HAYATEQUANT: MORE GEOMETRY PER BIT WITH FOLDED ROTATIONS FOR KV CACHE QUANTIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Compressing the key–value (KV) cache can substantially reduce memory traffic during language-model decoding. However, improved quantization geometry is useful only if it can be realized with low encoding and serving overhead. We investigate whether short-block quantization can improve a data-free cache codec while preserving a compact, fixed-rate representation. We introduce HayateQuant (HQ), which replaces independent scalar assignments with compact joint codebooks. A folded rotation promotes uniform within-block directions, while precomputed candidate lists keep online assignment efficient. To account for the distinct effects of key and value quantization errors, HQ further employs role-specific scaling and key and value bit allocation. Across diverse model families, HayateQuant achieves a favorable quality-compression tradeoff, particularly at moderate rates. Its optimized packed serving implementation also improves decoding throughput over the existing baselines.

1 INTRODUCTION

A key–value (KV) cache lets a language model reuse information from earlier tokens instead of recomputing it at every generation step. This reuse incurs a growing cost: each new token adds key and value vectors at every attention layer. Long contexts therefore increase both the memory occupied by the cache and the data read during decoding (Pope et al., 2023; Kwon et al., 2023). Quantizing these vectors can reduce storage and memory traffic, but the codec itself lies on the inference path: new entries must be encoded as they arrive, and attention must repeatedly read their compressed representations (Shazeer, 2019; Liu et al., 2024). Better compression geometry is useful when these operations remain inexpensive.

Random-rotation quantization provides a natural starting point: an orthogonal transform redistributes large coordinates, and a fixed codebook then maps each coordinate to a low-bit index. TurboQuant (TQ) follows this principle, constructing quantizers from a model-independent source distribution rather than from activations (Zandieh et al., 2026). Its scalar reconstruction path is data-free and simple, storing code indices and one scale per cached vector. Here, *data-free* means that codebooks are fit without model activations, while development experiments choose a single shared configuration.

Scalar assignments, however, impose a geometric constraint: independently encoding coordinates restricts a block's reconstruction points to a Cartesian product. In contrast, a joint codebook can place the same number of reconstruction points directly in the block space, enabling a richer family of quantization cells without increasing the code length (Gersho, 1979; Gray & Neuhoff, 1998). Recent KV-cache quantization methods likewise exploit the geometry of joint blocks (Lee & Kim, 2026; Ann et al., 2026). The key practical challenge is to realize this flexibility under the storage and computational constraints of an online cache. Increasing the block size causes the codebook size to grow exponentially, while exhaustive assignment introduces additional computation at every cache update. Reconstruction error alone is therefore insufficient to determine the appropriate design.

This trade-off motivates the following question:

> *Can short-block quantization improve a KV-cache codec within the code-and-scale budget of scalar quantization while preserving efficient packed serving?*

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

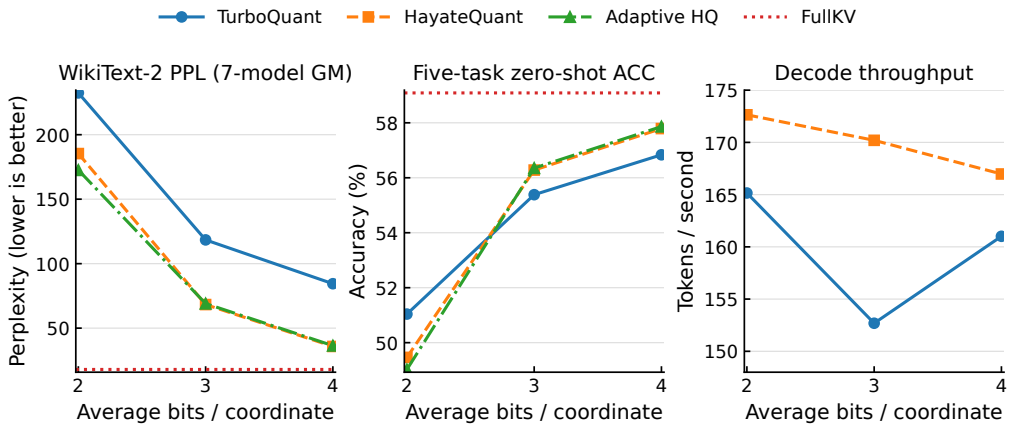

Figure 1. Quality and serving efficiency. Left: geometric-mean WikiText-2 perplexity (PPL) across seven models. Middle: mean task accuracy (ACC). Right: packed-cache decoding throughput on Llama-3.1-8B. FullKV denotes an uncompressed cache; A-HQ is evaluated for quality only. Tables 1 and 2 give the underlying compressed results. Quality and serving protocols are specified separately in Section 4.

We evaluate this trade-off along two dimensions: quality at a fixed logical rate and throughput under packed serving.

We introduce **HY**brid **A**ttention-aware **T**able-based **E**ncoding (**HayateQuant; HQ**), which replaces scalar decisions with compact joint codebooks and prepares their expensive structure offline (Figure 2). Synthetic samples determine the codebooks, within-block rotations are folded into a precomputed transform, and candidate lists replace a full codebook scan with a short online search. These choices keep the code length fixed while avoiding a separate mixing pass and an exhaustive online search.

Keys control attention weights, whereas values supply the content being aggregated. HQ therefore uses role-specific scales and a fixed allocation of the shared bit budget. Adaptive HQ (A-HQ) optionally selects between predefined allocations using causal cache statistics. Our analysis separates the benefits of block codebooks, rotation-induced distributions, and role-specific reconstruction.

Across the evaluated model families, HQ reduces aggregate language-modeling perplexity at every tested rate. At moderate and higher rates, these gains accompany improved average downstream accuracy. Its optimized packed implementation also improves measured decoding throughput over the corresponding TQ controls (Figure 1). Together, the quality and serving results support the central finding: short-block geometry can improve an online KV-cache codec when its representation, assignment procedure, and attention computation are designed together.

2 Background and Problem Setup

What is quantized? For one attention head of positive integer dimension d, let $q \in \mathbb{R}^d$ be the current query and let $k_i, v_i \in \mathbb{R}^d$ be the key and value for cached token $i \in \{1, \dots, S\}$, where $S \geq 1$ is an integer. Attention computes

$$a_i = \frac{\exp(q^\top k_i/\sqrt{d})}{\sum_{j=1}^{S} \exp(q^\top k_j/\sqrt{d})}, \qquad o = \sum_{i=1}^{S} a_i v_i. \tag{1}$$

Keys determine *which* tokens receive attention; values determine *what* is aggregated. KV-cache quantization stores compact representations of k_i and v_i, then uses their reconstructions in this computation. It does not quantize model weights or remove tokens. We describe one head and layer; the codec applies across the cache. The scores $q^\top k_i/\sqrt{d}$ are *logits*, and their exponential normalization is *softmax*. A hat denotes a quantized reconstruction. Vectors are columns, $\langle u, v\rangle = u^\top v$, and $\|\cdot\|$ is the Euclidean norm. For any positive integer n, I_n is the identity matrix and $\mathcal{N}(0, I_n)$ the standard Gaussian law in $\mathbb{R}^n$.

TurboQuant: prepare once. A *codebook* is a shared collection of reconstruction values; an *index* identifies one of them. TurboQuant's mean-squared-error (MSE) variant first prepares a random orthogonal transform $\boldsymbol{W} \in \mathbb{R}^{d\times d}$, with $\boldsymbol{W}^\top \boldsymbol{W} = I_d$, and 2^β scalar levels $\ell_1, \ldots, \ell_{2^\beta}$ for an integer rate $\beta \geq 0$ (Zandieh et al., 2026). A Haar transform is sampled uniformly from the orthogonal group. For any fixed nonzero input independent of that transform, rotation makes its normalized direction uniform on a sphere. The resulting coordinate distribution depends on d, not on model activations. Lloyd–Max fitting alternates nearest-level regions and their mean reconstruction values to fit levels to this known distribution (Lloyd, 1982; Max, 1960).

TurboQuant: encode and decode. For an arriving vector $x \in \mathbb{R}^d \setminus \{0\}$, first rotate and normalize it: $y = \boldsymbol{W}x$, $s_0 = \|y\|/\sqrt{d}$, and $z = y/s_0$. Here s_0 is the root-mean-square (RMS) scale. Next, each coordinate z_j independently selects the nearest level, storing its β-bit index. Let $\hat{c}_j$ be that selected level and $\hat{c} = (\hat{c}_1, \ldots, \hat{c}_d)^\top$. Store the indices and one scale $s = \mathrm{fp16}(s_0)$, where fp16 rounds to a signed 16-bit floating-point value. To decode, look up the levels, restore the scale, and undo the rotation: $\widehat{x} = \boldsymbol{W}^\top(s\hat{c})$. Thus, TQ needs no codebook fitting when a token arrives. This RMS notation rescales TQ's unit-sphere levels by $\sqrt{d}$; with exact scales, it describes the same quantizer. Zero inputs use fixed indices and scale zero.

The TQ path compared here. TQ's inner-product variant reserves one code bit per coordinate for random-projection signs of the residual—the input minus its MSE reconstruction—and stores its norm (Zandieh et al., 2026). Our quality reference is the fixed-length scalar reconstruction path above, without that residual channel. Its normalized vector z lies on the sphere of radius $\sqrt{d}$. For $1 \leq b \leq d$, any b selected coordinates have the *projected-sphere law*; for fixed b, this approaches $\mathcal{N}(0, I_b)$ as d grows. Section 3.1 uses this source for block-codebook fitting.

