BEYOND SHADOW WEIGHTS: QUANTIZATION-AWARE TRAINING AS QUANTIZED-ENDPOINT DESCENT

Sheng-An Xu[1] **Hanyang Li**[1] **Jianhao Ma**[2] **Ying Cui**[1]
[1]Department of Industrial Engineering and Operations Research, University of California, Berkeley
[2]Department of Statistics and Data Science, University of Pennsylvania

ABSTRACT

Quantization-aware training (QAT) updates a full-precision shadow weight $\mathbf{x}$ but deploys the quantized endpoint $Q(\mathbf{x})$. Existing explanations for QAT largely view its success through the lens of shadow weights: QAT can move $\mathbf{x}$ toward flatter basins, gain robustness from quantization-induced oscillations, or balance the shadow loss $f(\mathbf{x})$ against the quantization error $\|\mathbf{x} - Q(\mathbf{x})\|_2$. These perspectives do not directly explain the empirical observation that the deployed endpoint loss $f(Q(\mathbf{x}))$ improves while the shadow loss $f(\mathbf{x})$ does not, and can even increase substantially. In this paper, we offer a different explanation by treating QAT as finite-grid endpoint dynamics. Motivated by the approximate normality of rescaled pretrained weights, we propose an idealized model for the residual phase, which records where each shadow weight sits inside its quantization cell as a fraction of the cell width. This model leads to a crossing law that determines which coordinates cross quantization boundaries after a shadow update. Inspired by the idealized model and signal-imbalance phenomenon in QAT, we further propose *QAR (Quantization with Amplified Routing)*, an algorithmic framework that directly operates on the quantization code. In contrast to QAT, QAR is both theoretically grounded and memory-efficient: it admits feasible-gradient bounds for a family of power amplifiers up to unavoidable finite-grid floors without retaining a full-precision shadow weight copy. Experiments on post-training of large language models provide evidence consistent with the endpoint view and show that QAR can be comparable to or better than QAT with smaller memory cost.

1 INTRODUCTION

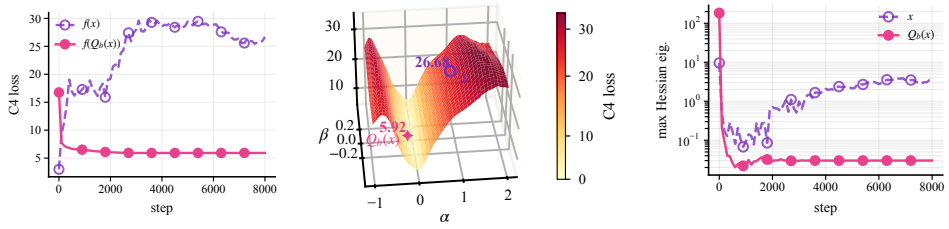
Modern deep neural networks have achieved broad empirical success across language, vision, and decision-making tasks (Brown et al., 2020; He et al., 2016; Silver et al., 2016), by scaling parameters, data, and compute (Kaplan et al., 2020; Hoffmann et al., 2022). But this scaling comes at a cost. As the model becomes larger, deployment is increasingly constrained by memory footprint, and inference slows down due to high memory bandwidth and compute throughput. This phenomenon is highly pronounced for large language models (LLMs). Low-bit quantization addresses this issue by replacing high-precision weights with a small set of representable values. Post-training quantization methods compress a pretrained model with little additional training, and they have become a central tool for LLM deployment. Representative examples include GPTQ (Frantar et al., 2022), SmoothQuant (Xiao et al., 2023), and AWQ (Lin et al., 2024). However, at more aggressive bit widths, the quantized model can lose accuracy substantially. In this regime, quantization-aware training (QAT) becomes the standard method for repair. For example, LLM-QAT (Liu et al., 2024) and EfficientQAT (Chen et al., 2025) make this repair step practical for LLMs with data-free distillation or reduced memory cost.

Conceptually, QAT tries to solve the deployment problem. Let $f : \mathbb{R}^d \to \mathbb{R}$ be the full-precision loss, Q_b be a b-bit quantizer, and $\mathcal{Q}_b$ be the finite set of deployed endpoints. To achieve the best deployment performance, the training objective should be a discrete constrained problem

$$\min_{\mathbf{q} \in \mathcal{Q}_b} f(\mathbf{q}). \tag{1}$$

By introducing a full-precision shadow weight $\mathbf{x}$, we can equivalently write the above problem as

$$\min_{\mathbf{x} \in \mathbb{R}^d} F_b(\mathbf{x}) = f(Q_b(\mathbf{x})).$$



(a) 2-bit evaluation loss **(b)** Loss landscape at the final step **(c)** 2-bit max Hessian eigenvalue

Figure 1: Qwen-2.5-0.5B 2-bit loss and curvature diagnostics using seed 0. (a) contrasts the increasing shadow loss with the low endpoint loss. (b) plots evaluation loss on $Q_b(\mathbf{x}) + \alpha(\mathbf{x} - Q_b(\mathbf{x})) + \beta\|\mathbf{x} - Q_b(\mathbf{x})\|_2\mathbf{v}$ for a random unit $\mathbf{v} \perp (\mathbf{x} - Q_b(\mathbf{x}))$. (c) shows curvature better aligned with the deployed endpoint in 2 bits.

Since Q_b is constant on each quantization cell, the exact gradient of $f(Q_b(\mathbf{x}))$ is zero almost everywhere; direct differentiation therefore gives no useful descent direction. QAT bypasses this degeneracy through the straight-through estimator (STE), which treats Q_b as the identity in the backward pass (Bengio et al., 2013). In the notation above, one step of QAT has the form

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \eta_t \mathbf{u}_t, \qquad \mathbf{u}_t = \mathcal{U}_t(\nabla f(Q_b(\mathbf{x}_t))),$$

where $\mathcal{U}_t(\cdot)$ is a time-dependent update map. STE-based QAT has been effective in practice (Courbariaux et al., 2015; Zhou et al., 2016; Jacob et al., 2018; Choi et al., 2018; Esser et al., 2020), but it remains largely heuristic for solving the problem (1). From a classical optimization viewpoint, QAT is unusual because the variable being updated is not the variable being deployed. During training, QAT maintains a full-precision shadow weight $\mathbf{x}$, also called the latent weight, but at deployment the model uses the quantized endpoint $\mathbf{q}$. Thus, to analyze QAT, **the central analytical question is not how the shadow trajectory behaves, but how the deployed endpoint trajectory evolves.**

The main analytical difficulty lies in the discreteness induced by quantization. Standard first-order arguments alone fail to match the motion of the deployed endpoint. By L-smoothness of f, the classical gradient update $\mathbf{x}_{t+1} = \mathbf{x}_t - \eta\nabla f(\mathbf{x}_t)$ yields $f(\mathbf{x}_{t+1}) - f(\mathbf{x_t}) \le -\eta\big(1 - \frac{L\eta}{2}\big)\|\nabla f(\mathbf{x_t})\|_2^2$, so choosing η small enough guarantees a descent in the objective value. However, the loss at the QAT endpoint $f(Q_b(\mathbf{x}_t))$ behaves differently. Although the change in shadow weights $\mathbf{x}_{t+1} - \mathbf{x}_t$ is proportional to the stepsize η_t, once $\mathbf{x}_{t+1}$ crosses a quantization boundary, the change in the deployed endpoint $Q_b(\mathbf{x}_{t+1}) - Q_b(\mathbf{x}_t)$ is a grid jump. Shrinking η cannot reduce the size of this jump, hence descent of the endpoint loss no longer holds.

Existing Explanations. For QAT with weight-only quantization, explanations in the existing literature can be grouped into three categories. The first is a flatness view: QAT can drive the shadow weight into a flat region where replacing $\mathbf{x}$ by $Q_b(\mathbf{x})$ changes the loss only mildly. For example, Javed et al. (2025) model quantization as adding noise for the shadow weight. In this way, QAT has an implicit regularizer for the second-order term and can steer the shadow point toward flatter regions. This perspective is inspired by the broader literature which links flatter regions of the loss landscape to better generalization (Hochreiter & Schmidhuber, 1997; Keskar et al., 2017; Jiang et al., 2020; Foret et al., 2021). A related explanation builds on the river-valley picture of loss landscapes (Wen et al., 2025). For instance, Li et al. (2026) argue that when the deployed endpoint leaves a low-loss basin, the STE update can include an inward component that pulls the quantized endpoint back toward the basin, where $f(Q_b(\mathbf{x}))$ stays close to $f(\mathbf{x})$. The second view interprets QAT as minimizing the shadow loss $f(\mathbf{x})$ plus a regularizer that penalizes distance of the shadow weight to the quantized set. ProxQuant (Bai et al., 2019) proposes a continuous regularized problem that encourages the shadow weight to approach the quantized set, and STE-based QAT corresponds to the hard-projection limiting case of the proximal framework. The third view emphasizes oscillation robustness gained from quantization. In a toy model, Wenshøj et al. (2025) argue that the STE update contains a residual-driven component that pushes shadow weights away from their nearest quantization level and toward cell boundaries. And this effect can be achieved by adding a regularization term to the shadow loss $f(\mathbf{x})$.

Figure 1 illustrates a regime that is not described by any account requiring the shadow loss $f(\mathbf{x}_t)$ to decrease. In this 2-bit Qwen-2.5-0.5B run, the loss at the shadow weight $f(\mathbf{x}_t)$ increases substantially while the deployed endpoint loss $f(Q_b(\mathbf{x}_t))$ remains much lower. Moreover, the loss landscape

around the deployed endpoint $Q_b(\mathbf{x}_t)$ appears to be flatter, rather than around $\mathbf{x}_t$. These observations suggest a stronger conclusion: QAT can improve the deployed quantized endpoint, even at the expense of sacrificing the performance of the full-precision objective. This motivates us to analyze the deployed endpoint loss directly.

Contribution. Our contributions are two-fold. First, we analyze QAT through the residual $\mathbf{r}_t = \mathbf{x}_t - Q_b(\mathbf{x}_t)$, which records the position of the shadow point inside its quantization cell. Motivated by the approximate normality of the rescaled pretrained weights, we introduce an idealized residual-phase model that explains why pooled phases can approach uniformity at higher bit widths. This model expresses endpoint motion through random boundary-crossing indicators, allowing the dynamics to be analyzed directly at the quantized endpoint without explicitly tracking the shadow weights. It offers an approximate perspective on when QAT can help training: the bound certifies expected endpoint descent when the first-order signal exceeds the upper bound on the finite-jump penalty.

Second, we identify asymmetric oscillations caused by signal imbalance in QAT and use this mechanism to develop **QAR** (**Q**uantization with **A**mplified **R**outing), a memory-efficient endpoint-training framework that updates quantized codes directly and prioritizes transitions associated with stronger signals. We establish convergence guarantee for QAR and characterize when signal amplification remains beneficial under stochastic gradient noise. Post-training experiments on Qwen-2.5-0.5B and Llama-3.2-3B support finite-grid endpoint view and show that QAR can achieve performance comparable to or better than QAT with up to **26.3%** lower peak training memory.

2 PRELIMINARIES

To begin with, quantization maps high-precision floating-point values to low-precision discrete representations, thereby reducing memory footprint. We use the standard uniform affine quantization notation (Nagel et al., 2021). Throughout the paper, unless stated otherwise, we use a signed symmetric uniform b-bit quantizer $Q_b(\cdot)$.[1] Let $\mathbf{s} = (s_1, s_2, \ldots, s_d) > 0$ be the fixed local scale. Define $K_b = 2^{b-1} - 1$ and $\mathcal{K}_b = \{-K_b, -K_b+1, \ldots, K_b\}$. For shadow weight $\mathbf{x}$, the i-th coordinate of the deployed endpoint is

$$Q_b(\mathbf{x})_i = s_i \ \mathrm{clamp}\left(\left\lfloor \frac{x_i}{s_i} \right\rceil ; -K_b, K_b \right),$$

where $\lfloor \cdot \rceil$ denotes rounding to the nearest integer and $\mathrm{clamp}(a; \ell, u) = \min\{\max\{a, \ell\}, u\}$. When the bit width is clear from context, we write $Q(\mathbf{x})$ for $Q_b(\mathbf{x})$ for simplicity. Then the deployed values in coordinate i are $s_i\mathcal{K}_b$, with adjacent interior spacing s_i. For an interior code $k \in \{-K_b + 1, \ldots, K_b - 1\}$, the non-saturated cell is $C_{i,k} = \left[s_i\left(k - \frac{1}{2}\right), s_i\left(k + \frac{1}{2}\right)\right)$. We call coordinate i non-saturated when its deployed code $k_i = Q(\mathbf{x})_i / s_i$ satisfies $-K_b < k_i < K_b$; coordinates with $k_i = \pm K_b$ are saturated.

Let $\mathbf{g} = \nabla f(\mathbf{q})$ be the endpoint gradient. A one-step QAT shadow update with stepsize η is

$$\mathbf{x}^+ = \mathbf{x} - \eta \mathbf{u} \quad \text{with } \mathbf{u} = \mathcal{U}_t(\mathbf{g}). \tag{2}$$

Here $\mathcal{U}_t(\cdot)$ is the update rule applied to the endpoint gradient at step t. For example, GD takes $\mathcal{U}_t(\mathbf{g}) = \mathbf{g}$ and signGD corresponds to $\mathcal{U}_t(\mathbf{g}) = \mathrm{sign}(\mathbf{g})$. We further write $\mathbf{q}^+ = Q(\mathbf{x}^+)$, and call coordinate i active in this step when $u_i \neq 0$. Throughout, we use the convention $\mathrm{sign}(0) = 0$.

Assumption 1 (Bounded update direction)**.** *There exists a constant $U < \infty$ such that the update direction $\mathbf{u} = \mathcal{U}_t(\mathbf{g})$ satisfying $\|\mathbf{u}\|_\infty \leq U$.*

Assumption 2 (Smoothness)**.** *The loss f is differentiable, and there exists a nonnegative vector $\mathbf{L} = (L_1, \ldots, L_d)$ such that for all $\mathbf{x}, \mathbf{y}$,*

$$f(\mathbf{y}) \leq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{1}{2} \sum_{i=1}^{d} L_i (y_i - x_i)^2.$$

This coordinatewise smoothness is commonly used in sign-gradient analyses such as Bernstein et al. (2018). Standard L-smoothness implies this assumption by taking $L_i = L$ for all coordinates.

[1] In this convention, we drop the most negative code so that the positive and negative endpoints have equal magnitude. The resulting codebook has $2^b - 1$ levels. For example, the $b = 2$ setting in this paper has levels $\{-1, 0, 1\}$, so it corresponds to the ternary setting, or 1.58-bit, in common terminology.

