHUROPT uses the reduced curvature both to fit the grid and to choose the integer codes themselves.

3.5 AUGMENTING THE LOCAL OBJECTIVE

Even an exact optimizer can overfit a layer-wise surrogate. We therefore augment its sufficient statistics with a reference-weight anchor, a residual-add target, and teacher-decision token weighting while leaving SCHUROPT unchanged.

Reference-weight anchor. Following QEP (Arai & Ichikawa, 2025), the regularizer in Eq. 6 contributes λI to G and λW^{ref} to C without changing the optimizer.

Residual-add target. Following LoaQ (Lin & Wan, 2026), we use the clean residual output $Y = h + f(h)$ rather than the projection-local output, with student-side statistics collected from the quantized prefix.

Teacher-decision token weighting. Before quantizing Transformer layer ℓ, we compare the logits $s_i^{Q,\ell}$ of the current quantized-prefix model with cached clean-teacher logits s_i^T. We define

$$\tau_i = \arg\max_v s_{i,v}^T, \qquad m_i^{(\ell)} = \mathbf{1}\left[\arg\max_v s_{i,v}^{Q,\ell} \neq \tau_i\right], \qquad \alpha_i^{(\ell)} = 1 + (\rho - 1) m_i^{(\ell)}. \tag{17}$$

Algorithm 1 SCHUROPT for a single linear layer

Input: Sufficient statistics (G, C), reference weights W^{ref}, bit width b, group size g, refinement count R, and damping coefficient ϵ_{damp}
Output: Quantized weights W

$P \leftarrow \left(G + \epsilon_{\mathrm{damp}}\ \mathrm{diag}(G) I\right)^{-1}$
Initialize the working matrix $W \leftarrow W^{\mathrm{ref}}$
for $t = 0, 1, \ldots, d_{\mathrm{in}}/g - 1$ **do**
  Partition $P = \begin{bmatrix} P_{cc} & P_{cr} \\ P_{cr}^\top & P_{rr} \end{bmatrix}$ according to the current chunk
  $S \leftarrow P_{cc}^{-1}$
  $K \leftarrow -P_{cr}^\top S$
  Compute C_c^{eff} and C_r^{eff}
  $T \leftarrow C_c^{\mathrm{eff}} - C_r^{\mathrm{eff}} K$
  Initialize (Z, a, o) by row-wise min–max quantization of W_c^{ref} to b bits
  for $1, 2, \ldots, R$ **do**
    Fit (a, o) by Eq. 15
    $Q \leftarrow \mathrm{diag}(a)(Z - o\mathbf{1}^\top)$
    Sweep all entries of Z using Eq. 16
  end for
  Refit (a, o); set $W_c \leftarrow \mathrm{diag}(a)(Z - o\mathbf{1}^\top)$
  $P \leftarrow P_{rr} - P_{cr}^\top S P_{cr}$
end for
return W

Thus a flipped top-1 prediction receives ρ times the mass of an unchanged prediction. We use $\rho = 8$ and normalize $\alpha_i^{(\ell)}$ to unit mean.

The mask is shared across projections in a Transformer layer. For each projection, we compute the detached residual $r_i = \|W^{\mathrm{ref}} x_i - y_i\|_2$ and use

$$\omega_i = \frac{\bar{\alpha}_i^{(\ell)}}{\max(r_i, \epsilon)} \tag{18}$$

in Ω. The resulting statistics remain $X\Omega X^\top$ and $Y\Omega X^\top$ in Eq. 7. Weights are recomputed before each Transformer layer while clean logits are cached once. No gradient update or change to the stored affine weight format is required.

3.6 COMPLETE ALGORITHM

Each chunk is initialized by row-wise min–max quantization of W_c^{ref}, then alternates the exact update of (a, o) with coordinate descent over Z for R refinement steps, followed by a final parameter refit because the last sweep changes Z. Each block update is non-increasing in $\mathcal{L}_{\mathrm{local}}$, and the finite code space implies convergence to a coordinate-wise fixed point. Unlike GPTQ, SCHUROPT needs no explicit suffix error-propagation step. The optimal response is already incorporated into (S, T), and the next chunk's effective linear term reflects the fixed prefix. If $g = 1$, (a, o) are held fixed, and refinement is restricted to a single nearest-level decision, the procedure reduces to GPTQ's column-wise update. Algorithm 1 states the full procedure.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Quantization. Unless stated otherwise, all weight-only experiments use group size $g = 128$, a row-wise asymmetric quantizer, and float16 computation. We follow the standard GPTQ calibration protocol and quantize the linear attention and MLP projections inside Transformer blocks. Full quantization and calibration settings are provided in Appendix D.

