MXATTENTION: DATA-FREE OPTIMAL SCALING AND PRE-NORMALIZATION QUANTIZATION FOR MXFP4 ATTENTION

Anonymous authors
Paper under double-blind review

ABSTRACT

The quadratic cost of attention limits diffusion-based video generation. MXFP4 attention offers a promising path toward lower-cost inference, but direct quantization often degrades generation quality. We identify two numerical causes: power-of-two shared scaling creates a clipping–underflow trade-off within MXFP4 blocks, while quantization in the online softmax loop breaks row-wise normalization, so the attention weights no longer sum to one. We propose **MXAttention**, a data-free post-training quantization framework with two components. First, **Universal Optimal Scaling (UOS)** exploits the periodic structure of power-of-two microscaling to minimize a global MXFP4 quantization-error objective. It yields the closed-form, distribution-independent boundary $Q_{\max} = 7.25$, without calibration or per-layer search. Second, **Pre-Normalization Quantization (PNQ)** uses the same quantized exponential tiles in the normalizer and output updates, preserving row-wise normalization. On Wan2.2 and HunyuanVideo, MXAttention substantially improves frame-level similarity to FP16 outputs. It preserves **FP16-level generation quality** with less than 0.01 absolute degradation on all reported VBench metrics and achieves video quality competitive with strong NVFP4-based baselines. We further evaluate LLM quantization on Qwen3-14B, observing accuracy gains for attention quantization, weight–activation quantization, and their combination. On one Ascend 950-series NPU, MXAttention delivers these accuracy gains at nearly the same attention latency as the PNQ-enabled MXFP4 baseline (0.13% difference). MXAttention retains a $2.70\times$ attention speedup and a $1.84\times$ total video-generation task speedup over BF16 attention.



Figure 1: **Recovering video quality with MXAttention.** Matched video frames with FP16 as reference. Settings: Table 1; full comparison: Figure 6.

1 INTRODUCTION

Video diffusion Transformers process long sequences of video tokens at every denoising step, making attention a major inference cost (Kong et al., 2024; Wan Team et al., 2025; Yang et al., 2024). Native MXFP4 GEMMs can accelerate both $QK^\top$ and PV, but direct MXFP4 quantization severely degrades video quality. MXFP4 uses 32-element blocks with shared power-of-two E8M0 scales (Open Compute Project, 2023; Rouhani et al., 2023) and is supported across accelerator families (NVIDIA, 2025a; Advanced Micro Devices, Inc., 2025; Huawei, 2025). Compared

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.

with NVFP4's 16-element blocks, E4M3 block scales, and additional tensor scale, it reduces scale metadata but restricts scaling choices (NVIDIA, 2025b;c). Our goal is to recover attention accuracy while preserving the efficiency of native MXFP4 computation.

To reduce the accuracy loss from low-bit quantization, prior work suppresses outliers with rotations (Ashkboos et al., 2024; Liu et al., 2025b), combines low-bit attention with smoothing and scaling (Zhang et al., 2025c;b), or adapts models through quantization-aware training (Zhang et al., 2026). Work on microscaling quantization has explored overflow-aware scaling, empirical scale selection, and additional metadata (Chhugani et al., 2026; Gupta et al., 2026; Hu et al., 2026); Appendix M provides further context.

We identify two numerical causes of quality loss in direct MXFP4 attention. First, shared power-of-two scales force a trade-off between clipping large values and preserving small ones. Second, quantization changes the exponentials used to compute the output, while the normalization factor is still computed before quantization. The resulting attention weights may therefore no longer sum to one. **MXAttention** addresses both without calibration, training, or per-layer search:

- **Universal Optimal Scaling (UOS).** UOS selects each MXFP4 block's shared scale using a fixed boundary, $Q_{\max} = 7.25$. We derive this value in closed form by exploiting the periodic structure of power-of-two scaling. It minimizes our global quantization-error objective, and the optimal boundary is independent of the block-maximum distribution.
- **Pre-Normalization Quantization (PNQ).** PNQ restores the row-wise normalization lost under direct MXFP4 quantization. It uses the same quantized exponentials in the normalizer and output updates, so the attention weights sum to one. PNQ works within the online softmax loop without storing the full attention matrix.
- **Quality recovery with negligible overhead.** MXAttention preserves FP16-level video generation quality while retaining the speed of MXFP4 attention. It also improves LLM quantization accuracy on Qwen3-14B. Our efficiency evaluation shows nearly the same attention latency as the PNQ-enabled MXFP4 baseline, with a difference of just 0.13%. MXAttention retains a $2.70\times$ attention kernel speedup and a $1.84\times$ total video-generation task speedup over BF16 attention.

2 Preliminaries

Online softmax. Given query, key, and value matrices Q, K, V, scaled dot-product attention computes $S = QK^\top/\sqrt{d}$, $P = \mathrm{Softmax}(S)$, and $O = PV$. Each row of P sums to one and weights the value vectors. To avoid storing this large matrix, FlashAttention processes tiles with online softmax (Dao et al., 2022; Dao, 2024; Shah et al., 2024); we preserve this tiled computation. For query tile i, it sums unnormalized exponentials into ℓ_i and uses them to accumulate weighted value vectors in $\widetilde{O}_i$. Dividing $\widetilde{O}_i$ row-wise by ℓ_i gives the output. A running row maximum m_i ensures numerical stability. We denote each exponential tile by $\widetilde{P}_{ij}$; Appendix L gives the full update equations.

MXFP4. MXFP4 stores each block as 32 4-bit E2M1 values with one shared power-of-two scale $\sigma = 2^e$, encoded in 8-bit E8M0. Quantization divides each value by σ, rounds it to the nearest finite E2M1 value, and multiplies by σ to reconstruct it. We denote this rounding operation by $\Pi_{\mathcal{G}}$, where $\mathcal{G}_+ = \{0, 0.5, 1, 1.5, 2, 3, 4, 6\}$ and $\mathcal{G}$ includes both signs. With block maximum magnitude M, OCP uses

$$\mathcal{Q}_\sigma(v) = \sigma \Pi_{\mathcal{G}}(v/\sigma), \qquad e_{\mathrm{OCP}} = \lfloor \log_2 M \rfloor - 2, \tag{1}$$

where $\Pi_{\mathcal{G}}$ rounds to the nearest finite grid value, saturating at ± 6 (Open Compute Project, 2023; Rouhani et al., 2023).

3 Failure Modes of Standard MXFP4 Attention

Clipping–underflow trade-off. The shared scale controls a trade-off: a larger scale reduces clipping of large values but makes smaller values more likely to round to zero. Under the OCP rule, the scaled block maximum M/σ lies in $[4, 8)$, although the largest E2M1 value is 6. The threshold 7 is the midpoint between 6 and the next value 8 on an extended E2M1 grid. Above 7, rounding would

select 8, but finite-range saturation forces the result back to 6. Truncation-Free Scaling (TFS) keeps the scaled maximum in $(3, 6]$ (Chen et al., 2025). This avoids overflow but can double the scale, reducing the resolution available to smaller values (Figure 2).

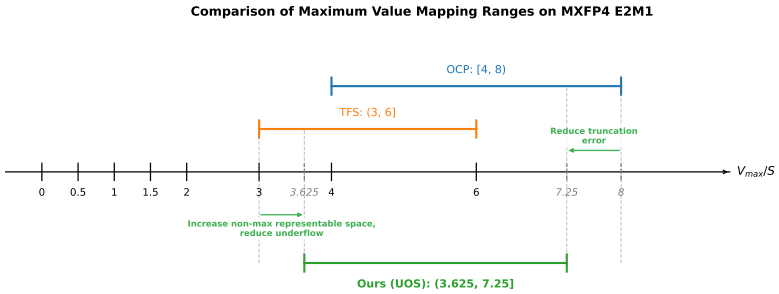


Figure 2: **The shared-scale trade-off.** The horizontal axis is the scaled block maximum M/σ. OCP allows overflow rounding above 7; TFS avoids it at the cost of small-value resolution. UOS selects $(3.625, 7.25]$ by minimizing a global quantization-error objective.

Normalization mismatch. To use MXFP4 in the second attention GEMM, we quantize the softmax exponentials before multiplying them by V. If the normalizer still sums the original exponentials, it no longer matches the weights used to form the output. The resulting attention row sums are generally $\neq 1$, adding a row-dependent scale error to the output. This mismatch persists after online rescaling (Appendix I).

Even an already normalized P need not remain normalized after quantization. Across 80 Wan2.2 attention tensors, quantization reduces its average row sum from 1 to 0.9266 (offline measurement; averaging details in Appendix R). PNQ addresses the online mismatch by normalizing with the quantized exponentials themselves.

4 MXAttention

MXAttention addresses the two failure modes with UOS for shared-scale selection and PNQ for consistent normalization. It also applies a fixed Hadamard rotation to Q and K to suppress outliers before quantization (Figure 3).

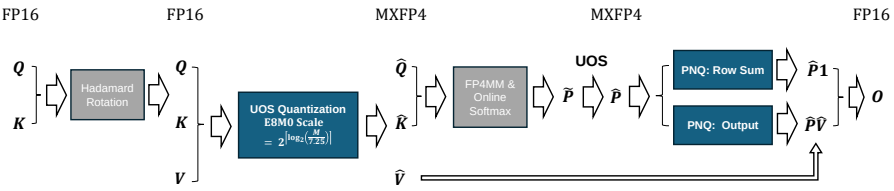


Figure 3: **MXAttention.** Hadamard rotation acts on Q/K, and UOS selects shared scales. PNQ sends the same quantized exponential tile to both the row-sum and weighted-value updates.

4.1 Universal Optimal Scaling

To resolve the clipping–underflow trade-off, we seek a fixed scale-selection boundary that minimizes quantization error across blocks without calibration. We first define a normalized error objective, then show how power-of-two scaling allows us to solve it without knowing the block-maximum distribution.

Let $q = Q_{\max}$ bound the scaled maximum $X_q = M/\sigma$ of a nonzero block with maximum magnitude M. Choosing the smallest power-of-two scale that satisfies this bound gives

$$e_q(M) = \lceil \log_2(M/q) \rceil, \qquad X_q = \frac{M}{2^{e_q(M)}} \in (q/2, q], \qquad q = Q_{\max} > 0. \tag{2}$$

Increasing q can select a smaller scale, improving small-value resolution while exposing larger values to saturation.

A global quantization-error objective. To compare blocks of different magnitudes, we measure squared error relative to each block's maximum. By symmetry, consider nonnegative values. For a scaled maximum x, rounding changes a relative magnitude $t \in [0, 1]$ to $\Pi_{\mathcal{G}_+}(xt)/x$. Averaging uniformly over this unit interval gives

$$D(x) = \int_0^1 \left(t - \frac{\Pi_{\mathcal{G}_+}(xt)}{x} \right)^2 dt = \frac{E(x)}{x^3}, \qquad E(x) = \int_0^x \left(v - \Pi_{\mathcal{G}_+}(v) \right)^2 dv. \tag{3}$$

The factor x^{-3} follows by substituting $v = xt$; $E(x)$ is the cumulative squared rounding error. This uniform interval average is a data-free reference, while actual within-block distributions may be nonuniform. To evaluate q across blocks, we average $D(X_q)$ over the density g_q of their scaled maxima:

$$\mathcal{J}(q) = \mathbb{E}[D(X_q)] = \int_{q/2}^{q} D(x) g_q(x)\, dx. \tag{4}$$

Power-of-two scaling induces exact periodicity. The objective appears to require g_q, which varies across tensors and models. To avoid estimating it, we examine the effect of power-of-two scaling. Doubling a maximum doubles its scale and leaves X_q unchanged: maxima $M = 2^k x$ map to the same $x \in (q/2, q)$. Logarithms turn these rescalings into integer shifts. Let $U = \log_2 M$ have any absolutely continuous density f_U. We sum the density f_U over these integer shifts and convert back to the scaled maximum x:

$$g(x) = \frac{H(\log_2 x)}{x \ln 2}, \qquad H(t) = \sum_{k \in \mathbb{Z}} f_U(t + k), \qquad H(t + 1) = H(t). \tag{5}$$

Here H is the unit-mean, one-periodic density of $U \bmod 1$. Changing q only selects a different interval $(q/2, q)$ from the same global function g: g_q equals g there, up to null sets (Appendix A). The bounds satisfy $\log_2(q/2) = \log_2 q - 1$, so the one-periodic function H takes the same value at both. The factor $1/x$ in g then gives the following relation:

Lemma 1 (Boundary density ratio)**.** *For almost every $q > 0$, $g(q/2) = 2g(q)$. Proof.* By Eq. (5), $g(q/2) = H(\log_2 q - 1)/[(q/2) \ln 2] = 2g(q)$. Appendix D.1 gives a Fourier interpretation. ■

Finding the optimal boundary. We now use this boundary relation to minimize $\mathcal{J}$ without estimating the distribution. Since g is fixed, changing q affects only the integration endpoints. Local integrability of Dg makes $\mathcal{J}$ locally absolutely continuous; differentiation and the boundary ratio give, for almost every q,

$$\begin{aligned} \mathcal{J}'(q) &= D(q)g(q) - \tfrac{1}{2} D(q/2) g(q/2) \\ &= g(q)\big[D(q) - D(q/2)\big] = \frac{g(q)}{q^3} \underbrace{\big[E(q) - 8E(q/2)\big]}_{\Delta(q)}. \end{aligned} \tag{6}$$

The distribution now appears only in the nonnegative factor $g(q)$. It changes the derivative's magnitude but cannot reverse its sign. Locating the optimum therefore reduces to comparing the grid errors at q and $q/2$. Where the derivative exists and $g(q) > 0$, setting it to zero gives

$$D(q) = D(q/2) \quad \Longleftrightarrow \quad E(q) = 8E(q/2). \tag{7}$$

In Eq. (7), the normalized rounding errors at the lower and upper bounds, $q/2$ and q, are equal. Appendix C.1 justifies the almost-everywhere differentiation without assuming continuity of g.