Lemma 1 (First-order descent alignment)**.** *For the one-step QAT update (2) at point* $\mathbf{x}$ *with stepsize* η*, we have* $\mathbf{u} \odot (\mathbf{q}^+ - \mathbf{q}) \le 0$*. If in addition* $\mathbf{g} \odot \mathbf{u} \ge 0$*, then* $\mathbf{g} \odot (\mathbf{q}^+ - \mathbf{q}) \le 0$*, and consequently,*

$$\langle \mathbf{g}, \mathbf{q}^+ - \mathbf{q} \rangle \le 0.$$

Here $\odot$ *denotes the Hadamard product, and the inequalities should be understood coordinatewise.*

The proof is deferred to Appendix D.1. Lemma 1 shows that each realized endpoint jump moves opposite to the update direction. If the update is coordinatewise aligned with the endpoint gradient, the jump is also aligned with first-order descent. However, to get the descent of the loss, we need a more fine-grained bound. To achieve the bound, we assume that the stepsize is small enough that $\eta|u_i| < s_i$ for every coordinate i. In this case, each coordinate crosses at most one interior threshold per step, so $(\mathbf{q}^+ - \mathbf{q})_i \in \{-s_i, 0, s_i\}$. Let $\mathbf{J} = (J_1, \dots, J_d)$ with $J_i = \mathbf{1}\{q_i^+ \ne q_i\}$ be the crossing indicator, where $J_i = 1$ indicates that a one-cell endpoint jump occurs. Now we can describe the grid-jump mechanism of the QAT.

Proposition 2 (Realized endpoint loss bound)**.** *Suppose Assumptions 1 and 2 hold, and the stepsize is small enough that* $\eta U < \min_i s_i$*. Then*

$$\mathbf{q}^+ - \mathbf{q} = -\mathbf{s} \odot \operatorname{sign}(\mathbf{u}) \odot \mathbf{J},$$

and

$$f(\mathbf{q}^+) - f(\mathbf{q}) \le - \sum_{\{i: J_i = 1\}} s_i \operatorname{sign}(u_i) g_i + \frac{1}{2} \sum_{\{i: J_i = 1\}} L_i s_i^2. \tag{3}$$

Proposition 2 illustrates the finite-grid descent mechanism: once the crossed coordinates are known, the bound certifies endpoint descent whenever the first-order signal exceeds the second-order penalty. The proof is deferred to Appendix D.2.

3 Residual Homogeneity and Endpoint Motion

The bound in Proposition 2 controls the endpoint-loss change for a realized crossing set, but it does not specify how often each coordinate crosses. To quantitatively understand how the crossing mechanism of QAT aggregates to a steady decrease of the endpoint loss and to illustrate when QAT stops making progress, we also need to know how often such crossings occur. Thus, we model the crossing indicators J_i conditionally on the current endpoint and update direction. The resulting conditional laws provide an approximate perspective on QAT by linking the frequency of boundary crossings to expected descent at the quantized endpoint.

Model 1 (Conditional crossing model)**.** *Fix an endpoint* $\mathbf{q} = Q(\mathbf{x})$*, an update direction* $\mathbf{u}$*, and a stepsize* η *in a regime where each active coordinate can cross at most one neighboring quantization boundary. Let* $J_i \in \{0, 1\}$ *be the crossing indicator defined above. We model* J_i*, conditional on* $(\eta, \mathbf{q}, \mathbf{u})$*, as a Bernoulli random variable with conditional mean*

$$p_i(\eta, \mathbf{q}, \mathbf{u}) = \mathbb{E}[J_i \mid \eta, \mathbf{q}, \mathbf{u}] = \Pr(J_i = 1 \mid \eta, \mathbf{q}, \mathbf{u}).$$

The deterministic realized case is included by taking $p_i(\eta, \mathbf{q}, \mathbf{u}) \in \{0, 1\}$*.*

To further illustrate the inner structure of the endpoint motion, we introduce the residual phase. For $\mathbf{q} = Q(\mathbf{x})$, define the residual $\mathbf{r} = \mathbf{x} - Q(\mathbf{x})$. In a non-saturated cell, $|r_i| \le \frac{s_i}{2}$, and the normalized residual $\theta_i = r_i / s_i$ lies in $\left[-\frac{1}{2}, \frac{1}{2}\right)$. This phase is the state variable that decides whether the next shadow update changes the deployed endpoint. If x_i is near the center of its cell, a small shadow update leaves q_i unchanged; if x_i is within a boundary window of width $\eta|u_i|$, the same update produces a one-cell endpoint jump. We make this precise in Corollary 6 (see Appendix D.3): the crossing event is exactly the event that θ_i falls in a window of normalized width $\rho_i = \eta|u_i|/s_i$ at the boundary the update moves toward, so p_i is the mass the conditional phase law assigns to that window.

In LLM quantization, it is empirically observed that after layerwise or blockwise rescaling, the central mass of pretrained weights is often close to zero-centered normal distributions. QLoRA (Dettmers et al., 2023) uses this observation to motivate NormalFloat (NF4), a low-bit datatype designed for normally distributed weights. This observation is also related to mean-field analyses of

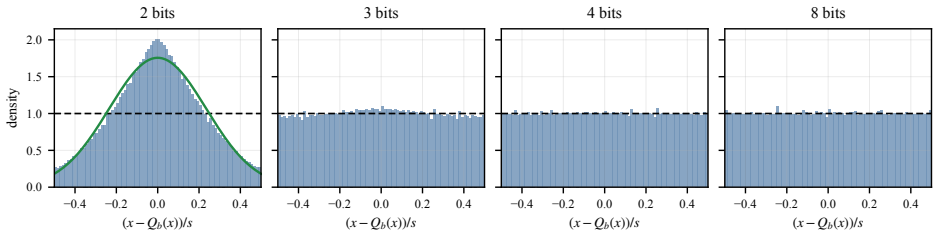


Figure 2: Distributions of the normalized residual $\theta_i = (x_i - Q(\mathbf{x})_i)/s_i$ for Qwen-2.5-0.5B at the pretrained point, pooled over non-saturated linear weights. The solid curve in the 2-bit panel is a fitted zero-centered truncated normal. The dashed reference is the uniform cell-phase density on $[-\frac{1}{2}, \frac{1}{2})$. The 3-, 4-, and 8-bit cases are close to the uniform residual-phase calculation underlying Model 2.

wide neural-network training, where the empirical measure of the parameter admits a large-width limiting description and cross-particle dependence becomes weak under suitable scaling (Sirignano & Spiliopoulos, 2020; De Bortoli et al., 2020). Corresponding pretrained-weight diagnostics for Qwen-2.5-0.5B are reported in Appendix C.2.

Based on this observation, we reason about the residual phase induced by folding weights through a finite grid. Consider a scalar pretrained weight $X \sim \mathcal{N}(0, \tau^2)$ with $\tau > 0$, and a symmetric b-bit quantizer with grid spacing $s_b > 0$. When the weights are measured in units of one quantization step, X/s_b has standard deviation $\lambda_b = \tau/s_b$. Aggregating the normalized residual phase across all non-saturated interior cells gives a folded density

$$\pi(\theta) \propto \sum_{k=-K_b+1}^{K_b-1} \exp\left(-\frac{(k+\theta)^2}{2\lambda_b^2}\right), \qquad \theta \in \left[-\frac{1}{2}, \frac{1}{2}\right).$$

For this zero-centered folded model, the mass near the upper and lower cell boundaries is the same. We therefore write the one-sided boundary mass as $B(\rho) = \int_{1/2-\rho}^{1/2} \pi(\theta)\,\mathrm{d}\theta = \int_{-1/2}^{-1/2+\rho} \pi(\theta)\,\mathrm{d}\theta$ for $0 \le \rho < 1$. For $b = 2$, we have $K_2 = 1$, so the only non-saturated interior cell is the zero cell and the pooled residual phase is center-concentrated. Such a phase places less mass near the cell boundary than the uniform case, so $B(\rho) < \rho$ for small ρ. For $b > 2$, pooling over many interior cells makes the folded sum nearly flat within one cell, so $B(\rho) \approx \rho$. This is why the 3-, 4-, and 8-bit histograms in Figure 2 are much closer to a uniform cell-phase density. The arguments above start from a normally distributed weight parameter, but the resulting near-uniformity holds for a broad class of weight densities (see Appendix C.1 for more discussion). In Appendix C.3, we also report the predicted folded densities together with their empirical counterparts.

To obtain a tractable description throughout training, we introduce an approximation in which the residual phase remains representative after conditioning on the current endpoint: among non-saturated coordinates, the phase laws do not vary systematically with $\mathbf{q}$. This approximation implies that the update-selected crossing probabilities exhibit the same near-uniform behavior. Figure 3 gives the corresponding smooth-density intuition: conditioning on $\mathbf{q}$ restricts each active coordinate x_i to its own quantization cell centered at q_i. If the conditional marginal density of x_i varies little across that cell, then the phase $\theta_i = (x_i - q_i)/s_i$ is approximately uniform on $[-\frac{1}{2}, \frac{1}{2})$. Since the update-selected boundary window has normalized width $\rho_i = \eta|u_i|/s_i$, the uniform phase calculation gives crossing probability ρ_i. This gives the simplified conditional model.

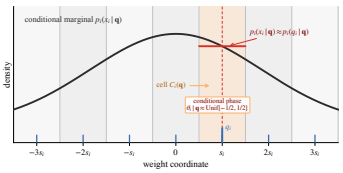


Figure 3: Illustration for residual homogeneity.

Model 2 (Simplified crossing model)**.** *Suppose Model 1 holds and that, conditional on the current endpoint* $\mathbf{q} = Q(\mathbf{x})$*, update direction* $\mathbf{u}$*, and stepsize* η*, each non-saturated coordinate has crossing probability*

$$p_i(\eta, \mathbf{q}, \mathbf{u}) = \Pr(J_i = 1 \mid \eta, \mathbf{q}, \mathbf{u}) = \frac{\eta|u_i|}{s_i}.$$

With the crossing model, we can translate random boundary crossings into an expected one-step bound on the endpoint loss.

Theorem 3 (Conditional endpoint descent under the crossing model)**.** *Suppose Assumptions 1 and 2 hold, the crossings follow Model 1, and the stepsize* η *satisfies* $\eta U < s_i$ *for every coordinate* i*. Given*

the endpoint $\mathbf{q} = Q(\mathbf{x})$ *and update direction* $\mathbf{u} = \mathcal{U}(\nabla f(\mathbf{q}))$*, write* $p_i = p_i(\eta, \mathbf{q}, \mathbf{u})$*. The endpoint update can be written as*

$$\mathbf{q}^+ - \mathbf{q} = -\mathbf{s} \odot \operatorname{sign}(\mathbf{u}) \odot \mathbf{J} \tag{4}$$

where $J_i \in \{0, 1\}$ *satisfies* $\Pr(J_i = 1 \mid \eta, \mathbf{q}, \mathbf{u}) = p_i$*. Consequently,*

$$\mathbb{E}[f(\mathbf{q}^+) - f(\mathbf{q}) \mid \eta, \mathbf{q}, \mathbf{u}] \le -\sum_i p_i s_i \operatorname{sign}(u_i) g_i + \frac{1}{2} \sum_i p_i L_i s_i^2. \tag{5}$$

If, in addition, Model 2 holds and $\mathbf{q}$ *is non-saturated in each coordinate, then*

$$\mathbb{E}[f(\mathbf{q}^+) - f(\mathbf{q}) \mid \eta, \mathbf{q}, \mathbf{u}] \le -\eta \langle \mathbf{g}, \mathbf{u} \rangle + \frac{\eta}{2} \sum_i L_i s_i |u_i|. \tag{6}$$

The proof is deferred to Appendix D.4. Theorem 3 is the one-step certificate for quantized-endpoint descent. Under the uniform crossing model, the first-order term $-\eta \langle \mathbf{g}, \mathbf{u} \rangle$ matches that of continuous optimization. However, the second-order term is also linear in η, rather than $O(\eta^2)$ as in continuous optimization. This difference arises because a smaller η reduces the probability of a grid jump but not the size of an accepted jump. Therefore, making the stepsize arbitrarily small does not by itself guarantee descent. The bound therefore certifies descent when the first-order signal is large enough to dominate the second-order effect of one-cell jumps.

4 QAR: A Direct Endpoint Algorithmic Framework

4.1 Signal-Imbalance Induced Oscillation in QAT Dynamics

So far, we have used the crossing model to describe how shadow updates induce transitions between deployed endpoints. The simplified crossing model is reasonably well calibrated at higher bit widths, as shown by Figure 11 in Appendix B.4. However, this pooled view can overlook the temporal dependence in QAT dynamics. In particular, adjacent endpoints can have different update signals, causing the shadow weight to spend unequal amounts of time at them. Consider $f(q) = \frac{1}{2}(q - q^\star)^2$, adjacent endpoints $q_1 < q^\star < q_2$, and the shadow update $x_{t+1} = x_t - \eta g(Q(x_t))$, where $g(q) = q - q^\star$. If π_j denotes the residence fraction at q_j, the small-step two-cell dynamics approximately balance as

$$\pi_1 g(q_1) + \pi_2 g(q_2) \approx 0, \qquad \frac{\pi_2}{\pi_1} \approx \frac{|g(q_1)|}{|g(q_2)|}.$$

Thus, the weaker the signal at an endpoint, the longer the shadow resides there. In Figure 4, the shadow drifts slowly away from q_1, but returns rapidly after crossing to q_2. Such signal-imbalanced oscillations have been observed in QAT (Nagel et al., 2022).

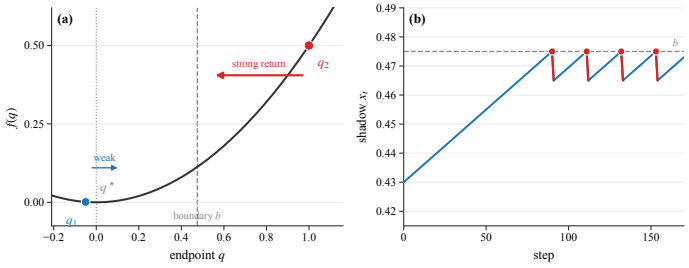


Figure 4: Signal-imbalanced QAT dynamics for $f(q) = \frac{1}{2}q^2$, $q_1 = -0.05$, and $q_2 = 1$. The weak signal at q_1 causes slow drift toward the boundary, whereas the larger signal at q_2 quickly pulls the shadow back.

Stochastic gradients can further reinforce this asymmetry later in training, when gradient signals become small relative to mini-batch noise. At a weak-signal endpoint such as q_1, noise can flip the effective update direction and undo part of the accumulated motion toward the boundary, further lengthening the residence time. Once the shadow crosses to q_2, the stronger opposing signal is less easily reversed by noise and quickly drives it back toward q_1. This history-dependent cycle biases how long the shadow remains at each endpoint and which crossings persist, even when the pooled crossing rate is well predicted.

Algorithm 1 QAR framework: one endpoint step on the finite codebook

Input: code $\mathbf{k}_t \in \mathcal{K} = \mathcal{K}_1 \times \cdots \times \mathcal{K}_d$, spacings $\mathbf{s}$, stepsize η, route map $\mathcal{U}_t$, amplifier map $\mathcal{A}$
1: recover endpoint $\mathbf{q}_t = \mathbf{s} \odot \mathbf{k}_t$
2: compute endpoint-gradient estimate $\widehat{\mathbf{g}}_t$ at $\mathbf{q}_t$
3: set route $\mathbf{u}_t = \mathcal{U}_t(\widehat{\mathbf{g}}_t)$
4: set direction $\mathbf{z}_t = \operatorname{sign}(\mathbf{u}_t)$
5: set feasibility mask $\mathbf{M}_t = \mathbf{1}_{\mathcal{K}}(\mathbf{k}_t - \mathbf{z}_t)$
6: set probability $\mathbf{p}_t = \min\{\mathcal{A}(\eta, \mathbf{u}_t, \mathbf{s}, \mathbf{M}_t), \mathbf{1}\}$
7: sample $\mathbf{J}_t \sim \text{Bernoulli}(\mathbf{p}_t)$ coordinatewise
8: set $\mathbf{k}_{t+1} = \mathbf{k}_t - \mathbf{z}_t \odot \mathbf{J}_t$
Output: updated code $\mathbf{k}_{t+1}$

This asymmetry can favor the better endpoint, but it is realized indirectly: weak signals move the shadow slowly through a cell, while strong opposite signals can quickly undo a crossing. This suggests prioritizing endpoint transitions associated with stronger signals.