Models. We evaluate Llama2-7B/13B (Touvron et al., 2023), Llama3.2-1B/3B and Llama3-8B (Grattafiori et al., 2024), and Qwen3-0.6B/4B/8B (Yang et al., 2025). They span 0.6B–13B parameters and include distinct tokenizers and normalization designs.

Metrics. We evaluate perplexity (PPL) on the complete WikiText-2 test split (Merity et al., 2017) using windows of length 2,048:

$$\mathrm{PPL} = \exp\left(-\frac{1}{N}\sum_{i=1}^{N}\log p(x_i \mid x_{<i})\right). \tag{19}$$

We evaluate zero-shot accuracy on PIQA (Bisk et al., 2020), ARC-Easy and ARC-Challenge (Clark et al., 2018), HellaSwag (Zellers et al., 2019), WinoGrande (Sakaguchi et al., 2020), and BoolQ (Clark et al., 2019). We use length-normalized accuracy for ARC-Easy, ARC-Challenge, and HellaSwag and standard accuracy otherwise, and report the unweighted mean across the six tasks.

Baselines and environment. GPTQ, QEP, FOEM, and SCHURQUANT are evaluated within the same quantization codebase and calibration protocol. Experiments use Python 3.12, PyTorch 2.11.0, Transformers 5.14.1, GPTQModel 7.3.4, Triton 3.6.0, and CUDA runtime 13.0.

4.2 WIKITEXT-2 PERPLEXITY

Table 1: WikiText-2 test perplexity (lower is better). FP16 is the full-precision reference and SchQ denotes SCHURQUANT. For each model and bit-width, the best quantized result is bold.

		Llama2		Llama3.2		Llama3	Qwen3		
Bits	Method	7B	13B	1B	3B	8B	0.6B	4B	8B
–	FP16	5.46	4.88	9.76	7.82	6.14	20.96	13.64	9.72
2	SchQ	**10.54**	**8.28**	**79.19**	**29.39**	**36.69**	**74.72**	**32.40**	**29.48**
	GPTQ	37.63	16.69	3432.88	10549.22	486.51	12027.65	258.96	77.13
	FOEM	68.73	15.42	6034.37	4331.26	554.52	106866.61	580.99	165.42
	QEP	29.84	13.50	1248.04	378.70	221.76	659.24	80.61	57.47
3	SchQ	**6.07**	**5.33**	**13.25**	**9.39**	**7.55**	**26.90**	**15.48**	11.16
	GPTQ	6.32	5.42	15.41	12.07	8.31	50.53	15.79	**10.92**
	FOEM	6.38	5.43	68.22	253.99	15.72	39.16	15.85	11.66
	QEP	6.39	5.43	15.26	18.05	9.79	35.39	15.54	11.37
4	SchQ	**5.59**	4.99	**10.42**	**8.12**	**6.46**	**22.63**	14.22	10.02
	GPTQ	5.62	4.98	10.63	8.24	6.57	26.22	**14.16**	**9.91**
	FOEM	5.64	**4.97**	43.25	17.33	11.04	24.13	14.31	10.03
	QEP	5.71	4.98	10.70	8.51	6.64	23.77	14.40	10.17

Table 1 summarizes WikiText-2 perplexity across all models and bit widths. At 2 bits, SCHURQUANT improves substantially over GPTQ on all eight models. It prevents the extreme collapses observed for Llama3.2-1B (3432.88 to 79.19), Llama3.2-3B (10549.22 to 29.39), and Qwen3-0.6B (12027.65 to 74.72), and gives the lowest PPL among all methods on every model. At 3 bits, SCHURQUANT is best on seven of the eight models; only Qwen3-8B favors GPTQ (10.92 versus 11.16). At 4 bits, SCHURQUANT is best on five models, while FOEM is best on Llama2-13B and GPTQ is best on Qwen3-4B/8B. As Section 4.3 shows, this reversal is specific to perplexity.