The storage constraint. Let β_K, β_V be nonnegative integer code bits per coordinate and define the average code rate $\bar{\beta} = (\beta_K + \beta_V)/2$. With one 16-bit scale for each role vector, the logical stored rate is

$$R_{\mathrm{kv}} = \bar{\beta} + \frac{16}{d} \quad \text{bits per coordinate, averaged over keys and values.} \tag{2}$$

Each token and head therefore stores $d\beta_K + d\beta_V + 32$ bits for its key–value pair. Codebooks, rotations, and search tables are shared persistent state, not repeated per token. A *packed* cache stores indices in their low-bit fields and reads them directly during attention. Kx/Vy means $\beta_K = x$ and $\beta_V = y$. For fixed HQ, reallocating code bits preserves this rate when their sum and the scale format are unchanged.

3 HayateQuant

Relative to TQ's independent scalar assignments, HQ retains data-free preparation and the code-and-scale interface, but combines joint block codebooks with folded mixing, candidate-table search, and attention-specific scaling and bit allocation. Figure 2 shows the encoding and serving path. We describe each component below.

3.1 Short Blocks at an Unchanged Code Rate

Choose a positive integer block size b dividing d, and an ordered codebook $\mathcal{C}_{b,\beta} = \{c_1, \ldots, c_K\} \subset \mathbb{R}^b$ with $K = 2^{b\beta}$ representatives. Each representative is a *codeword*. Write $z^{(m)} \in \mathbb{R}^b$ for block $m \in \{1, \ldots, d/b\}$ of the normalized vector. The exact reference assignment and reconstruction are

$$i_m = \min \arg\min_{1\leq i\leq K} \left\| z^{(m)} - c_i \right\|^2, \qquad \widehat{z}^{(m)} = c_{i_m}, \qquad \widehat{x} = \boldsymbol{W}^\top(s\widehat{z}). \tag{3}$$

The minimum-index rule resolves ties. There are d/b indices, each requiring $b\beta$ bits, so their total remains $d\beta$ bits. One scale s is fitted to the full vector, so grouping changes the reconstruction points but neither the code length nor the scale count. The role-specific refinement below can update the initial codes. For $x = 0$, canonical index-one assignments and a zero scale reconstruct zero exactly.

A scalar codebook's Cartesian product is feasible for block coding, giving the following finite-rate guarantee.

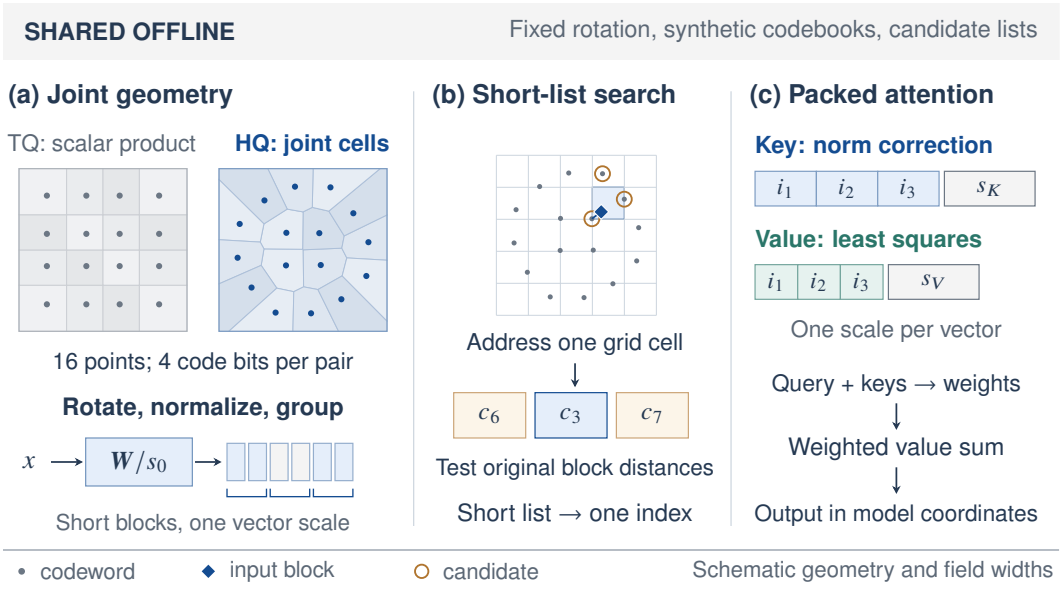

Figure 2. HayateQuant at a fixed code budget. (a) Illustrative scalar-product and joint codebooks, each with 16 points; lines show nearest-codeword cells. (b) A grid cell selects candidates; distances use the original block (diamond). (c) Fixed HQ stores indices and one scale per role vector for packed attention. Rotations, codebooks, and tables are shared offline. Sections 3.3 and 3.5 describe value reassignment and optional A-HQ.

Theorem 3.1 (Finite-rate product inclusion)**.** *For integers $b \geq 1$ and $\beta \geq 0$, let $D_b^*(\beta)$ be the infimum of per-coordinate MSE for $\mathcal{N}(0, I_b)$ over nonempty codebooks with at most $2^{b\beta}$ points. Then*

$$D_b^*(\beta) \leq D_1^*(\beta). \tag{4}$$

At the same code rate, joint codebooks can match or improve the best scalar-product distortion.

Appendix B.1 proves this class-level guarantee, separately from fitting and candidate search. With $b = 1$ and matched encoding conventions, HQ recovers the scalar codec itself.

Fitting without model activations. HQ fits codebooks by Lloyd iterations on synthetic projected-sphere samples or their Gaussian approximation, using multiple restarts. The fitted codebooks are shared across models. Let $P_{d,b}$ be the law of b coordinates of a uniform vector on the sphere of radius $\sqrt{d}$ in $\mathbb{R}^d$. For a source law P with finite second moment, define $D_P(\mathcal{C}) = \mathbb{E}_{Z\sim P} \min_{c\in\mathcal{C}} \|Z - c\|^2$, where $\mathbb{E}$ denotes expectation.

Theorem 3.2 (Source approximation for a fixed codebook)**.** *For integers $1 \leq b \leq d$ and any nonempty finite codebook $\mathcal{C} \subset \mathbb{R}^b$,*

$$\left| \sqrt{\frac{D_{P_{d,b}}(\mathcal{C})}{b}} - \sqrt{\frac{D_{\mathcal{N}(0,I_b)}(\mathcal{C})}{b}} \right| \leq \sqrt{\frac{2}{d}}. \tag{5}$$

Thus, the same codebook has similar root MSE under the spherical and Gaussian sources as the head dimension grows. Appendix B.5 proves the bound before candidate truncation or scale fitting. The spherical source is exact for global Haar rotation; for the hybrid below, both sources serve as model-independent fitting distributions.

3.2 Folded Rotations and Table-Based Assignment

Preparing block directions offline. Assume d is a power of two. Let $\boldsymbol{H}_d$ be the unnormalized Sylvester–Hadamard matrix, with entries in $\{-1, +1\}$ and $\boldsymbol{H}_d\boldsymbol{H}_d^\top = dI_d$. Let $\boldsymbol{D}_\sigma =$

$\mathrm{diag}(\sigma_1, \ldots, \sigma_d)$, with independent signs satisfying $\Pr(\sigma_j = 1) = \Pr(\sigma_j = -1) = 1/2$, and draw independent Haar matrices $Q_m \in O(b)$, also independent of those signs. HQ's hybrid transform is

$$\boldsymbol{W}_0 = \mathrm{bdiag}(Q_1, \ldots, Q_{d/b}) \frac{\boldsymbol{H}_d \boldsymbol{D}_\sigma}{\sqrt{d}}. \tag{6}$$

Here $O(b)$ is the group of orthogonal $b \times b$ matrices, and bdiag forms their block-diagonal matrix. The Hadamard factor spreads energy, and each Q_m mixes one block. Conditional on the signs and a nonzero block, its direction is uniform on the block's sphere, but its radius remains input-dependent (Appendix B.2). Uniform block directions alone do not determine a global Haar or Gaussian law. This distributional statement concerns an input independent of the offline randomness; deployment holds the transform fixed.

Materializing the product in equation 6 offline gives one dense transform, with the same arithmetic structure as the dense-rotation reference and no separate block-mixing pass. HQ chooses globally between dense Haar and hybrid transforms according to each role's rate; the evaluated choice is specified in Section 4.1. Appendix B.3 gives the projection-folding identities.

Replacing exhaustive search by a candidate table. Exact block search compares against all K representatives. HQ instead partitions a bounded region of $\mathbb{R}^b$ into a Cartesian grid and precomputes a list of τ codeword indices for each cell, with integer $1 \le \tau \le K$. At a cache update, the grid address selects a candidate list; distances from the original block to those candidates determine the stored index. Clipping is used only to form the address, not to replace the block in the distance calculation.

With G grid cells per coordinate, where $G \ge 1$ is an integer, the table stores $O(G^b \tau)$ indices and the online search costs $O(b\tau)$ arithmetic operations per block under constant-time table access. The cost is independent of K when b and τ are fixed, at the expense of persistent table memory. The ordered assignment is exact when the list includes the minimum-index nearest codeword (Appendix B.4). Short lists therefore decouple online search from codebook size.