4.2 The Proposed Algorithm

The analysis in Section 3 shows how residual phases convert shadow-weight boundary crossings into probabilistic one-cell endpoint transitions. The signal-imbalance mechanism in Section 4.1 further suggests assigning more transition probability to coordinates with stronger signals. Together, these two observations motivate a direct realization of quantized-endpoint dynamics without maintaining a persistent shadow weight. This leads to **QAR** (**Q**uantization with **A**mplified **R**outing), an algorithmic framework that samples transitions directly on the quantized codebook. In this framework, a route map selects the update signal, while an amplifier map converts that signal into coordinatewise jump intensities. Formally, let

$$\mathcal{K}_i = \{\underline{k}_i, \underline{k}_i + 1, \ldots, \overline{k}_i\} \subset \mathbb{Z}, \quad \mathcal{Q} = \{\mathbf{q} : \ q_i = s_i k_i, \ k_i \in \mathcal{K}_i, \ i = 1, \ldots, d\},$$

where $\underline{k}_i, \overline{k}_i \in \mathbb{Z}$ and $\underline{k}_i \le \overline{k}_i$. The symmetric b-bit codebook introduced above is the special case $\mathcal{K}_i = \mathcal{K}_b$ for every coordinate.

QAR implements these transitions using the integer code $\mathbf{k}_t$ instead of maintaining a persistent full-precision shadow copy. Under the groupwise quantization used for LLMs, its persistent weight state consists of $\mathbf{k}_t$ and one shared scale s_G per group, while $\mathbf{q}_t = \mathbf{s} \odot \mathbf{k}_t$ is materialized only for computation. QAR therefore requires less persistent weight state than QAT; detailed memory accounting is provided in Appendix B.5.

For a candidate code vector $\mathbf{v}$, define the coordinatewise feasibility indicator $\mathbf{1}_{\mathcal{K}}(\mathbf{v})_i = \mathbf{1}\{v_i \in \mathcal{K}_i\}$. Algorithm 1 uses an amplifier map $\mathcal{A} : \mathbb{R}_+ \times \mathbb{R}^d \times \mathbb{R}^d_{++} \times \{0,1\}^d \to \mathbb{R}^d_+$ to convert the gradient magnitude into jump intensities. Because the movement direction is represented separately by $\mathbf{z}_t = \operatorname{sign}(\mathbf{u}_t)$, the amplifier returns only nonnegative intensities and assigns zero intensity to infeasible coordinates through $\mathbf{M}_t$. This masking guarantees that $\mathbf{k}_{t+1} \in \mathcal{K}$ whenever $\mathbf{k}_t \in \mathcal{K}$, so every QAR iterate is well-defined. Both the minimum and the Bernoulli sampling in Algorithm 1 are applied coordinatewise.

A natural baseline is obtained by reproducing the simplified crossing law in Model 2. Applying the feasibility mask therefore gives the uniform amplifier

$$\mathcal{A}_{\text{unif}}(\eta, \mathbf{u}, \mathbf{s}, \mathbf{M}) = \eta \mathbf{M} \odot |\mathbf{u}| \oslash \mathbf{s}, \tag{7}$$

where $\oslash$ denotes elementwise division. With this choice, Algorithm 1 realizes the boundary-projected uniform crossing model: the amplifier assigns zero intensity to outward jumps, while the cap ensures $\mathbf{p}_t \in [0,1]^d$. Away from the boundary, it recovers the transition law in Model 2 whenever $\eta|u_{t,i}| \le s_i$. With $\mathcal{A} = \mathcal{A}_{\text{unif}}$, the SGD route $\mathbf{u}_t = \widehat{\mathbf{g}}_t$, and the one-cell condition $\eta|\widehat{g}_{t,i}| \le s_i$, QAR recovers the DQT stochastic-rounding update (Zhao et al., 2025) on a fixed uniform codebook: coordinate i moves one level in direction $-\operatorname{sign}(\widehat{g}_{t,i})$ with probability $\eta|\widehat{g}_{t,i}|/s_i$, with matching clipping at codebook boundaries. The detailed comparison with DQT and other direct low-precision methods can be found in Appendix A.

The QAR framework specifies how to sample endpoint transitions, but it leaves open how the routing intensity should be distributed within each scale-sharing group. The signal-imbalance

Table 1: Llama-3.2-3B results after 10,000 training steps. QAR-γ uses $\gamma = (8, 1, 0)$ at $(2, 4, 8)$ bits.

Bits	Method	C4 PPL ↓	Wiki PPL ↓	HellaSwag	PIQA	ARC-E	ARC-C	Wino.	BoolQ	OBQA	LAMBADA	Avg. ↑
2	QAT	358.89	779.54	25.32	52.77	26.77	24.83	50.36	38.17	26.00	0.00	30.53
2	QAR-γ	**201.90**	**393.14**	25.56	52.12	28.66	23.81	49.33	49.69	25.20	0.08	**31.81**
4	QAT	**14.15**	**12.13**	70.22	77.15	68.86	42.32	64.40	72.72	37.60	64.22	**62.19**
4	QAR-γ	**14.41**	**12.32**	69.82	76.44	68.60	40.27	67.09	71.74	38.20	64.43	**62.07**
8	QAT	**11.09**	**10.12**	73.83	77.69	72.35	46.25	70.09	73.43	42.40	69.47	**65.69**
8	QAR-γ	**11.06**	**10.12**	73.72	77.97	71.89	46.42	69.85	73.36	42.80	69.77	**65.72**

analysis suggests assigning larger amplification scores to coordinates with stronger feasible gradient magnitudes. For each scale-sharing group G and every $i \in G$, let $s_G = s_i$ denote the common scale within this group and $a_i = M_i|u_i|$ be the feasible gradient magnitude. Define the normalized power score for $\gamma \geq 0$ by

$$b_{\gamma,i} = \begin{cases} \dfrac{a_i^\gamma}{|G|^{-1}\sum_{j\in G} a_j^\gamma}, & \gamma > 0 \text{ and } \sum_{j\in G} a_j^\gamma > 0, \\ 1, & \text{else}, \end{cases} \qquad i \in G,$$

and define the power amplifier

$$\mathcal{A}_{\gamma,i}(\eta, \mathbf{u}, \mathbf{s}, \mathbf{M}) = \frac{\eta M_i |u_i|}{s_G} b_{\gamma,i}, \qquad i \in G. \tag{8}$$

For any $\gamma \geq 0$, we call Algorithm 1 instantiated with $\mathcal{A}_\gamma$ by *QAR-γ*. At $\gamma = 0$, it reduces to the uniform-routing update in (7) by treating coordinates within each scale-sharing group equally; for $\gamma > 0$, we amplify the crossing probability for coordinates with larger gradient magnitude $a_i = M_i|u_i|$.

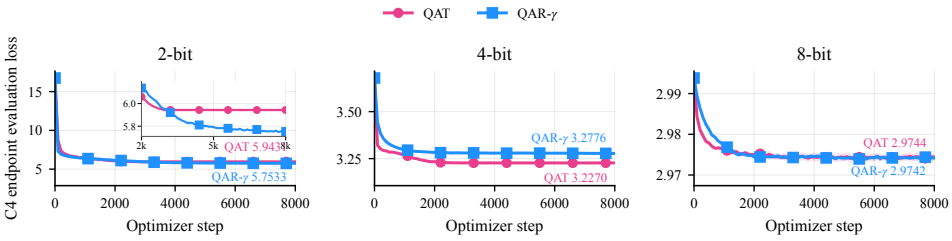


Figure 5: Qwen-2.5-0.5B evaluation-loss curves for the best settings of QAT and QAR-γ at 2, 4, and 8 bits. Curves show the mean over three seeds, and shading shows ± 1 sample standard deviation.

Experimental Results. Figure 5 shows that QAR-γ helps most at 2 bits on Qwen-2.5-0.5B. At higher bits, QAR remains competitive with QAT. The selected amplification is strongest at 2 bits, weaker at 4 bits, and absent at 8 bits. This suggests that favoring stronger update signals is useful when grid jumps are coarse, but offers little benefit on finer grids. The Llama-3.2-3B results in Table 1 show the same trend at a larger scale: QAR-γ performs better at 2 bits on both language-modeling and average downstream metrics, while the differences are small at 4 and 8 bits. Table 2 further shows that the memory advantage grows as the code width decreases. QAR therefore provides its clearest quality and memory benefits under aggressive quantization. Detailed experimental settings and ablations are deferred to Appendix B.

Table 2: Peak CUDA memory cost (MiB) for QAT and QAR-γ. Parentheses show reductions relative to QAT.

	QAT	QAR-γ		
		8-bit	4-bit	2-bit
Qwen-2.5-0.5B	3,238	2,904 (10.3%)	2,733 (15.6%)	**2,648 (18.2%)**
Llama-3.2-3B	17,743	15,100 (14.9%)	13,756 (22.5%)	**13,084 (26.3%)**

4.3 Finite-Grid Feasible-Gradient Bounds for QAR

At a codebook boundary, the full gradient need not vanish because one descent direction can be infeasible. For $q_i = s_i k_i$, define the feasible endpoint gradient by

$$\big(\mathbf{g}_{\mathcal{Q}}(\mathbf{q})\big)_i = g_i \, \mathbf{1}\{k_i - \operatorname{sign}(g_i) \in \mathcal{K}_i\}. \tag{9}$$

It retains exactly the components with a feasible one-cell descent move and equals the full gradient away from the boundary.

Theorem 4 (Feasible-gradient bound for QAR-γ)**.** *Suppose Assumptions 1 and 2 hold. Write $f^*_{\mathcal{Q}} = \min_{\mathbf{q}\in\mathcal{Q}} f(\mathbf{q})$. For each scale-sharing group G with $s_i = s_G$, let $L_G = \max_{i\in G} L_i$. For any $\gamma \geq 0$, run QAR-γ according to Algorithm 1, and let $b_{t,i}$ denote the power score of coordinate i at step t. Assume almost surely that $\eta U b_{t,i} \leq s_G$ for every $t < T$ and $i \in G$.*

If $\mathbf{u}_t = \nabla f(\mathbf{q}_t)$ (GD), then

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\big[\|\mathbf{g}_{\mathcal{Q}}(\mathbf{q}_t)\|_2^2\big] \leq \frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\left[\sum_G \sum_{i\in G} b_{t,i}\big(\mathbf{g}_{\mathcal{Q}}(\mathbf{q}_t)\big)_i^2\right] \leq \frac{2(f(\mathbf{q}_0) - f^*_{\mathcal{Q}})}{\eta T} + \frac{1}{4}\sum_G |G| L_G^2 s_G^2. \tag{10}$$

If $\mathbf{u}_t = \operatorname{sign}(\nabla f(\mathbf{q}_t))$ (signGD), then

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}[\|\mathbf{g}_{\mathcal{Q}}(\mathbf{q}_t)\|_1] \leq \frac{f(\mathbf{q}_0) - f^*_{\mathcal{Q}}}{\eta T} + \frac{1}{2}\sum_G |G| L_G s_G. \tag{11}$$

Remark 1 (Effect of signal imbalance). The first inequality in (10) follows from Lemma 7. It is an equality for $\gamma = 0$ and is strict for $\gamma > 0$ whenever the feasible-gradient magnitudes are unequal within at least one group. This shows how leveraging signal imbalance can accelerate convergence relative to uniform routing.

For comparison, let $\mathbf{y}_{t+1} = \mathbf{y}_t - \eta \nabla f(\mathbf{y}_t)$. For an L-smooth objective, classical nonconvex GD with fixed $0 < \eta \leq 1/L$ satisfies

$$\frac{1}{T}\sum_{t=0}^{T-1} \|\nabla f(\mathbf{y}_t)\|_2^2 \leq \frac{2(f(\mathbf{y}_0) - \inf_y f(y))}{\eta T} = O(T^{-1}).$$

Equation (10) replaces the full gradient by the feasible endpoint gradient and adds a grid floor, while $\eta U b_{t,i} \leq s_G$ ensures that the transition probabilities are not truncated. For gradient update, the bound of QAR decreases like T^{-1} until it reaches the fixed-grid floor $\frac{1}{4}\sum_G |G| L_G^2 s_G^2$. The sign update instead controls the average feasible ℓ_1 gradient by $O(T^{-1})$ plus its grid floor. Both finite-grid floors are unavoidable under the stated assumptions. The proof and a construction showing that these finite-grid floors are tight appear in Appendix E. The stochastic extension is deferred to Appendix F, where we further characterize the tradeoff between exploiting signal imbalance and amplifying stochastic-gradient noise.

5 Conclusion and Limitations

QAT updates a full-precision shadow weight but deploys a quantized endpoint, and the relevant object of analysis is therefore the deployed endpoint trajectory. We show how homogeneous residual model turns endpoint motion into a probabilistic transition on the finite grid and how signal imbalance can create asymmetric oscillations, motivating QAR's update on the quantization code without keeping a full-precision shadow weight. We derive feasible-gradient bounds for QAR with power amplifier and discuss the tradeoff between the gain from signal imbalance and the penalty of strengthening stochastic noise (see Appendix F). Experiments support the endpoint view, especially in low-bit LLM post-training, and show that QAR is comparable to or better than QAT at lower peak memory.

Limitations. This work focuses on weight-only quantization. In this setting, the deployed model lives on a finite weight grid. Our analysis tracks how STE updates move the shadow weights across the cells of this grid. Practical low-bit QAT often quantizes activations as well, where the quantized values depend on the input data and the endpoint can change through both cell crossings and activation-level changes. Extending the finite-grid endpoint view to joint weight-activation QAT is an important direction for future work.

REFERENCES

Yu Bai, Yu-Xiang Wang, and Edo Liberty. ProxQuant: Quantized neural networks via proximal operators. In *Proceedings of the 7th International Conference on Learning Representations (ICLR)*, 2019.

Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint*, arXiv:1308.3432, 2013.

Jeremy Bernstein, Yu-Xiang Wang, Kamyar Azizzadenesheli, and Animashree Anandkumar. signSGD: Compressed optimisation for non-convex problems. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pp. 560–569, 2018.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33 (NeurIPS)*, pp. 1877–1901, 2020.

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 *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10081–10100. Association for Computational Linguistics, 2025.

Jungwook Choi, Zhuo Wang, Swagath Venkataramani, Pierce I-Jen Chuang, Vijayalakshmi Srinivasan, and Kailash Gopalakrishnan. PACT: Parameterized clipping activation for quantized neural networks. *arXiv preprint*, arXiv:1805.06085, 2018.

Matthieu Courbariaux, Yoshua Bengio, and Jean-Pierre David. BinaryConnect: Training deep neural networks with binary weights during propagations. In *Advances in Neural Information Processing Systems 28 (NeurIPS)*, 2015.

Valentin De Bortoli, Alain Durmus, Xavier Fontaine, and Umut Şimşekli. Quantitative propagation of chaos for SGD in wide neural networks. In *Advances in Neural Information Processing Systems 33 (NeurIPS)*, pp. 278–288, 2020.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. QLoRA: Efficient finetuning of quantized LLMs. In *Advances in Neural Information Processing Systems 36 (NeurIPS)*, pp. 10088–10115, 2023.