A closed-form optimum of 7.25. It remains to compute the error difference Δ for E2M1. We first consider $q \in [6, 8]$, between the largest finite E2M1 value, 6, and the next power of two, 8. At $q = 7$, the lower endpoint $q/2$ crosses the rounding midpoint 3.5. Integrating over the E2M1 grid therefore gives two branches (Appendix B):

$$\Delta(q) = E(q) - 8E(q/2) = \begin{cases} -1/8, & 6 \le q \le 7, \\ (4q - 29)(4q - 27)/8, & 7 \le q \le 8. \end{cases} \tag{8}$$

The only admissible root is $q^\star = 29/4 = 7.25$; 6.75 lies outside the second branch. To establish global optimality, we also check the derivative outside this interval. The complete sign analysis confirms that no other boundary gives a lower objective value (Appendix C).

Theorem 1 (Global optimality of UOS). *Consider nonzero blocks under the scale family in Eq. (2), with the selected shared-scale exponents within the E8M0 range. For any absolutely continuous distribution of $U = \log_2 M$, the objective in Eq. (4) has the global minimizer*

$$Q^\star_{\max} = q^\star = \frac{29}{4} = 7.25. \tag{9}$$

Different layers or heads need not contribute equally to the error objective. The same boundary remains optimal when some blocks receive greater importance:

Corollary 1 (Optimality with unequal block weights). *Under the assumptions of Theorem 1, $q^\star = 7.25$ also minimizes $\mathbb{E}[W\,D(X_q)]$ for any nonnegative weights with finite mean whose rule is fixed as q varies (Appendix C.2).*

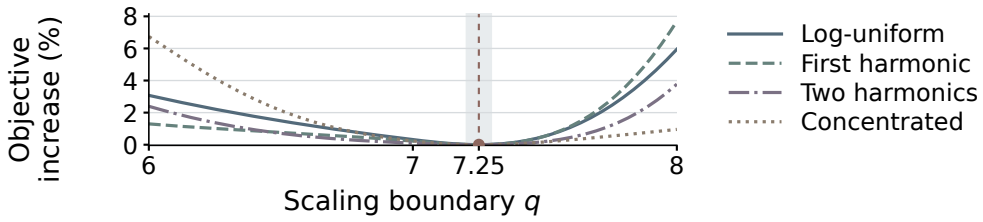


Figure 4: **Different block-maximum distributions yield the same optimum at 7.25.** The curves show the analytical objective $\mathcal{J}(q)$ for four wrapped block-maximum distributions. Each reports the percentage increase from its own minimum at 7.25. Settings: Appendix K.

This result gives a direct scale-selection rule: use $q = 7.25$ in Eq. (2) for every block. UOS changes only the boundary in the existing scale calculation, retaining the E2M1 grid and 32-element format. Appendices N and G give the quantizer and comparisons with OCP and OAS (Chhugani et al., 2026).

4.2 Pre-Normalization Quantization

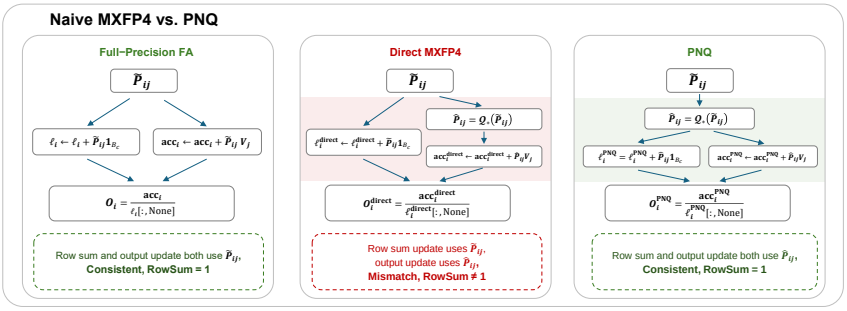


Figure 5: **Why PNQ preserves normalization.** PNQ replaces $\widetilde{P}_{ij}\mathbf{1}$ by $\widehat{P}_{ij}\mathbf{1}$ in the normalizer, so both updates use the same quantized tile. Here $\mathrm{acc}_i = \widetilde{O}_i$; running-max rescaling and previous tiles are omitted. Full illustration: Appendix O.

PNQ resolves the mismatch in Section 3 by computing the normalizer from the quantized exponentials used for the output. Suppose quantization changes a row's exponential sum from 2 to 1.875:

Method	Quantized sum	Divisor	Weight sum
Direct quantization	1.875	2	0.9375
PNQ	1.875	1.875	1

In online softmax, this consistency must hold across tiles and running-maximum rescaling. For query tile i and successive key/value tiles j, let $m_i^{(j)} = \max(m_i^{(j-1)}, \mathrm{rowmax}(S_{ij}))$ and $\widetilde{P}_{ij} = \exp(S_{ij} - m_i^{(j)}\mathbf{1}^\top)$. With the UOS quantizer $\mathcal{Q}_\star$, set $\widehat{P}_{ij} = \mathcal{Q}_\star(\widetilde{P}_{ij})$ and $\widehat{V}_j = \mathcal{Q}_\star(V_j)$. The factor $\alpha_i^{(j)} = \exp(m_i^{(j-1)} - m_i^{(j)})$ rescales both accumulators to the updated row maximum. Initialize $m_i^{(0)} = -\infty$, $\ell_i^{(0)} = 0$, and $\widetilde{O}_i^{(0)} = 0$. PNQ then updates

$$\begin{aligned} \ell_i^{(j)} &= \alpha_i^{(j)} \odot \ell_i^{(j-1)} + \widehat{P}_{ij}\mathbf{1}, \\ \widetilde{O}_i^{(j)} &= \mathrm{Diag}(\alpha_i^{(j)})\widetilde{O}_i^{(j-1)} + \widehat{P}_{ij}\widehat{V}_j. \end{aligned} \tag{10}$$

The two additions use the same quantized exponential tile: $\widehat{P}_{ij}\mathbf{1}$ gives its row sums, and $\widehat{P}_{ij}\widehat{V}_j$ gives its weighted-value sums. The baseline instead uses the pre-quantization sum $\widetilde{P}_{ij}\mathbf{1}$ for the normalizer; the other updates are unchanged.

At every step, both accumulators undergo the same rescaling and receive contributions from the same quantized exponentials. Dividing $\widetilde{O}_i$ row-wise by ℓ_i therefore normalizes the weights actually used to form the output: each row sums to one in exact arithmetic. PNQ requires no additional pass. Appendices H–J give the complete derivation, including why the denominator remains nonzero.

Implementation. Applying the same Hadamard rotation to Q and K preserves $QK^\top$ before quantization. The algorithm applies UOS to the rotated Q/K, to V, and to the online exponential tiles, all with the fixed boundary 7.25. Algorithm 2 in Appendix O gives the complete forward pass.

5 Experiments

5.1 Setup and Baselines

We evaluate Wan2.2-14B and HunyuanVideo-13B at 720p on the complete Open-Sora evaluation prompt set (48 prompts) (Zheng et al., 2024b). All quantized video configurations use 4-bit attention throughout, except the Wan2.2 main comparison, which keeps only steps 38–39 in high precision. We leave the rest of the model at its original precision.



Figure 6: **Recovering detail with MXAttention.** Compared with direct MXFP4, MXAttention restores beard and coat textures and better-defined street backgrounds. Wan2.2 (left) and HunyuanVideo (right); configurations follow Table 1.

Table 1: **Video quality on the full Open-Sora prompt set (48 prompts).** Wan2.2 quantized methods retain high precision only at steps 38–39; HunyuanVideo quantized methods use fully 4-bit attention. The first three metrics are VBench scores; the last five use FP16 outputs as reference. Bold: best quantized result.

Method	Subject↑	Imaging↑	Aesthetic↑	Cosine↑	SSIM↑	PSNR↑	FID↓	LPIPS↓
Wan2.2								
FP16	0.9472	0.6914	0.6359	–	–	–	–	–
MXFP4 (OCP)	0.9479	0.5973	0.6198	0.9026	0.4466	14.05	190.25	0.6012
NVFP4	0.9322	0.6733	0.6135	0.8583	0.4157	12.07	168.22	0.5480
NVFP4 + Sage	0.9459	**0.6900**	0.6348	0.8988	0.4766	13.93	126.18	0.4378
MXAttention	**0.9491**	0.6847	**0.6355**	**0.9216**	**0.5190**	**15.16**	**112.60**	**0.3861**
HunyuanVideo								
FP16	0.9556	0.6135	0.6024	–	–	–	–	–
MXFP4 (OCP)	**0.9616**	0.4098	0.5511	0.9406	0.6313	15.47	228.78	0.5514
NVFP4	0.9533	**0.6140**	0.5983	0.9535	0.6697	16.97	144.75	0.3840
NVFP4 + Sage	0.9575	0.6119	0.6029	**0.9710**	**0.7524**	**19.74**	**105.95**	**0.2679**
MXAttention	0.9563	0.6050	**0.6080**	0.9655	0.7198	18.59	136.28	0.3089

We compare exact-floor OCP MXFP4, NVFP4, NVFP4 with SageAttention smoothing and scaling (NVIDIA, 2025b;c; Zhang et al., 2025b), and MXAttention. We report VBench scores (Huang et al., 2024) and similarity to FP16 reference videos generated with the same prompts and seeds. Appendix P defines the metrics, generation settings, and baselines.

5.2 Video Generation Quality

MXAttention recovers 92.9% of the Imaging Quality loss from direct OCP quantization on Wan2.2 and 95.8% on HunyuanVideo (Table 1; Figure 6). We measure the recovered fraction of the FP16–OCP gap as $(s_{\text{MXAttention}} - s_{\text{OCP}})/(s_{\text{FP16}} - s_{\text{OCP}})$. Every reported VBench score is less than 0.01 below FP16 or exceeds it. Compared with NVFP4, MXAttention improves all five FP16-reference metrics on both models. Among the quantized methods, it leads these metrics on Wan2.2, while NVFP4 + SageAttention leads on HunyuanVideo.

5.3 Component Ablation

Video diffusion. We evaluate UOS, PNQ, and Q/K rotation with 4-bit attention at every denoising step (Table 2; Figure 7). With rotation enabled, PNQ improves Imaging Quality by 0.0542 on Wan2.2 and 0.1007 on HunyuanVideo and improves all five FP16-reference metrics. UOS further improves similarity to FP16 outputs across all five reference metrics on both models. Together, these components recover fine visual details and improve similarity to FP16 outputs.

Table 2: **Sequential ablation with fully 4-bit attention.** No high-precision denoising steps; other settings follow Table 1. H/P/U denote Hadamard/PNQ/UOS. Bold: best per model, including ties.

Configuration	H	P	U	Subject↑	Imaging↑	Aesthetic↑	Cosine↑	SSIM↑	PSNR↑	FID↓	LPIPS↓
Wan2.2											
Full	✓	✓	✓	**0.9357**	**0.6730**	**0.6216**	**0.9215**	**0.5047**	**14.98**	**120.96**	**0.4010**
w/o UOS	✓	✓	–	**0.9357**	0.6702	0.6198	0.9188	0.4892	14.97	129.05	0.4309
w/o P, U	✓	–	–	0.9250	0.6160	0.6025	0.8865	0.4256	13.30	194.35	0.6193
OCP	–	–	–	0.9343	0.5247	0.5801	0.9034	0.4377	14.06	218.09	0.6762
HunyuanVideo											
Full	✓	✓	✓	0.9563	**0.6050**	**0.6080**	**0.9655**	**0.7198**	**18.59**	**136.28**	**0.3089**
w/o UOS	✓	✓	–	0.9617	0.5989	0.6071	0.9588	0.6969	17.90	151.48	0.3520
w/o P, U	✓	–	–	**0.9641**	0.4982	0.5836	0.9490	0.6702	16.72	197.31	0.4681
OCP	–	–	–	0.9616	0.4098	0.5511	0.9406	0.6313	15.47	228.78	0.5514

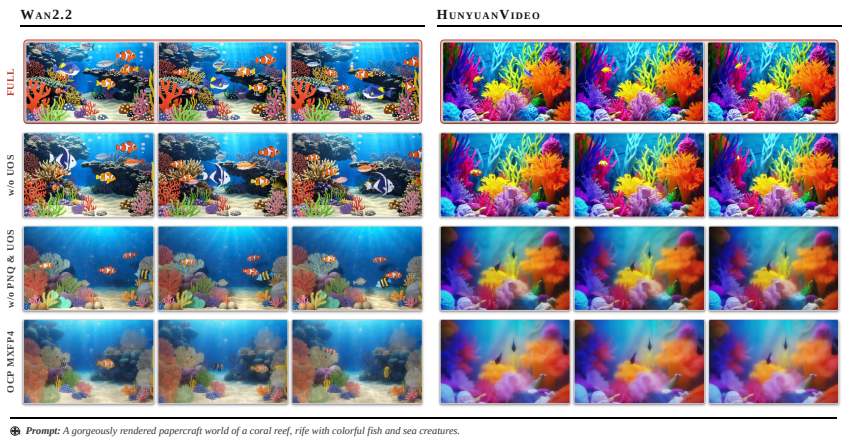


Figure 7: **Visual effects of PNQ and UOS.** PNQ sharpens fish and coral boundaries; the full Wan2.2 result also shows clearer papercraft shapes. UOS further improves similarity to FP16 outputs on both models (Table 2). All denoising steps use 4-bit attention. Wan2.2 (left), HunyuanVideo (right).

Language models. Table 3 compares three quantization settings on Qwen3-14B (Yang et al., 2025) against BF16: attention only, all linear-layer weights and input activations (W4A4) only, and both together. UOS applies to MXFP4 block quantization in either setting, whereas PNQ applies to attention's online softmax. The combined MXAttention + UOS configuration uses MXAttention for attention and UOS for W4A4 linear layers. The UOS + UOS configuration applies only UOS in both settings, without rotation or PNQ.