3.3 Attention-Aware Scales and Bit Allocation

Why distinguish the two roles? Equation 1 is linear in the values but nonlinear in the keys. For nonnegative error bounds ϵ_K, ϵ_V and a fixed query, changing each value by Euclidean distance at most ϵ_V changes the output by at most ϵ_V. If only keys change, by at most ϵ_K each, the perturbed output $\widehat{o}$ satisfies

$$\|\widehat{o} - o\| \le \frac{\|q\|\, \epsilon_K}{\sqrt{d}} \inf_{c \in \mathbb{R}^d} \max_i \|v_i - c\|. \tag{7}$$

Theorem B.16 proves both inequalities in Appendix B.7. Value errors enter a weighted sum directly; key errors act through attention weights, with sensitivity governed by the query norm and value spread.

One scale, chosen by role. For $y \neq 0$ and the nonzero concatenation $\hat{c} = \widehat{z}$ of assigned codewords, HQ uses the norm-correcting (NC) and least-squares (LS) scales

$$s_{\mathrm{nc}} = \frac{\|y\|}{\|\hat{c}\|}, \qquad s_{\mathrm{ls}} = \frac{\langle y, \hat{c} \rangle}{\|\hat{c}\|^2}. \tag{8}$$

NC preserves the vector norm before scale rounding. LS minimizes $\|y - s\hat{c}\|^2$ over real s for fixed codes. Keys use NC; values use one LS refinement: fit a scale, reassign using the scaled input, and refit the scale to the resulting codes. The stored scale is signed fp16. A fixed numerical tolerance handles zero or near-zero denominators as specified in Appendix A.3.

If θ is the angle between y and $\hat{c}$, the difference in *total squared error* before rounding is

$$\|y - s_{\mathrm{nc}}\hat{c}\|^2 - \|y - s_{\mathrm{ls}}\hat{c}\|^2 = \|y\|^2 (1 - \cos\theta)^2. \tag{9}$$

For small angular error, preserving the key norm costs only fourth-order additional squared error. Values instead use LS to minimize reconstruction error for their assigned codes. Appendix B.6 proves the identity and separates the contribution of scale rounding.

Sharing the bit budget. At average rate $\bar{\beta}$, fixed HQ uses a predefined key-heavy split $(\bar{\beta}+\delta_{\bar{\beta}}, \bar{\beta}-\delta_{\bar{\beta}})$, with $0 \le \delta_{\bar{\beta}} \le \bar{\beta}$ and integer role rates. The same split is used across models and tasks at each rate. A high-resolution proxy, $a_K 4^{-\beta_K} + a_V 4^{-\beta_V}$, models how each role's distortion decreases with its rate. The positive weights a_K, a_V combine distortion and sensitivity; the continuous optimum is key-heavy when $a_K > a_V$. Appendix B.8 derives the constrained optimum. This proxy motivates allocation; the experiments assess the fixed finite-rate recipe.

3.4 Packed Attention in Rotated Coordinates

The reconstruction in equation 3 defines the quantized vector; it need not be materialized for every cached token. For fixed HQ, let $\boldsymbol{W}_K$ and $\boldsymbol{W}_V$ be the shared key and value transforms. For token i and role $r \in \{K, V\}$, let $s_{r,i}$ be the stored scale, $\widehat{z}_{r,i}$ the concatenated codewords, and $u_{r,i} = s_{r,i}\widehat{z}_{r,i}$. Attention on the reconstructed cache can be written as

$$\widehat{a}_i = \frac{\exp((\boldsymbol{W}_K q)^\top u_{K,i}/\sqrt{d})}{\sum_{j=1}^{S} \exp((\boldsymbol{W}_K q)^\top u_{K,j}/\sqrt{d})}, \qquad \widehat{o} = \boldsymbol{W}_V^\top \sum_{i=1}^{S} \widehat{a}_i u_{V,i}. \tag{10}$$

This is exact attention on $\widehat{k}_i = \boldsymbol{W}_K^\top u_{K,i}$ and $\widehat{v}_i = \boldsymbol{W}_V^\top u_{V,i}$ in real arithmetic (Proposition B.6); it adds no quantization approximation.

The query is transformed once per head and decode step; cached indices select codewords as needed. Values are aggregated in shared rotated coordinates, avoiding a separate inverse transform per cached value. A fixed value transform and its inverse can be absorbed into the adjacent projections. Key and query transforms follow positional transformations; no commutation is assumed.

The serving path therefore uses the compressed entries without constructing a full uncompressed cache. Section 4.3 measures this integrated computation.

3.5 Optional Causal Adaptation

At integer average rates, Adaptive HQ chooses between the fixed HQ split and a symmetric split with the same code-bit sum. Before quantizing an update, it estimates causal mean power

$$\widehat{\sigma}_r^2 = \frac{1}{n_r d} \sum_{j=1}^{n_r} \|x_{r,j}\|^2, \qquad r \in \{K, V\}, \tag{11}$$

from raw vectors $x_{r,j} \in \mathbb{R}^d$ available up to and including the current update, with positive integer count n_r. This second moment is *causal*: it uses no future vectors. For a candidate split (t_K, t_V), it evaluates

$$\widehat{J}(t_K, t_V) = \lambda_K \widehat{\sigma}_K^2 D_K(t_K) + \lambda_V \widehat{\sigma}_V^2 D_V(t_V), \tag{12}$$

where $D_r(t)$ is the precomputed per-coordinate distortion estimate on the synthetic source for role r at code rate t, and $\lambda_r > 0$ is a fixed sensitivity weight. The symmetric split is selected only when its proxy improves on the default by a fixed safety margin; ties retain fixed HQ. Quantization runs once after selection. Both candidates have the same code-bit sum. Section 4.2 evaluates this causal allocation through model quality at matched code rates.

4 Experiments

We evaluate quality at a fixed logical cache budget and efficiency in packed serving, then examine the rotation and allocation choices. The quality and serving protocols below address these distinct parts of the research question.

4.1 Common Settings

Models and codecs. Table 1 covers seven models spanning the Llama, Qwen, and DeepSeek-R1 distill families. We compare the TQ scalar reconstruction reference, fixed HQ, and A-HQ at average code rates of 2, 3, and 4 bits per coordinate.

Table 1. Seven-model quality comparison. PPL is full WikiText-2 test perplexity; ACC is the five-task zero-shot primary-score mean in percent (`acc_norm` for ARC-Easy, ARC-Challenge, PIQA, and HellaSwag; `acc` for WinoGrande). Aggregate PPL is a geometric mean and aggregate ACC an arithmetic mean. Bits are average code bits; R1 abbreviates DeepSeek-R1-Distill and A-HQ denotes Adaptive HQ. Best compressed values, including displayed ties, are bold.

		PPL ↓			ACC ↑		
model	bits	TQ	HQ	A-HQ	TQ	HQ	A-HQ
Llama-2-7B	2	**6.079**	8.151	8.145	**64.9**	59.8	59.5
	3	**5.060**	5.071	5.072	**68.0**	67.5	67.8
	4	4.937	**4.934**	4.935	**68.7**	68.5	68.1
Llama-3.1-8B	2	**10.654**	11.141	11.130	**68.0**	66.4	65.9
	3	6.947	**6.821**	6.826	72.9	**73.0**	72.8
	4	6.582	6.556	**6.554**	73.5	**73.9**	73.8
Qwen3-4B	2	117.543	46.900	**39.031**	49.5	**50.0**	49.9
	3	17.103	**12.121**	12.647	62.8	**65.8**	**65.8**
	4	**11.767**	12.095	12.214	66.5	**67.8**	**67.8**
Qwen3-8B	2	18.114	18.836	**17.788**	**61.9**	57.6	56.9
	3	9.894	**8.993**	9.203	68.8	69.2	**69.3**
	4	8.948	**8.673**	8.706	70.4	**71.2**	**71.2**
R1-Llama-8B	2	**45.618**	46.790	46.839	**40.6**	40.4	40.2
	3	24.934	**23.089**	23.128	43.6	**46.4**	46.0
	4	23.188	22.632	**22.606**	**46.8**	46.6	46.3
R1-Qwen-1.5B	2	37926.974	**26744.607**	28326.832	35.9	**36.1**	35.8
	3	31255.779	**12881.925**	13345.570	35.9	**36.1**	35.6
	4	17950.156	**3866.156**	4254.030	35.8	35.6	**36.5**
R1-Qwen-7B	2	153393.349	74901.519	**54570.953**	**36.4**	35.9	34.9
	3	70447.547	6170.956	**6020.934**	35.7	35.9	**37.1**
	4	21495.516	**254.027**	256.554	36.0	40.9	**41.4**
Aggregate	2	232.388	185.365	**172.566**	**51.0**	49.5	49.0
Aggregate	3	118.423	**68.282**	69.057	55.4	56.3	**56.4**
Aggregate	4	84.442	**35.803**	36.412	56.8	57.8	**57.9**

Shared HQ configuration. All heads have $d = 128$; HQ uses $b = 2$, NC keys, and one LS reassignment with a final refit for values. Its fixed splits are K3/V1, K4/V2, and K5/V3. A role uses a dense Haar rotation at up to two code bits per coordinate and the folded hybrid transform above that rate. This configuration is shared across the model grid. For fixed HQ and the scalar quality reference, one fp16 scale per role gives logical rates of 2.125, 3.125, and 4.125 bits per coordinate. Appendix A collects the codebook sizes and adaptive candidates.

