CONTEXT MODELING FOR EFFICIENT LOSS-RESILIENT NEURAL VIDEO COMPRESSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Rich context modeling gives modern neural video codecs strong rate-distortion performance, while resilience to packet loss has largely been bought by giving that modeling up. We pursue both at once. The obstacle we identify is receiver-side *entropy decoding*. Because the coding distribution is inferred from hyperprior, temporal, and spatial context, an erasure can cause the receiver to use a different distribution, corrupting symbols decoded from intact payloads and propagating damage to later frames. Holding retained payloads byte-identical and correcting only the entropy condition recovers $+5.64$ dB at 2% target packet loss. We formalize the property that excludes this failure, *entropy-condition locality*: the distribution used to decode a retained payload must not depend on what was erased elsewhere. Our codec, SEAL (**S**ender-decided **E**ntropy with **A**ssured **L**ocality), restores it by serializing the sender's final entropy decision instead of the dependencies that produced it. Sender-only staged refinement preserves rich conditional modeling, while the resulting scale is compressed into *packet-local entropy side information* bound to its residual payload. Both endpoints recover the same coding distribution from this information, without requiring the receiver to reproduce the sender's context. Locality settles interpretation, but the receiver must still reconstruct missing content from a reference state that can diverge from the sender's. Dual-trajectory training therefore optimizes reconstruction along a clean sender trajectory and a separate packet-erased receiver trajectory. With side-information cost included, SEAL lies on the measured rate-robustness frontier against GRACE at its reported rate and the evaluated H.264/H.265+FEC configurations under both independent and bursty loss, with gains across UVG, MCL-JCV, and HEVC sequences.

1 INTRODUCTION

Video delivered over a packet network must be reconstructed from whatever arrives, at a transmitted rate the channel can afford. These two demands have largely been met separately. Neural codecs that condition on rich context reach strong rate-distortion (RD) performance, but they assume the decoder can reproduce the conditions the encoder used; codecs designed for packet loss keep received content decodable, but adopt representations and architectures that give up much of that efficiency. What remains open is the combination: a codec whose context modeling remains usable over a lossy channel, so that good reconstruction under loss is obtained at a competitive transmitted rate.

The efficiency of modern codecs comes from conditioning. Hyperpriors, temporal features, and spatial context improve conditional entropy modeling (Ballé et al., 2018; Minnen et al., 2018; Li et al., 2021; 2023), with real-time designs such as DCVC-RT (Jia et al., 2025) making such conditioning practical for real-time delivery. The same conditioning ties entropy decoding to information a packet network may not deliver. Current hyperprior loss or reference-state divergence can make the receiver select a different discrete distribution, corrupting symbols decoded from intact payloads and affecting later frames. Section 2 develops this receiver-side failure; the sender still encodes correctly under its own intact state.

Codecs built for packet loss address the complementary half. GRACE (Cheng et al., 2024) independently entropy-codes packets with per-packet distribution information and uses loss-aware reconstruction training; NeuralMDC (Hu et al., 2024) independently codes descriptions and infers

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.

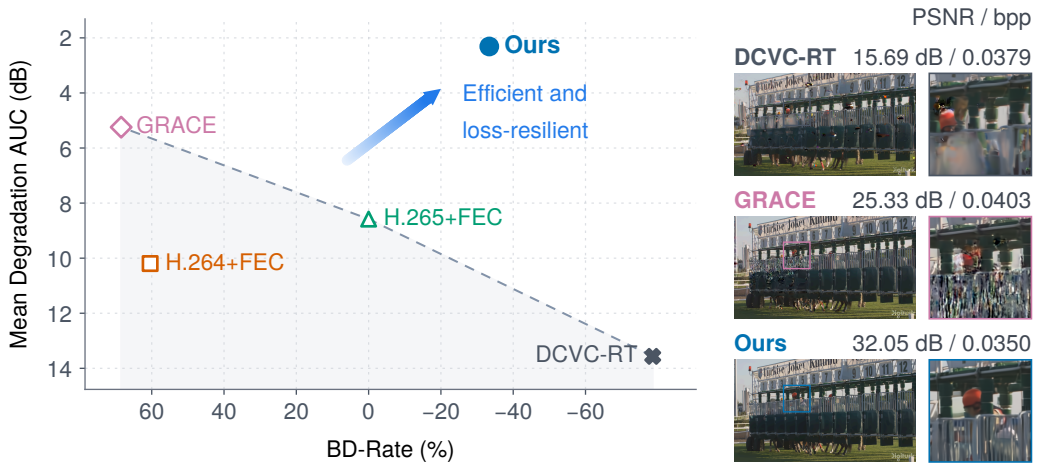

Figure 1: Clean-rate efficiency versus packet-loss resilience. Rightward indicates greater rate saving over H.265+FEC; upward indicates lower mean degradation AUC. GRACE uses reported rate; metric definitions are in Appendix A.2.

missing tokens. GRACE, built on a DVC-generation backbone (Lu et al., 2019), has no cross-packet entropy condition for an erasure to disturb, so it is not an instance of the failure analyzed here. The question is whether this packet-local interpretability can coexist with the rich conditional modeling of the DCVC family.