Table 3: **Qwen3-14B: BF16 and MXFP4.** Combined rows list attention + W4A4; MXAttn denotes MXAttention. Bold: best within each quantized group. GSM: GSM8K; Wino: Winogrande; Wiki/Lamb: Wikitext/Lambada perplexity.

Configuration	GSM↑	MMLU↑	CEval↑	ARC-C↑	Wino↑	PIQA↑	BoolQ↑	Wiki↓	Lamb↓
BF16	0.926	0.771	0.821	0.590	0.726	0.800	0.893	11.664	3.672
Attention only									
OCP	0.857	0.715	0.757	0.526	0.680	0.768	0.869	13.355	**4.069**
MXAttention	**0.890**	**0.749**	**0.794**	**0.559**	**0.711**	**0.770**	**0.884**	**11.943**	4.332
Weight–activation only (W4A4)									
OCP	0.865	0.717	0.765	0.543	0.689	0.760	0.870	13.898	4.954
UOS	**0.885**	**0.729**	**0.771**	**0.555**	**0.695**	**0.763**	**0.873**	**13.458**	**4.659**
Attention + weight–activation									
OCP + OCP	0.734	0.649	0.653	0.456	0.627	0.732	0.835	16.962	6.139
UOS + UOS	0.789	0.671	0.704	0.493	0.665	0.718	0.835	15.187	**5.449**
MXAttn + UOS	**0.810**	**0.697**	**0.736**	**0.504**	**0.695**	**0.741**	**0.853**	**13.724**	5.510

MXAttention improves eight of the nine attention-only metrics over OCP. UOS also improves all nine metrics when applied to W4A4 linear layers, supporting its use beyond attention. Combining MXAttention with UOS for W4A4 linear layers gives the best results on eight of the nine metrics among the combined configurations.

5.4 SCALING BOUNDARY VALIDATION

UOS derives 7.25 from a data-free error objective. We test whether it also minimizes reconstruction error on real attention tensors. For each Transformer layer, we sweep $Q_{\max}$ separately for Q, K, and V, and record the boundary with the lowest tensor MSE. Table 4 reports the frequency of these choices across layers.

Table 4: **Layer-wise $Q_{\max}$ selected by tensor-MSE minimization.** Percentages of layers; bold marks the most frequent choices, including ties.

	Wan2.2			HunyuanVideo			Qwen3-14B		
$Q_{\max}$	Q	K	V	Q	K	V	Q	K	V
≤ 7.15	0.0	0.0	0.0	45.0	0.0	0.0	0.0	0.0	0.0
7.20	5.0	10.0	10.0	0.0	**62.5**	**50.0**	27.5	35.0	12.5
7.25	**95.0**	**90.0**	**90.0**	**52.5**	32.5	**50.0**	**72.5**	**65.0**	**77.5**
≥ 7.30	0.0	0.0	0.0	2.5	5.0	0.0	0.0	0.0	10.0

The value 7.25 is the most frequent choice in eight of the nine model–tensor combinations, including one tie. These searches are used only for validation; MXAttention uses the fixed boundary 7.25 without calibration. Appendix Q gives the search settings.

5.5 Efficiency

Table 5: **Efficiency on one Ascend 950-series NPU.** (a) Attention latency, including preprocessing. (b) Mean total task time, including startup, model loading, generation, and checks. Speedups use the corresponding BF16 attention reference.

(a) Attention latency: median of 48 samples

Method	Latency (ms)	Speedup
BF16 attention	588.86	1.00×
MXFP8	405.66	1.45×
Fused MXFP4 (PNQ enabled)	217.72	2.70×
MXAttention	218.01	2.70×

(b) Video generation: two runs per method

Method	Mean task time (s)	Speedup
BF16 attention	2780.0	1.00×
MXAttention	1507.5	1.84×

Attention latency. We time non-causal attention on synthetic BF16 $Q/K/V$ with $B = 1$, $H = 40$, $D = 128$, and $S = 75{,}776$. Table 5(a) shows nearly the same latency as the PNQ-enabled MXFP4 baseline: 218.01 versus 217.72 ms, a difference of 0.13%. MXAttention retains a 2.70× speedup over BF16 attention. Latency includes layout conversion, quantization, and rotation. Both MXFP4 configurations use the same fused kernel and native exponential quantizer, without UOS on these tiles. MXAttention applies UOS to $Q/K/V$ and fixed Hadamard rotation to Q/K. UOS only replaces the existing scale-selection boundary; it adds no processing step.

Total task time. Table 5(b) reports a 1.84× speedup (45.8% reduction) in mean total task time for Wan2.2-T2V-A14B over SGLang's native BF16 attention backend (Zheng et al., 2024a). We use one fixed prompt and seed, two runs per method, and the main quality comparison's precision policy (Table 1). Appendix S details the attention and video-generation timing procedures, individual runs, and integration.

6 Conclusion

MXAttention combines data-free optimal scaling with consistent normalization to recover FP16-level video generation quality and improve LLM quantization accuracy. UOS fixes the scaling boundary at 7.25, while PNQ computes the normalizer from the quantized weights. These accuracy gains retain nearly the same attention latency as the PNQ-enabled MXFP4 baseline in the tested NPU configuration (0.13% difference). In conclusion, MXAttention makes MXFP4 attention practical for video generation by recovering quality while retaining its speed.

AI USE STATEMENT

We used generative AI tools to assist with polishing the manuscript and to identify relevant literature. The authors take responsibility for the final content.

REFERENCES

Advanced Micro Devices, Inc. AMD Instinct MI350 Series GPUs. https://www.amd.com/en/products/accelerators/instinct/mi350.html, 2025. Native MXFP4 and MXFP6 datatype support.

Saleh Ashkboos, Amirkeivan Mohtashami, Maximilian L. Croci, Bo Li, Pashmina Cameron, Martin Jaggi, Dan Alistarh, Torsten Hoefler, and James Hensman. Quarot: Outlier-free 4-bit inference in rotated LLMs. In *Advances in Neural Information Processing Systems*, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/b5b939436789f76f08b9d0da5e81af7c-Abstract-Conference.html.

William Bell and Saralees Nadarajah. A review of wrapped distributions for circular data. *Mathematics*, 12(16):2440, 2024. doi: 10.3390/math12162440.

Yuxiang Chen, Haocheng Xi, Jun Zhu, and Jianfei Chen. Oscillation-reduced MXFP4 training for vision transformers. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, 2025.

Jatin Chhugani, Geonhwa Jeong, Bor-Yiing Su, Yunjie Pan, Hanmei Yang, Aayush Ankit, Jiecao Yu, Summer Deng, Yunqing Chen, Nadathur Satish, and Changkyu Kim. Unveiling the potential of quantization with MXFP4: Strategies for quantization error reduction. In *International Conference on Machine Learning*, 2026.

Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. In *International Conference on Learning Representations*, 2024.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with IO-awareness. In *Advances in Neural Information Processing Systems*, volume 35, pp. 10448–10459, 2022.

Tanmaey Gupta, Hayden Prairie, Xiaoxia Wu, Reyna Abhyankar, Qingyang Wu, Austin Silveria, Pragaash Ponnusamy, Jue Wang, Ben Athiwaratkun, Leon Song, Tri Dao, Daniel Y. Fu, and Chris De Sa. Search your block floating point scales! *arXiv preprint arXiv:2605.12464*, 2026.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Weiming Hu, Zihan Zhang, Haoyan Zhang, Chen Zhang, Cong Guo, Yu Feng, Tianchi Hu, Guanglin Li, Guipeng Hu, Junsong Wang, and Jingwen Leng. M^2xfp: A metadata-augmented microscaling data format for efficient low-bit quantization. *arXiv preprint arXiv:2601.19213*, 2026.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, Yaohui Wang, Xinyuan Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Huawei. Leading a new paradigm for ai infrastructure. https://www.huawei.com/en/news/2025/9/hc-xu-keynote-speech, 2025. Announced MXFP4 support for the Ascend 950 series.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

Lauwerens Kuipers and Harald Niederreiter. *Uniform Distribution of Sequences*. John Wiley & Sons, New York, 1974.

Xingyang Li, Samuel Tesfai, Zhekai Zhang, Haocheng Xi, Shuo Yang, Lvmin Zhang, Yufei Sun, Kelly Peng, Maneesh Agrawala, Ion Stoica, Kurt Keutzer, Jun-Yan Zhu, Song Han, Yujun Lin, and Muyang Li. Deltaquant: 4-bit video diffusion models with spatiotemporal delta smoothing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 43578–43588, 2026.

Feng Liu, Shiwei Zhang, Xiaofeng Wang, Yujie Wei, Haonan Qiu, Yuzhong Zhao, Yingya Zhang, Qixiang Ye, and Fang Wan. Timestep embedding tells: It's time to cache for video diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7353–7363, 2025a.

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*, 2025b.

Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference. In *International Conference on Learning Representations*, 2024.

Kanti V. Mardia and Peter E. Jupp. *Directional Statistics*. Wiley Series in Probability and Statistics. John Wiley & Sons, Chichester, 2000.

NVIDIA. Openai triton on nvidia blackwell boosts ai performance and programmability. https://developer.nvidia.com/blog/openai-triton-on-nvidia-blackwell-boosts-ai-performance-and-programmability/, 2025a. Demonstrates hardware-accelerated MXFP4 GEMM support on NVIDIA Blackwell.

NVIDIA. Introducing NVFP4 for efficient and accurate low-precision inference. https://developer.nvidia.com/blog/introducing-nvfp4-for-efficient-and-accurate-low-precision-inference/, 2025b.

NVIDIA. Using FP8 and FP4 with transformer engine. https://docs.nvidia.com/deeplearning/transformer-engine/user-guide/examples/fp8_primer.html, 2025c.

Open Compute Project. OCP Microscaling Formats (MX) Specification Version 1.0. https://www.opencompute.org/documents/ocp-microscaling-formats-mx-v1-0-spec-final-pdf, 2023.

Bita Darvish Rouhani, Ritchie Zhao, Ankit More, Mathew Hall, Alireza Khodamoradi, Summer Deng, Dhruv Choudhary, Marius Cornea, Eric Dellinger, Kristof Denolf, et al. Microscaling data formats for deep learning. *arXiv preprint arXiv:2310.10537*, 2023.

Jay Shah, Ganesh Bikshandi, Ying Zhang, Vijay Thakkar, Pradeep Ramani, and Tri Dao. Flashattention-3: Fast and accurate attention with asynchrony and low-precision. *arXiv preprint arXiv:2407.08608*, 2024.

Elias M. Stein and Rami Shakarchi. *Fourier Analysis: An Introduction*, volume 1 of *Princeton Lectures in Analysis*. Princeton University Press, Princeton, NJ, 2003.

Wan Team, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Fu-Yun Wang, Zhaoyang Huang, Alexander William Bergman, Dazhong Shen, Peng Gao, Michael Lingelbach, Keqiang Sun, Weikang Bian, Guanglu Song, Yu Liu, Xiaogang Wang, and Hongsheng Li. Phased consistency models. In *Advances in Neural Information Processing Systems*, 2024.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024.

Jintao Zhang, Haofeng Huang, Pengle Zhang, Jia Wei, Jun Zhu, and Jianfei Chen. Sageattention2: Efficient attention with thorough outlier smoothing and per-thread INT4 quantization. In *International Conference on Machine Learning*, 2025a.

Jintao Zhang, Jia Wei, Pengle Zhang, Xiaoming Xu, Haofeng Huang, Haoxu Wang, Kai Jiang, Jun Zhu, and Jianfei Chen. Sageattention3: Microscaling FP4 attention for inference and an exploration of 8-bit training. *arXiv preprint arXiv:2505.11594*, 2025b.

Jintao Zhang, Jia Wei, Pengle Zhang, Jun Zhu, and Jianfei Chen. Sageattention: Accurate 8-bit attention for plug-and-play inference acceleration. In *International Conference on Learning Representations*, 2025c.

Jintao Zhang, Xiaoming Xu, Jia Wei, Haofeng Huang, Pengle Zhang, Chendong Xiang, Jun Zhu, and Jianfei Chen. Sageattention2++: A more efficient implementation of sageattention2. *arXiv preprint arXiv:2505.21136*, 2025d.

Peiyuan Zhang, Matthew Noto, Wenxuan Tan, Chengquan Jiang, Will Lin, Wei Zhou, and Hao Zhang. Attn-qat: 4-bit attention with quantization-aware training. *arXiv preprint arXiv:2603.00040*, 2026.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 586–595, 2018.

Xuanlei Zhao, Xiaolong Jin, Kai Wang, and Yang You. Real-time video generation with pyramid attention broadcast. *arXiv preprint arXiv:2408.12588*, 2024.

Lianmin Zheng, Liangsheng Yin, Zhiqiang Xie, Chuyue Sun, Jeff Huang, Cody Hao Yu, Shiyi Cao, Christos Kozyrakis, Ion Stoica, Joseph E. Gonzalez, Clark Barrett, and Ying Sheng. SGLang: Efficient execution of structured language model programs. In *Advances in Neural Information Processing Systems*, volume 37, 2024a.

Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all. *arXiv preprint arXiv:2412.20404*, 2024b.

Appendix guide. Appendices A–C give the UOS derivation and scalar-weighted extension. Appendices D–F provide supplementary distributional interpretations and measurements; their approximations are not needed for the UOS proof. Appendices G and N compare scaling rules, and Appendix K specifies the theoretical curves. Appendices H–J prove PNQ normalization, while Appendix O gives the full algorithm. Appendices L–M provide background; Appendices P–S give evaluation and implementation details.

A Exact Density under Power-of-Two Scaling

We derive the density of the scaled block maximum and show why changing q selects a different interval of one fixed function g. This gives the boundary relation $g(q/2) = 2g(q)$ used in the main proof.