Quality metrics. Language-modeling quality is measured by perplexity (PPL) on the complete WikiText-2 raw test split (Merity et al., 2017), using bfloat16 (a 16-bit floating-point format) for model computation, 2048-token windows, and a stride of 1024. PPL is the exponentiated mean negative log likelihood, so lower values indicate better next-token prediction. Downstream quality is the equally weighted zero-shot mean over ARC-Easy, ARC-Challenge, PIQA, HellaSwag, and WinoGrande, reported as ACC in percent (Clark et al., 2018; Bisk et al., 2020; Zellers et al., 2019; Sakaguchi et al., 2020). Zero-shot evaluation provides no task demonstrations. We use Language Model Evaluation Harness (lm-eval) templates. It reports `acc_norm`, accuracy after answer-length normalization of log likelihoods, for the first four tasks, and `acc`, ordinary accuracy, for WinoGrande (Gao et al., 2023). Higher ACC is better. Across models, PPL is aggregated geometrically and ACC arithmetically. The uncompressed reference, FullKV, uses the same quality protocol.

Table 2. Packed serving throughput. Median of three vLLM runs on one B200. E2E denotes end-to-end throughput, including prefill (processing the input prompt); decode throughput covers generation. Comparisons are at the same nominal code rate under the serving settings in Section 4.3. Higher is better; the best value at each rate is bold.

bits	method	E2E tok/s	decode tok/s
2	TQ	153.359	165.154
2	HQ	**159.373**	**172.641**
3	TQ	142.601	152.687
3	HQ	**157.692**	**170.198**
4	TQ	149.835	161.010
4	HQ	**154.466**	**166.975**

4.2 Quality at a Fixed Logical Budget

This comparison tests fixed HQ against the scalar reference at matched code-and-scale rates. A-HQ evaluates causal allocation at the same code rates. Table 1 reports every model and rate; its two metric groups show PPL and ACC, and its final rows summarize the model grid. Figure 1 visualizes these aggregates. Appendix C provides FullKV values and complementary summaries of the per-model changes.

HQ lowers aggregate PPL at every evaluated rate. At 3 and 4 bits, the improvement extends across most models, while the largest relative reductions occur in the R1-Qwen rows. The downstream average also improves at these two rates, connecting the language-modeling gains to multiple-choice task performance. A-HQ provides a further reduction in aggregate PPL at the lowest rate and achieves the highest aggregate ACC at the two higher rates. These results support joint coding at a fixed budget and show the additional quality benefit of causal allocation at selected rates.

The moderate-rate PPL improvement also extends to the non-R1 subset. Appendix C reports this subset and per-model relative changes; the geometric-mean aggregate weights models equally on a log scale.

4.3 Packed-Cache Serving Efficiency

This experiment measures the cost of block quantization in the online decoding loop. The cache remains packed: update kernels write compressed entries and attention reads them directly.

We benchmark fixed HQ and TQ using vLLM 0.24 on one NVIDIA B200, with Llama-3.1-8B, fp16 serving tensors, batch size one, a 4096-token prompt, and 128 generated tokens. Both paths quantize all 32 layers. TQ uses native presets at rates 3 and 4 and a packed scalar plugin at rate 2. The comparison uses matched nominal code rates.

Table 2 reports end-to-end and decode tokens per second. HQ is faster in both measures at all three rates, with the largest separation at the intermediate rate. The gain therefore holds for the integrated codec, including cache updates and attention over packed entries.

4.4 Why Distinguish Rotations and Roles?

These development comparisons determine the shared rotation and allocation rules. Each varies one design choice while holding the other codec components fixed; the seven-model evaluation then tests the resulting recipe.

Rotation. The purpose is to distinguish coordinate mixing from its effect on language-modeling quality. Table 3 compares global Hadamard, hybrid folded, and dense Haar transforms on Llama-3.1-8B at two rates. The paired columns show the key-side MSE diagnostic (KV-MSE) and WikiText-2 PPL, both with lower values preferred. The hybrid yields the lowest reconstruction error at both rates. PPL favors dense Haar at the lower rate and the hybrid at the higher rate, supporting the rate-dependent rotation rule.

Table 3. Rotation design check. Llama-3.1-8B, with the remaining codec components fixed within this development comparison. KV-MSE denotes the key-side mean-squared reconstruction-error diagnostic; PPL is WikiText-2 perplexity. Lower values are better.

	2 bits		3 bits	
rotation	KV-MSE	PPL	KV-MSE	PPL
global Hadamard	0.1095	10.161	0.0321	7.097
hybrid folded	**0.0983**	9.123	**0.0290**	**7.013**
global dense Haar	0.0996	**8.915**	0.0300	7.034

Bit allocation. The purpose is to test how a fixed total code rate should be divided between keys and values. Table 4 compares symmetric, key-heavy, and value-heavy splits on two development models. The key-heavy split gives the lowest PPL in both rows, supporting one shared allocation rule rather than a separate split for each main-grid model. Together with the rotation comparison, this motivates the fixed recipe evaluated in Table 1: joint assignments with a rate-conditioned rotation and a key-heavy allocation.

Table 4. Allocation at three average code bits. Equal code budgets; the remaining codec is fixed within each comparison. Lower PPL is better.

model	K3/V3	K4/V2	K2/V4
Llama-3.1-8B	7.020	**6.933**	8.444
Qwen3-8B	8.800	**8.648**	9.715

5 Related Work

Rotation and quantization geometry. QuIP and QuIP# exploit incoherence and structured codebooks for weight quantization (Chee et al., 2023; Tseng et al., 2024); QuaRot and SpinQuant develop fixed and learned rotations for low-bit inference (Ashkboos et al., 2024; Liu et al., 2025). TQ uses model-independent rotation for online quantization (Zandieh et al., 2026). HQ replaces its scalar decisions with short-block assignments, whose larger codebook class is motivated by classical vector-quantization theory (Zador, 1963; Gersho, 1979; Gray & Neuhoff, 1998).

Vector codecs for the KV cache. FibQuant fits shared radial–angular codebooks to the spherical source induced by rotation and supports fine-grained rates (Lee & Kim, 2026). BlockQuant likewise exploits block-spherical geometry and studies reconstruction and inner-product distortion (Ann et al., 2026). Both are closely related vector codecs. HQ's focus is the combined realization of short-block coding, folded transforms, table assignment, role-specific scales, and packed serving. We evaluate this design against TQ's scalar reference.

Complementary cache-compression approaches. KIVI, KVQuant, ZipCache, and GEAR use asymmetric grouping, calibration, salient-token treatment, or low-rank correction (Liu et al., 2024; Hooper et al., 2024; He et al., 2024; Kang et al., 2024). KVarN studies scale errors during autoregressive decoding, while UltraQuant, SAW-INT4, and OSCAR address practical low-bit serving constraints (Muller et al., 2026; Chakrabarti et al., 2026; Jia et al., 2026; Zhou et al., 2026). Eviction methods instead reduce the number of retained tokens (Zhang et al., 2023; Xiao et al., 2024; Li et al., 2024). HQ instead compresses retained vectors, using data-free codebooks and a globally shared configuration.

6 Conclusion

HayateQuant combines short-block codebooks, folded rotations, candidate search, and role-specific scales and allocation. It improves aggregate language-modeling quality at every tested rate and increases measured packed-serving throughput. These results answer our research question: richer quantization geometry can improve a data-free codec within its code-and-scale budget when the representation and its online computation are designed together.

Reproducibility Statement

The codec is specified in Section 3; Appendix B states the assumptions and proofs supporting its mathematical claims. Experimental configurations and reconstruction conventions are given in Section 4 and Appendix A.

Statement of AI Use

The research question, core ideas, proposed method, and experiments were developed by the author. A large language model assisted with drafting and editing the manuscript, including mathematical exposition and figure preparation, and with checking arguments and references. It was not used to generate experimental results. The author is responsible for the manuscript's content and claims.

References

Heesang Ann, Joongkyu Lee, and Min-hwan Oh. Block-sphere vector quantization. *arXiv preprint arXiv:2605.19972*, 2026. doi: 10.48550/arXiv.2605.19972. URL https://arxiv.org/abs/2605.19972.

Saleh Ashkboos, Amirkeivan Mohtashami, Maximilian L. Croci, Bo Li, Pashmina Cameron, Martin Jaggi, Dan Alistarh, Torsten Hoefler, and James Hensman. QuaRot: Outlier-free 4-bit inference in rotated LLMs. *Advances in Neural Information Processing Systems*, 37, 2024. URL https://arxiv.org/abs/2404.00456.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. PIQA: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 7432–7439, 2020. doi: 10.1609/aaai.v34i05.6239.

Inesh Chakrabarti, David Limpus, Aditi Ghai Rana, Bowen Bao, Spandan Tiwari, Thiago Crepaldi, and Ashish Sirasao. UltraQuant: 4-bit KV caching for context-heavy agents. *arXiv preprint arXiv:2606.20474*, 2026. doi: 10.48550/arXiv.2606.20474. URL https://arxiv.org/abs/2606.20474.

Jerry Chee, Yaohui Cai, Volodymyr Kuleshov, and Christopher M De Sa. QuIP: 2-bit quantization of large language models with guarantees. *Advances in Neural Information Processing Systems*, 36: 4396–4429, 2023.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018. doi: 10.48550/arXiv.1803.05457. URL https://arxiv.org/abs/1803.05457.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation. Zenodo, December 2023. URL https://zenodo.org/records/10256836. Version v0.4.0.

Allen Gersho. Asymptotically optimal block quantization. *IEEE Transactions on Information Theory*, 25(4):373–380, 1979.

Robert M. Gray and David L. Neuhoff. Quantization. *IEEE Transactions on Information Theory*, 44 (6):2325–2383, 1998.