4.3 ZERO-SHOT ACCURACY

Table 2 reports the mean zero-shot accuracy across the six downstream tasks for all models and bit widths. The results reveal three distinct regimes across quantization precisions. At 2 bits, SCHURQUANT is best on all eight models and averages 50.83%, exceeding QEP by 9.65 pp and GPTQ by 11.58 pp. Search error from greedy rounding is dominant on this coarse grid. At 3 bits, SCHURQUANT averages 65.11%, 2.16 pp above QEP, and wins on seven of eight models.

Table 2: Mean zero-shot accuracy (%; higher is better) across the six evaluated tasks. For each model and bit-width, the best quantized result is bold.

		Llama2		Llama3.2		Llama3	Qwen3			
Bits	Method	7B	13B	1B	3B	8B	0.6B	4B	8B	Mean
–	FP16	70.45	73.29	60.35	69.04	74.61	54.19	71.17	74.09	68.40
2	SchQ	**56.58**	**62.55**	**45.05**	**47.90**	**48.53**	**44.19**	**51.50**	**50.31**	**50.83**
	GPTQ	41.38	48.33	37.60	36.72	38.04	36.63	36.77	38.53	39.25
	FOEM	39.47	40.35	36.45	36.90	37.20	36.33	36.36	36.95	37.50
	QEP	43.73	50.14	37.07	41.11	37.73	37.50	40.31	41.81	41.18
3	SchQ	**68.13**	71.59	**55.49**	**65.98**	**72.20**	**49.82**	**65.79**	**71.89**	**65.11**
	GPTQ	67.03	**71.84**	53.94	63.05	66.15	44.43	65.53	69.56	62.69
	FOEM	67.04	71.08	52.07	60.54	66.72	46.04	62.82	68.92	61.90
	QEP	67.53	71.63	53.51	60.56	68.22	45.75	65.44	70.94	62.95
4	SchQ	69.54	**73.11**	**59.28**	68.04	**74.27**	**52.43**	70.19	73.43	**67.54**
	GPTQ	**70.07**	73.06	59.19	**68.11**	72.94	48.37	69.50	**73.60**	66.86
	FOEM	69.89	73.04	57.88	66.52	73.03	48.25	**70.39**	72.70	66.46
	QEP	69.93	73.09	58.56	67.81	73.27	49.76	69.00	73.47	66.86

On Llama3-8B, it exceeds GPTQ by 6.05 pp. Notably, the 3-bit Qwen3-8B perplexity reversal in Table 1 is flipped again in zero-shot accuracy. SCHURQUANT is 2.33 pp above GPTQ. At 4 bits, SCHURQUANT remains best on average at 67.54%, but its 0.68-point advantage over the strongest baselines is much smaller than at lower precision, consistent with the diminishing difference between nearest rounding and a better discrete assignment on a finer grid.

4.4 OPTIMIZER COMPONENT ABLATION

Table 3 holds the 2-bit current-input self-reconstruction objective, code descent, initialization, $g = 128$, $R = 16$, windows, and seed fixed on Qwen3-4B. "Raw" holds the suffix fixed, "Schur" eliminates its continuous response. "Fixed" retains the initial grid, "refit" applies Eq. 15.

Table 3: Controlled optimizer ablation on Qwen3-4B at 2 bits. $\Delta\mathcal{L}$ is the normalized layer-0 Q/K/V local-loss increase in percent.

Curvature	Grid	$\Delta\mathcal{L}$ (%) $\downarrow$	PPL $\downarrow$
Raw	fixed	1.645	2344.82
Raw	refit	0.872	152.95
Schur	fixed	1.165	324.81
Schur	refit	**0.550**	**86.64**

With the grid fixed, Schur conditioning reduces the controlled loss from 1.645% to 1.165% and PPL from 2344.82 to 324.81. With the same refit on both curvatures, it reduces the loss from 0.872% to 0.550% (36.9% relative) and PPL from 152.95 to 86.64. Thus, the gain is not attributable to coordinate descent or grid refitting alone.

4.5 OBJECTIVE ABLATION

Table 4 varies the optimizer and sufficient statistics on Qwen3-4B. Rows 1–2 share the GPTQ objective. The remaining rows use quantized-prefix inputs and clean teacher outputs; residual-add rows extend the target to the sublayer output. Weighted rows use $\rho = 8$ and Eq. 18, and $\lambda = 1$ denotes the reference-weight anchor.