Steven K. Esser, Jeffrey L. McKinstry, Deepika Bablani, Rathinakumar Appuswamy, and Dharmendra S. Modha. Learned step size quantization. In *Proceedings of the 8th International Conference on Learning Representations (ICLR)*, 2020.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *Proceedings of the 9th International Conference on Learning Representations (ICLR)*, 2021.

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.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

Koen Helwegen, James Widdicombe, Lukas Geiger, Zechun Liu, Kwang-Ting Cheng, and Roeland Nusselder. Latent weights do not exist: Rethinking binarized neural network optimization. In *Advances in Neural Information Processing Systems 32 (NeurIPS)*, pp. 7531–7542, 2019.

Sepp Hochreiter and Jürgen Schmidhuber. Flat minima. *Neural Computation*, 9(1):1–42, 1997.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models. In *Advances in Neural Information Processing Systems 35 (NeurIPS)*, pp. 30016–30030, 2022.

Benoit Jacob, Skirmantas Kligys, Bo Chen, Menglong Zhu, Matthew Tang, Andrew Howard, Hartwig Adam, and Dmitry Kalenichenko. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2704–2713, 2018.

Saqib Javed, Hieu Le, and Mathieu Salzmann. QT-DoG: Quantization-aware training for domain generalization. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, pp. 26981–27004, 2025.

Yiding Jiang, Behnam Neyshabur, Hossein Mobahi, Dilip Krishnan, and Samy Bengio. Fantastic generalization measures and where to find them. In *Proceedings of the 8th International Conference on Learning Representations (ICLR)*, 2020.

Taejong Joo, Wenhan Xia, Cheolmin Kim, Ming Zhang, and Eugene Ie. On surprising effectiveness of masking updates in adaptive optimizers. *arXiv preprint*, arXiv:2602.15322, 2026.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint*, arXiv:2001.08361, 2020.

Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. In *Proceedings of the 5th International Conference on Learning Representations (ICLR)*, 2017.

Junghyup Lee, Dohyung Kim, and Bumsub Ham. Network quantization with element-wise gradient scaling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6448–6457, 2021.

Junghyup Lee, Jeimin Jeon, Dohyung Kim, and Bumsub Ham. Scheduling weight transitions for quantization-aware training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 23466–23475, 2025.

Guoqi Li, Lei Deng, Lei Tian, Haotian Cui, Wentao Han, Jing Pei, and Luping Shi. Training deep neural networks with discrete state transition. *Neurocomputing*, 272:154–162, 2018.

Hanyang Li, Jianhao Ma, and Ying Cui. Understanding quantization-aware training: Gradients at quantized weights bias to the low-loss basin. *arXiv preprint*, arXiv:2606.09012, 2026.

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 (MLSys)*, volume 6, pp. 87–100, 2024.

Zechun Liu, Barlas Oguz, Changsheng Zhao, Ernie Chang, Pierre Stock, Yashar Mehdad, Yangyang Shi, Raghuraman Krishnamoorthi, and Vikas Chandra. LLM-QAT: Data-free quantization aware training for large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 467–484. Association for Computational Linguistics, 2024.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *Proceedings of the 5th International Conference on Learning Representations (ICLR)*, 2017.

Markus Nagel, Marios Fournarakis, Rana Ali Amjad, Yelysei Bondarenko, Mart van Baalen, and Tijmen Blankevoort. A white paper on neural network quantization. *arXiv preprint*, arXiv:2106.08295, 2021.

Markus Nagel, Marios Fournarakis, Yelysei Bondarenko, and Tijmen Blankevoort. Overcoming oscillations in quantization-aware training. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, pp. 16318–16330, 2022.

Mahdi Nikdan, Amir Zandieh, Dan Alistarh, and Vahab Mirrokni. ECO: Quantized training without full-precision master weights. *arXiv preprint*, arXiv:2601.22101, 2026.

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.

David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

Justin Sirignano and Konstantinos Spiliopoulos. Mean field analysis of neural networks: A law of large numbers. *SIAM Journal on Applied Mathematics*, 80(2):725–752, 2020.

Daria Soboleva, Faisal Al-Khateeb, Robert Myers, Jacob R. Steeves, Joel Hestness, and Nolan Dey. SlimPajama: A 627B token cleaned and deduplicated version of RedPajama. https://cerebras.ai/blog/slimpajama-a-627b-token-cleaned-and-deduplicated-version-of-redpajama, 2023.

Kaiyue Wen, Zhiyuan Li, Jason Wang, David Hall, Percy Liang, and Tengyu Ma. Understanding warmup-stable-decay learning rates: A river valley loss landscape view. In *Proceedings of the 13th International Conference on Learning Representations (ICLR)*, pp. 42840–42885, 2025.

Jonathan Wenshøj, Bob Pepin, and Raghavendra Selvan. Oscillations make neural networks robust to quantization. *Transactions on Machine Learning Research*, 2025.

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.

Kaiyan Zhao, Tsuguchika Tabaru, Kenichi Kobayashi, Takumi Honda, Masafumi Yamazaki, and Yoshimasa Tsuruoka. Direct quantized training of language models with stochastic rounding. In *Proceedings of the 17th Asian Conference on Machine Learning (ACML)*, pp. 1150–1165, 2025.

Shuchang Zhou, Yuxin Wu, Zekun Ni, Xinyu Zhou, He Wen, and Yuheng Zou. DoReFa-Net: Training low bitwidth convolutional neural networks with low bitwidth gradients. *arXiv preprint*, arXiv:1606.06160, 2016.

APPENDIX A RELATED WORK

Direct low-precision weight updates. Direct updates of discrete weights predate recent low-bit language-model training. Discrete State Transition (DST) uses probabilistic projection to keep weights in a configurable multilevel discrete space (Li et al., 2018). In the one-cell regime, DST's nonlinear projection can be represented within QAR by a tanh amplifier. Bop instead stores binary weights and an exponential moving average of their gradients, then deterministically flips a weight when the accumulated signal exceeds a threshold and points in the flipping direction (Helwegen et al., 2019). Direct Quantized Training (DQT) applies stochastic rounding to a tentative optimizer update on a low-bit language-model weight grid (Zhao et al., 2025). On a fixed uniform codebook, QAR-0 with the SGD route and $\eta|u_{t,i}| \leq s_i$ recovers its coordinatewise transition law and boundary clipping. ECO removes master weights through an error-feedback loop in optimizer momentum (Nikdan et al., 2026). QAR approaches direct discrete updates from a different starting point: it derives a transition law from QAT's boundary-crossing dynamics and factors that law into a route and an amplifier. This formulation supports signal-dependent power amplification and a finite-codebook analysis in terms of feasible endpoint gradients.

Transition-rate scheduling. One way to regulate endpoint motion in QAT is to control how often shadow weights cross quantization boundaries. Lee et al. (2025) adapt the shadow-weight learning rate to track a prescribed fraction of quantized weights that change levels. QAR instead samples coordinatewise endpoint transitions directly.

Update scaling. Boundary crossings can also be influenced by changing the magnitude or frequency of shadow-weight updates. Lee et al. (2021) use the gradient sign and quantization residual to rescale each STE update, with a Hessian-based rule for adapting the scaling factor. Joo et al. (2026) randomly mask blockwise optimizer updates while retaining dense moment updates and use momentum–gradient alignment to control the masking probability. These approaches modify shadow-weight updates, whereas QAR leverages signal imbalance to assign larger transition probabilities to feasible one-cell endpoint updates with stronger gradient signals.

APPENDIX B EXPERIMENTAL RESULTS

Table B.1: Experimental settings for the Qwen-2.5-0.5B and Llama-3.2-3B runs

setting	Qwen-2.5-0.5B	Llama-3.2-3B
compute resource	1 NVIDIA A40 GPU per run	1 NVIDIA A100 GPU per run
training budget	8000 steps	10,000 steps
sequence length	512	512
quantizer	fixed symmetric per-output-channel uniform weight quantization, initialized by RTN; language-model head excluded	
bit widths	$2, 4, 8$	$2, 4, 8$
scale selection	$s_c = \max_j \lvert W_{cj}\rvert/(2^{b-1}-1)$, fixed during training	
training scope	quantized linear weights only; biases, embeddings, normalizations, and all other parameters frozen	
training data	tokens from SlimPajama-627B (Soboleva et al., 2023)	
batch size	1	4
seeds	$0, 1, 2$	0
evaluation data	C4 validation (Raffel et al., 2020)	C4 and WikiText-2 (Merity et al., 2017) perplexity; eight zero-shot tasks
optimizer	AdamW with QAT $(\beta_1, \beta_2) = (0.9, 0.95)$ and QAR-γ $(\beta_1, \beta_2) = (0.95, 0.9)$, $\epsilon = 10^{-8}$	
weight decay	0	
gradient clipping	0 (disabled)	
learning-rate schedule	no warmup; cosine decay to $0.1\eta_0$ over the first half of the training budget, then held fixed	

B.1 EXPERIMENTAL SETTINGS

We use Qwen-2.5-0.5B and Llama-3.2-3B to evaluate the finite-grid predictions on post-training QAT runs and compare QAT with QAR-γ. For each model, both methods start from the same pretrained checkpoint and run at bit widths $b \in \{2, 4, 8\}$. We quantize linear-layer weights except the language-model head using fixed symmetric per-output-channel uniform quantization initialized by round-to-nearest (RTN). For each output channel c, we set $s_c = \max_j |W_{cj}|/q_{\max}$, where W_{cj} is

the weight in output channel c and input coordinate j, and $q_{\max} = 2^{b-1} - 1$, then round and clip the integer levels to $[-q_{\max}, q_{\max}]$. The scales remain fixed for both methods. We use the same quantizer family at every bit width so that bit width is the only factor that varies across the comparisons. At 2 bits this is coarser than the sub-channel group sizes typically used for 2-bit deployment, and the absolute quality of the 2-bit endpoints is correspondingly low (Table 1). The 2-bit rows should therefore be read as a comparison of endpoint optimization on the coarsest grid rather than as a 2-bit deployment result.

We train both models on SlimPajama-627B (Soboleva et al., 2023) and use C4 validation (Raffel et al., 2020) to evaluate the endpoint. For Llama-3.2-3B, we additionally report WikiText-2 perplexity (Merity et al., 2017) and eight zero-shot tasks. All runs use sequence length 512 and gradient accumulation 1. The Qwen runs use minibatch size 1 for 8000 steps on an NVIDIA A40, whereas the Llama runs use minibatch size 4 for 10,000 steps on an NVIDIA A100.

All runs use AdamW with QAT $(\beta_1, \beta_2) = (0.9, 0.95)$ and QAR-γ $(\beta_1, \beta_2) = (0.95, 0.9)$, $\epsilon = 10^{-8}$, zero weight decay, and no gradient clipping. The learning rate decays without warmup to $0.1\eta_0$ over the first half of the training budget and remains fixed thereafter.

Table B.2: Final C4 endpoint evaluation loss at each method's selected best learning rate. Every entry is the mean $\pm$ sample standard deviation over three seeds (lower is better).

Bits	QAT	QAR-γ	γ
2	5.9431 ± 0.0175	$\mathbf{5.7533 \pm 0.0120}$	16
4	$\mathbf{3.2270 \pm 0.0022}$	3.2776 ± 0.0037	2
8	2.9744 ± 0.0008	2.9742 ± 0.0007	0

Table B.2 reports the final-step values at the selected matched cosine-schedule settings.

B.2 Separate Endpoint Diagnostics

Figure 6 directly compares QAT's shadow and deployed-endpoint losses at the selected best learning rate for each bit width. At 2 and 4 bits, the C4 evaluation loss decreases, whereas the shadow loss increases substantially.

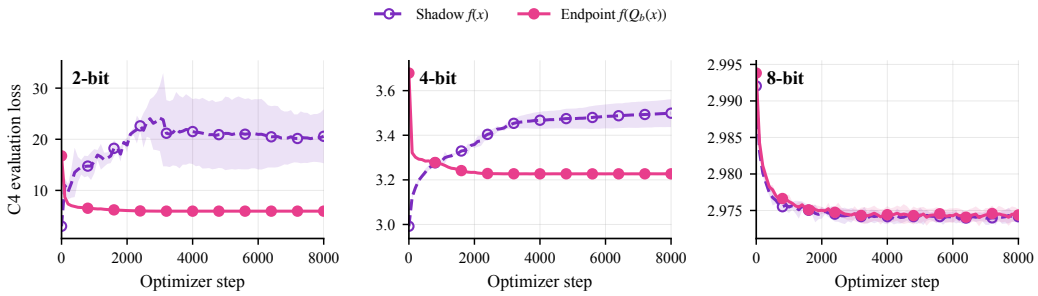


Figure 6: Qwen-2.5-0.5B QAT shadow loss $f(x)$ versus deployed-endpoint loss $f(Q_b(x))$. We plot the mean value across 3 seeds, and the shading is ± 1 sample standard deviation over three seeds at every bit width.

Figure 7 compares the initial and final QAT checkpoints. At initialization, the shadow point has lower loss than the quantized endpoint, especially at 2 bits. After QAT, this ordering is reversed: the endpoint loss falls from 16.75 to 5.92 at 2 bits and from 3.68 to 3.23 at 4 bits, while the corresponding shadow loss rises from 2.99 to 26.68 and 3.47, respectively. Thus QAT can improve the deployed endpoint even as its shadow moves into a higher-loss region, with the separation becoming most pronounced under aggressive quantization.

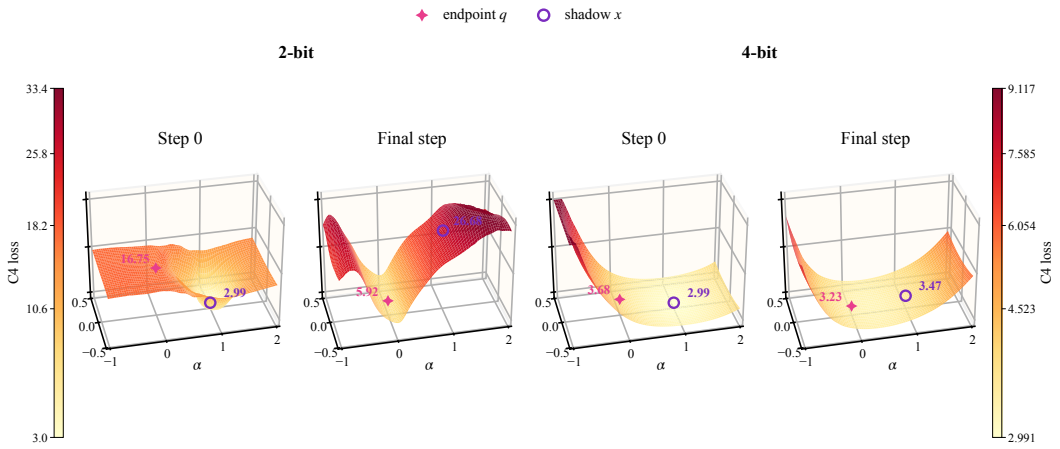


Figure 7: Qwen-2.5-0.5B C4 evaluation-loss landscape at step 0 and at final step for seed 0. The surface evaluates $Q_b(\mathbf{x}) + \alpha(\mathbf{x} - Q_b(\mathbf{x})) + \beta\|\mathbf{x} - Q_b(\mathbf{x})\|_2\mathbf{v}$ for a random unit $\mathbf{v} \perp (\mathbf{x} - Q_b(\mathbf{x}))$.