Yefei He, Luoming Zhang, Weijia Wu, Jing Liu, Hong Zhou, and Bohan Zhuang. ZipCache: Accurate and efficient KV cache quantization with salient token identification. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Coleman Hooper, Sehoon Kim, Hiva Mohammadzadeh, Michael W. Mahoney, Yakun Sophia Shao, Kurt Keutzer, and Amir Gholami. KVQuant: Towards 10 million context length LLM inference with KV cache quantization. *Advances in Neural Information Processing Systems*, 37, 2024.

Jinda Jia, Jisen Li, Zhongzhu Zhou, Jung Hwan Heo, Jue Wang, Tri Dao, Shuaiwen Leon Song, Ben Athiwaratkun, Chenfeng Xu, Tianyi Zhang, and Xiaoxia Wu. SAW-INT4: System-aware 4-bit KV-cache quantization for real-world LLM serving. *arXiv preprint arXiv:2604.19157*, 2026. doi: 10.48550/arXiv.2604.19157. URL https://arxiv.org/abs/2604.19157.

Hao Kang, Qingru Zhang, Souvik Kundu, Geonhwa Jeong, Zaoxing Liu, Tushar Krishna, and Tuo Zhao. GEAR: An efficient KV cache compression recipe for near-lossless generative inference of LLM. *arXiv preprint arXiv:2403.05527*, 2024. URL https://arxiv.org/abs/2403.05527.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th ACM Symposium on Operating Systems Principles*, pp. 611–626, 2023. doi: 10.1145/3600006.3613165.

Namyoon Lee and Yongjune Kim. FibQuant: Universal vector quantization for random-access KV-cache compression. *arXiv preprint arXiv:2605.11478*, 2026. doi: 10.48550/arXiv.2605.11478. URL https://arxiv.org/abs/2605.11478.

Yuhong Li, Yingbing Huang, Bowen Yang, Bharat Venkitesh, Acyr Locatelli, Hanchen Ye, Tianle Cai, Patrick Lewis, and Deming Chen. SnapKV: LLM knows what you are looking for before generation. *Advances in Neural Information Processing Systems*, 37, 2024.

Zechun Liu, Changsheng Zhao, Igor Fedorov, Bilge Soran, Dhruv Choudhary, Raghuraman Krishnamoorthi, Vikas Chandra, Yuandong Tian, and Tijmen Blankevoort. SpinQuant: LLM quantization with learned rotations. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2405.16406.

Zirui Liu, Jiayi Yuan, Hongye Jin, Shaochen Zhong, Zhaozhuo Xu, Vladimir Braverman, Beidi Chen, and Xia Hu. KIVI: A tuning-free asymmetric 2bit quantization for KV cache. In *International Conference on Machine Learning*, pp. 32332–32344. PMLR, 2024.

Stuart Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2): 129–137, 1982.

Joel Max. Quantizing for minimum distortion. *IRE Transactions on Information Theory*, 6(1):7–12, 1960.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *International Conference on Learning Representations*, 2017.

Lorenz K. Muller, Philippe Bich, Chiara Boretti, Hyun-Min Chang, Jiawei Zhuang, and Lukas Cavigelli. KVarN: Variance-normalized KV-cache quantization mitigates error accumulation in reasoning tasks. *arXiv preprint arXiv:2606.03458*, 2026. doi: 10.48550/arXiv.2606.03458. URL https://arxiv.org/abs/2606.03458.

Reiner Pope, Sholto Douglas, Aakanksha Chowdhery, Jacob Devlin, James Bradbury, Jonathan Heek, Kefan Xiao, Shivani Agrawal, and Jeff Dean. Efficiently scaling transformer inference. *Proceedings of Machine Learning and Systems*, 5, 2023. URL https://proceedings.mlsys.org/paper_files/paper/2023/hash/c4be71ab8d24cdfb45e3d06dbfca2780-Abstract-mlsys2023.html.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. WinoGrande: An adversarial winograd schema challenge at scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 8732–8740, 2020. doi: 10.1609/aaai.v34i05.6399.

Noam Shazeer. Fast transformer decoding: One write-head is all you need. *arXiv preprint arXiv:1911.02150*, 2019. URL https://arxiv.org/abs/1911.02150.

Albert Tseng, Jerry Chee, Qingyao Sun, Volodymyr Kuleshov, and Christopher De Sa. QuIP#: Even better LLM quantization with Hadamard incoherence and lattice codebooks. In *International Conference on Machine Learning*, 2024. URL https://arxiv.org/abs/2402.04396.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *International Conference on Learning Representations*, 2024.

Paul Laszlo Zador. *Development and evaluation of procedures for quantizing multivariate distributions*. PhD thesis, Stanford University, 1963.

Amir Zandieh, Majid Daliri, Majid Hadian, and Vahab Mirrokni. TurboQuant: Online vector quantization with near-optimal distortion rate. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/5c802ef38ab6e366c2ea06eee554c088-Abstract-Conference.html.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800, 2019. doi: 10.18653/v1/P19-1472.

Zhenyu Zhang, Ying Sheng, Tianyi Zhou, Tianlong Chen, Lianmin Zheng, Ruisi Cai, Zhao Song, Yuandong Tian, Christopher Ré, Clark Barrett, et al. H2O: Heavy-hitter oracle for efficient generative inference of large language models. *Advances in Neural Information Processing Systems*, 36, 2023.

Zhongzhu Zhou, Donglin Zhuang, Jisen Li, Ziyan Chen, Shuaiwen Leon Song, Ben Athiwaratkun, and Xiaoxia Wu. OSCAR: Offline spectral covariance-aware rotation for 2-bit KV cache quantization. *arXiv preprint arXiv:2605.17757*, 2026. doi: 10.48550/arXiv.2605.17757. URL https://arxiv.org/abs/2605.17757.

A Evaluation and Codec Conventions

This appendix records the operating points and reconstruction conventions used with the common settings in Section 4.1. The development comparisons are presented in Section 4.4.

A.1 Fixed Operating Points

Table 5. Evaluated operating points. The block size is $b = 2$ and the head dimension is $d = 128$. A-HQ chooses between the two listed allocations; its second candidate is fixed HQ. Keys use NC and values use one LS reassignment followed by a refit. The final column gives fixed HQ's logical code-and-scale rate from equation 2.

average bits	HQ K/V bits per coordinate	HQ K/V bits per block	A-HQ block candidates	largest codebook	stored bits per coordinate
2	K3/V1	K6/V2	K4/V4, K6/V2	2^6	2.125
3	K4/V2	K8/V4	K6/V6, K8/V4	2^8	3.125
4	K5/V3	K10/V6	K8/V8, K10/V6	2^{10}	4.125

A.2 Metric Aggregation

For model perplexities $p_1, \ldots, p_M > 0$, the aggregate is $\exp(M^{-1} \sum_j \log p_j)$, with $M = 7$. For two codecs, its log ratio is the mean of their per-model log ratios. Consequently, each model contributes equally on the log scale. ACC first averages the five task scores within each model, then averages the model scores with equal weights. Appendix C supplies the uncompressed reference and complementary summaries of the same results.

A.3 Reconstruction Conventions

Encoder and decoder share the ordered codebook, rotation, block size, rates, and numerical conventions. Exact search breaks ties by the smallest global codeword index; candidate search applies the same order within its list. A zero input stores index one for each block and scale zero.

Let $\epsilon > 0$ be the fixed numerical tolerance. A concatenated codeword vector $\hat{c}$ with $\|\hat{c}\| \le \epsilon$ uses scale zero. Otherwise, keys use NC. Values first fit LS; if $|s_{\mathrm{ls}}| > \epsilon$, they reassign the blocks of y/s_{ls} and refit LS to the new codes, again using scale zero for a near-zero concatenation. If $|s_{\mathrm{ls}}| \le \epsilon$, the initial fit is retained without division. Thus, the nonfallback path has two scale fits separated by one reassignment. The final scale is rounded to signed fp16. The mathematical results separate exact real-arithmetic scales from their finite rounded values.

A.4 Serving and Storage Accounting

Fixed HQ writes packed code indices and one signed fp16 scale per role vector. Attention consumes this representation directly, as described in Section 3.4; fixed value rotations are absorbed into the adjacent projections. Codebooks, rotations, and candidate tables are persistent shared state, separate from the per-token budget.

Serving compares fixed HQ with TQ at matched nominal code rates using the paths in Section 4.3. Table 2 reports medians of three repetitions with the same configuration and generated-token sequence. A-HQ is evaluated in the quality comparison.

B Mathematical Foundations

All statements use exact real arithmetic unless rounding is explicit. Codebooks are nonempty finite ordered sets; nearest-neighbor ties use that order. Randomized statements concern a fixed input independent of the sampled codec randomness. The rotation is held fixed after offline preparation. For vectors, $\|u\|_1 = \sum_j |u_j|$ and $\|u\|_\infty = \max_j |u_j|$; tr denotes the matrix trace, and Var denotes variance. All dimensions are positive integers. Unless a base is shown, $\log$ is the natural logarithm.