With the GPTQ objective fixed, SCHUROPT changes 2-bit PPL from 258.96 to 81.95 and accuracy from 36.77% to 48.65% ($+11.88$ pp). At 3 and 4 bits, it worsens PPL while changing mean accuracy by -0.12 and $+0.08$ pp, respectively. For the projection-local teacher target, weighting improves

Table 4: Objective ablation on Qwen3-4B. Mean is six-task zero-shot accuracy (%). Teacher denotes the projection-local teacher target; Residual denotes the residual-add target. Weights indicates token weighting.

				2 bits		3 bits		4 bits	
Optimizer	Target	Weights	λ	PPL ↓	Mean ↑	PPL ↓	Mean ↑	PPL ↓	Mean ↑
GPTQ	GPTQ	–	–	258.96	36.77	15.79	65.53	**14.16**	69.50
SCHUROPT	GPTQ	–	–	81.95	48.65	21.26	65.41	15.52	69.58
SCHUROPT	Teacher	–	1	75.01	42.97	16.23	65.13	14.29	69.06
SCHUROPT	Teacher	Yes	0	64.81	43.62	18.92	63.77	15.69	69.02
SCHUROPT	Teacher	Yes	1	67.70	46.20	16.07	**66.83**	14.31	70.00
SCHUROPT	Residual	–	1	38.72	44.84	15.99	62.30	14.22	68.80
SCHUROPT	Residual	Yes	0	38.94	48.17	18.63	65.87	16.01	69.54
SCHURQUANT	Residual	Yes	1	**32.40**	**51.50**	**15.48**	65.79	14.22	**70.19**

mean accuracy at every precision, and adding the anchor on top of weighting gives a further $+2.58$, $+3.06$, and $+0.98$ pp at 2, 3, and 4 bits. With weights and $\lambda = 1$ fixed, the residual-add target lowers PPL at all three precisions and changes mean accuracy by $+5.30$, -1.04, and $+0.19$ pp, respectively. The full configuration therefore gives the best 2-bit mean, the best 2/3-bit PPL, and the best 4-bit mean; the GPTQ optimizer gives the best 4-bit PPL, while the projection-local weighted objective remains best in 3-bit mean accuracy.

4.6 CONVERGENCE WITH REFINEMENT STEPS

Table 5: WikiText-2 perplexity versus refinement count R for 2-bit Llama2-7B under the fixed GPTQ objective. $R = 0$ uses only min–max initialization.

R	0	1	2	4	8	16	32
PPL ↓	70.46	37.93	28.75	**20.54**	21.41	21.19	21.21

Table 5 reports WikiText-2 perplexity across different refinement counts R. Perplexity improves rapidly and saturates around $R = 4$. Variation over $R \geq 4$ is only 0.87, compared with a 49.92 improvement from $R = 0$ to $R = 4$. The local quadratic objective is monotone in R, whereas perplexity is not, again illustrating surrogate over-optimization. We use $R = 16$ elsewhere because it was stable across models and bit widths.

5 CONCLUSION

SCHUROPT analytically eliminates the optimal continuous suffix response and optimizes each group under the resulting Schur-reduced quadratic, including a closed-form row-wise scale/zero-point refit. With the GPTQ objective fixed on 2-bit Qwen3-4B, replacing GPTQ's optimizer reduces PPL from 258.96 to 81.95 and raises mean accuracy by 11.88 pp. The component ablation separately tests the conditional curvature. SCHURQUANT pairs this optimizer with quantized-prefix teacher reconstruction, reference-weight regularization, residual-add targets, and teacher-decision token weighting, reaching mean accuracy of 50.83%, 65.11%, and 67.54% at 2, 3, and 4 bits across eight models.

6 LIMITATIONS

Our method has two main limitations. First, SCHURQUANT incurs higher offline quantization cost than existing PTQ methods due to repeated calibration and discrete refinement, limiting its scalability to larger models. Second, optimizing the layer-wise surrogate does not always translate to improved end-task performance, and our evaluation is limited to asymmetric quantization, leaving broader quantization settings for future work.

Reproducibility statement

The weighted objective, token-weight construction, sufficient statistics, Schur recurrence, quantizer updates, and code-sweep complexity are specified in Sections 3.1–3.6. Section 4 reports the models, calibration data, evaluation protocol, software environment, and all principal hyperparameters. Complete derivations are provided in Appendix B.

AI use statement

Generative AI was utilized for minor tasks like grammar checking.