B.3 Gradient Diagnostics

Figure 8 tracks the average magnitude of active gradient coordinates, $\|g\|_1/\|\operatorname{sign}(g)\|_1$. At the quantized endpoint, both QAT and QAR-γ reduce this quantity, most clearly at 2 bits, showing that both methods remove strong endpoint-gradient signals. In contrast, QAT's shadow-space value can remain much larger at low precision, reinforcing that the shadow trajectory is not a faithful diagnostic of endpoint progress.

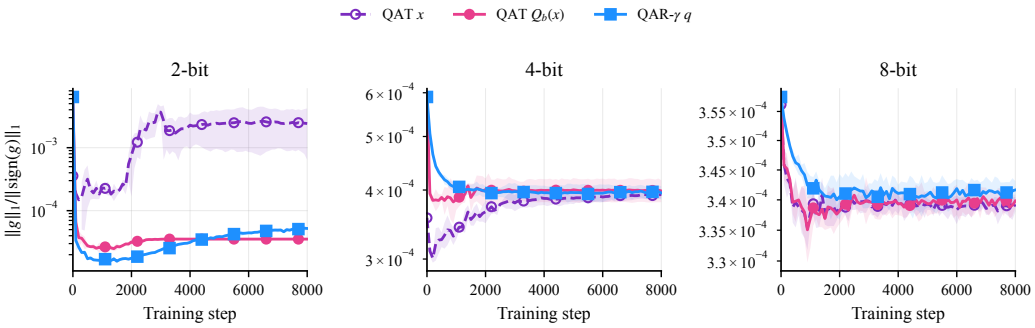


Figure 8: Qwen-2.5-0.5B active-gradient ratio at 2, 4, and 8 bits: QAT at x, QAT at $Q_b(x)$, and QAR-γ at q, with $\gamma = (16, 2, 0)$, respectively. We plot the mean value across 3 seeds, and the shading is ± 1 sample standard deviation over three seeds at every bit width.

B.4 Crossing-Rate Diagnostics

For the realized shadow step from $\mathbf{x}_t$ to $\mathbf{x}_{t+1}$, let $\mathcal{I}_t$ contain the active non-saturated coordinates and define the predicted and observed aggregate crossing rates by

$$\widehat{p}_t = \frac{1}{|\mathcal{I}_t|}\sum_{i\in\mathcal{I}_t}\min\left\{\frac{|(\mathbf{x}_{t+1})_i - (\mathbf{x}_t)_i|}{s_i}, 1\right\}, \qquad \widehat{c}_t = \frac{1}{|\mathcal{I}_t|}\sum_{i\in\mathcal{I}_t}\mathbf{1}\{Q_b(\mathbf{x}_{t+1})_i \neq Q_b(\mathbf{x}_t)_i\}\,.$$

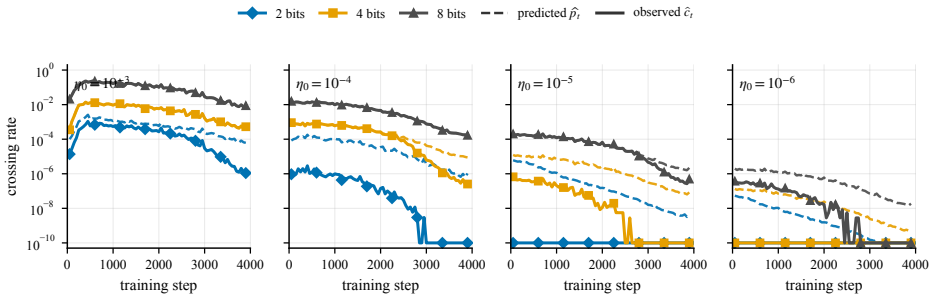


Figure 9: QAT crossing rates on Qwen-2.5-0.5B across four learning rates and 2, 4, and 8 bits. Dashed curves show the prediction $\widehat{p}_t$ and solid curves show the observed rate $\widehat{c}_t$. Zero observations are displayed at 10^{-10}.

Figures 9 and 10 compare predicted or planned crossing rates with realized fixed-grid crossings. Figure 9 sweeps $\eta_0 \in \{10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}\}$ for QAT, while Figure 10 shows the best QAT and QAR-γ settings at 2, 4, and 8 bits. Once the updates become sufficiently small, the realized QAT rate can collapse even though its prediction remains nonzero.

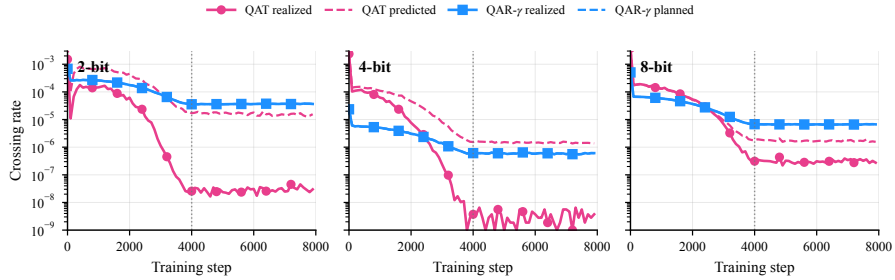


Figure 10: Predicted or planned versus realized crossing rates for the best QAT and QAR-γ settings. Dashed lines are predicted or planned; solid lines are realized.

The late-stage collapse in Figures 9 and 10 does not indicate a failure of the residual-phase model; it comes from BF16 rounding. BF16 values are not evenly spaced, and the gap between adjacent values grows with their magnitude. When an update is smaller than half the local gap around x_i, round-to-nearest maps $x_i + \Delta x_i$ back to x_i, so the shadow weight neither moves nor crosses a quantization boundary. More coordinates enter this regime as the learning rate decays. Figure 11 repeats the diagnostic with FP32 shadow weights to separate this numerical effect from residual-phase calibration.

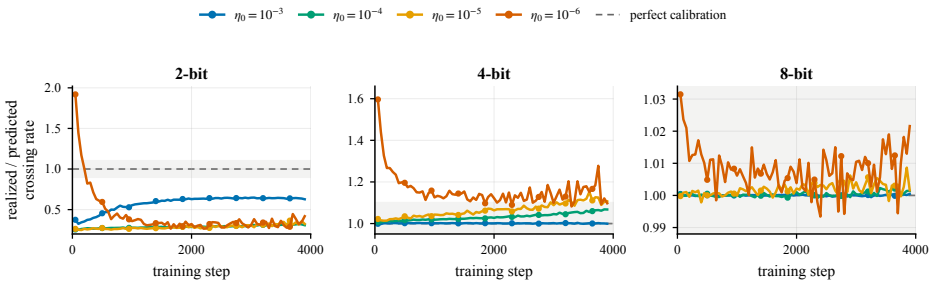


Figure 11: FP32 control for Figure 9. Curves show the ratio of the realized to the predicted boundary-crossing rate for each initial learning rate; the dashed line marks perfect calibration and the shaded band marks $\pm 10\%$.

With FP32, the late-training collapse disappears. At 4 and 8 bits, the realized-to-predicted ratios stay near one or modestly above it. Their gradual upward drift after the initial transient is consistent with the toy-model analysis of Wenshøj et al. (2025), in which a residual-driven STE component pushes shadow weights away from quantization levels and toward cell boundaries. This boundary concentration puts more residual mass in the crossing region than the uniform-phase model predicts, causing the realized crossing rate to exceed the prediction. The 2-bit ratios remain below one, consistent with the center-concentrated phase of its single non-saturated interior cell.

At $\eta_0 = 10^{-6}$, Figure 11 shows an initial spike because the run starts from a BF16 checkpoint. BF16 stores weights on a discrete lattice, placing about 0.1%–0.2% of them exactly on quantization boundaries. Even a tiny first update can move these weights across a boundary, producing more crossings than the continuous phase model predicts. The spike fades once training moves the weights off the BF16 lattice. Thereafter, the realized-to-predicted ratios stay near one or slightly above it at 4 and 8 bits.

QAR does not depend on shadow-weight precision because it samples and applies code transitions directly, without accumulating updates in a shadow weight. It therefore follows the crossing law more faithfully than QAT.

B.5 Memory Diagnostic

Persistent state. Let P be the total number of model parameters, N the number of quantized coordinates updated by both methods, and R the number of quantization groups. Both methods freeze the remaining $P - N$ parameters. If a shadow value, optimizer moment, and group scale each use w, m, and r bytes, then

$$M_{\mathrm{QAT}} = w(P - N) + wN + 2mN + rR, \qquad M_{\mathrm{QAR}} = w(P - N) + \frac{b}{8}N + 2mN + rR,$$

where QAR-γ stores b-bit endpoint codes instead of a shadow tensor. Its persistent-state saving is therefore $(w - b/8)N$ bytes.

Table B.3: Numbers of total parameters P, quantized parameters N, and quantization groups R.

Model	P	N	R
Qwen-2.5-0.5B	494,032,768	357,826,560	304,128
Llama-3.2-3B	3,212,749,824	2,818,572,386	774,144

With BF16 weights and moments and FP32 group scales, the two totals are $2(P - N) + 6N + 4R$ and $2(P - N) + (4 + b/8)N + 4R$. Our implementation stores 2- and 4-bit endpoint codes in packed form and updates them with fused packed CUDA kernels. The resulting persistent-state savings at 2, 4, and 8 bits are 25.9%, 22.2%, and 14.8% on Qwen-2.5-0.5B and 27.9%, 23.9%, and 15.9% on Llama-3.2-3B, respectively.

Peak memory. Peak allocation also depends on transient gradient buffers. In our implementation, we adopt a streamed update in which QAR consumes and releases each layer gradient during backward, while its fused sampler avoids model-shaped temporaries. For a fair comparison, we additionally stream the QAT AdamW update in the same way.

B.6 Learning-Rate Ablation

Figure 12 shows that QAR-γ operates at a systematically smaller learning-rate scale. It is also less sensitive to the learning rate over each displayed sweep.

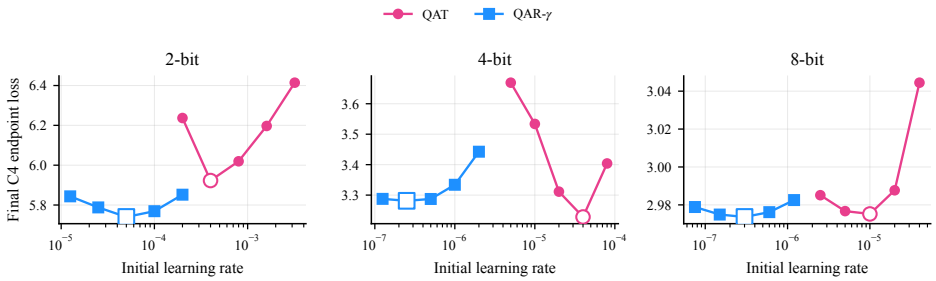


Figure 12: Learning-rate sensitivity. Hollow markers denote the lowest final C4 endpoint loss within each displayed method curve.

B.7 Power-Exponent Ablation

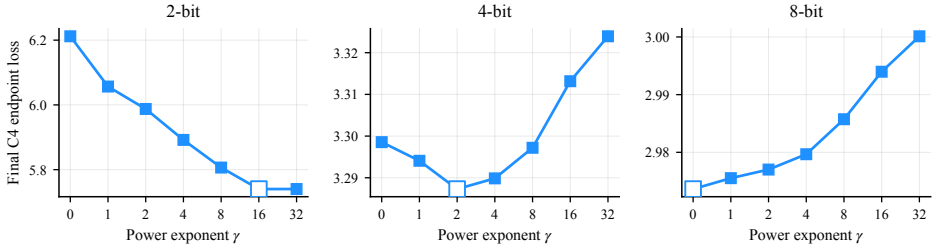


Figure 13: QAR-γ exponent ablation for $(2, 4, 8)$ bits. Hollow markers denote the lowest final C4 endpoint loss in each evaluated grid. Strong amplification helps on the 2-bit grid, moderate amplification is best at 4 bits, and uniform routing ($\gamma = 0$) is best at 8 bits.

Figure 13 gives the selection sweep over $\gamma \in \{0, 1, 2, 4, 8, 16, 32\}$. Final loss is minimized at $\gamma = 16, 2, 0$ for 2, 4, and 8 bits, respectively. Thus stronger signal concentration is useful on the coarsest grid, but its value disappears as the grid becomes finer. This reflects the tradeoff between the signal imbalance and the stochastic noise. Because the stepsizes used here satisfy the one-cell condition $\eta|u_{t,i}| \le s_i$ at every step, the $\gamma = 0$ arm is exactly the DQT update (Section 4.2), so the sweep also compares against a master-weight-free stochastic-rounding baseline.

Appendix C Residual-Phase Distributions

C.1 A High-Resolution Source of Uniform Phases

The main text uses residual phases as a conditional model for endpoint crossings. The following elementary bound gives a sufficient condition under which an unclipped uniform scalar quantizer has nearly uniform residual phases.

Proposition 5 (Bounded-variation phase error)**.** *Let X have density p on $\mathbb{R}$, and let*

$$Q_s(x) = s\left\lfloor \frac{x}{s} + \frac{1}{2} \right\rfloor$$

be nearest-neighbor uniform quantization with step size $s > 0$. Define

$$\theta_s = \frac{X - Q_s(X)}{s} \in \left[-\frac{1}{2}, \frac{1}{2}\right).$$

If p has bounded variation on $\mathbb{R}$, then θ_s has density

$$\pi_s(\theta) = s\sum_{k\in\mathbb{Z}} p\big(s(k+\theta)\big), \qquad \theta \in [-\tfrac{1}{2}, \tfrac{1}{2}),$$

and

$$\sup_{\theta\in[-1/2,1/2)} |\pi_s(\theta) - 1| \le s\,\mathrm{TV}(p).$$

Here $\mathrm{TV}(p)$ *denotes the total variation of* p *on* $\mathbb{R}$*, defined as* $\sup_{\{x_j\}} \sum_j |p(x_{j+1}) - p(x_j)|$*, where the supremum is over finite ordered partitions. If* p *is absolutely continuous, then* $\mathrm{TV}(p) = \int_{\mathbb{R}} |p'(x)|\,\mathrm{d}x$*. Consequently,*

$$d_{\mathrm{TV}}\big(\mathrm{Law}(\theta_s), \mathrm{Unif}[-\tfrac{1}{2}, \tfrac{1}{2})\big) \le \frac{s\,\mathrm{TV}(p)}{2},$$

where $d_{\mathrm{TV}}(\mu, \nu)$ *means the total variation distance between two probability distributions* μ *and* ν*. Moreover, the boundary-window mass*

$$B_s(\rho) = \int_{\frac{1}{2}-\rho}^{\frac{1}{2}} \pi_s(\theta)\,\mathrm{d}\theta$$

satisfies

$$|B_s(\rho) - \rho| \le \rho\, s\,\mathrm{TV}(p), \qquad 0 \le \rho \le 1.$$

For fixed p*, these bounds vanish linearly as* $s \to 0$*. Thus* θ_s *converges in total variation to* $\mathrm{Unif}\big[-\frac{1}{2}, \frac{1}{2}\big)$*, and* $B_s(\rho) \to \rho$ *uniformly for* $0 \le \rho \le 1$*.*

Proof. The residual-phase map folds each cell $\big[s\big(k - \frac{1}{2}\big), s\big(k + \frac{1}{2}\big)\big)$ onto $\big[-\frac{1}{2}, \frac{1}{2}\big)$. On the cell centered at sk, the inverse branch is $x = s(k + \theta)$, so a change of variables contributes $s\,p(s(k + \theta))$ to the density of θ_s. Summing over $k \in \mathbb{Z}$ gives the displayed density. Now fix θ and write $x_k = s(k + \theta)$. The intervals $I_k = \big[x_k - \frac{s}{2}, x_k + \frac{s}{2}\big)$ partition $\mathbb{R}$. Since $\int p(x)\,\mathrm{d}x = 1$,

$$\pi_s(\theta) - 1 = \sum_{k\in\mathbb{Z}} \int_{I_k} \big(p(x_k) - p(x)\big)\,\mathrm{d}x.$$

For each interval I_k,

$$|p(x_k) - p(x)| \le \mathrm{TV}(p; I_k), \qquad x \in I_k,$$

where $\mathrm{TV}(p; I_k) = \sup_{\{y_j\}\subset I_k} \sum_j |p(y_{j+1}) - p(y_j)|$ is the variation of p on I_k, with the supremum over finite ordered partitions of I_k. Hence

$$|\pi_s(\theta) - 1| \le \sum_{k\in\mathbb{Z}} s\,\mathrm{TV}(p; I_k) \le s\,\mathrm{TV}(p).$$

The total-variation and boundary-window bounds follow by integrating this pointwise density bound over intervals of length 1 and ρ, respectively. □

C.2 Normality of Pretrained Weights

This subsection gives additional evidence for the pretrained normal-weight phenomenon for Qwen-2.5-0.5B. We collect linear weight tensors, exclude embeddings and the LM head, rescale each layer, and draw parameter-weighted samples before overlaying the standard-normal density. Figure 14 reports the pooled diagnostic. This plot supports the modeling step used in the main text: after local rescaling, the central pretrained weight mass is well approximated by a centered normal law, which induces the folded residual phase model for finite grids.