We analyze nonzero blocks for which the required shared-scale exponent lies within the E8M0 exponent range. The cases $M = 0$ and exponents outside this range are handled by the MXFP4 conversion routine and are omitted because they do not affect the derivation of q. Let $M > 0$ be the unscaled block maximum, let $U = \log_2 M$ have density f_U, and define

$$e_q(M) = \left\lceil \log_2 \left(\frac{M}{q} \right) \right\rceil, \qquad X_q = \frac{M}{2^{e_q(M)}}. \tag{11}$$

For $x \in (q/2, q)$, every preimage of x has the form

$$M = 2^k x, \qquad U = \log_2 x + k, \qquad k \in \mathbb{Z}. \tag{12}$$

Indeed,

$$e_q(2^k x) = \left\lceil k + \log_2 \left(\frac{x}{q} \right) \right\rceil = k, \tag{13}$$

because $\log_2(x/q) \in (-1, 0)$. The Jacobian of $U = \log_2 x + k$ is

$$\left| \frac{dU}{dx} \right| = \frac{1}{x \ln 2}. \tag{14}$$

Summing over all preimages gives

$$g_q(x) = \frac{1}{x \ln 2} \sum_{k \in \mathbb{Z}} f_U(\log_2 x + k), \qquad x \in (q/2, q). \tag{15}$$

Define the periodized log-domain density

$$H(t) = \sum_{k \in \mathbb{Z}} f_U(t + k). \tag{16}$$

The function H has period one: $H(t + 1) = H(t)$. Define the fixed function g on $x > 0$ by

$$g(x) = \frac{H(\log_2 x)}{x \ln 2}. \tag{17}$$

Equation (15) gives $g_q(x) = g(x)\mathbf{1}_{\{q/2 < x < q\}}$ almost everywhere. Thus, q selects a factor-of-two interval of the fixed function g. The relation $g(x/2) = 2g(x)$ concerns g, not the truncated density g_q.

The density integrates to one over every such interval. Let $a = \log_2 q$. Then

$$\int_{q/2}^{q} g(x)\, dx = \int_{a-1}^{a} H(t)\, dt \tag{18}$$

$$= \int_{0}^{1} H(t)\, dt \tag{19}$$

$$= \sum_{k \in \mathbb{Z}} \int_{k}^{k+1} f_U(u)\, du \tag{20}$$

$$= 1. \tag{21}$$

Table 6: Exact cumulative quantization error $E(x)$ at E2M1 decision boundaries and selected grid points.

$E(0)$	$E(0.25)$	$E(0.75)$	$E(1.25)$	$E(1.75)$	$E(2.5)$	$E(3.5)$	$E(5)$	$E(6)$
0	$\frac{1}{192}$	$\frac{1}{64}$	$\frac{5}{192}$	$\frac{7}{192}$	$\frac{1}{12}$	$\frac{1}{6}$	$\frac{13}{24}$	$\frac{7}{8}$

For almost every $q > 0$, periodicity gives

$$g\left(\frac{q}{2}\right) = \frac{H(\log_2 q - 1)}{(q/2)\ln 2} \tag{22}$$

$$= 2\frac{H(\log_2 q)}{q \ln 2} \tag{23}$$

$$= 2g(q). \tag{24}$$

This proves Lemma 1. The result requires only the exact periodicity of H, not a log-uniform, Gaussian, or small-perturbation assumption.

B Integrated Quantization Error of the E2M1 Grid

For nonnegative magnitudes, the finite E2M1 grid and its round-to-nearest decision boundaries are

$$\mathcal{G}_{\text{E2M1}} = \{0, 0.5, 1, 1.5, 2, 3, 4, 6\}, \tag{25}$$

$$\mathcal{T}_{\text{E2M1}} = \{0.25, 0.75, 1.25, 1.75, 2.5, 3.5, 5.0\}. \tag{26}$$

The value 6 is the largest finite E2M1 value, and 5 is the decision boundary between 4 and 6. Hence, every normalized value above 5 projects to 6. The threshold 7 has a different role: it is the midpoint between 6 and the next value 8 on the unbounded E2M1 ladder. Values above 7 would round upward to 8 before finite-range saturation forces them to 6. We refer to this subset as the overflow-rounding region.

For an interval $[a, b]$ mapped to a quantization value r,

$$\int_a^b (v - r)^2 \, dv = \frac{(b - r)^3 - (a - r)^3}{3}. \tag{27}$$

Applying Eq. (27) successively gives the cumulative errors in Table 6.

The relevant cumulative values are $E(3) = 1/8$, $E(4) = 5/24$, and $E(6) = 7/8$. For $q \geq 6$, every value above 5 maps to 6, giving

$$E(q) = E(6) + \int_6^q (v - 6)^2 \, dv = \frac{7}{8} + \frac{1}{3}(q - 6)^3. \tag{28}$$

For $6 \leq q \leq 7$, $q/2 \in [3, 3.5]$, where values map to 3. Hence,

$$E\left(\frac{q}{2}\right) = \frac{1}{8} + \frac{1}{3}\left(\frac{q}{2} - 3\right)^3. \tag{29}$$

For $7 \leq q \leq 8$, $q/2 \in [3.5, 4]$, where values map to 4. Hence,

$$E\left(\frac{q}{2}\right) = \frac{5}{24} + \frac{1}{3}\left(\frac{q}{2} - 4\right)^3. \tag{30}$$

Substitution into $\Delta(q) = E(q) - 8E(q/2)$ gives

$$\Delta(q) = \begin{cases} -\dfrac{1}{8}, & 6 \leq q \leq 7, \\ \dfrac{(4q - 29)(4q - 27)}{8}, & 7 \leq q \leq 8. \end{cases} \tag{31}$$

C Global Optimality of the UOS Boundary

Since $D(x) = E(x)/x^3$,

$$D(q) - D\left(\frac{q}{2}\right) = \frac{E(q) - 8E(q/2)}{q^3} = \frac{\Delta(q)}{q^3}. \tag{32}$$

C.1 Justification of Differentiation

We justify differentiating the moving-boundary integral without requiring H or g to be continuous. The key is to express the objective as the difference of two values of an absolutely continuous integral primitive.

Let $F(x) = D(x)g(x)$. Since D is continuous on $(0, \infty)$ and g is locally integrable, $F \in L^1_{\text{loc}}(0, \infty)$. Define

$$G(t) = \int_1^t F(x)\, dx. \tag{33}$$

Then G is absolutely continuous on every compact subinterval of $(0, \infty)$ and $G'(t) = F(t)$ for almost every t. Since

$$\mathcal{J}(q) = G(q) - G(q/2), \tag{34}$$

$\mathcal{J}$ is locally absolutely continuous and, for almost every q,

$$\mathcal{J}'(q) = F(q) - \frac{1}{2}F(q/2). \tag{35}$$

Using Eq. (24),

$$\mathcal{J}'(q) = D(q)g(q) - \frac{1}{2}D\left(\frac{q}{2}\right) g\left(\frac{q}{2}\right) \tag{36}$$

$$= g(q)\left[D(q) - D\left(\frac{q}{2}\right)\right] \tag{37}$$

$$= \frac{g(q)}{q^3}\Delta(q). \tag{38}$$

The nonnegative factor $g(q)$ preserves the sign of $\Delta(q)$ wherever it is positive and makes the derivative zero wherever it vanishes. Local integrability and absolute continuity establish this derivative relation almost everywhere. Density values on null sets leave both $\mathcal{J}$ and its almost-everywhere derivative unchanged.

Table 7 gives the exact piecewise form of $\Delta(q)$ over $q > 0$.

The quadratic in the final row has discriminant $-18240 < 0$ and a positive leading coefficient, so it is strictly positive. Therefore,

$$\Delta(q) \begin{cases} = 0, & 0 < q \le \frac{1}{4}, \\ < 0, & \frac{1}{4} < q < \frac{29}{4}, \\ = 0, & q = \frac{29}{4}, \\ > 0, & q > \frac{29}{4}. \end{cases} \tag{39}$$

Combining Eqs. (38) and (39), $\mathcal{J}(q)$ is nonincreasing before $q = 29/4$ and nondecreasing afterward. Hence $q^\star = 29/4$, equivalently $Q^\star_{\max} = 7.25$, is a global minimizer for every absolutely continuous law of $\log_2 M$ within the stated scale family.

If H is positive almost everywhere over one period, $\mathcal{J}(q)$ is constant on $0 < q \le 1/4$, strictly decreasing on $(1/4, 29/4)$, and strictly increasing on $(29/4, \infty)$. Therefore, $Q^\star_{\max} = 29/4$ is the unique global minimizer.

C.2 Optimality under Scalar Block Weighting

The weighted case changes the nonnegative density factor but preserves its period-one structure. We prove Corollary 1 by repeating the boundary-ratio argument with g_W, under the assumptions of

Table 7: Exact piecewise form and sign of $\Delta(q) = E(q) - 8E(q/2)$.

Interval	$\boldsymbol{\Delta(q)}$	**Sign**
$0 < q \le \frac{1}{4}$	0	0
$\frac{1}{4} < q \le \frac{1}{2}$	$-\dfrac{(4q-1)^2}{32}$	< 0
$\frac{1}{2} < q \le \frac{3}{4}$	$\dfrac{16q^2 - 24q + 7}{32}$	< 0
$\frac{3}{4} < q \le \frac{5}{4}$	$-\dfrac{1}{16}$	< 0
$\frac{5}{4} < q \le \frac{3}{2}$	$-\dfrac{16q^2 - 40q + 27}{32}$	< 0
$\frac{3}{2} < q \le \frac{7}{4}$	$\dfrac{(4q-9)(4q-5)}{32}$	< 0
$\frac{7}{4} < q \le 7$	$-\dfrac{1}{8}$	< 0
$7 < q \le 10$	$\dfrac{(4q-29)(4q-27)}{8}$	< 0 before $29/4$, $= 0$ at $29/4$, > 0 afterward
$q > 10$	$\dfrac{48q^2 - 864q + 3983}{8}$	> 0

Theorem 1. Let $W \ge 0$ satisfy $\mathbb{E}[W] < \infty$. The joint law of $U = \log_2 M$ and W is fixed as q varies; W may otherwise depend on the block magnitude or any other context. Define

$$\mathcal{J}_W(q) = \mathbb{E}[W D(X_q)], \qquad h(u) = f_U(u)\mathbb{E}[W \mid U = u]. \tag{40}$$

Then $h \ge 0$ and $\int_{\mathbb{R}} h(u)\,du = \mathbb{E}[W] < \infty$. Its periodization and the corresponding weighted density are

$$H_W(t) = \sum_{k\in\mathbb{Z}} h(t+k), \qquad g_W(x) = \frac{H_W(\log_2 x)}{x \ln 2}. \tag{41}$$

Tonelli's theorem gives $\int_0^1 H_W(t)\,dt = \mathbb{E}[W]$, so H_W is integrable over one period and g_W is locally integrable on $(0,\infty)$. Reindexing the nonnegative sum yields $H_W(t+1) = H_W(t)$, and hence

$$g_W(x/2) = 2g_W(x) \qquad \text{for almost every } x > 0. \tag{42}$$

Conditioning on U and applying the change of variables from Appendix A gives

$$\mathcal{J}_W(q) = \int_{q/2}^{q} D(x) g_W(x)\,dx. \tag{43}$$

Since D is continuous, Dg_W is locally integrable. The integral-primitive argument in Appendix C.1 therefore shows that $\mathcal{J}_W$ is locally absolutely continuous and, for almost every $q > 0$,

$$\begin{aligned} \mathcal{J}_W'(q) &= D(q)g_W(q) - \tfrac{1}{2}D(q/2)g_W(q/2) \\ &= g_W(q)\big[D(q) - D(q/2)\big] = \frac{g_W(q)}{q^3}\Delta(q). \end{aligned} \tag{44}$$

Because $g_W \ge 0$, the sign analysis in Eq. (39) applies unchanged: $\mathcal{J}_W$ is constant on $(0, 1/4]$, nonincreasing on $(1/4, 29/4)$, and nondecreasing on $(29/4, \infty)$. Thus $q^\star = 29/4 = 7.25$ is a global minimizer, proving the corollary. If $H_W > 0$ almost everywhere over one period, the decrease and increase on these latter intervals are strict, so the minimizer is unique. This argument also includes $\mathbb{E}[W] = 0$, for which $\mathcal{J}_W \equiv 0$.

D Wrapped-Distribution and Harmonic Interpretation

The wrapped phase is the fractional part of $U = \log_2 M$: block maxima differing by a power of two have the same phase. Its density on one period is a wrapped distribution (Mardia & Jupp, 2000; Bell & Nadarajah, 2024), which can be expressed as a sum of translated densities or through Fourier

coefficients. This appendix explains those representations. The Gaussian and broad-distribution examples below illustrate possible density shapes; the UOS proof in Appendices A–C uses exact periodicity and does not require either approximation.

To relate the phase to a boundary q, subtract $\log_2 q$ before taking the fractional part:

$$Y_q = (U - \log_2 q) \bmod 1 \in [0, 1). \tag{45}$$

For noninteger $U - \log_2 q$, which occurs almost surely under an absolutely continuous law,

$$X_q = q\, 2^{Y_q - 1}. \tag{46}$$

At integer endpoints, the ceiling convention maps the value to q; this measure-zero distinction does not affect the density or the quantization-error integrals.

If Y_q is uniformly distributed, the Jacobian gives

$$f_{X_q}(x) = \frac{1}{x \ln 2}, \qquad x \in (q/2, q]. \tag{47}$$

For the ceiling-based $q = 8$ interval $(4, 8]$, which differs from the exact OCP interval $[4, 8)$ only at measure-zero endpoints, this approximation gives

$$\Pr(X_8 > 7) = \int_7^8 \frac{1}{x \ln 2}\, dx = \log_2 \frac{8}{7} \approx 19.27\%. \tag{48}$$

The uniform wrapped-phase approximation is used only for this estimate; Theorem 1 does not require it.