Our objective is the combination these two lines of work leave open: a codec that keeps useful context modeling and its RD benefit while remaining decodable and reconstructable under packet loss. Section 2 turns that into a checkable requirement we call *entropy-condition locality*: the distribution a receiver uses to decode a payload it did receive must not depend on what the channel erased elsewhere. Locality is a property of a design, not of a channel: a codec whose packets are independently interpretable has it, and a codec that infers its coding distribution from cross-packet context generally does not.

The principle we use to obtain it without surrendering context is to *serialize the decision, not the dependency*. Conditioning is cheap and reliable at the sender, which always holds its own intact state; what is fragile is requiring the receiver to re-derive that conditioning from a state the channel may have damaged. The sender can therefore keep its context and transmit the entropy decision that context produced, instead of asking the receiver to reproduce the inference behind it. Correct interpretation is necessary but not sufficient. The receiver still reconstructs from incomplete content, and from a reference state that no longer matches the sender's, so a second mechanism is needed past the entropy boundary.

Our codec, SEAL (**S**ender-decided **E**ntropy with **A**ssured **L**ocality), follows that order. First, the sender compresses its refined scale into packet-local entropy side information, binds it to the residual, and decodes that information itself so that both endpoints use the recovered CDF (Figure 2). Second, because the receiver no longer needs to replay this conditioning, sender-only staged refinement can preserve rich context-dependent modeling. Third, dual-trajectory training optimizes reconstruction along a clean sender trajectory and a separate packet-erased receiver trajectory, which addresses reconstruction, the part CDF agreement does not cover.

With side information and headers counted, SEAL lies on the measured rate-robustness frontier against GRACE at its reported rate and the evaluated H.264/H.265+FEC configurations (Figure 1). Matched ablations, cross-dataset evaluation, and a separate unified-MTU comparison test the combination and its transport assumptions (Section 4).

Contributions.

1. **A packet-loss failure specific to context-conditioned coding, and the condition that excludes it.** We identify corruption of retained payloads under receiver-side entropy-condition mismatch and formalize entropy-condition locality. A held-payload intervention quantifies the value of restoring that condition.

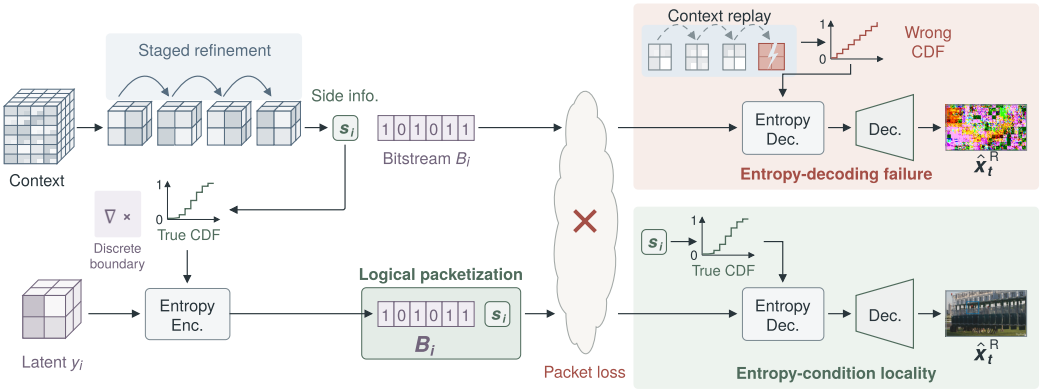

Figure 2: Context-dependent entropy decoding and the proposed design. Cross-packet context can cause CDF mismatch under loss (top right). The sender compresses its staged-refined scale into side information s_i, decodes s_i itself to select the CDF, and binds s_i with the residual bytes in one logical packet (left, center). The receiver recovers the same CDF from s_i without replaying context (bottom right); this agreement covers entropy interpretation only.