References

Yamato Arai and Yuma Ichikawa. Quantization Error Propagation: Revisiting Layer-Wise Post-Training Quantization. In *Advances in Neural Information Processing Systems*, volume 38, pp. 151916–151951. Curran Associates, Inc., 2025.

Saleh Ashkboos, Amirkeivan Mohtashami, Maximilian L. Croci, Bo Li, Pashmina Cameron, Martin Jaggi, Dan Alistarh, Torsten Hoefler, and James Hensman. QuaRot: Outlier-Free 4-Bit Inference in Rotated LLMs. In *Advances in Neural Information Processing Systems*, volume 37, pp. 100213–100240, 2024.

Kayhan Behdin, Ayan Acharya, Aman Gupta, Qingquan Song, Siyu Zhu, Sathiya Keerthi, and Rahul Mazumder. QuantEase: Optimization-based quantization for language models. *arXiv preprint arXiv:2309.01885*, 2023.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. PIQA: Reasoning about Physical Commonsense in Natural Language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 7432–7439, 2020.

Jerry Chee, Yaohui Cai, Volodymyr Kuleshov, and Christopher M. De Sa. QuIP: 2-Bit Quantization of Large Language Models with Guarantees. *Advances in Neural Information Processing Systems*, 36:4396–4429, 2023.

Jiale Chen, Yalda Shabanzadeh, Elvir Crnčević, Torsten Hoefler, and Dan Alistarh. The Geometry of LLM Quantization: GPTQ as Babai's Nearest Plane Algorithm. In *International Conference on Learning Representations*, 2026.

Mengzhao Chen, Wenqi Shao, Peng Xu, Jiahao Wang, Peng Gao, Kaipeng Zhang, and Ping Luo. EfficientQAT: Efficient Quantization-Aware Training for Large Language Models. In *Annual Meeting of the Association for Computational Linguistics*, pp. 10081–10100, 2025.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the Surprising Difficulty of Natural Yes/No Questions. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2924–2936, 2019.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think You Have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge. arXiv preprint arXiv:1803.05457, 2018.

Vage Egiazarian, Andrei Panferov, Denis Kuznedelev, Elias Frantar, Artem Babenko, and Dan Alistarh. Extreme Compression of Large Language Models via Additive Quantization. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, 2024.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. GPTQ: Accurate Quantization for Generative Pre-trained Transformers. In *International Conference on Learning Representations*, 2023.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The Llama 3 Herd of Models. arXiv preprint arXiv:2407.21783, 2024.

Yuma Ichikawa, Yudai Fujimoto, and Akira Sakai. LPCD: Unified Framework from Layer-Wise to Submodule Quantization. arXiv preprint arXiv:2512.01546, 2025.

Jinuk Kim, Marwa El Halabi, Wonpyo Park, Clemens JS Schaefer, Deokjae Lee, Yeonhong Park, Jae W. Lee, and Hyun Oh Song. GuidedQuant: Large Language Model Quantization via Exploiting End Loss Guidance. In *Forty-second International Conference on Machine Learning*, 2025.

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, M. C., Jianfeng Cai, Xinyuan Cai, Peizhou Cao, Yuxuan Cao, Ziwei Chai, et al. Kimi K3: Open Frontier Intelligence, 2026.

Yuhang Li, Ruokai Yin, Donghyun Lee, Shiting Xiao, and Priyadarshini Panda. GPTAQ: Efficient Finetuning-Free Quantization for Asymmetric Calibration. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, 2025.

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 Machine Learning and Systems*, volume 6, 2024.

Li Lin and Xiaojun Wan. LoaQ: Layer-Wise Output Approximation Quantization. arXiv preprint arXiv:2509.06297, 2026.

Zechun Liu, Changsheng Zhao, Igor Fedorov, Bilge Soran, Dhruv Choudhary, Raghuraman Krishnamoorthi, Vikas Chandra, Yuandong Tian, and Tijmen Blankevoort. SpinQuant: LLM Quantization with Learned Rotations. In *International Conference on Learning Representations*, 2025.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer Sentinel Mixture Models. In *International Conference on Learning Representations*, 2017.

Pranav Ajit Nair and Arun Sai Suggala. CDQuant: Greedy coordinate descent for accurate LLM quantization. *arXiv preprint arXiv:2406.17542*, 2024.

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. *Journal of Machine Learning Research*, 21(140):1–67, 2020.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. WinoGrande: An Adversarial Winograd Schema Challenge at Scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 8732–8740, 2020.