D.1 Fourier Representation

The Fourier representation explains why nonuniformity does not alter the boundary relation in Lemma 1. Under standard convergence conditions (Mardia & Jupp, 2000; Bell & Nadarajah, 2024; Stein & Shakarchi, 2003),

$$H(t) = 1 + \sum_{n \neq 0} c_n e^{i2\pi n t}, \quad e^{i2\pi n \log_2(q/2)} = e^{i2\pi n(\log_2 q - 1)} = e^{i2\pi n \log_2 q}. \tag{49}$$

The constant term is uniform, and every harmonic takes the same value at both endpoints regardless of amplitude. No near-uniform approximation is needed; the lemma relies only on periodicity, without requiring Fourier convergence.

The Fourier coefficients of the periodized density H are

$$c_n = \int_0^1 H(t) e^{-i2\pi n t}\, dt \tag{50}$$

$$= \int_{-\infty}^{\infty} f_U(u) e^{-i2\pi n u}\, du \tag{51}$$

$$= \phi_U(-2\pi n), \tag{52}$$

where $\phi_U(\omega) = \mathbb{E}[e^{i\omega U}]$. The zeroth coefficient is $c_0 = 1$. Under standard Fourier-convergence conditions (Stein & Shakarchi, 2003),

$$H(t) = 1 + \sum_{n \neq 0} \phi_U(-2\pi n) e^{i2\pi n t}. \tag{53}$$

For a general integrable density, the coefficient identity remains valid even when pointwise convergence of the Fourier series is not assumed.

Equation (53) gives the harmonic interpretation of Lemma 1. For every integer n,

$$e^{i2\pi n(\log_2 q - 1)} = e^{i2\pi n \log_2 q}. \tag{54}$$

Thus, the uniform term and every nonzero Fourier mode take the same value at the two scale boundaries. The harmonic components can change the objective value and the magnitude of its derivative, but they cannot reverse the derivative sign or shift the UOS minimizer.

Table 8: First-harmonic amplitude under a Gaussian log-domain model, expressed as a percentage of the uniform density $H(t) = 1$. The values describe the leading Fourier component, not the complete pointwise deviation of the wrapped density.

σ	σ^2	First-harmonic amplitude $\alpha(\sigma)$ (%)
0.10	0.01	164.17
0.20	0.04	90.81
0.30	0.09	33.84
0.40	0.16	8.50
0.50	0.25	1.44
0.60	0.36	0.164
1.00	1.00	5.35×10^{-7}

As a concrete example, if $U \sim \mathcal{N}(\mu, \sigma^2)$, the Fourier series becomes

$$H(t) = 1 + 2 \sum_{n=1}^{\infty} e^{-2\pi^2 n^2 \sigma^2} \cos\left(2\pi n(t-\mu)\right). \tag{55}$$

The first-harmonic amplitude under this Gaussian model is

$$\alpha(\sigma) = 2e^{-2\pi^2\sigma^2}. \tag{56}$$

Higher harmonics may remain non-negligible when σ^2 is small, so $\alpha(\sigma)$ should not be interpreted as a bound on the total deviation from uniformity. Substituting a fitted variance yields a Gaussian-model estimate of this amplitude, not a bound on the empirical density's deviation from uniformity.

E Wrapped-Phase Convergence for Location–Scale Families

We next describe when the wrapped phase approaches a uniform distribution as the log-domain distribution broadens.

Let $U_\sigma = \mu + \sigma Z$, where Z has an absolutely continuous density. For every nonzero integer n,

$$|\phi_{U_\sigma}(-2\pi n)| = |\phi_Z(-2\pi n\sigma)|. \tag{57}$$

Since Z has an integrable density, the Riemann–Lebesgue lemma gives $\lim_{|t|\to\infty} \phi_Z(t) = 0$ (Stein & Shakarchi, 2003). Therefore, $\phi_{U_\sigma}(-2\pi n) \to 0$ for every $n \neq 0$. By the Fourier characterization of the uniform distribution modulo one (Kuipers & Niederreiter, 1974),

$$U_\sigma \bmod 1 \xrightarrow{d} \mathcal{U}(0,1). \tag{58}$$

Thus, broad absolutely continuous location–scale families approach a uniform wrapped phase. Without additional regularity assumptions, this weak convergence does not imply uniform convergence of the corresponding densities.

F Empirical Wrapped-Phase Distributions

Figure 8 compares Wan2.2 block maxima, their logarithms, scaled values, and fractional log-domain phases. The empirical phases are visibly nonuniform; their harmonics illustrate how real distributions can differ from the uniform-phase example in Appendix D.

Figure 9 further examines how Hadamard rotation changes the empirical block-maximum distributions. Although the rotation modifies the log-domain harmonic structure, the periodic boundary relation used by UOS remains unchanged.

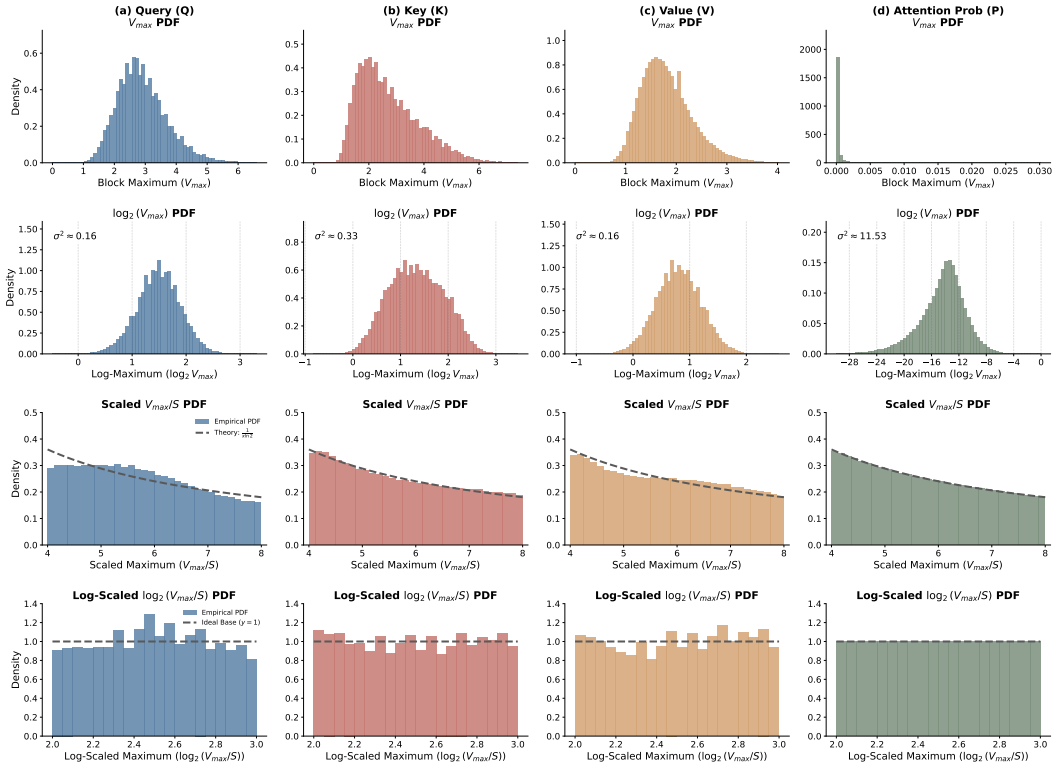


Figure 8: **Empirical block-maximum distributions in Wan2.2.** The rows show the original block maxima, their logarithms, the corresponding normalized linear-domain values, and the wrapped log-domain phases for Q, K, V, and the normalized attention probability matrix P. The visible nonuniformity corresponds to distribution-specific harmonic components but does not change the UOS boundary condition. The P column is included only to examine its distribution; PNQ quantizes the unnormalized softmax exponential tile.

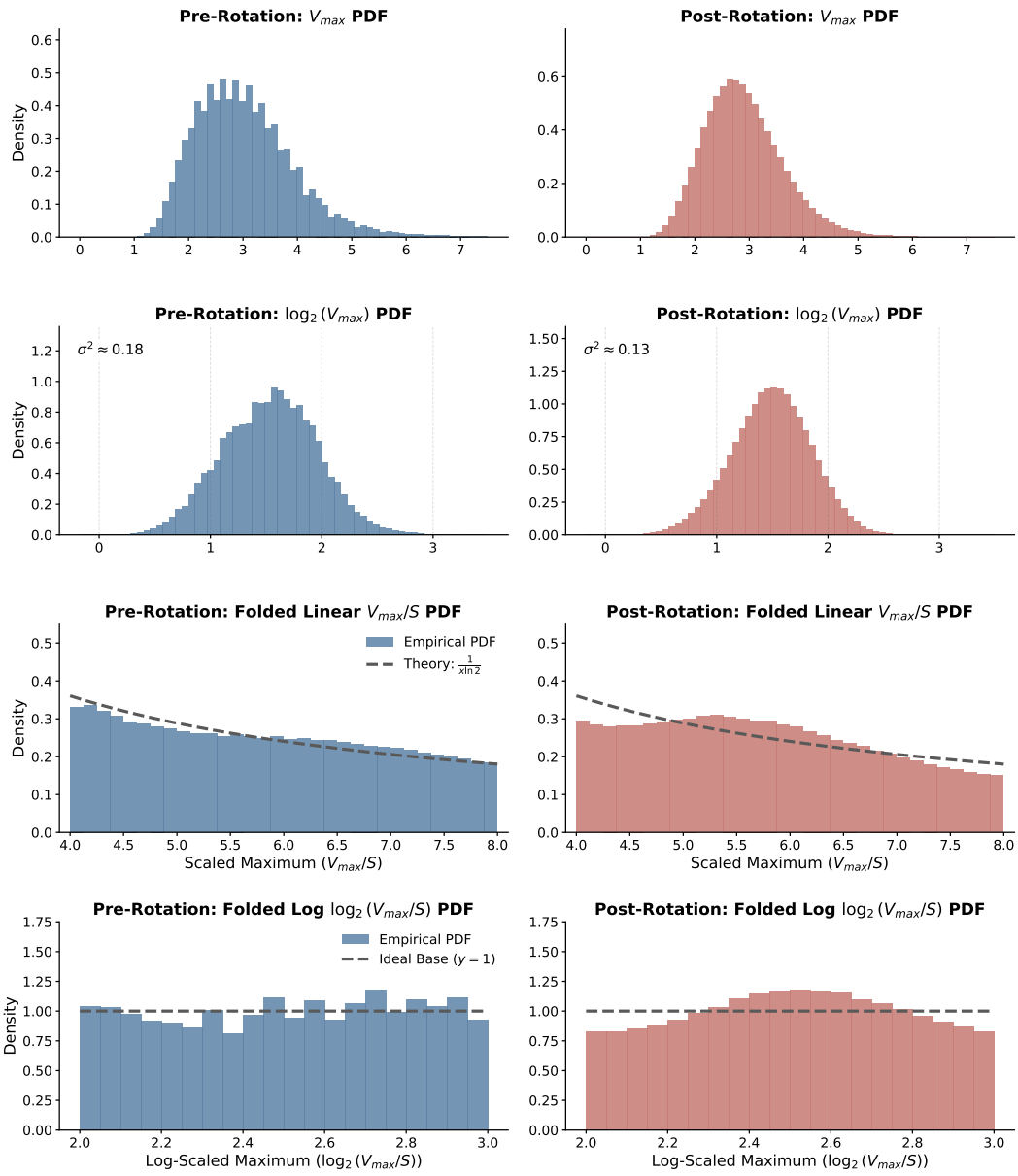


Figure 9: **Effect of Hadamard rotation on block-maximum distributions.** Rotation changes the log-domain distribution and its harmonic content; the boundary density ratio remains unchanged.

G Relation to OAS and the OCP Endpoint Convention

We consider two distinct comparisons. First, we relate OAS to the boundary $q = 7$ at a fixed block size of 32. We then distinguish the exact OCP floor rule used in the experiments from its ceiling-based counterpart in the continuous analysis.

At the common 32-element block granularity, the OAS scale-selection rule is equivalent to the ceiling-based boundary $q = 7$. Let

$$e_6 = \left\lceil \log_2 \left(\frac{M}{6} \right) \right\rceil, \qquad x_6 = \frac{M}{2^{e_6}} \in (3, 6]. \tag{59}$$

Then

$$e_7 = e_6 + \left\lceil \log_2 \left(\frac{x_6}{7} \right) \right\rceil = \begin{cases} e_6 - 1, & 3 < x_6 \le 3.5, \\ e_6, & 3.5 < x_6 \le 6. \end{cases} \tag{60}$$

Thus, $q = 7$ halves the TFS scale when $x_6 \in (3, 3.5]$, mapping this interval to $(6, 7]$, and otherwise leaves the scale unchanged. This is the OAS scale-selection rule at the same block granularity (Chhugani et al., 2026). The comparison does not include the smaller block size used in the full OAS configuration.

The experimental OCP baseline uses the exact floor-based conversion rule

$$e_{\mathrm{OCP}} = \lfloor \log_2 M \rfloor - 2 = \left\lfloor \log_2 \left(\frac{M}{8} \right) \right\rfloor + 1. \tag{61}$$

The ceiling-based $q = 8$ rule used in the continuous boundary analysis is

$$e_8(M) = \left\lceil \log_2 \left(\frac{M}{8} \right) \right\rceil. \tag{62}$$

The two expressions agree whenever $\log_2(M/8) \notin \mathbb{Z}$, since $\lceil z \rceil = \lfloor z \rfloor + 1$ for noninteger z. They differ when $M/8$ is an exact power of two. Under an absolutely continuous law, these endpoints have measure zero and do not affect the quantization-error integrals or Theorem 1. All reported OCP experiments use the exact floor-based rule.