2. **Serializing the entropy decision instead of its dependencies.** Packet-local side information preserves received residuals' interpretation while sender-only staged refinement retains conditional modeling, with the side-information cost counted in rate.
3. **Dual-trajectory training for what locality does not cover.** Separate clean sender and packet-erased receiver trajectories optimize reconstruction under state divergence, without feedback or state resynchronization.
4. **SEAL, an efficient loss-resilient codec.** Matched ablations, rate-quality comparisons, cross-dataset results, and a separate unified-MTU evaluation place SEAL on the measured rate-robustness frontier, with a +4.87 dB average BD-PSNR gain over H.265+FEC.

2 Problem: Context Dependencies under Packet Loss

We analyze the context-conditioned backbone before adding the entropy side information of Section 3. The failure applies when entropy conditions span loss units (through hyperprior, temporal, or cross-group context), not to codecs whose packets already carry the parameters needed for entropy decoding.

2.1 How an erasure corrupts bytes that arrived

In a conditional neural codec, the sender entropy-codes a retained unit i into bytes B_i using a discrete CDF sequence. The receiver must select the same sequence to recover those symbols. Packet loss leaves the sender's coding state intact but can perturb the receiver's selection inputs (Figure 3).

There are two routes. Current hyperprior loss can change the inferred entropy parameters despite intact residual bytes. Past losses can instead make the receiver's decoded-picture buffer (DPB) diverge from the sender's, changing temporal context even when all relevant current-frame packets arrive. This second route does not require a preceding entropy-decoding failure.

A scale perturbation changes the coding distribution only if it crosses a discrete CDF-selection boundary. A mismatched CDF can corrupt both the decoded symbol and the rANS decoder state, so later symbols in the same retained stream can be wrong even when their own CDF selection is correct. The damaged reconstruction then updates the receiver's DPB, perturbing later temporal context and potentially the entropy conditions used for subsequent frames. This closes a cross-frame propagation loop: the damage need not be confined to the missing content or the frame that lost it. Appendix E.1 gives the full propagation analysis, and Section 4.2 tests retained payloads directly.

2.2 Entropy-condition locality

Formally, let c_i^S and $c_i^R(\ell)$ be the entropy-modeling contexts of unit i at the sender and at a receiver under loss history ℓ, comprising the hyperprior, DPB-derived temporal information, and spatial or cross-group information actually used by the model. Let F predict scales from context and $\mathcal{I}$ map

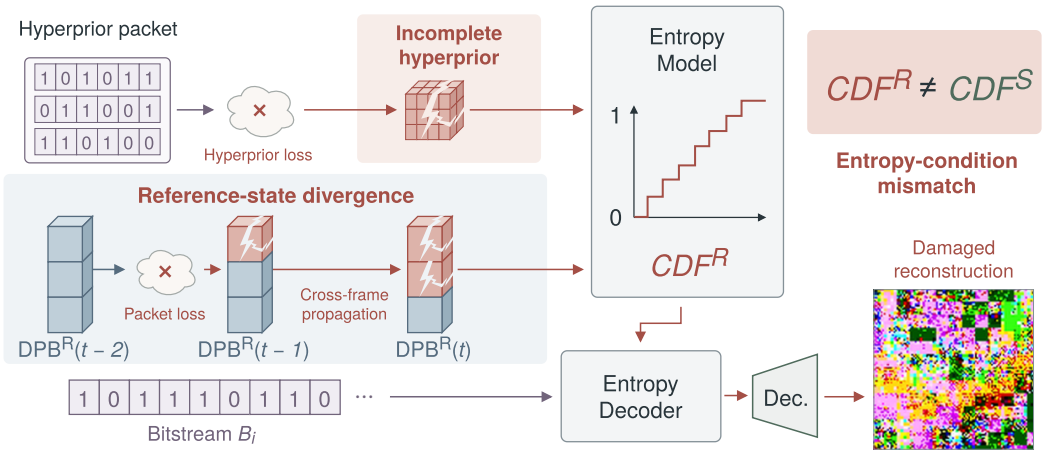

Figure 3: The entropy-decoding failure boundary. Hyperprior loss or reference-state divergence can change the selected CDF and corrupt residual symbols decoded from an intact bitstream.

scales to discrete CDF indexes. A context-conditioned codec selects

$$C_i^S = \mathcal{I}\big(F(c_i^S)\big)\,, \qquad C_i^R(\ell) = \mathcal{I}\big(F(c_i^R(\ell))\big)\,, \tag{1}$$

so an erasure that moves $c_i^R(\ell)$ across a selection boundary changes the CDFs applied to intact bytes.

Negating this failure gives a property a design can be checked against. Fix a retained unit i and its payload bytes B_i. Let C_i^S be the CDF-index sequence the sender actually used to encode B_i, and let $C_i^R(\ell)$ be the sequence a receiver uses to decode those same bytes under a loss history ℓ that erases only content outside unit i. Under a shared coding configuration and valid framing, unit i satisfies *entropy-condition locality* when

$$C_i^R(\ell) = C_i^S \qquad \text{for every admissible } \ell \text{ that retains unit } i, \tag{2}$$

and a codec has the property when this holds for every unit it emits.