C.3 Two-Bit and Folded Residual Phase Distributions

The main text uses residual phases to explain when shadow motion becomes endpoint motion. We now spell out a simple folded-normal model for the pooled residual phase distribution. Let $X \sim \mathcal{N}(0, \tau^2)$ be a scalar pretrained-weight model, where τ is the weight standard deviation. For a symmetric b-bit quantizer with clipping threshold $c > 0$, write

$$K_b = 2^{b-1} - 1, \qquad s_b = \frac{c}{K_b}, \qquad \lambda_b = \frac{\tau}{s_b}.$$

The grid levels are $s_b k$ for integers $-K_b \le k \le K_b$, so $c = K_b s_b$ is the largest positive representable value. Thus λ_b measures the pretrained weight scale in grid-step units. The pooled interior residual phase has density

$$\pi(\theta) = \frac{\sum_{k=-K_b+1}^{K_b-1} \exp\left(-\frac{(k+\theta)^2}{2\lambda_b^2}\right)}{\int_{-1/2}^{1/2} \sum_{k=-K_b+1}^{K_b-1} \exp\left(-\frac{(k+v)^2}{2\lambda_b^2}\right) \mathrm{d}v}, \qquad \theta \in \left[-\frac{1}{2}, \frac{1}{2}\right).$$

At 2 bits, $K_2 = 1$, so the sum contains only the zero cell and $\pi(\cdot)$ is the density of a truncated normal on $\left[-\frac{1}{2}, \frac{1}{2}\right)$. For higher bit widths, the histogram pools many interior cells modulo the grid spacing. With fixed τ and endpoint c, $\lambda_b = K_b \lambda_2$, and the folded sum becomes nearly flat within a cell.

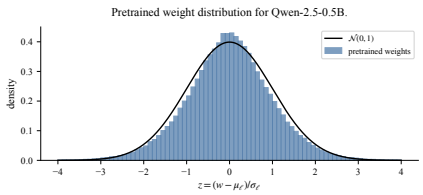


Figure 14: Pooled pretrained weight distribution for Qwen-2.5-0.5B. The plot uses linear weight tensors excluding embeddings and the LM head, standardizes by layer, and draws parameter-weighted samples.

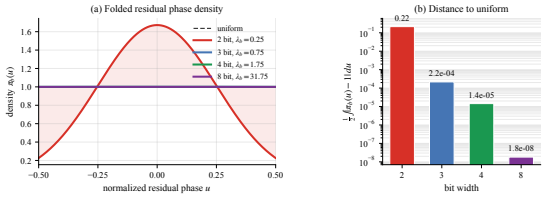


Figure 15: Predicted folded residual phase density $\pi_b(\theta)$ compared with the uniform density on $[-1/2, 1/2)$. The left panel uses $\lambda_2 = 0.25$ and $\lambda_b = K_b \lambda_2$, and the right panel reports the total-variation distance from the uniform model.

Figure 15 makes the scale separation explicit. With the 2-bit fit $\lambda_2 = 0.25$, $\pi(\cdot)$ is center-concentrated and has substantial distance from the uniform cell-phase model. The same absolute weight scale makes the 3-, 4-, and 8-bit folded densities nearly uniform, with total-variation distance dropping by roughly three orders of magnitude already at 3 bits. Figure 2 reports the corresponding empirical Qwen-2.5-0.5B residual histograms at the initial checkpoint.

Figure 16 shows that QAT changes the residual distribution only modestly after the initial adjustment. The dominant difference is across bit widths: 2-bit remains the least uniform, while 4 and 8 bits stay close to uniform.

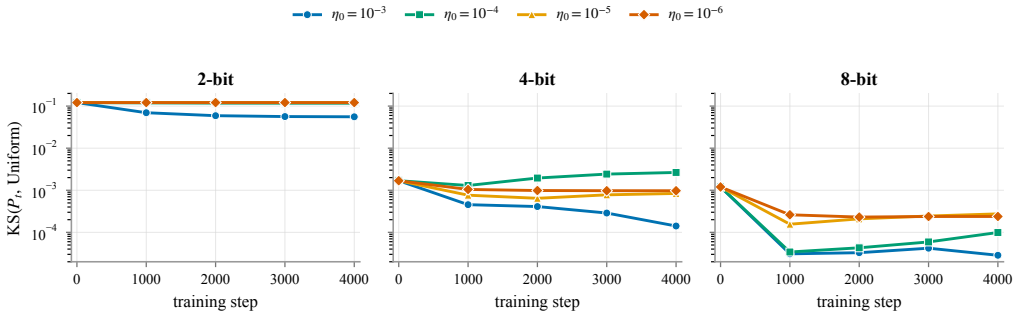


Figure 16: KS distance from the uniform residual-phase model over QAT training steps.

APPENDIX D PROOFS FOR CROSSING AND ENDPOINT MOTION

D.1 PROOF OF LEMMA 1

Proof of Lemma 1. Fix coordinate i. If $u_i > 0$, then

$$x_i^+ = x_i - \eta u_i < x_i.$$

For the fixed coordinatewise uniform quantizer, decreasing x_i cannot increase the endpoint, so

$$q_i^+ = Q_b(\mathbf{x}^+)_i \le Q_b(\mathbf{x})_i = q_i.$$

Hence $q_i^+ - q_i \le 0$, and multiplying by $u_i > 0$ gives $u_i(q_i^+ - q_i) \le 0$. If $u_i < 0$, then $x_i^+ > x_i$, so the same fixed grid ordering gives $q_i^+ \ge q_i$, and again $u_i(q_i^+ - q_i) \le 0$. If $u_i = 0$, the product is zero. If $g_i u_i \ge 0$ coordinatewise, then g_i has the same sign as u_i whenever both are nonzero, so $g_i(q_i^+ - q_i) \le 0$ as well. Summing over coordinates proves the inner-product statement. □

D.2 Proof of Proposition 2

Proof of Proposition 2. Under the stated small-step condition, a crossed active coordinate moves by one neighboring grid level in the direction opposite to u_i, while a non-crossed coordinate does not move. Hence

$$q_i^+ - q_i = -s_i \operatorname{sign}(u_i) J_i.$$

Applying Assumption 2 with $\mathbf{x} = \mathbf{q}$ and $\mathbf{y} = \mathbf{q}^+$ gives

$$f(\mathbf{q}^+) - f(\mathbf{q}) \le \langle \mathbf{g}, \mathbf{q}^+ - \mathbf{q} \rangle + \frac{1}{2} \sum_i L_i (q_i^+ - q_i)^2.$$

Substituting the realized jump representation yields

$$\langle \mathbf{g}, \mathbf{q}^+ - \mathbf{q} \rangle = -\sum_i J_i s_i \operatorname{sign}(u_i) g_i, \qquad (q_i^+ - q_i)^2 = J_i s_i^2,$$

which proves (3). If $g_i u_i \ge 0$ and $J_i = 1$, then $u_i \ne 0$ and $\operatorname{sign}(u_i) g_i = |g_i|$. □

D.3 Crossing probability of a single coordinate

Corollary 6 (One-coordinate crossing probability)**.** *Consider one non-saturated coordinate i with*

$$r_i \in \left[-\frac{s_i}{2}, \frac{s_i}{2} \right), \qquad x_i^+ = x_i - \eta u_i, \qquad \eta |u_i| < s_i.$$

The endpoint crosses only when r_i lies in the boundary window

$$I_i(u_i) = \begin{cases} \left[-\frac{s_i}{2}, -\frac{s_i}{2} + \eta u_i \right), & u_i > 0, \\ \left[\frac{s_i}{2} + \eta u_i, \frac{s_i}{2} \right), & u_i < 0, \\ \varnothing, & u_i = 0. \end{cases}$$

On this event,

$$q_i^+ - q_i = -s_i \operatorname{sign}(u_i).$$

Hence, if $\pi_i(\cdot \mid \mathbf{q})$ is the conditional density of r_i, then the crossing probability is

$$p_i = \int_{I_i(u_i)} \pi_i(v \mid \mathbf{q}) \, \mathrm{d}v, \qquad \mathbb{E}[q_i^+ - q_i \mid \mathbf{q}] = -s_i \operatorname{sign}(u_i) \, p_i. \tag{12}$$

Proof of Corollary 6. Fix coordinate i. If $u_i > 0$, the update moves left. A left crossing occurs precisely when

$$r_i - \eta u_i < -\frac{s_i}{2},$$

which is equivalent to the stated interval. Since one threshold is crossed, the endpoint moves by $q_i^+ - q_i = -s_i$. The case $u_i < 0$ is symmetric: the update moves right, and the crossing condition is

$$r_i - \eta u_i \ge \frac{s_i}{2}.$$

The expectation follows by multiplying the signed jump by its crossing probability. Integrating the conditional density $\pi_i(\cdot \mid \mathbf{q})$ over the crossing interval gives (12). □

D.4 Proof of Theorem 3

Proof of Theorem 3. Model 1 gives crossing indicators satisfying

$$\mathbb{E}[J_i \mid \eta, \mathbf{q}, \mathbf{u}] = p_i(\eta, \mathbf{q}, \mathbf{u}).$$

Under the stated stepsize condition, a crossing in coordinate i has signed jump $-s_i \operatorname{sign}(u_i)$, and no crossing has jump zero. Therefore

$$q_i^+ - q_i = -s_i \operatorname{sign}(u_i) J_i.$$

Collecting these coordinatewise identities proves (4). Moreover, $(q_i^+ - q_i)^2 = s_i^2 J_i$. Applying Assumption 2 to the full endpoint jump gives

$$\begin{aligned}\mathbb{E}[f(\mathbf{q}^+) - f(\mathbf{q}) \mid \eta, \mathbf{q}, \mathbf{u}] &\le \mathbb{E}\left[\left\langle \mathbf{g}, \mathbf{q}^+ - \mathbf{q}\right\rangle + \frac{1}{2}\sum_i L_i (q_i^+ - q_i)^2 \,\middle|\, \eta, \mathbf{q}, \mathbf{u}\right] \\ &= -\sum_i p_i(\eta, \mathbf{q}, \mathbf{u}) s_i \operatorname{sign}(u_i) g_i + \frac{1}{2}\sum_i p_i(\eta, \mathbf{q}, \mathbf{u}) L_i s_i^2.\end{aligned}$$

This proves (5). Under Model 2, $p_i(\eta, \mathbf{q}, \mathbf{u}) = \eta |u_i|/s_i$, so the last display becomes (6). □

Appendix E Power-Amplifier Guarantees

E.1 Proof of Theorem 4

For each scale-sharing group G, at an endpoint $q_i = s_G k_i$, define

$$M_i = \mathbf{1}\{k_i - \operatorname{sign}(u_i) \in \mathcal{K}_i\}, \qquad a_i = M_i |u_i|, \qquad i \in G.$$

$$b_{\gamma,i} = \begin{cases} \dfrac{a_i^\gamma}{|G|^{-1}\sum_{j\in G} a_j^\gamma}, & \gamma > 0 \text{ and } \sum_{j\in G} a_j^\gamma > 0, \\ 1, & \text{else}, \end{cases} \qquad i \in G. \tag{13}$$

Thus the uncapped transition probability is

$$p_i = \frac{\eta a_i b_{\gamma,i}}{s_G}. \tag{14}$$

Lemma 7 (Power-score properties)**.** *For every group G and $\gamma \ge 0$,*

$$\sum_{i\in G} b_{\gamma,i} = |G|, \qquad \sum_{i\in G} b_{\gamma,i} a_i \ge \sum_{i\in G} a_i, \qquad \sum_{i\in G} b_{\gamma,i} a_i^2 \ge \sum_{i\in G} a_i^2. \tag{15}$$

For $\gamma > 0$, both inequalities are strict exactly when the signals in the group are not all equal.