H Full Online-Softmax Form of PNQ

We first express the final accumulators using a common set of rescaled tile weights. Appendix I shows why direct placement instead uses different weights in the numerator and denominator. Appendix J establishes that PNQ's denominator is positive.

Let T_c be the number of key/value tiles processed for query tile i. For each tile j, define the row-wise rescaling factor from its current running maximum to the final running maximum as

$$\beta_i^{(j)} = \exp\left(m_i^{(j)} - m_i^{(T_c)} \right). \tag{63}$$

Because

$$\alpha_i^{(r)} = \exp\left(m_i^{(r-1)} - m_i^{(r)} \right), \tag{64}$$

the element-wise product of all subsequent rescaling factors telescopes:

$$\prod_{r=j+1}^{T_c} \alpha_i^{(r)} = \exp\left(m_i^{(j)} - m_i^{(T_c)} \right) = \beta_i^{(j)}, \tag{65}$$

where the empty product for $j = T_c$ is the all-ones vector.

Lemma 2 (Effective Full-Row Form)**.** *After all key/value tiles have been processed, PNQ satisfies*

$$\begin{aligned} \ell_i^{(T_c)} &= \sum_{j=1}^{T_c} \beta_i^{(j)} \odot \left(\widehat{P}_{ij} \mathbf{1}_{B_c} \right), \\ \widetilde{O}_i^{(T_c)} &= \sum_{j=1}^{T_c} \mathrm{Diag}\left(\beta_i^{(j)} \right) \widehat{P}_{ij} \widehat{V}_j. \end{aligned} \tag{66}$$

Proof. Unrolling Eq. (10), the contribution of tile j is multiplied by every subsequent rescaling factor $\alpha_i^{(r)}$, for $r = j+1, \ldots, T_c$. Equation (65) reduces their element-wise product to $\beta_i^{(j)}$, yielding Eq. (66). ■

To identify the weights actually used in the final output, we express every quantized tile relative to the same final row maximum. Define the rescaled tile

$$\widehat{P}_{ij}^{\text{eff}} = \text{Diag}\Big(\beta_i^{(j)}\Big)\, \widehat{P}_{ij}, \tag{67}$$

and concatenate these tiles in key order. The resulting matrix records all coefficients used in the final weighted sum; it is a mathematical representation that the kernel never stores:

$$\widehat{P}_i^{\text{eff}} = \left[\widehat{P}_{i1}^{\text{eff}}, \ldots, \widehat{P}_{iT_c}^{\text{eff}}\right]. \tag{68}$$

Let $\widehat{V}$ denote the corresponding vertical concatenation of $\widehat{V}_1, \ldots, \widehat{V}_{T_c}$. Lemma 2 then gives

$$\ell_i^{(T_c)} = \widehat{P}_i^{\text{eff}} \mathbf{1}, \qquad \widetilde{O}_i^{(T_c)} = \widehat{P}_i^{\text{eff}} \widehat{V}. \tag{69}$$

These equalities express both accumulators in terms of the same effective quantized exponential values.

I Full-Row Mismatch under Direct Placement

For direct placement, we use the same final-maximum rescaling to compare the quantized values in the output accumulator with the unquantized values in the normalizer. Define

$$\widetilde{P}_{ij}^{\text{eff}} = \text{Diag}\Big(\beta_i^{(j)}\Big)\, \widetilde{P}_{ij}, \qquad \widehat{P}_{ij}^{\text{eff}} = \text{Diag}\Big(\beta_i^{(j)}\Big)\, \widehat{P}_{ij}, \tag{70}$$

and concatenate the tiles across the key sequence:

$$\widetilde{P}_i^{\text{eff}} = \left[\widetilde{P}_{i1}^{\text{eff}}, \ldots, \widetilde{P}_{iT_c}^{\text{eff}}\right], \qquad \widehat{P}_i^{\text{eff}} = \left[\widehat{P}_{i1}^{\text{eff}}, \ldots, \widehat{P}_{iT_c}^{\text{eff}}\right]. \tag{71}$$

The final direct-placement states are

$$\ell_i^{\text{direct}} = \widetilde{P}_i^{\text{eff}} \mathbf{1}, \qquad \widetilde{O}_i^{\text{direct}} = \widehat{P}_i^{\text{eff}} \widehat{V}. \tag{72}$$

Hence, the final weights induced by the direct placement are

$$\widehat{P}_i^{\text{direct}} = \text{Diag}\Big(\widetilde{P}_i^{\text{eff}} \mathbf{1}\Big)^{-1}\, \widehat{P}_i^{\text{eff}}, \tag{73}$$

with row sums

$$\widehat{P}_i^{\text{direct}} \mathbf{1} = \Big(\widehat{P}_i^{\text{eff}} \mathbf{1}\Big) \oslash \Big(\widetilde{P}_i^{\text{eff}} \mathbf{1}\Big). \tag{74}$$

Here, $\oslash$ denotes element-wise division between the two row-wise sum vectors. The running-max rescaling is the same in both states, but the exponential tiles are not; the final row sum therefore need not equal one.

J Positive PNQ Normalization Factor

To show that the final division is well defined, we identify one contribution to each normalizer that survives quantization exactly. Fix any row r of query tile i. All attention rows considered in this work have a nonempty key set. Let tile $j^\star$ contain an entry $c^\star$ attaining the final maximum of row r:

$$S_{ij^\star}[r, c^\star] = m_i^{(T_c)}[r]. \tag{75}$$

When tile $j^\star$ is processed, the running maximum for this row reaches its final value:

$$m_i^{(j^\star)}[r] = m_i^{(T_c)}[r]. \tag{76}$$

Therefore,

$$\widetilde{P}_{ij^\star}[r, c^\star] = \exp\Big(S_{ij^\star}[r, c^\star] - m_i^{(j^\star)}[r]\Big) = 1, \tag{77}$$

and

$$\beta_i^{(j^\star)}[r] = 1. \tag{78}$$

Every entry of $\widetilde{P}_{ij^\star}$ lies in $(0, 1]$, and the MXFP4 block containing $\widetilde{P}_{ij^\star}[r, c^\star]$ therefore has maximum magnitude one. Under the UOS boundary $q^\star = 7.25$,

$$e_{q^\star}(1) = \left\lceil \log_2 \left(\frac{1}{7.25} \right) \right\rceil = -2. \tag{79}$$

The shared scale is $2^{-2} = 1/4$, so the normalized value is

$$\frac{1}{2^{-2}} = 4, \tag{80}$$

which is exactly representable in E2M1. Hence,

$$\mathcal{Q}_\star(1) = 1. \tag{81}$$

Since this entry also has $\beta_i^{(j^\star)}[r] = 1$, it contributes exactly one to the final row-wise sum. Therefore,

$$\ell_i^{(T_c)}[r] \geq 1 > 0. \tag{82}$$

The final division can now be written using the effective weights defined in Appendix H:

$$P_i^{\mathrm{PNQ}} = \mathrm{Diag}\Big(\widehat{P}_i^{\mathrm{eff}}\mathbf{1}\Big)^{-1} \widehat{P}_i^{\mathrm{eff}}. \tag{83}$$

Proposition 1 (Row-wise normalization)**.** *For the nonempty attention rows considered here,* $P_i^{\mathrm{PNQ}}\mathbf{1} = \mathbf{1}$ *in exact arithmetic.*

Proof. The preceding argument gives $\ell_i^{(T_c)}[r] > 0$ for every row. Equation (83) divides each row of $\widehat{P}_i^{\mathrm{eff}}$ by its own positive sum, yielding $P_i^{\mathrm{PNQ}}\mathbf{1} = \mathbf{1}$. ■

The normalization guarantee concerns the weights induced by the PNQ recurrence in exact arithmetic. It does not assert that the quantized weights equal the full-precision softmax weights. In an implementation, finite-precision accumulation and the final division may introduce ordinary floating-point rounding. PNQ removes the additional structural mismatch caused by updating ℓ_i with $\widetilde{P}_{ij}$ while updating $\widetilde{O}_i$ with $\widehat{P}_{ij}$.

K Numerical Settings for the UOS Objective Curves

The objective curves in Figure 4 use the following unit-period wrapped densities, corresponding to the plot legends:

$$\begin{aligned} H_{\text{log-uniform}}(t) &= 1, \\ H_{\text{first}}(t) &= 1 + 0.70\cos\big(2\pi(t - 0.15)\big), \\ H_{\text{two}}(t) &= 1 + 0.45\cos\big(2\pi(t - 0.36)\big) + 0.25\cos\big(4\pi(t - 0.08)\big), \\ H_{\text{concentrated}}(t) &= \frac{\exp\big(2\cos(2\pi(t - 0.62))\big)}{I_0(2)}, \end{aligned}$$

where I_0 is the modified Bessel function of the first kind of order zero. For each density, we evaluate

$$J_H(q) = \int_0^1 D\big(q2^{y-1}\big) H\big(\log_2 q + y - 1\big)\, dy, \tag{84}$$

using the composite trapezoidal rule with 20,001 equally spaced phase points and float64 arithmetic. The sweep contains 1,201 equally spaced values of q on $[6, 8]$. The distribution-comparison panel reports the relative objective increase $100[J_H(q)/J_H(7.25) - 1]$ in percent, normalized separately for each density.

L Standard Attention and Scaling Background

Attention and FlashAttention. Given query, key, and value matrices Q, K, and V, scaled dot-product attention computes

$$S = \frac{QK^\top}{\sqrt{d}}, \qquad P = \mathrm{Softmax}(S), \qquad O = PV, \tag{85}$$

where every row of P satisfies $\sum_j P_{ij} = 1$. A conventional implementation materializes the score matrix S and probability matrix P, whose sizes grow quadratically with sequence length. FlashAttention avoids writing these matrices to high-bandwidth memory (HBM) by partitioning Q, K, and V into tiles and evaluating attention through an online-softmax recurrence in on-chip memory (Dao et al., 2022; Dao, 2024; Shah et al., 2024).

We use the FlashAttention-2 recurrence, which normalizes the output after all key/value tiles have been processed. Here, P denotes the final normalized attention matrix, $\widetilde{P}_{ij}$ the unnormalized exponential tile, and $\widetilde{O}_i^{(j)}$ the unnormalized output accumulator.

For query tile Q_i and key/value tiles K_j, V_j, define

$$S_{ij} = \frac{Q_i K_j^\top}{\sqrt{d}}. \tag{86}$$

FlashAttention maintains a running row maximum $m_i^{(j)}$, a row-sum normalizer $\ell_i^{(j)}$, and an output accumulator $\widetilde{O}_i^{(j)}$. The same exponentials contribute their row sums to $\ell_i^{(j)}$ and their weighted-value sums to $\widetilde{O}_i^{(j)}$. Starting from $m_i^{(0)} = -\infty$, $\ell_i^{(0)} = 0$, and $\widetilde{O}_i^{(0)} = 0$, it updates

$$m_i^{(j)} = \max\left(m_i^{(j-1)}, \mathrm{rowmax}(S_{ij})\right), \tag{87}$$

$$\alpha_i^{(j)} = \exp\left(m_i^{(j-1)} - m_i^{(j)}\right), \tag{88}$$

$$\widetilde{P}_{ij} = \exp\left(S_{ij} - m_i^{(j)} \mathbf{1}_{B_c}^\top\right), \tag{89}$$

$$\ell_i^{(j)} = \alpha_i^{(j)} \odot \ell_i^{(j-1)} + \widetilde{P}_{ij} \mathbf{1}_{B_c}, \tag{90}$$

$$\widetilde{O}_i^{(j)} = \mathrm{Diag}\left(\alpha_i^{(j)}\right) \widetilde{O}_i^{(j-1)} + \widetilde{P}_{ij} V_j. \tag{91}$$

Here, $\widetilde{P}_{ij}$ contains the unnormalized softmax exponentials and enters the second attention GEMM; it is not the final normalized probability tile. After all T_c key/value tiles have been processed,

$$O_i = \mathrm{Diag}\left(\ell_i^{(T_c)}\right)^{-1} \widetilde{O}_i^{(T_c)}. \tag{92}$$

FlashAttention therefore computes the normalized output from the row-wise sum $\ell_i^{(T_c)}$ and the unnormalized output accumulator $\widetilde{O}_i^{(T_c)}$, without materializing the full probability matrix P. Materializing P for quantization would reintroduce quadratic intermediate storage and memory traffic. A FlashAttention-compatible low-bit implementation must instead integrate quantization into the tiled online-softmax loop.

MXFP4 Block Quantization. MXFP4 is a microscaling format defined by the OCP MX specification (Open Compute Project, 2023; Rouhani et al., 2023). Each block contains $B = 32$ 4-bit E2M1 values that share a power-of-two scale $\sigma = 2^e$, encoded in 8-bit E8M0. The E2M1 grid is

$$\mathcal{G}_{\mathrm{E2M1}} = \{0, \pm 0.5, \pm 1, \pm 1.5, \pm 2, \pm 3, \pm 4, \pm 6\}. \tag{93}$$

For an element v, quantization divides by σ and rounds to the E2M1 grid. Multiplying by σ gives the reconstructed value, which we denote by

$$\mathcal{Q}_\sigma(v) = \sigma\, \Pi_{\mathcal{G}_{\mathrm{E2M1}}}\left(\frac{v}{\sigma}\right), \tag{94}$$

where $\Pi_{\mathcal{G}_{\mathrm{E2M1}}}$ denotes round-to-nearest projection onto the finite E2M1 grid, with out-of-range values saturated at ± 6.

Let $M = \max_{v \in \mathcal{B}} |v|$ be the maximum magnitude of a nonzero block $\mathcal{B}$. The standard MX conversion rule computes

$$e_{\mathrm{OCP}} = \lfloor \log_2 M \rfloor - e_{\max}^{\mathrm{elem}}, \qquad \sigma_{\mathrm{OCP}} = 2^{e_{\mathrm{OCP}}}, \tag{95}$$

where $e_{\max}^{\mathrm{elem}}$ is the exponent of the largest power-of-two value in the element format (Rouhani et al., 2023). For E2M1, $e_{\max}^{\mathrm{elem}} = 2$, corresponding to the grid value 4.