J. Schur. Über Potenzreihen, die im Innern des Einheitskreises beschränkt sind. *Journal für die reine und angewandte Mathematik*, 148:122–145, 1918.

Wenqi Shao, Mengzhao Chen, Zhaoyang Zhang, Peng Xu, Lirui Zhao, Zhiqian Li, Kaipeng Zhang, Peng Gao, Yu Qiao, and Ping Luo. OmniQuant: Omnidirectionally Calibrated Quantization for Large Language Models. In *International Conference on Learning Representations*, 2024.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open Foundation and Fine-Tuned Chat Models. arXiv preprint arXiv:2307.09288, 2023.

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*, volume 235, 2024.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 Technical Report. arXiv preprint arXiv:2505.09388, 2025.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a Machine Really Finish Your Sentence? In *Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800, 2019.

Shihao Zhang, Haoyu Zhang, Ian Colbert, and Rayan Saab. Qronos: Correcting the Past by Shaping the Future... in Post-Training Quantization. In *International Conference on Learning Representations*, 2026.

Xingyu Zheng, Haotong Qin, Yuye Li, Haoran Chu, Jiakai Wang, Jinyang Guo, Michele Magno, and Xianglong Liu. First-order error matters: Accurate compensation for quantized large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 28883–28891, 2026.

A Calibration

The purpose of quantization is not to preserve weights in isolation, but to preserve the model's outputs. Let $E = W^{\mathrm{ref}} - W$ be the weight error and let $X = [x_1, \ldots, x_N]$ contain calibration inputs. The total squared output error is

$$\|EX\|_F^2 = \sum_{k=1}^{N} \|Ex_k\|_2^2 = \mathrm{tr}\big(E\,XX^\top E^\top\big)\,. \tag{20}$$

The Gram matrix $XX^\top$ weights errors by the activation distribution: errors in strongly activated directions matter more than equally large errors in nearly inactive directions. Data-free round-to-nearest minimizes $\|E\|_F^2$, which implicitly replaces $XX^\top$ by the identity and treats every direction equally. Calibration estimates the missing curvature from a small sample of activations. Standard calibration assigns equal explicit mass to all token columns. With nonnegative token weights ω_i, the same derivation becomes

$$\sum_{i=1}^{N} \omega_i \|Ex_i\|_2^2 = \mathrm{tr}\big(EX\Omega X^\top E^\top\big)\,, \qquad \Omega = \mathrm{diag}(\omega_1, \ldots, \omega_N). \tag{21}$$

Thus token weighting changes the empirical activation distribution seen by the quantizer without changing the quadratic form of the problem. This weighted curvature is the sufficient-statistics modification used by teacher-decision weighting in Section 3.5. The residual-add target independently modifies the weighted cross-Gram $Y\Omega X^\top$. The unweighted special cases recover the Hessians used by GPTQ (Frantar et al., 2023) and QEP (Arai & Ichikawa, 2025).

B Detailed Derivations

B.1 Proof of Proposition 1

Terms containing only the fixed prefix W_F are constant. Expanding the quadratic form in Eq. 8, the cross terms involving W_F and W_c are

$$\frac{1}{2}\,\mathrm{tr}\big(W_c G_{cF} W_F^\top + W_F G_{Fc} W_c^\top\big) = \mathrm{tr}(W_F G_{Fc} W_c^\top), \tag{22}$$

where symmetry gives $G_{cF} = G_{Fc}^\top$. Combining this expression with $-\,\mathrm{tr}(C_c W_c^\top)$ yields the effective linear coefficient $C_c^{\mathrm{eff}} = C_c - W_F G_{Fc}$. The same argument gives $C_r^{\mathrm{eff}} = C_r - W_F G_{Fr}$ for the suffix.

Collecting all terms that depend on (W_c, W_r) gives

$$\begin{aligned}\mathcal{L}(W_c, W_r) = \frac{1}{2}\,\mathrm{tr}\left(\begin{bmatrix} W_c & W_r \end{bmatrix} \begin{bmatrix} G_{cc} & G_{cr} \\ G_{rc} & G_{rr} \end{bmatrix} \begin{bmatrix} W_c & W_r \end{bmatrix}^\top\right) \\ - \mathrm{tr}(C_c^{\mathrm{eff}} W_c^\top) - \mathrm{tr}(C_r^{\mathrm{eff}} W_r^\top) + \mathrm{const}.\end{aligned} \tag{23}$$