Proof. The claim is immediate for $\gamma = 0$ and for an all-zero group. Otherwise, the normalization in (13) gives $\sum_i b_{\gamma,i} = |G|$. For $r \in \{1, 2\}$, the maps $z \mapsto z^\gamma$ and $z \mapsto z^r$ are nondecreasing on $\mathbb{R}_+$, so the finite-sum Chebyshev inequality gives

$$\frac{1}{|G|}\sum_{i\in G} a_i^{\gamma+r} \ge \left(\frac{1}{|G|}\sum_{i\in G} a_i^\gamma\right)\left(\frac{1}{|G|}\sum_{i\in G} a_i^r\right).$$

Dividing by the first parenthesized factor proves both inequalities in (15). Strictness follows when two signals differ. □

For each group, let $L_G = \max_{i\in G} L_i$. Assumption 2 implies the block majorizer

$$f(\mathbf{y}) \le f(\mathbf{q}) + \langle \nabla f(\mathbf{q}), \mathbf{y} - \mathbf{q}\rangle + \frac{1}{2}\sum_G L_G \|\mathbf{y}_G - \mathbf{q}_G\|_2^2. \tag{16}$$

Proposition 8 (One-step power-amplifier bound)**.** *Suppose the route is the exact gradient and the probabilities are uncapped. For $\mathcal{A}_\gamma$ with any $\gamma \geq 0$, let $b_i = b_{\gamma,i}$. Then*

$$\begin{aligned}\mathbb{E}[f(\mathbf{q}^+) - f(\mathbf{q}) \mid \mathbf{q}] &\leq -\eta \sum_G \sum_{i\in G} b_i a_i^2 + \frac{\eta}{2} \sum_G L_G s_G \sum_{i\in G} b_i a_i \\ &\leq -\frac{\eta}{2} \sum_G \sum_{i\in G} b_i a_i^2 + \frac{\eta}{8} \sum_G |G| L_G^2 s_G^2 \\ &\leq -\frac{\eta}{2} \|\mathbf{g}_{\mathcal{Q}}(\mathbf{q})\|_2^2 + \frac{\eta}{8} \sum_G |G| L_G^2 s_G^2.\end{aligned} \tag{17}$$

Proof. For the exact-gradient route, $a_i = |(\mathbf{g}_{\mathcal{Q}}(\mathbf{q}))_i|$. An accepted jump in coordinate $i \in G$ contributes $-s_G a_i$ to the linear term of (16) and s_G^2 to its squared displacement. Substituting $p_i = \eta a_i b_i / s_G$ gives the first line. For $c_G = L_G s_G$, the scalar inequality

$$-a_i^2 + \frac{c_G}{2} a_i \leq -\frac{1}{2} a_i^2 + \frac{1}{8} c_G^2$$

and $\sum_{i\in G} b_i = |G|$ give the second line. The final line follows from Lemma 7, it is an equality at $\gamma = 0$. □

Proposition 9 (One-step signGD bound)**.** *Suppose $\mathbf{u} = \operatorname{sign}(\nabla f(\mathbf{q}))$ and the probabilities are uncapped. For $\mathcal{A}_\gamma$ with any $\gamma \geq 0$, let $b_i = b_{\gamma,i}$. Then*

$$\begin{aligned}\mathbb{E}[f(\mathbf{q}^+) - f(\mathbf{q}) \mid \mathbf{q}] &\leq -\eta \sum_G \sum_{i\in G} b_i \left|\left(\mathbf{g}_{\mathcal{Q}}(\mathbf{q})\right)_i\right| + \frac{\eta}{2} \sum_G L_G s_G \sum_{i\in G} b_i a_i \\ &\leq -\eta \|\mathbf{g}_{\mathcal{Q}}(\mathbf{q})\|_1 + \frac{\eta}{2} \sum_G |G| L_G s_G.\end{aligned} \tag{18}$$

Proof. Here $a_i = M_i |u_i| \in \{0, 1\}$, and $a_i = 1$ exactly when the corresponding component of $\mathbf{g}_{\mathcal{Q}}(\mathbf{q})$ is nonzero. Substituting $p_i = \eta a_i b_i / s_G$ into (16) gives the first line. For $\gamma = 0$, active coordinates have $b_i = 1$ and $\sum_{i\in G} b_i a_i \leq |G|$. For $\gamma > 0$, if $m_G = \sum_{i\in G} a_i > 0$, every active coordinate has $b_i = |G|/m_G \geq 1$ and $\sum_i b_i a_i = |G|$; when $m_G = 0$, the group does not move. Thus the weighted first-order signal is at least the unweighted feasible-gradient ℓ_1 norm, while the curvature mass is at most $|G|$ in every group. □

Proof of Theorem 4. Since $a_{t,i} = M_{t,i} |u_{t,i}| \leq U$, the condition $\eta U b_{t,i} \leq s_G$ implies $\eta a_{t,i} b_{t,i} / s_G \leq 1$. Hence the coordinatewise minimum in Algorithm 1 is inactive. Therefore Proposition 8 applies to the GD route and Proposition 9 applies to the signGD route at every step. For GD, keep the weighted second line of (17), take total expectation, and sum over $t = 0, \ldots, T-1$. Since $f(\mathbf{q}_T) \geq f_{\mathcal{Q}}^*$, this gives

$$\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\left[\sum_i b_{t,i} \left(\mathbf{g}_{\mathcal{Q}}(\mathbf{q}_t)\right)_i^2\right] \leq \frac{2(f(\mathbf{q}_0) - f_{\mathcal{Q}}^*)}{\eta T} + \frac{1}{4} \sum_G |G| L_G^2 s_G^2.$$

This proves the second inequality in (10); the first follows by applying Lemma 7 within each group before taking expectations. For signGD, the same summation applied to (18) proves (11). □

E.2 Tightness of the finite-grid floors

For arbitrary dimension d and positive L_i, s_i, consider

$$\mathcal{Q} = \prod_{i=1}^d \{-s_i, 0, s_i\}, \qquad f(\mathbf{x}) = \frac{1}{2} \sum_{i=1}^d L_i \left(x_i - \frac{s_i}{2}\right)^2.$$

Its Hessian is $\operatorname{diag}(L_1, \ldots, L_d)$, so the coordinate-smoothness constants are exactly L_i. Every $\mathbf{q} \in \prod_i \{0, s_i\}$ is a global minimizer over $\mathcal{Q}$. Moreover, the descent-sign move toggles coordinate i between 0 and s_i, and hence

$$\mathbf{g}_{\mathcal{Q}}(\mathbf{q}) = \nabla f(\mathbf{q}), \qquad \|\mathbf{g}_{\mathcal{Q}}(\mathbf{q})\|_2^2 = \frac{1}{4} \|\mathbf{L} \odot \mathbf{s}\|_2^2, \qquad \|\mathbf{g}_{\mathcal{Q}}(\mathbf{q})\|_1 = \frac{1}{2} \|\mathbf{L} \odot \mathbf{s}\|_1.$$

With exact gradients and $0 < \eta \le 2/\max_i L_i$, the QAR-0 SGD route moves coordinate i with probability $\eta L_i/2$, so a trajectory initialized in $\prod_i\{0, s_i\}$ remains there. The initial finite-grid optimality gap and noise terms are zero, while the averaged feasible-gradient measure equals the SGD floor for every T. The same trajectory under the QAR-0 signSGD route, with $0 < \eta \le \min_i s_i$, attains its ℓ_1 floor for every T.

The exact construction has tied neighboring minimizers, but the scaling does not rely on exact ties. Replacing $s_i/2$ above by $s_i/2 - \delta_i$, where $0 < \delta_i < s_i/2$, makes $\mathbf{0}$ the unique finite-grid minimizer. On $\prod_i\{0, s_i\}$, for a sufficiently small positive stepsize, the exact-gradient SGD route has coordinate transition probabilities proportional to $L_i(s_i/2-\delta_i)$ and $L_i(s_i/2+\delta_i)$, its stationary feasible-gradient energy is

$$\mathbb{E}_\pi\|\mathbf{g}_{\mathcal{Q}}(\mathbf{q})\|_2^2 = \sum_{i=1}^{d} L_i^2\left(\frac{s_i^2}{4} - \delta_i^2\right),$$

which approaches $\|\mathbf{L}\odot\mathbf{s}\|_2^2/4$ as $\boldsymbol{\delta}\to\mathbf{0}$. In particular, in the tied construction with $L_i = L$ and $s_i = s$, the exact floor is $dL^2s^2/4$, even though $\nabla^2 f = LI_d$ and the objective is globally L-smooth. Thus the linear dimension dependence is unavoidable for the unnormalized feasible-gradient measure under the stated assumptions.

APPENDIX F STOCHASTIC QAR GUARANTEES

To analyze QAR under stochastic gradient noise, we follow Bernstein et al. (2018) and adopt the standard assumption that the mini-batch gradient $\widehat{\mathbf{g}}(\mathbf{q})$ is unbiased with bounded coordinatewise variance.

Assumption 3 (Stochastic gradient noise)**.** *At each endpoint $\mathbf{q}$, the mini-batch gradient is unbiased and has coordinate variance*

$$\mathbb{E}[(\widehat{g}_i(\mathbf{q}) - g_i(\mathbf{q}))^2 \mid \mathbf{q}] \le \sigma_i^2/n.$$

Given $(\mathbf{q}_t, \widehat{\mathbf{g}}_t)$, Algorithm 1 samples the jump variables coordinatewise with probabilities $\mathbf{p}_t$, using fresh routing randomness.

F.1 BASELINE STOCHASTIC QAR BOUNDS

Theorem 10 (Stochastic QAR finite-grid bounds)**.** *Suppose Assumptions 2 and 3 hold, and write $f^*_{\mathcal{Q}} = \min_{\mathbf{q}\in\mathcal{Q}} f(\mathbf{q})$. Fix $\eta > 0$ and an integer $T \ge 1$. For each scale-sharing group G, let $s_i = s_G$ for $i \in G$, $L_G = \max_{i\in G} L_i$, and $\boldsymbol{\sigma}_G = (\sigma_i)_{i\in G}$. For the SGD route $\mathbf{u}_t = \widehat{\mathbf{g}}(\mathbf{q}_t)$, use $\mathcal{A}_\gamma$ with $\gamma = 0$ and suppose $\eta|\widehat{g}_{t,i}| \le s_i$ almost surely. Then*

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\mathbf{g}_{\mathcal{Q}}(\mathbf{q}_t)\|_2^2 \le \frac{2(f(\mathbf{q}_0) - f^*_{\mathcal{Q}})}{\eta T} + \frac{1}{4}\|\mathbf{L}\odot\mathbf{s}\|_2^2 + \frac{\langle \mathbf{L}\odot\mathbf{s}, \boldsymbol{\sigma}\rangle}{\sqrt{n}} + \frac{2\|\boldsymbol{\sigma}\|_2^2}{n}. \tag{19}$$

For the signSGD route $\mathbf{u}_t = \operatorname{sign}(\widehat{\mathbf{g}}(\mathbf{q}_t))$, use $\mathcal{A}_\gamma$ for any $\gamma \ge 0$, with $\eta \le \min_G s_G$ when $\gamma = 0$ and $\eta \le \min_G(s_G/|G|)$ when $\gamma > 0$. Then

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\mathbf{g}_{\mathcal{Q}}(\mathbf{q}_t)\|_1 \le \frac{f(\mathbf{q}_0) - f^*_{\mathcal{Q}}}{\eta T} + \begin{cases} \dfrac{1}{2}\|\mathbf{L}\odot\mathbf{s}\|_1 + \dfrac{2\|\boldsymbol{\sigma}\|_1}{\sqrt{n}}, & \gamma = 0, \\ \dfrac{1}{2}\displaystyle\sum_G |G|L_G s_G + \frac{2}{\sqrt{n}}\sum_G |G|\|\boldsymbol{\sigma}_G\|_1, & \gamma > 0, \end{cases} \tag{20}$$

Proof. Fix $\mathbf{q}_t$, write $\mathbf{g}_t = \nabla f(\mathbf{q}_t)$, $\widehat{\mathbf{g}}_t = \mathbf{g}_t + \boldsymbol{\xi}_t$, and $\mathbf{h}_t = \mathbf{g}_{\mathcal{Q}}(\mathbf{q}_t)$. Also write $\mathbf{z}_t = \operatorname{sign}(\widehat{\mathbf{g}}_t)$. The coordinate mask $M_{t,i}$ keeps only feasible one-cell moves, so the endpoint always remains in $\mathcal{Q}$.

SGD route. For $\mathbf{u}_t = \widehat{\mathbf{g}}_t$, conditioning on $(\mathbf{q}_t, \widehat{\mathbf{g}}_t)$ and applying Assumption 2 gives

$$\mathbb{E}_{\mathbf{J}}[f(\mathbf{q}_{t+1}) - f(\mathbf{q}_t) \mid \mathbf{q}_t, \widehat{\mathbf{g}}_t] \le -\eta\sum_i g_{t,i}\widehat{g}_{t,i}M_{t,i} + \frac{\eta}{2}\sum_i L_i s_i|\widehat{g}_{t,i}|M_{t,i}.$$

We first lower bound the first-order term. For a coordinate with $h_{t,i} \ne 0$, the true descent sign is feasible. If the coordinate is interior, $M_{t,i} = 1$ whenever $\widehat{g}_{t,i} \ne 0$, and hence $g_{t,i}\widehat{g}_{t,i}M_{t,i} = g_{t,i}\widehat{g}_{t,i}$.

If the coordinate is at a boundary, the mask removes only stochastic signs for which $g_{t,i}\widehat{g}_{t,i} \le 0$. Therefore, in both cases,

$$\mathbb{E}[g_{t,i}\widehat{g}_{t,i}M_{t,i} \mid \mathbf{q}_t] \ge g_{t,i}^2 = h_{t,i}^2.$$

For a coordinate with $h_{t,i} = 0$, either $g_{t,i} = 0$, in which case the term is zero, or the true descent sign is infeasible at a boundary. In the latter case $g_{t,i}\widehat{g}_{t,i}M_{t,i}$ is nonzero only after a stochastic sign flip. On that event, $|g_{t,i}| \le |\xi_{t,i}|$ and $|\widehat{g}_{t,i}| \le |\xi_{t,i}|$, so

$$g_{t,i}\widehat{g}_{t,i}M_{t,i} \ge -\xi_{t,i}^2.$$

Using the variance bound, for every coordinate,

$$\mathbb{E}[g_{t,i}\widehat{g}_{t,i}M_{t,i} \mid \mathbf{q}_t] \ge h_{t,i}^2 - \frac{\sigma_i^2}{n}.$$

The curvature term satisfies

$$|\widehat{g}_{t,i}|M_{t,i} \le |h_{t,i}| + |\xi_{t,i}|.$$

Indeed, if $h_{t,i} \neq 0$, this follows from $|\widehat{g}_{t,i}| \le |g_{t,i}| + |\xi_{t,i}|$. If $h_{t,i} = 0$ and $g_{t,i} \neq 0$, the product $|\widehat{g}_{t,i}|M_{t,i}$ can be nonzero only on a sign flip, where $|\widehat{g}_{t,i}| \le |\xi_{t,i}|$. Hence

$$\mathbb{E}[|\widehat{g}_{t,i}|M_{t,i} \mid \mathbf{q}_t] \le |h_{t,i}| + \frac{\sigma_i}{\sqrt{n}}.$$

Taking expectation over $\widehat{\mathbf{g}}_t$ in the one-step bound gives

$$\mathbb{E}[f(\mathbf{q}_{t+1}) - f(\mathbf{q}_t) \mid \mathbf{q}_t] \le -\eta\|\mathbf{h}_t\|_2^2 + \frac{\eta\|\boldsymbol{\sigma}\|_2^2}{n} + \frac{\eta}{2}\langle \mathbf{L}\odot\mathbf{s}, |\mathbf{h}_t|\rangle + \frac{\eta}{2\sqrt{n}}\langle \mathbf{L}\odot\mathbf{s}, \boldsymbol{\sigma}\rangle.$$

By Young's inequality,

$$\frac{1}{2}\langle \mathbf{L}\odot\mathbf{s}, |\mathbf{h}_t|\rangle \le \frac{1}{2}\|\mathbf{h}_t\|_2^2 + \frac{1}{8}\|\mathbf{L}\odot\mathbf{s}\|_2^2.$$

Thus

$$\mathbb{E}[f(\mathbf{q}_{t+1}) - f(\mathbf{q}_t)] \le -\frac{\eta}{2}\mathbb{E}\|\mathbf{h}_t\|_2^2 + \frac{\eta}{8}\|\mathbf{L}\odot\mathbf{s}\|_2^2 + \frac{\eta}{2\sqrt{n}}\langle \mathbf{L}\odot\mathbf{s}, \boldsymbol{\sigma}\rangle + \frac{\eta\|\boldsymbol{\sigma}\|_2^2}{n}.$$

Summing over $t = 0, \dots, T-1$, using $f(\mathbf{q}_T) \ge f_{\mathcal{Q}}^*$, and dividing by $\eta T/2$ proves (19).