We express different scale-selection rules through a common boundary $Q_{\max}$:

$$e_{\mathrm{floor}}(Q_{\max}) = \left\lfloor \log_2 \left(\frac{M}{Q_{\max}} \right) \right\rfloor + 1, \qquad \sigma_{\mathrm{floor}}(Q_{\max}) = 2^{e_{\mathrm{floor}}(Q_{\max})}, \tag{96}$$

$$e_{\mathrm{ceil}}(Q_{\max}) = \left\lceil \log_2 \left(\frac{M}{Q_{\max}} \right) \right\rceil, \qquad \sigma_{\mathrm{ceil}}(Q_{\max}) = 2^{e_{\mathrm{ceil}}(Q_{\max})}. \tag{97}$$

The exact OCP rule used in our experiments is $e_{\mathrm{floor}}(8)$, which maps the normalized block maximum into $[4, 8)$. Because 8 is not representable in E2M1, normalized block maxima above 7 fall into an overflow-rounding region and are saturated to 6.

TetraJet's Truncation-Free Scaling (TFS) corresponds to $e_{\mathrm{ceil}}(6)$, giving

$$3 < \frac{M}{\sigma_{\mathrm{ceil}}(6)} \leq 6, \tag{98}$$

which avoids overflow of the block maximum but can increase rounding and underflow errors for smaller values (Chen et al., 2025). The floor- and ceiling-based forms agree except when $M/Q_{\max}$ is an exact power of two; this measure-zero endpoint distinction does not affect the continuous analysis in Section 4.1.

UOS uses the ceiling-based rule with the fixed boundary $Q_{\max} = 7.25$, derived from the global quantization-error objective in Eq. (4). It requires no calibration or per-layer search. For an all-zero block, we set $\sigma = 1$ and quantize every element to zero.

M Extended Context

Video diffusion Transformers process video-token sequences whose length grows with spatial resolution and video duration (Kong et al., 2024; Wan Team et al., 2025; Yang et al., 2024). At every denoising step, the model performs a forward pass through its Transformer blocks. Fast sampling, distillation, and caching reduce the number of model evaluations or reuse intermediate computation (Luo et al., 2024; Wang et al., 2024; Liu et al., 2025a; Zhao et al., 2024). Low-precision attention reduces the cost of the $QK^\top$ and PV GEMMs within each forward pass, provided that Q, K, V, and the online-softmax exponential tiles can be quantized accurately (Zhang et al., 2025b; 2026).

MXFP4 uses 32-element E2M1 blocks with a shared E8M0 power-of-two scale. Defined by the OCP MX specification, this open format is supported by NVIDIA Blackwell GPUs, AMD Instinct MI350 series, and Ascend 950 series (Open Compute Project, 2023; NVIDIA, 2025a; Advanced Micro Devices, Inc., 2025; Huawei, 2025). NVIDIA's NVFP4 uses 16-element blocks, E4M3 block scales, and an additional tensor-level scale (NVIDIA, 2025b;c). MXFP4 uses less scale metadata and simpler scale representation, but values within each larger block must share a power-of-two scale. This restriction creates the clipping–underflow trade-off analyzed in Section 3.

QuaRot and SpinQuant use orthogonal transformations, including Hadamard transforms, to suppress activation outliers (Ashkboos et al., 2024; Liu et al., 2025b). The SageAttention series combines smoothing, scaling, and kernel optimization for low-bit attention (Zhang et al., 2025c;a;d;b). Quantization-aware training (QAT) adapts the model to FP4 attention during training (Zhang et al., 2026). These approaches can improve the values presented to the quantizer; UOS selects the MXFP4 shared scale using a fixed boundary derived from the format's error objective.

Work on microscaling quantization also uses overflow-aware scaling, empirical scale selection, and additional metadata (Chhugani et al., 2026; Gupta et al., 2026; Hu et al., 2026). UOS derives its boundary without calibration or per-layer search. Video-diffusion quantization can exploit spatiotemporal structure in model weights and activations (Li et al., 2026); PNQ addresses the different problem of preserving normalization inside the fused attention softmax loop. MXAttention combines UOS and PNQ with a fixed Hadamard rotation, without calibration or training.

N UOS Quantizer and Scaling Comparisons

Algorithm 1 UOS-Based MXFP4 Block Quantization

Input: A block $\mathcal{B} = \{v_i\}_{i=1}^{B}$, $B = 32$; UOS boundary $q^\star = 7.25$
Output: E8M0 shared scale σ; E2M1 elements $\{p_i\}_{i=1}^{B}$

$M \leftarrow \max_{1 \le i \le B} |v_i|$
if $\mathcal{B}$ is an all-zero block **then**
  $\sigma \leftarrow 1$
  $p_i \leftarrow 0, \quad i = 1, \ldots, B$
  return $\sigma, \{p_i\}_{i=1}^{B}$
end if
$e \leftarrow \lceil \log_2(M/q^\star) \rceil$
$\sigma \leftarrow 2^e$
for $i = 1$ to B **do**
  $p_i \leftarrow \Pi_{\mathcal{G}_{\mathrm{E2M1}}}(v_i/\sigma)$
end for
return $\sigma, \{p_i\}_{i=1}^{B}$

Comparison with Existing Scaling Rules. Table 9 compares OCP, TFS, and UOS at the standard 32-element MXFP4 block granularity. The exact OCP baseline uses a floor-based conversion rule, whereas TFS and UOS belong to the ceiling-based family. The relation to OAS is given separately in Appendix G.

Table 9: **Comparison of MXFP4 scaling rules at $B = 32$.** All normalized maxima above 6 map to the largest finite E2M1 value. The reported overflow-rounding interval is the stricter subset in which the nearest value on an unbounded E2M1 ladder would be 8 before finite-range saturation. The endpoint at 7 depends on the tie-breaking convention.

Method	Shared-scale exponent	Normalized maximum range	Overflow-rounding interval
OCP	$\lfloor \log_2 M \rfloor - 2$	$[4, 8)$	$(7, 8)$
TFS	$\lceil \log_2(M/6) \rceil$	$(3, 6]$	None
UOS	$\lceil \log_2(M/7.25) \rceil$	$(3.625, 7.25]$	$(7, 7.25]$

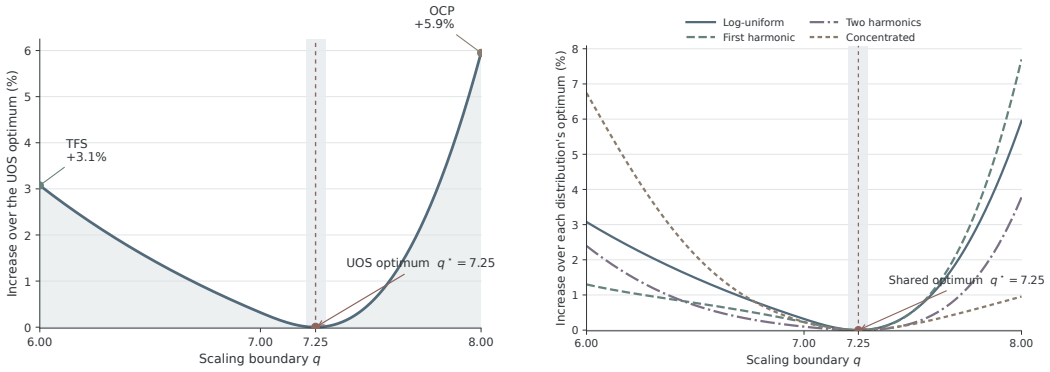
Relative to TFS, UOS selects a smaller shared scale for a subset of blocks, improving the resolution available to smaller values while admitting only the narrow overflow-rounding interval $(7, 7.25]$. Compared with OCP, the overflow-rounding interval shrinks from $(7, 8)$ to $(7, 7.25]$. The optimum does not come from setting a clipping term equal to an underflow term. Instead, Eq. (7) balances the total normalized projection errors at the two endpoints associated with adjacent power-of-two scale choices.

Figure 2 in the main text compares the three scaling ranges.

Figure 10 compares the analytical objective under a log-uniform density and representative nonuniform wrapped densities.

O Complete Algorithms and PNQ Illustration

Figure 3 gives the overall workflow; Algorithm 2 specifies the corresponding forward pass. Hats denote the numerical values represented by quantized tensors and their shared scales. They do not imply a separate full-precision materialization before each matrix product.



(a) Objective under a log-uniform density **(b) Objectives under wrapped densities**

Figure 10: **UOS objective under different wrapped distributions.** (a) Percentage increase in the objective relative to its value at $q = 7.25$ under a log-uniform density. (b) The same quantity under representative nonuniform wrapped densities. Distribution-specific harmonics change the curve shapes, but all cases share the minimum $q = 7.25$. Density definitions and numerical settings are given in Appendix K.

Algorithm 2: MXAttention Forward Pass

Input: Query, key, and value tensors Q, K, V; fixed Hadamard rotation R_{H}; UOS boundary $q^\star = 7.25$
Output: Attention output O

$Q^{\mathrm{rot}} \leftarrow QR_{\mathrm{H}}$
$K^{\mathrm{rot}} \leftarrow KR_{\mathrm{H}}$
$\widehat{Q} \leftarrow \mathcal{Q}_\star(Q^{\mathrm{rot}})$
$\widehat{K} \leftarrow \mathcal{Q}_\star(K^{\mathrm{rot}})$
$\widehat{V} \leftarrow \mathcal{Q}_\star(V)$
for each query tile i **do**
  $m_i \leftarrow -\infty, \quad \ell_i \leftarrow 0, \quad \widetilde{O}_i \leftarrow 0$
  for each key/value tile j **do**
    $S_{ij} \leftarrow \widehat{Q}_i \widehat{K}_j^\top / \sqrt{d}$
    $m_i^{\mathrm{new}} \leftarrow \max(m_i, \mathrm{rowmax}(S_{ij}))$
    $\alpha_i \leftarrow \exp(m_i - m_i^{\mathrm{new}})$
    $\widetilde{P}_{ij} \leftarrow \exp\big(S_{ij} - m_i^{\mathrm{new}} \mathbf{1}_{B_c}^\top\big)$
    $\widehat{P}_{ij} \leftarrow \mathcal{Q}_\star(\widetilde{P}_{ij})$
    $\ell_i \leftarrow \alpha_i \odot \ell_i + \widehat{P}_{ij} \mathbf{1}_{B_c}$
    $\widetilde{O}_i \leftarrow \mathrm{Diag}(\alpha_i)\widetilde{O}_i + \widehat{P}_{ij}\widehat{V}_j$
    $m_i \leftarrow m_i^{\mathrm{new}}$
  end for
  $O_i \leftarrow \mathrm{Diag}(\ell_i)^{-1}\widetilde{O}_i$
end for
return O

P Video Evaluation Details

P.1 Experimental Setup

Models. We evaluate two large-scale open-source text-to-video models with distinct attention architectures: `Wan2.2-14B` and `HunyuanVideo-13B`. `Wan2.2` employs separate self-attention and cross-attention modules, whereas `HunyuanVideo` adopts dual-stream and single-stream Transformer blocks with full attention.

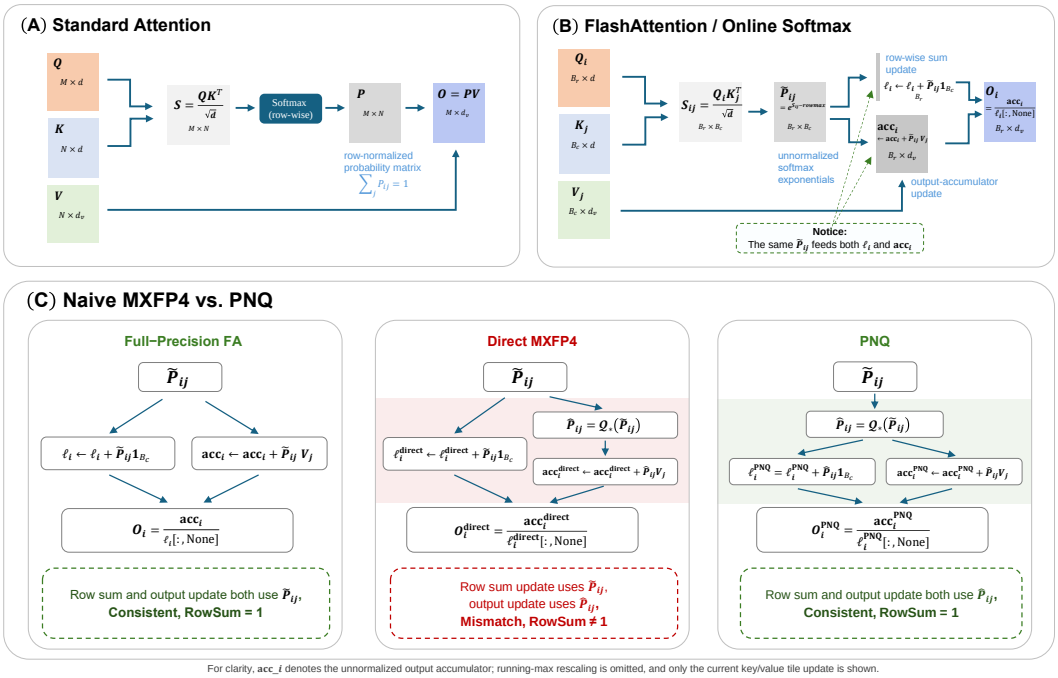


Figure 11: **Pre-Normalization Quantization (PNQ).** (A) Standard attention forms the normalized probability matrix P before computing $O = PV$. (B) FlashAttention avoids materializing P and updates both the row-wise sum of exponentials ℓ_i and the unnormalized output accumulator $\widetilde{O}_i$ from the same exponential tile $\widetilde{P}_{ij}$. (C) A direct placement uses the unquantized tile $\widetilde{P}_{ij}$ in the ℓ_i update but the quantized tile $\widehat{P}_{ij}$ in the $\widetilde{O}_i$ update. PNQ uses the same $\widehat{P}_{ij}$ in both updates. Running-max rescaling and previous key/value tiles are omitted for clarity.