B.1 Scalar Containment and Finite-Rate Block Geometry

Proposition B.1 (Containment of the scalar reference)**.** *Let $b = 1$. Suppose the scalar and block paths share their ordered codebook, orthogonal transform, normalization, exact assignment, scale rule, reconstruction, and zero-input convention. Their indices and reconstructions coincide for every input. Their stored bits also coincide if arithmetic operations, rounding, and packing are identical and executed in the same order.*

Proof. At a nonzero input, both paths form the same normalized vector. Each one-dimensional block is a scalar coordinate, so both solve the same ordered nearest-neighbor problem. Their indices coincide, and the shared scale and reconstruction give the same vector. The zero case follows from the common convention. The additional implementation hypotheses make every arithmetic and serialization operation identical, which proves the bitwise claim. □

For a positive integer b, a nonnegative integer β, and $Z_b \sim \mathcal{N}(0, I_b)$, define the optimal per-coordinate distortion

$$D_b^*(\beta) = \inf_{1 \le |\mathcal{C}| \le 2^{b\beta}} \frac{1}{b} \mathbb{E} \min_{c \in \mathcal{C}} \|Z_b - c\|^2 . \tag{13}$$

The infimum is over finite codebooks in $\mathbb{R}^b$ and is finite and nonnegative.

Theorem B.2 (Product inclusion at finite rate)**.** *For positive integers b_1, b_2 with $b_1 \mid b_2$ and every nonnegative integer β, $D_{b_2}^*(\beta) \le D_{b_1}^*(\beta)$. In particular, $D_b^*(\beta) \le D_1^*(\beta) \le D_\beta$, where D_β is the Gaussian distortion of any specified scalar codebook with at most 2^β points.*

Proof. Write $b_2 = mb_1$. Given $\eta > 0$, choose a b_1-dimensional codebook with at most $2^{b_1\beta}$ points and per-coordinate distortion at most $D_{b_1}^*(\beta) + \eta$. Its m-fold Cartesian product contains at most $2^{b_2\beta}$ points. Squared distance to that product is the sum of squared distances to its factors. Each Gaussian subblock has law $\mathcal{N}(0, I_{b_1})$, so the product has the same per-coordinate distortion as the chosen factor. Taking the infimum in dimension b_2 and then $\eta \downarrow 0$ proves the first claim. Set $b_1 = 1$ for the scalar case. The specified scalar codebook is feasible for $D_1^*(\beta)$, giving the last inequality. □

The scalar case proves Theorem 3.1: joint coding enlarges the feasible reconstruction geometry at fixed rate. The guarantee concerns the codebook class; scalar-codec equivalence additionally uses the matched operations in Proposition B.1.

B.2 Hybrid Rotations: Direction and Block Energy

Lemma B.3 (Conditional block directions)**.** *Let d be a power of two, $b \mid d$, and $x \in \mathbb{R}^d$ be fixed. Let $\boldsymbol{D}_\sigma = \mathrm{diag}(\sigma)$ contain independent uniform signs and let $Q_m \in O(b)$ be independent Haar matrices, independent of those signs. Set $u = \boldsymbol{H}_d \boldsymbol{D}_\sigma x/\sqrt{d}$ and let $u^{(m)}$ be block m. The hybrid matrix in equation 6 is orthogonal. Conditional on the signs, each nonzero block has uniform direction $Q_m u^{(m)} / \left\|u^{(m)}\right\|$ on the unit sphere in $\mathbb{R}^b$; these directions are conditionally independent across blocks.*

Proof. Both factors of equation 6 are orthogonal. Conditional on the signs, $u^{(m)}$ is deterministic. For every $V \in O(b)$, Haar invariance gives $VQ_m \overset{d}{=} Q_m$. Hence $Q_m u^{(m)}$ has fixed norm and a rotation-invariant law, the uniform law on that sphere. Independence follows from independence of the Q_m. A zero block stays zero and has no defined direction. □

Proposition B.4 (Signed-Hadamard block energy)**.** *Under the preceding assumptions, let $P_I \in \mathbb{R}^{b\times d}$ select any b distinct coordinates, and set $E_I = \left\| P_I \boldsymbol{H}_d \boldsymbol{D}_\sigma x/\sqrt{d} \right\|^2$. Then*

$$\mathbb{E}_\sigma E_I = \frac{b}{d}\|x\|^2, \qquad \mathrm{Var}_\sigma(E_I) \le 2\frac{b}{d}\|x\|_\infty^2 \|x\|^2. \tag{14}$$

For $x \neq 0$, the standard deviation divided by the mean is at most $\sqrt{2d/b}\ \|x\|_\infty / \|x\|$.

Proof. Let $B = d^{-1}\boldsymbol{H}_d^\top P_I^\top P_I \boldsymbol{H}_d$. Orthogonality of the selected normalized rows gives $B^2 = B = B^\top$, and every diagonal entry equals b/d. With $D_x = \mathrm{diag}(x)$ and $A = D_x B D_x$, we have $E_I = \sigma^\top A \sigma$ and $\mathbb{E}E_I = \mathrm{tr}\, A = (b/d)\|x\|^2$. Moreover,

$$E_I - \mathrm{tr}\, A = 2\sum_{j<k} A_{jk}\sigma_j\sigma_k.$$

Distinct unordered pairs have zero cross expectation, including those sharing one index. Therefore $\mathrm{Var}(E_I) = 4\sum_{j<k} A_{jk}^2 \le 2\|A\|_F^2$, where $\|\cdot\|_F$ is the Frobenius norm. Finally,

$$\|D_x B D_x\|_F^2 \le \|x\|_\infty^2 \|BD_x\|_F^2 = \|x\|_\infty^2 \,\mathrm{tr}(D_x B^2 D_x) = \frac{b}{d}\|x\|_\infty^2\|x\|^2.$$

Combining the inequalities proves the variance bound. Dividing its square root by the positive mean proves the relative bound. □

The two results separate uniform within-block directions from input-dependent block energy. They do not identify the hybrid source with the projected-sphere law below; equal coordinate second moments alone are insufficient for that conclusion.

B.3 Folding and Packed Attention

Proposition B.5 (Materializing a fixed transform)**.** *Let A and B be the normalized signed-Hadamard and block-diagonal factors in equation 6, so that $\boldsymbol{W}_0 = BA$. For every $x \in \mathbb{R}^d$, the materialized matrix $\boldsymbol{W}_0 = BA$ satisfies $\boldsymbol{W}_0 x = B(Ax)$. For S input vectors, dense application uses $O(Sd^2)$ arithmetic operations and $O(d^2)$ persistent matrix entries, with no subsequent block-mixing pass.*

Proof. The identity is associativity of matrix multiplication. The stored matrix has d^2 entries, and its dense application evaluates Sd inner products of length d. The factor B is already included in those products. □

Proposition B.6 (Attention in fixed rotated coordinates)**.** *Let $\boldsymbol{W}_K, \boldsymbol{W}_V \in O(d)$ be fixed. For $S \ge 1$, $q \in \mathbb{R}^d$, and any $u_{K,i}, u_{V,i} \in \mathbb{R}^d$, define $\widehat{k}_i = \boldsymbol{W}_K^\top u_{K,i}$ and $\widehat{v}_i = \boldsymbol{W}_V^\top u_{V,i}$. Attention on these reconstructions equals equation 10. Furthermore, if $v = Ah + a$ for $h \in \mathbb{R}^p$, with $A \in \mathbb{R}^{d\times p}$, $a \in \mathbb{R}^d$, and the attention output is followed by $B \in \mathbb{R}^{p'\times d}$, then replacing*

$$A \text{ by } \boldsymbol{W}_V A, \qquad a \text{ by } \boldsymbol{W}_V a, \qquad B \text{ by } B\boldsymbol{W}_V^\top$$

absorbs a fixed value rotation and its inverse without changing the unquantized computation.

Proof. For each token, $q^\top \widehat{k}_i = (\boldsymbol{W}_K q)^\top u_{K,i}$, so the logits, and hence the softmax weights, coincide. Linearity gives $\sum_i \widehat{a}_i \widehat{v}_i = \boldsymbol{W}_V^\top \sum_i \widehat{a}_i u_{V,i}$, proving equation 10. For projection folding,

$$(\boldsymbol{W}_V A)h + \boldsymbol{W}_V a = \boldsymbol{W}_V v, \qquad (B\boldsymbol{W}_V^\top)\sum_i a_i(\boldsymbol{W}_V v_i) = B\sum_i a_i v_i.$$

The second equality uses $\boldsymbol{W}_V^\top \boldsymbol{W}_V = I_d$ and holds for any attention weights a_i. In multihead attention, the same argument applies to each matching head slice of the output projection. □

These identities concern fixed linear transforms. They do not move rotations through nonlinearities or positional transformations, and floating-point reassociation need not be bitwise identical. Dense materialization is distinct from a fast Hadamard implementation.

B.4 Candidate-Table Assignment

Partition $[-L, L]^b$ into G^b equal-width Cartesian cells, where $L > 0$ and $G \geq 1$ is an integer. Each cell stores the τ codewords nearest its center, ordered by distance and then global index, with $1 \leq \tau \leq K$. Fix boundary conventions and clip only to form the address. For the resulting candidate set $A(z)$, define

$$i_A(z) = \min \arg\min_{i \in A(z)} \|z - c_i\|^2, \qquad i^*(z) = \min \arg\min_{1 \leq i \leq K} \|z - c_i\|^2.$$

Distances always use the original block z.