signSGD route. Set $a_{t,i} = M_{t,i}|z_{t,i}| \in \{0,1\}$. For $\gamma = 0$, $b_{t,i} = 1$. For $\gamma > 0$, if $m_{t,G} = \sum_{i\in G} a_{t,i} > 0$, then the binary scores give $b_{t,i} = |G|/m_{t,G}$ on active coordinates; if $m_{t,G} = 0$, the group does not move. Hence every active score lies in $[1, w_G]$, where $w_G = 1$ for $\gamma = 0$ and $w_G = |G|$ for $\gamma > 0$, and

$$\sum_{i\in G} b_{t,i}a_{t,i} \le |G|.$$

The stated stepsize conditions therefore keep all probabilities uncapped. Conditioning on $(\mathbf{q}_t, \widehat{\mathbf{g}}_t)$ and applying Assumption 2 gives

$$\mathbb{E}_{\mathbf{J}}[f(\mathbf{q}_{t+1}) - f(\mathbf{q}_t) \mid \mathbf{q}_t, \widehat{\mathbf{g}}_t] \le -\eta\sum_i b_{t,i}g_{t,i}z_{t,i}M_{t,i} + \frac{\eta}{2}\sum_G s_G \sum_{i\in G} L_i b_{t,i}a_{t,i}.$$

For every $i \in G$, a case split over a correct sign, an incorrect sign, and $z_{t,i} = 0$ gives

$$b_{t,i}g_{t,i}z_{t,i}M_{t,i} \ge |h_{t,i}| - 2w_G|g_{t,i}|\mathbf{1}\{z_{t,i} \neq \operatorname{sign}(g_{t,i})\}.$$

This also covers a boundary where the true descent direction is infeasible: then $h_{t,i} = 0$, and a harmful inward move requires a sign error. For $g_{t,i} \neq 0$,

$$\Pr(z_{t,i} \neq \operatorname{sign}(g_{t,i}) \mid \mathbf{q}_t) \le \Pr(|\widehat{g}_{t,i} - g_{t,i}| \ge |g_{t,i}| \mid \mathbf{q}_t) \le \frac{\sigma_i}{\sqrt{n}\,|g_{t,i}|},$$

where the last step uses Markov's inequality on $|\widehat{g}_{t,i} - g_{t,i}|$ and Cauchy–Schwarz with the coordinate variance bound. Coordinates with $g_{t,i} = 0$ contribute zero, so

$$\mathbb{E}\left[\sum_i b_{t,i}g_{t,i}z_{t,i}M_{t,i}\,\middle|\,\mathbf{q}_t\right] \ge \|\mathbf{g}_{\mathcal{Q}}(\mathbf{q}_t)\|_1 - \frac{2}{\sqrt{n}}\sum_G w_G\|\boldsymbol{\sigma}_G\|_1.$$

For the curvature term, $\gamma = 0$ retains the sharper bound $\sum_i L_i s_i$, while $\gamma > 0$ is bounded by $\sum_G |G| L_G s_G$. Taking expectation over the stochastic gradient therefore gives

$$\mathbb{E}[f(\mathbf{q}_{t+1}) - f(\mathbf{q}_t) \mid \mathbf{q}_t] \le -\eta \|\mathbf{h}_t\|_1 + \begin{cases} \frac{\eta}{2}\|\mathbf{L} \odot \mathbf{s}\|_1 + \frac{2\eta\|\boldsymbol{\sigma}\|_1}{\sqrt{n}}, & \gamma = 0, \\ \frac{\eta}{2}\sum_G |G| L_G s_G + \frac{2\eta}{\sqrt{n}}\sum_G |G| \|\boldsymbol{\sigma}_G\|_1, & \gamma > 0. \end{cases}$$

Summing over $t = 0, \dots, T-1$, using $f(\mathbf{q}_T) \ge f^*_{\mathcal{Q}}$, and dividing by ηT proves (20). □

F.2 Noisy power-routing guarantee

Consider the stochastic full-magnitude route $\mathbf{u} = \widehat{\mathbf{g}}(\mathbf{q})$. Write $\widehat{\mathbf{g}} = \mathbf{g} + \boldsymbol{\xi}$, and construct the feasibility mask and power score from the same sampled gradient. For $i \in G$, let

$$\widehat{a}_i = M_i |\widehat{g}_i|, \qquad b_i = b_{\gamma,i}(\widehat{\mathbf{a}}).$$

We restrict this result to the uncapped regime $\eta \widehat{a}_i b_i \le s_G$. If this condition fails, the cap can truncate the largest power scores and the deterministic comparison in Lemma 7 need not be preserved.

The effect of amplification can be summarized by the following net margin:

$$\begin{aligned} \Delta_\gamma(\mathbf{q}) := \mathbb{E}\left[\sum_G \sum_{i \in G} (b_i - 1)\left(\widehat{a}_i^2 - \frac{1}{2} L_G s_G \widehat{a}_i\right) \,\middle|\, \mathbf{q}\right] \\ - \frac{1}{\sqrt{n}} \sum_G \left(\mathbb{E}\left[\sum_{i \in G} (b_i - 1)^2 \widehat{a}_i^2 \,\middle|\, \mathbf{q}\right]\right)^{1/2} \|\boldsymbol{\sigma}_G\|_2. \end{aligned} \tag{21}$$

The first term is the additional first-order signal captured by power routing minus its curvature cost; the second accounts for selecting the routing scores from noisy gradients. At $\gamma = 0$, $b_i = 1$ and $\Delta_0(\mathbf{q}) = 0$.

Theorem 11 (Noisy QAR-γ trade-off)**.** *Suppose Assumptions 2 and 3 hold. Run Algorithm 1 with $\mathbf{u}_t = \widehat{\mathbf{g}}(\mathbf{q}_t)$ and $\mathcal{A}_\gamma$, $\gamma \ge 0$, and assume $\eta \widehat{a}_{t,i} b_{t,i} \le s_G$ almost surely for every $t < T$, every group G, and every $i \in G$. Let $f^*_{\mathcal{Q}} = \min_{\mathbf{q} \in \mathcal{Q}} f(\mathbf{q})$. Then*

$$\begin{aligned} \frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\|\mathbf{g}_{\mathcal{Q}}(\mathbf{q}_t)\|_2^2 \le \frac{2(f(\mathbf{q}_0) - f^*_{\mathcal{Q}})}{\eta T} + \underbrace{\frac{1}{4}\sum_G |G| L_G^2 s_G^2}_{\textit{finite-grid floor}} + \underbrace{\frac{1}{\sqrt{n}}\sum_G L_G s_G \|\boldsymbol{\sigma}_G\|_1 + \frac{2\|\boldsymbol{\sigma}\|_2^2}{n}}_{\textit{stochastic-noise floor}} \\ - \frac{2}{T}\sum_{t=0}^{T-1} \mathbb{E}\Delta_\gamma(\mathbf{q}_t). \end{aligned} \tag{22}$$

Because $\Delta_0 = 0$, the first three terms on the right-hand side form the corresponding groupwise QAR-0 certificate. Power amplification gives a strictly sharper certificate whenever

$$\sum_{t=0}^{T-1} \mathbb{E}\Delta_\gamma(\mathbf{q}_t) > 0.$$

When this quantity is nonpositive, the comparison does not certify an advantage from amplification.

Proof. Condition on $(\mathbf{q}, \widehat{\mathbf{g}})$, and let $v_i = M_i g_i \operatorname{sign}(\widehat{g}_i)$. Let $\mathcal{U}(\mathbf{q}, \widehat{\mathbf{g}})$ denote the one-step upper bound obtained with $b_i = 1$. The block majorizer (16), Bernoulli expectation, and the uncapped probability $p_i = \eta \widehat{a}_i b_i / s_G$ give

$$\begin{aligned} \mathbb{E}_{\mathbf{J}}[f(\mathbf{q}^+) - f(\mathbf{q}) \mid \mathbf{q}, \widehat{\mathbf{g}}] \le \mathcal{U}(\mathbf{q}, \widehat{\mathbf{g}}) - \eta \sum_G \sum_{i \in G} (b_i - 1)\left(\widehat{a}_i^2 - \frac{1}{2} L_G s_G \widehat{a}_i\right) \\ + \eta \sum_G \sum_{i \in G} |b_i - 1| \widehat{a}_i |\xi_i|. \end{aligned} \tag{23}$$

Indeed,

$$|v_i - \widehat{a}_i| = M_i |\operatorname{sign}(\widehat{g}_i)|\,|g_i - \widehat{g}_i| \le |\xi_i|.$$

Conditional Cauchy–Schwarz and Assumption 3 imply

$$\mathbb{E}\left[\sum_G \sum_{i\in G} |b_i - 1|\widehat{a}_i|\xi_i| \,\middle|\, \mathbf{q}\right] \le \frac{1}{\sqrt{n}} \sum_G \left(\mathbb{E}\left[\sum_{i\in G} (b_i-1)^2 \widehat{a}_i^2 \,\middle|\, \mathbf{q}\right]\right)^{1/2} \|\boldsymbol{\sigma}_G\|_2.$$

The coordinate estimates established in the SGD part of Theorem 10 give

$$\mathbb{E}[g_i \widehat{g}_i M_i \mid \mathbf{q}] \ge \big((\mathbf{g}_{\mathcal{Q}}(\mathbf{q}))_i\big)^2 - \frac{\sigma_i^2}{n},$$
$$\mathbb{E}[M_i |\widehat{g}_i| \mid \mathbf{q}] \le |(\mathbf{g}_{\mathcal{Q}}(\mathbf{q}))_i| + \frac{\sigma_i}{\sqrt{n}}.$$

Applying these estimates to $\mathcal{U}$, followed by Young's inequality, yields

$$\begin{aligned}\mathbb{E}[\mathcal{U}(\mathbf{q}, \widehat{\mathbf{g}}) \mid \mathbf{q}] \le\ & -\frac{\eta}{2}\|\mathbf{g}_{\mathcal{Q}}(\mathbf{q})\|_2^2 + \frac{\eta}{8}\sum_G |G| L_G^2 s_G^2 \\ & + \frac{\eta}{2\sqrt{n}} \sum_G L_G s_G \|\boldsymbol{\sigma}_G\|_1 + \frac{\eta \|\boldsymbol{\sigma}\|_2^2}{n}.\end{aligned}$$

Taking conditional expectation in (23) and using (21) therefore gives

$$\begin{aligned}\mathbb{E}[f(\mathbf{q}^+) - f(\mathbf{q}) \mid \mathbf{q}] \le\ & -\frac{\eta}{2}\|\mathbf{g}_{\mathcal{Q}}(\mathbf{q})\|_2^2 + \frac{\eta}{8}\sum_G |G| L_G^2 s_G^2 \\ & + \frac{\eta}{2\sqrt{n}} \sum_G L_G s_G \|\boldsymbol{\sigma}_G\|_1 + \frac{\eta \|\boldsymbol{\sigma}\|_2^2}{n} - \eta \Delta_\gamma(\mathbf{q}).\end{aligned}$$

Summing over $t < T$, using $f(\mathbf{q}_T) \ge f_{\mathcal{Q}}^*$, and dividing by $\eta T/2$ proves (22). □

F.3 Effect of the power exponent

Lemma 12 (Effect of the power exponent)**.** *Fix a nonconstant group of nonnegative magnitudes a_i. For $\gamma > 0$,*

$$\frac{\sum_{i\in G}(b_{\gamma,i} - 1)a_i^2}{\sum_{i\in G}(b_{\gamma,i} - 1)a_i}$$

has a positive denominator and is nondecreasing in γ; it is strictly increasing when the group has at least three distinct magnitudes. Moreover, $\max_i b_{\gamma,i}$ and $\sum_i (b_{\gamma,i} - 1)^2 a_i^2$ are nondecreasing in γ.

Proof. It suffices to treat positive a_i, zero entries follow by continuity. Let $m = |G|$ and $S_r = \sum_i a_i^r$. The ratio in the statement equals

$$\frac{mS_{\gamma+2} - S_2 S_\gamma}{mS_{\gamma+1} - S_1 S_\gamma}.$$

For $0 < \gamma_1 < \gamma_2$, the moment kernel S_{x+y} is totally positive. Indeed, Cauchy–Binet applied to $S_{x+y} = \sum_i e^{x \log a_i} e^{y \log a_i}$ expresses each minor as a sum of products of equally signed exponential Vandermonde determinants. Hence

$$\det\begin{pmatrix} S_0 & S_1 & S_2 \\ S_{\gamma_1} & S_{\gamma_1+1} & S_{\gamma_1+2} \\ S_{\gamma_2} & S_{\gamma_2+1} & S_{\gamma_2+2} \end{pmatrix} \ge 0.$$

Subtracting S_{γ_j}/m times the first row from row $j+1$ shows that this determinant is nonnegative exactly when the displayed ratio at γ_2 is at least its value at γ_1. The determinant is strict with at least three distinct magnitudes; with two distinct values the ratio is constant.

For the remaining claims,

$$\frac{\mathrm{d}}{\mathrm{d}\gamma} \log \max_i b_{\gamma,i} = \log(\max_i a_i) - \frac{\sum_i a_i^\gamma \log a_i}{S_\gamma} \ge 0.$$

Finally, set $Q_\gamma = \sum_i (b_{\gamma,i} - 1)^2 a_i^2$ and $\ell_r = S'_r/S_r$. Direct differentiation gives

$$\frac{Q'_\gamma}{2} = \frac{m^2 S_{2\gamma+2}}{S_\gamma^2}(\ell_{2\gamma+2} - \ell_\gamma) - \frac{m S_{\gamma+2}}{S_\gamma}(\ell_{\gamma+2} - \ell_\gamma).$$

The function ℓ_r is nondecreasing because ℓ'_r is the variance of $\log a_i$ under weights proportional to a_i^r. Also $mS_{2\gamma+2} \geq S_\gamma S_{\gamma+2}$ by the finite-sum Chebyshev inequality. The first coefficient and first parenthesized difference are therefore no smaller than the corresponding second quantities, proving $Q'_\gamma \geq 0$. □

The ratio in Lemma 12 is the first-order signal gained per unit of additional crossing mass. Because the power scores $b_{\gamma,i}$ depart from uniform routing only when the within-group signals are imbalanced, this ratio quantifies the signal imbalance exploited by the amplifier. For each realized group, its deterministic contribution to Δ_γ is positive exactly when the ratio exceeds the one-cell curvature threshold $L_G s_G/2$. Under stochastic gradients, $\Delta_\gamma > 0$ further requires these groupwise gains to dominate the selection-noise penalty in (21). Increasing γ favors larger signals and can strengthen the imbalance-driven gain, but it also tightens the uncapped condition $\eta \widehat{a}_i b_{\gamma,i} \leq s_G$ and increases sensitivity to noise. Thus amplification is beneficial only when the signal imbalance is sufficiently strong relative to both curvature and stochastic noise.

AI USE STATEMENT

In this work, we used generative AI tools to assist with developing the proof of the feasible-gradient bound under stochastic noise and identifying candidate related work. Additionally, we used generative AI tools to refine the wording throughout the paper. We have reviewed all AI-assisted work. The authors independently checked the AI-assisted proof for mathematical correctness and consistency with the stated assumptions, manually verified suggested related work against the original sources, and reviewed and edited all AI-assisted wording to preserve the intended technical meaning and claims. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.