Datasets & Generation. For the video-quality comparison and video ablation, all methods use the complete set of 48 prompts provided in Open-Sora's (Zheng et al., 2024b) `assets/texts/sora.csv`.[1] For each model–prompt pair, all methods share the same generation seed, enabling paired comparisons against the FP16 baseline.

Evaluation Metrics. We report video quality and agreement with the FP16 output separately. VBench (Huang et al., 2024) assesses generated-video quality through *Subject Consistency*, *Imaging Quality*, and *Aesthetic Quality*, averaged over all generated videos.

Reference-based metrics compare each quantized output with its matched FP16 output. Cosine similarity, Structural Similarity Index Measure (SSIM), Peak Signal-to-Noise Ratio (PSNR), and LPIPS (Zhang et al., 2018) are frame-level comparisons; the first three are averaged over frames. FID (Heusel et al., 2017) is computed separately for each generated video against its paired FP16 reference and then averaged across videos.

Implementation Details. We follow the recommended inference settings of both models: Wan2.2 generates 81-frame videos at 720p in 40 denoising steps, and HunyuanVideo generates 129-frame videos at 720p in 50 denoising steps. For video-quality evaluation, except for the Wan2.2 main comparison (Table 1), all quantized configurations use fully 4-bit attention, in which every attention block at every denoising step runs the 4-bit attention kernel while non-attention components retain their original precision. The Wan2.2 main comparison adopts a fixed hybrid-precision policy, keeping the final two denoising steps (38 and 39) in high precision while all remaining attention computations use 4-bit attention.

[1] https://github.com/hpcaitech/Open-Sora/blob/main/assets/texts/sora.csv

P.2 Baselines and Configurations

For video generation, we compare MXAttention with FP16 and representative 4-bit attention configurations:

- **FP16.** The unmodified attention implementation in FP16.
- **MXFP4 (OCP).** A direct MXFP4 attention baseline using the exact floor-based OCP conversion rule (Open Compute Project, 2023; Rouhani et al., 2023). The unnormalized softmax tile $\widetilde{P}$ is quantized into $\widehat{P} = \mathcal{Q}_{\mathrm{OCP}}(\widetilde{P})$ for the output-accumulator update, while the row-wise sum ℓ is updated from the unquantized tile.
- **NVFP4.** A direct NVFP4 implementation in which Q, K, V, and $\widetilde{P}$ are quantized for the two attention GEMMs. No additional smoothing or scaling of the exponential tiles is applied (NVIDIA, 2025b;c).
- **NVFP4 + SageAttention.** The NVFP4 baseline augmented with channel-wise Q/K smoothing and two-level scaling of the exponential tiles, following the SageAttention design (Zhang et al., 2025b).
- **MXAttention.** Our method applies a fixed Hadamard rotation to Q/K for outlier suppression (Ashkboos et al., 2024; Liu et al., 2025b), uses the UOS boundary $Q_{\max} = 7.25$ for all MXFP4 quantization steps, and applies PNQ so that the same quantized softmax tile updates both ℓ and the output-accumulator state.

Q Layer-Wise Scaling Search

We compare UOS with the OCP boundary $Q_{\max} = 8$, TFS with $Q_{\max} = 6$, and layer-wise empirical $Q_{\max}$ values selected by a grid search from 6.00 to 8.00 in increments of 0.05. For the video models, at the initial denoising step, we independently select $Q_{\max}$ for the Q, K, and V tensors in each of the 40 evaluated Transformer layers by minimizing the corresponding tensor MSE. The Qwen3-14B results use the same layer-wise search range and tensor-MSE criterion.

Each entry in Table 4 is the percentage of evaluated Transformer layers for which a boundary minimizes the corresponding tensor MSE within the tested grid. The Q, K, and V searches are independent. This sweep validates the fixed UOS boundary; it is not required when applying the method.

For Wan2.2, $Q_{\max} = 7.25$ minimizes tensor MSE within the grid in 95.0% of evaluated layers for Q, 90.0% for K, and 90.0% for V. On HunyuanVideo, 7.25 is the most frequent choice for Q and is tied in frequency with 7.20 for V; for K, 62.5% of layers select 7.20 and 32.5% select 7.25. On Qwen3-14B, 7.25 is the most frequent choice for all three tensors: 72.5% of layers for Q, 65.0% for K, and 77.5% for V. Across the three models, 7.25 is therefore the most frequent boundary, including one tie, in eight of the nine model–tensor pairs and the second-most frequent in the remaining pair.

The sweep minimizes finite-sample tensor MSE, whereas UOS minimizes the data-free analytical objective in Eq. (4). Their agreement supports using $Q_{\max} = 7.25$ without layer-wise calibration. Appendix R separately examines how quantization changes probability row sums.

R Row Sums after Probability Quantization

We store the normalized attention probability matrix P to examine how MXFP4 quantization changes its row sums. FlashAttention does not store P during inference, and PNQ quantizes the unnormalized exponential tile $\widetilde{P}$. This measurement captures rounding effects on an already normalized matrix; Section 4.2 establishes PNQ's guarantee for the online recurrence.

For each of the 80 tensors in Figure 12(A), we average the quantized row sums, then average these tensor means to obtain 0.9266. Every tensor mean is below one, although individual rows can have sums either below or above one. The aggregate therefore shows a negative bias across the evaluated layers and prompts.

Figure 12(B) separates the error by quantization outcome. Zeroing means that a positive probability becomes zero; saturation means that the rounded value in shared-scale units exceeds the E2M1

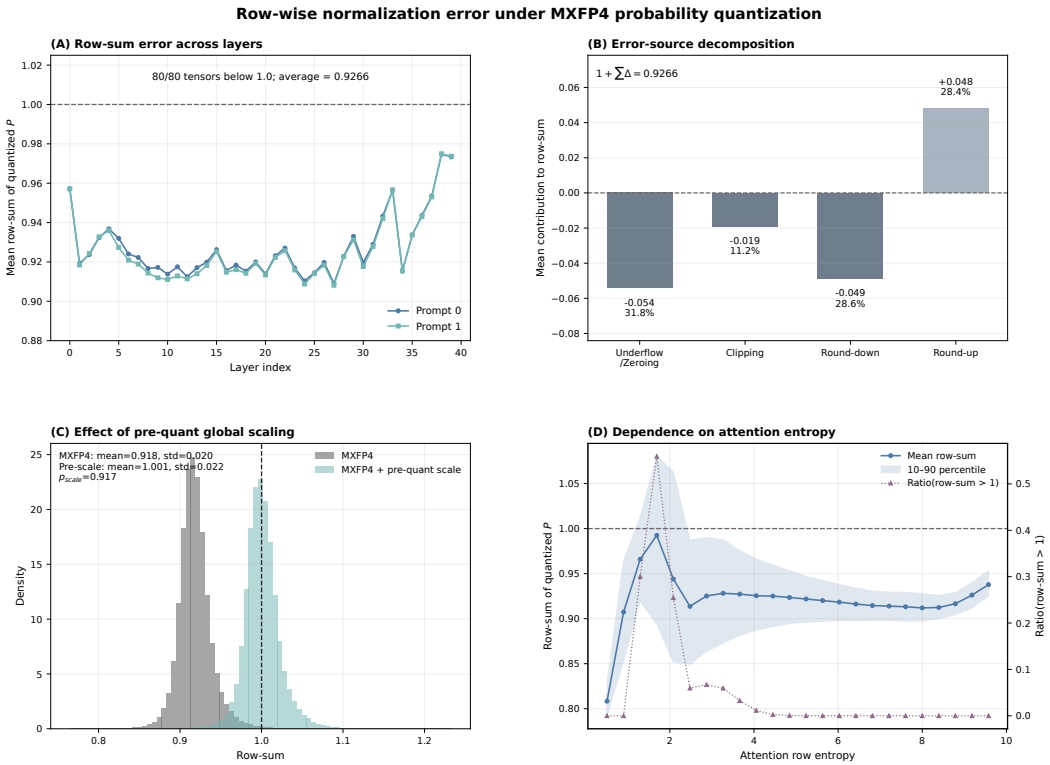


Figure 12: **Row sums after MXFP4 probability quantization.** (A) Across 80 Wan2.2 attention tensors at denoising step 9, covering two prompts and 40 layers, the mean row sum of each quantized tensor is below one. (B) The row-sum error is decomposed into zeroing, saturation, downward rounding, and upward rounding. (C) Pre-quant global scaling can restore the average row sum of a representative tensor but does not enforce normalization for individual rows. (D) High-entropy rows are predominantly under-normalized, whereas low-entropy rows may be under- or over-normalized.

range before clipping. Among the remaining entries, downward and upward rounding correspond to negative and positive reconstruction errors, respectively. For each category, we sum signed errors within each row, average over all rows of each tensor, and then average equally over the 80 tensors. The percentages in the figure are the absolute category contributions divided by their sum. Averaged across the tensors, zeroing contributes -0.0541, saturation contributes -0.0190, downward rounding contributes -0.0486, and upward rounding contributes $+0.0483$. On average, downward and upward rounding nearly cancel, while zeroing and saturation account for most of the row-sum deficit.

Figure 12(C) then examines whether one global pre-scaling factor can correct individual rows. The comparison uses $\widehat{P}_c = \mathcal{Q}_{\mathrm{OCP}}(P/c)$, with an empirically chosen scalar $c = 0.917$ shared across the tensor and no multiplication by c after quantization. In the illustrated tensor, pre-scaling restores the average row sum but increases the number of rows whose sums exceed one. Correcting the mean therefore does not normalize every row. This empirical factor is used only for this comparison and is not part of PNQ.

Finally, Figure 12(D) shows that row-sum behavior depends on attention entropy. Rows with entropy above 5 are almost always under-normalized, while lower-entropy rows exhibit errors in both directions. PNQ avoids this row-dependent normalization error by using the same quantized exponential tile in the row-wise sum and output-accumulator updates.

S Performance Protocol and Individual Runs

Attention latency. We compare BF16 attention, MXFP8, fused MXFP4, and MXAttention on a single Ascend 950-series NPU. We use synthetic BF16 $Q/K/V$ tensors with batch size 1, 40 heads, head dimension 128, and sequence length 75,776, matching the head configuration of Wan2.2-T2V-A14B. Attention is non-causal, and all sequence positions are valid. Inputs remain on the device. We measure attention latency from BF16 inputs to the output, including layout conversions, quantization, and rotation where applicable, and report median latency over 48 samples after warm-up using device-synchronized wall-clock timing.

Table 10: **Attention latency** on synthetic $Q/K/V$ tensors with $B = 1$, $H = 40$, $D = 128$, and $S = 75{,}776$. Latencies cover all computation from BF16 inputs to the attention output, including preprocessing. Speedups are computed relative to BF16 attention.

Method	Latency (ms)	Speedup vs. BF16 attention
BF16 attention	588.86	1.00×
MXFP8	405.66	1.45×
Fused MXFP4 baseline (PNQ enabled)	217.72	2.70×
MXAttention	218.01	2.70×

Table 10 shows that MXAttention takes 218.01 ms versus 217.72 ms for the PNQ-enabled MXFP4 baseline, a measured difference of 0.29 ms (0.13%). Both configurations use the same fused attention kernel and retain its native quantizer for the exponential tiles; this path does not use the UOS boundary 7.25. MXAttention uses the boundary $Q_{\max} = 7.25$ when quantizing $Q/K/V$, with a fixed Hadamard rotation applied to Q and K first. The reported latency includes all preprocessing in both configurations.

Kernel integration. UOS replaces the boundary in the existing shared-scale calculation. It retains the same block-maximum reduction, quantization granularity, and tensor traversal as direct MXFP4 quantization, with no additional processing step. PNQ obtains the row-sum normalizer ℓ from the quantized exponential tile $\widehat{P}$ already used by the output update. It requires neither another quantization operation nor another traversal of the attention tiles.

The fixed Hadamard transform is applied online only to Q and K. Rotation and quantization can overlap when scheduled on independent Q/K tiles using distinct compute units. Both operations act locally on the tiles and can be fused in Q/K preprocessing. Whether the operations overlap depends on the tile schedule; Table 10 reports attention latency including all preprocessing.

The implementation retains tiled attention without storing the full attention matrix or adding another pass over the key/value sequence.

End-to-end runtime. We also run Wan2.2-T2V-A14B in SGLang (Zheng et al., 2024a) on the same NPU, generating 81-frame videos at 1280×720 resolution with 40 denoising steps. We use one fixed prompt and seed, with two runs per method. MXAttention uses the same precision policy as the Wan2.2 main quality comparison: steps 38–39 retain high precision, and all remaining attention computations use 4-bit attention. The BF16 reference uses SGLang's native attention backend. We record total task time, including process startup, model loading, video generation, and validation checks.

Table 11: **Total task time** for Wan2.2-T2V-A14B video generation on one NPU. Both methods use the same prompt, seed, resolution, frame count, and denoising schedule. Speedup is the ratio of mean total task times over two runs per method.

Method	**Run 1 (s)**	**Run 2 (s)**	**Mean total task time (s)**	**Speedup vs. BF16 attention**
BF16 attention	2725	2835	2780.0	$1.00\times$
MXAttention	1504	1511	1507.5	$1.84\times$

As shown in Table 11, MXAttention reduces mean total task time from 2780.0 to 1507.5 s, a $1.84\times$ speedup (45.8% reduction) for this video-generation workload.