Proposition B.7 (Coverage and search cost)**.** *If $i^*(z) \in A(z)$, then $i_A(z) = i^*(z)$. Candidate search attains the global minimum distortion if and only if $A(z)$ contains a global minimizer, and otherwise has larger distortion. The table stores $O(G^b\tau)$ indices; addressing and search use $O(b\tau)$ arithmetic operations under constant-time random access. The codebook stores Kb real coordinates.*

Proof. A minimum over a subset is no smaller than the full-set minimum. For finite sets, equality holds precisely when the subset contains a global minimizer. If it contains the smallest-index one, the common tie rule also gives the same index. There are G^b cells with τ indices each. Addressing costs $O(b)$ and evaluating τ distances costs $O(b\tau)$; since $\tau \geq 1$, the latter bounds the sum. □

The table bound counts index words; for $K \geq 2$ its bit cost is $O(G^b\tau\lceil \log_2 K \rceil)$. A singleton codebook requires no choice of index. Exact coverage is a condition on the candidate list, not a consequence of its size alone.

B.5 Projected-Sphere Sources and Gaussian Transfer

Let U be uniform on $\mathbb{S}^{d-1} = \{u \in \mathbb{R}^d : \|u\| = 1\}$, and let $\Pi \in \mathbb{R}^{b \times d}$ select b distinct coordinates. Write $P_{d,b}$ for the law of $\sqrt{d}\,\Pi U$, for $1 \leq b \leq d$. A fixed nonzero vector, normalized by its RMS and transformed by an independent global Haar matrix, has this block law. Let Γ denote the gamma function.

Proposition B.8 (Finite-dimensional block law)**.** *For integers $1 \leq b < d$, $P_{d,b}$ has Lebesgue density*

$$f_{d,b}(z) = \frac{\Gamma(d/2)}{\Gamma((d-b)/2)(\pi d)^{b/2}} \left(1 - \frac{\|z\|^2}{d}\right)^{(d-b-2)/2} \qquad (\|z\| < \sqrt{d}), \tag{15}$$

and zero outside the ball, with arbitrary boundary values. Its mean is zero and its covariance is I_b. For $b = d$, the law is supported on a sphere and has no d-dimensional Lebesgue density.

Proof. For $g \sim \mathcal{N}(0, I_d)$, $U = g/\|g\|$ is uniform on the sphere. The independent variables $\|\Pi g\|^2$ and $\|(I_d - \Pi^\top \Pi)g\|^2$ have chi-squared laws with b and $d - b$ degrees of freedom. The first variable divided by their sum has the Beta law with parameters $b/2$ and $(d-b)/2$. The direction of Πg is uniform and independent of these norms. Polar coordinates therefore give the density $\Gamma(d/2)[\pi^{b/2}\Gamma((d-b)/2)]^{-1}(1 - \|u\|^2)^{(d-b-2)/2}$ on the unit ball. The change $z = \sqrt{d}\,u$ supplies the factor $d^{-b/2}$ in equation 15. Reflection symmetry gives zero means and cross moments; permutation symmetry and $\sum_j U_j^2 = 1$ give $\mathbb{E}U_i^2 = 1/d$. Thus the covariance is I_b. For $b = d$, all mass lies on a sphere of Lebesgue measure zero, proving the final assertion. □

For probability laws P, Q on $\mathbb{R}^b$ with finite second moments, their quadratic Wasserstein distance is

$$W_2(P, Q) = \inf_{(X,Y)} \left(\mathbb{E}\,\|X - Y\|^2\right)^{1/2},$$

where the infimum ranges over joint laws with marginals P, Q.

Proposition B.9 (Sphere-to-Gaussian coupling). *For $1 \le b \le d$, $W_2(P_{d,b}, \mathcal{N}(0, I_b)) \le \sqrt{2b/d}$.*

Proof. Write $g = RU$ for $g \sim \mathcal{N}(0, I_d)$, where $R = \|g\|$ is independent of U. Couple $X = \Pi g$ and $Z = \sqrt{d}\,\Pi U$. Independence gives

$$\mathbb{E}\,\|X - Z\|^2 = \frac{b}{d}\mathbb{E}(R - \sqrt{d})^2 \le \frac{b}{d^2}\mathbb{E}(R^2 - d)^2 = \frac{2b}{d}.$$

The inequality follows from $R + \sqrt{d} \ge \sqrt{d}$; the last equality uses the mean d and variance $2d$ of R^2. Taking the infimum over couplings proves the result. □

Theorem B.10 (Fixed-codebook distortion transfer). *Let P, Q have finite second moments on $\mathbb{R}^b$. For a nonempty finite codebook $\mathcal{C}$, define the total squared distortion $D_P(\mathcal{C}) = \mathbb{E}_{X\sim P} \min_{c\in\mathcal{C}} \|X - c\|^2$. Then*

$$\left|\sqrt{D_P(\mathcal{C})} - \sqrt{D_Q(\mathcal{C})}\right| \le W_2(P, Q). \tag{16}$$

For root per-coordinate distortion, the bound is $W_2(P, Q)/\sqrt{b}$, and is at most $\sqrt{2/d}$ between $P_{d,b}$ and $\mathcal{N}(0, I_b)$.

Proof. Distance to the codebook, $h(x) = \min_{c\in\mathcal{C}} \|x - c\|$, satisfies $|h(x) - h(y)| \le \|x - y\|$ by the triangle inequality. Its second moment is finite because distance to any fixed codeword bounds h. For any coupling of X, Y, with $\|T\|_{L^2} = (\mathbb{E}|T|^2)^{1/2}$, the reverse triangle inequality gives

$$\left|\|h(X)\|_{L^2} - \|h(Y)\|_{L^2}\right| \le \|h(X) - h(Y)\|_{L^2} \le \left(\mathbb{E}\,\|X - Y\|^2\right)^{1/2}.$$

Taking the infimum over couplings proves equation 16. Divide by $\sqrt{b}$ and apply Proposition B.9 for the spherical case, proving Theorem 3.2. □

Corollary B.11 (Transferring a fitted codebook). *Under the preceding theorem, fix an integer $N \ge 1$ and define $D_P^* = \inf_{1\le|\mathcal{C}|\le N} D_P(\mathcal{C})$ and D_Q^* analogously. Then $|\sqrt{D_P^*} - \sqrt{D_Q^*}| \le W_2(P, Q)$. If $1 \le |\mathcal{C}_Q| \le N$ and $\sqrt{D_Q(\mathcal{C}_Q)} \le \sqrt{D_Q^*} + \eta$ for $\eta \ge 0$, then*

$$\sqrt{D_P(\mathcal{C}_Q)} \le \sqrt{D_P^*} + 2W_2(P, Q) + \eta.$$

Proof. Apply equation 16 to every feasible codebook and take infima to obtain both directions of the first bound. Applying the theorem to $\mathcal{C}_Q$, then its assumed fit bound, and then the bound on the two infima proves the second claim. This argument does not require either infimum to be attained. □

Corollary B.12 (Scalar relative reconstruction error). *Let $x \in \mathbb{R}^d \setminus \{0\}$ be fixed, let $\boldsymbol{W}$ be an independent Haar matrix, and let Q_β be the exact nearest-level map of a fixed finite scalar codebook, applied coordinatewise. Set $s_0 = \|x\|/\sqrt{d}$ and $\widehat{x} = \boldsymbol{W}^\top s_0 Q_\beta(\boldsymbol{W}x/s_0)$. With $\mathrm{relMSE}(\widehat{x}; x) = \|\widehat{x} - x\|^2 / \|x\|^2$ and $D_\beta = \mathbb{E}_{G\sim\mathcal{N}(0,1)}(G - Q_\beta(G))^2$,*

$$\mathbb{E}_{\boldsymbol{W}}\mathrm{relMSE}(\widehat{x}; x) = \mathbb{E}_{Z\sim P_{d,1}}(Z - Q_\beta(Z))^2, \qquad \sqrt{\mathbb{E}_{\boldsymbol{W}}\mathrm{relMSE}(\widehat{x}; x)} \le \sqrt{D_\beta} + \sqrt{2/d}.$$

Proof. Orthogonality and $s_0^2 = \|x\|^2/d$ express the relative error as the average squared error of the normalized coordinates. Their common marginal law is $P_{d,1}$, so linearity of expectation gives the identity; independence is not required. The bound follows from Theorem B.10 and Proposition B.9 with $b = 1$. □

The coupling bound quantifies the change in exact root distortion when replacing the spherical source by a Gaussian. Candidate search and role-specific scale fitting remain separate operations.

B.6 Scale Fitting and Rounding

Proposition B.13 (Least squares and norm correction). *Let $y, \hat{c} \in \mathbb{R}^d \setminus \{0\}$ and $E(s) = \|y - s\hat{c}\|^2$. The unique minimizer over $s \in \mathbb{R}$ is s_{ls} in equation 8, while $s_{\mathrm{nc}}\hat{c}$ has norm $\|y\|$. With $\cos\theta = \langle y, \hat{c}\rangle / (\|y\|\,\|\hat{c}\|)$,*

$$E(s) = E(s_{\mathrm{ls}}) + \|\hat{c}\|^2 (s - s_{\mathrm{ls}})^2, \tag{17}$$

$$E(s_{\mathrm{nc}}) - E(s_{\mathrm{ls}}) = \|y\|^2 (1 - \cos\theta)^2. \tag{18}$$

For per-coordinate MSE, divide equation 18 by d. For $\theta \in [0, \pi]$, its right-hand side is at most $\|y\|^2 \theta^4/4$.