Holding W_c fixed and differentiating with respect to W_r gives

$$W_c G_{cr} + W_r G_{rr} - C_r^{\mathrm{eff}} = 0, \tag{24}$$

so the optimal continuous suffix response is

$$W_r^\star(W_c) = \big(C_r^{\mathrm{eff}} - W_c G_{cr}\big)\, G_{rr}^{-1}. \tag{25}$$

Let $U = C_r^{\text{eff}} - W_c G_{cr}$. The suffix-dependent terms can be completed to a square

$$\begin{aligned}&\frac{1}{2}\operatorname{tr}(W_r G_{rr} W_r^\top) - \operatorname{tr}(U W_r^\top)\\ &\quad= \frac{1}{2}\operatorname{tr}\big((W_r - U G_{rr}^{-1}) G_{rr} (W_r - U G_{rr}^{-1})^\top\big) - \frac{1}{2}\operatorname{tr}(U G_{rr}^{-1} U^\top).\end{aligned} \tag{26}$$

The first term vanishes at Eq. 25. Expanding the second gives

$$\begin{aligned}-\frac{1}{2}\operatorname{tr}(U G_{rr}^{-1} U^\top) = &-\frac{1}{2}\operatorname{tr}(C_r^{\text{eff}} G_{rr}^{-1} C_r^{\text{eff}\top})\\ &+\operatorname{tr}(C_r^{\text{eff}} G_{rr}^{-1} G_{rc} W_c^\top)\\ &-\frac{1}{2}\operatorname{tr}(W_c G_{cr} G_{rr}^{-1} G_{rc} W_c^\top).\end{aligned} \tag{27}$$

The first term is constant in W_c. Combining the remaining terms with Eq. 23 produces the curvature S and the linear coefficient T of Eq. 11, proving Proposition 1. □

B.2 Proof of Proposition 2

For a fixed zero-point o, substitute $q_i = a_i u_i(o)$ into Eq. 14:

$$\mathcal{L}_i(a_i; o) = \frac{1}{2} A_i(o) a_i^2 - d_i(o) a_i. \tag{28}$$

When $A_i(o) > 0$, differentiating gives the unconstrained optimum $d_i(o)/A_i(o)$; projecting it onto $a_i \geq \epsilon_a$ gives Eq. 15. Enumerating the finite set of b-bit zero-points and selecting the smallest substituted objective is globally exact for fixed codes. □

C Quantization runtime

Table 6 reports measured end-to-end wall-clock, including statistics collection and discrete optimization. SchurQuant performs $R = 16$ sweeps and one current full-model calibration pass before each Transformer layer in addition to the clean-cache pass. It is therefore an offline accuracy–quantization-time trade-off, not an inference-speed result.

Table 6: End-to-end quantization time in minutes on one NVIDIA H100 PCIe GPU.

Model	Method	2 bit	3 bit	4 bit
Qwen3-4B	GPTQ	5.5	6.9	5.5
	FOEM	6.4	7.6	6.6
	QEP	55.6	71.2	69.4
	SchurQuant	253.7	254.0	282.5
Llama2-7B	GPTQ	7.1	8.4	7.1
	FOEM	8.2	9.1	8.0
	QEP	38.6	54.1	59.3
	SchurQuant	224.2	220.9	235.6

At 2 bits, quantization with SchurQuant takes 45.8×/31.8× as long as GPTQ and 4.6×/5.8× QEP on Qwen3-4B/Llama2-7B; the 3/4-bit GPTQ ratios are 37.0/51.1× and 26.5/33.0×. Instrumentation now separates statistics and optimizer time and peak memory. Dense evaluator checkpoints are not packed-size evidence.

D Quantization setup

Unless stated otherwise, all weight-only experiments use group size $g = 128$, a row-wise asymmetric quantizer, $R = 16$ refinement steps, and float16 computation. We draw 128 random windows of length 2,048 from long documents in the C4 training split (Raffel et al., 2020), following the standard

calibration protocol of GPTQ. The final SCHURQUANT configuration constructs the top-mismatch weights from all 128 calibration windows before quantizing each Transformer layer. We use mismatch multiplier $\rho = 8$ and the detached projection-local residual normalization in Eq. 18 with residual floor $\epsilon = 10^{-8}$. We quantize the linear attention and MLP projections inside Transformer blocks while retaining the embeddings, language-model head, and normalization parameters in their original precision.