Proof. Expanding $E(s)$ gives $\|y\|^2 - 2s\langle y, \hat{c}\rangle + s^2 \|\hat{c}\|^2$. Its positive quadratic coefficient yields the unique minimizer, and completing the square yields equation 17. Norm preservation follows directly from s_{nc}. Since $s_{\mathrm{ls}} = (\|y\| / \|\hat{c}\|)\cos\theta$, substituting $s_{\mathrm{nc}} - s_{\mathrm{ls}} = (\|y\| / \|\hat{c}\|)(1 - \cos\theta)$ into equation 17 proves equation 18. The angular bound follows from $1 - \cos\theta \le \theta^2/2$. □

Corollary B.14 (Error from a finite rounded scale). *Under the preceding assumptions, let $s \in \mathbb{R}$, let $\widetilde{s} = \mathrm{fp16}(s)$ be finite, and put $\Delta = \widetilde{s} - s$. Then*

$$E(\widetilde{s}) - E(s) = -2\Delta\langle y - s\hat{c}, \hat{c}\rangle + \Delta^2 \|\hat{c}\|^2. \tag{19}$$

Its absolute value is at most $2|\Delta|\,\|y - s\hat{c}\|\,\|\hat{c}\| + \Delta^2 \|\hat{c}\|^2$. At $s = s_{\mathrm{ls}}$, the increase is exactly $\Delta^2 \|\hat{c}\|^2$. All these error identities are preserved by an inverse orthogonal transform.

Proof. Expand $\|(y - s\hat{c}) - \Delta\hat{c}\|^2$ and subtract $E(s)$. The triangle and Cauchy–Schwarz inequalities give the absolute bound. At the LS minimizer, $\langle y - s_{\mathrm{ls}}\hat{c}, \hat{c}\rangle = 0$. Orthogonal maps preserve the Euclidean norm, proving the last claim. □

NC preserves the norm before rounding, whereas LS minimizes error for fixed codes. The corollary isolates the additional error of a finite stored scale; zero and near-zero cases follow Appendix A.3.

B.7 Key and Value Perturbations in Attention

Lemma B.15 (Softmax perturbations). *Let $l, h \in \mathbb{R}^S$, $S \ge 1$, and define $[\mathrm{softmax}(l)]_i = e^{l_i} / \sum_j e^{l_j}$. Set $a = \mathrm{softmax}(l)$ and $\widetilde{a} = \mathrm{softmax}(l + h)$. Then*

$$\|\widetilde{a} - a\|_1 \le \frac{\max_i h_i - \min_i h_i}{2} \le \|h\|_\infty. \tag{20}$$

Proof. For $t \in [0, 1]$, set $a(t) = \mathrm{softmax}(l + th)$ and $\mu(t) = \sum_i a_i(t) h_i$. Differentiation gives $a_i'(t) = a_i(t)(h_i - \mu(t))$. Thus $\|a'(t)\|_1 = \mathbb{E}|H - \mathbb{E}H|$, where $H = h_i$ with probability $a_i(t)$. If $m \le H \le M$, the inequality $(H - m)(M - H) \ge 0$ yields

$$\mathrm{Var}(H) \le (M - \mathbb{E}H)(\mathbb{E}H - m) \le (M - m)^2/4.$$

Cauchy–Schwarz bounds $\mathbb{E}|H - \mathbb{E}H|$ by $(M - m)/2$. Integrate on $[0, 1]$ and use $M - m \le 2\|h\|_\infty$. This also covers $S = 1$, when the first bound is zero. □

Theorem B.16 (Attention-channel bounds). *Consider equation 1 with fixed $q, k_i, v_i \in \mathbb{R}^d$ and $\epsilon_K, \epsilon_V \geq 0$. If only values change and $\max_i \|\widehat{v}_i - v_i\| \leq \epsilon_V$, then $\|\sum_i a_i \widehat{v}_i - o\| \leq \epsilon_V$. If only keys change and $\max_i \left\|\widehat{k}_i - k_i\right\| \leq \epsilon_K$, the output with the perturbed weights satisfies equation 7. The value constant is attained, and the key coefficient is locally sharp.*

Proof. For values, $\|\sum_i a_i(\widehat{v}_i - v_i)\| \leq \sum_i a_i \epsilon_V = \epsilon_V$. For keys, put $h_i = q^\top(\widehat{k}_i - k_i)/\sqrt{d}$. Cauchy–Schwarz gives $\|h\|_\infty \leq \|q\| \, \epsilon_K/\sqrt{d}$, and Lemma B.15 gives the same bound on the ℓ_1 change of weights. Since both weight vectors sum to one, for every $c \in \mathbb{R}^d$,

$$\widehat{o} - o = \sum_i (\widetilde{a}_i - a_i)(v_i - c), \qquad \|\widehat{o} - o\| \leq \|\widetilde{a} - a\|_1 \max_i \|v_i - c\| .$$

Taking the infimum over c proves equation 7.

Adding one common perturbation vector to all values attains the value bound. For keys, use two equal logits, values $v, -v$ with $v \neq 0$, a nonzero query, and key perturbations $\epsilon q/\|q\|$ and $-\epsilon q/\|q\|$. The output change is $\|v\| \tanh(\epsilon \|q\| / \sqrt{d})$ in norm, where $\tanh$ is the hyperbolic tangent. The radius $\inf_c \max\{\|v - c\|, \|-v - c\|\}$ equals $\|v\|$, by the triangle inequality and $c = 0$. The ratio to the claimed bound tends to one as $\epsilon \downarrow 0$. □

For a fixed query, value error enters directly while key error is weighted by query norm and value spread. These are bounds for one attention operation.

B.8 A Fixed-Budget Allocation Model

Proposition B.17 (Continuous and integer allocations). *Let $a_K, a_V > 0$, $C \geq 0$, and $J(t) = a_K 4^{-t} + a_V 4^{-(C-t)}$ for $0 \leq t \leq C$. Its unique constrained minimizer is*

$$t^* = \mathrm{clip}_{[0,C]}\left(\frac{C}{2} + \frac{1}{2}\log_4 \frac{a_K}{a_V}\right), \tag{21}$$

where $\mathrm{clip}_{[0,C]}(u) = \min\{C, \max\{0, u\}\}$. For integer $C \geq 0$ and integer role rates, an optimum is among the feasible floor and ceiling of t^*. If $C > 0$ and $a_K > a_V$, the continuous optimum assigns strictly more bits to keys.

Proof. For $C > 0$, differentiation gives

$$J'(t) = (\log 4)[-a_K 4^{-t} + a_V 4^{-(C-t)}], \qquad J''(t) = (\log 4)^2 J(t) > 0.$$

Strict convexity yields a unique unconstrained minimizer from $a_K 4^{-t} = a_V 4^{-(C-t)}$, giving the expression inside the clip. Monotonicity on either side proves the constrained solution and restricts an integer optimum to the adjacent feasible integers. If $a_K > a_V$, the unconstrained optimum exceeds $C/2$, and clipping preserves that inequality for $C > 0$. For $C = 0$, the only feasible point is zero. □

In this proxy, greater role sensitivity shifts the optimum toward that role. The finite-rate experiments evaluate a fixed split, and A-HQ compares its predefined candidates causally.

C Additional Quality Summaries

The tables below summarize the seven-model quality evaluation from complementary perspectives. Table 6 summarizes relative PPL changes and counts improved models. Table 7 contrasts the full grid with the four non-R1 models, using codec-to-TQ geometric-mean ratios. Table 8 gives the uncompressed reference. Their purpose is to make the distribution of gains visible alongside the main aggregate.

Table 6. Per-model PPL summaries. Each row summarizes the seven models at one rate. Mean and median gains are relative reductions from TQ in percent; wins count models with lower PPL. The geometric-mean values repeat the corresponding aggregate in Table 1 for orientation.

bits	method	TQ PPL	method PPL	mean/median	wins
2	HQ	232.388	185.365	$13.7/-2.6\%$	$3/7$
	Adaptive HQ		**172.566**	$16.7/1.8\%$	$4/7$
3	HQ	118.423	**68.282**	$28.2/9.1\%$	$6/7$
	Adaptive HQ		69.057	$27.2/7.2\%$	$6/7$
4	HQ	84.442	**35.803**	$25.8/2.4\%$	$6/7$
	Adaptive HQ		36.412	$25.3/2.5\%$	$6/7$

Table 7. Relative PPL across model groups. Ratios are geometric-mean codec PPL divided by geometric-mean TQ PPL; $\Delta \log \mathrm{PPL}$ is their natural logarithm. Non-R1 denotes Llama-2-7B, Llama-3.1-8B, Qwen3-4B, and Qwen3-8B. Values below one favor the indicated codec.

bits	method	full-grid ratio	$\Delta \log \mathrm{PPL}$	non-R1 ratio	non-R1 $\Delta \log$
2	HQ	0.798	-0.226	0.873	-0.135
	Adaptive HQ	**0.743**	-0.298	**0.822**	-0.196
3	HQ	**0.577**	-0.551	**0.892**	-0.114
	Adaptive HQ	0.583	-0.539	0.907	-0.097
4	HQ	**0.424**	-0.858	**0.998**	-0.002
	Adaptive HQ	0.431	-0.841	1.001	0.001

Table 8. Uncompressed quality reference. FullKV uses the same models, data, precision, and quality scoring as Table 1. Aggregate PPL is a geometric mean and aggregate ACC an arithmetic mean.

model	FullKV PPL ↓	FullKV ACC ↑
Llama-2-7B	4.903	68.6
Llama-3.1-8B	6.484	74.1
Qwen3-4B	12.099	68.4
Qwen3-8B	8.687	71.5
R1-Llama-8B	22.616	46.6
R1-Qwen-1.5B	173.668	38.3
R1-Qwen-7B	46.204	46.2
Aggregate	17.975	59.1