Locality guarantees the interpretation of a delivered payload, not identical reconstruction. Predicted means, quantization steps, reference states, and pixels remain receiver-dependent (Appendix D.1, Equation 10); Section 3.4 addresses reconstruction beyond this boundary. Section 3.2 establishes locality by construction, while Section 4.2 reports a finite empirical audit.

Post-entropy latent masking presupposes that retained payloads decode correctly. It models erasure at the entropy-symbol layer when locality holds, but omits corruption of received symbols when it fails. Appendix E.2 develops this distinction.

2.3 Diagnostic evidence and the resulting design principle

Controlled interventions quantify the cost of violating locality. Holding retained payload bytes and the loss mask fixed, restoring sender-consistent CDF indexes (an *index oracle*) removes the observed index, symbol, boundary, and zero-fill mismatches and recovers +5.64 dB at 2% target packet loss (Section 4.2). This recovery from unchanged retained bytes identifies entropy interpretation as a major low-loss bottleneck of the context-conditioned backbone. The separate five-mode diagnosis examines the two loss routes and reference-state restoration (Appendix E.3).

Serialize the decision, not the dependency. Locality must be restored without giving up the sender's context. The sender can retain that context and transmit its final entropy decision rather than requiring receiver-side replay. The index oracle diagnoses the value of doing so but does not provide a deployable mechanism: the needed information must be transmitted at a counted rate cost. Reconstruction must additionally be learned on the receiver's evolving state. Section 3 takes these requirements in that order.

3 Method

3.1 Overview

The sender refines an entropy scale from its own reference state and staged context, compresses it into side information, and binds it to the residual payload. The receiver recovers this transmitted

condition, decodes received residuals, and reconstructs from surviving content and its own state. Dual-trajectory training optimizes reconstruction as the two states evolve separately. The entropy path and packet organization follow the design panels of Figure 2; Figure 4 shows training over time.

3.2 PACKET-LOCAL ENTROPY SIDE INFORMATION

To restore locality, the CDF used for a received residual must agree with the sender's despite losses elsewhere. This is the condition the index oracle of Section 2.3 imposed by injection; here it is obtained by transmission. We omit the frame index. For channel group i, z_i is the quantized hyperprior, s_i the quantized scale side information, and y_i the quantized residual symbols carried by the residual stream (Appendix D.1); shared models, QP-dependent parameters, CDF tables, and valid framing are implicit.

Let $\sigma_i^\star$ be the sender's staged-refined scale (Section 3.3), g_a and g_s the analysis and synthesis transforms of the side information, and Q_s its quantizer, with QP-dependent scaling absorbed into the transforms. The sender computes

$$\begin{aligned} s_i &= Q_s(g_a(\sigma_i^\star))\,, \quad C_i^S = \mathcal{I}(g_s(s_i))\,, \\ B_i^z &= \mathrm{Enc}_z(z_i), \qquad B_i = (B_i^s, B_i^y) = \big(\mathrm{Enc}_s(s_i),\, \mathrm{Enc}_y(y_i; C_i^S)\big)\,. \end{aligned} \tag{3}$$

The step that matters is the sender decoding its own side latent before it encodes: it selects the residual CDFs from the recovered scale $g_s(s_i)$ rather than directly from $\sigma_i^\star$ (Figure 2, left). Quantization error in s_i therefore costs rate efficiency, not correctness: a coarser s_i yields a less well matched distribution but never a mismatched one. The side latent is entropy-coded with a fully factorized model under shared codec configuration, so its decoding does not depend on the hyperprior, other channel groups, or the receiver DPB. B_i^z and B_i form two separate logical erasure units: a hyperprior unit and a bound (s_i, y_i) unit. The bound unit packages both streams with explicit boundaries and is retained or erased as a whole, so whenever the residual bytes arrive, the information needed to recover their CDF arrives with them (Figure 2, center; Appendix D.4).

Proposition 1 (Packet-local CDF agreement)**.** *Assume both endpoints apply the same deterministic scale recovery g_s and CDF selection $\mathcal{I}$, with matching entropy-coding configurations and valid framing. Then for every loss history ℓ that retains B_i intact,*

$$s_i^R = \mathrm{Dec}_s(B_i^s) = s_i, \quad C_i^R(\ell) = \mathcal{I}\big(g_s(s_i^R)\big) = C_i^S, \quad y_i^R(\ell) = \mathrm{Dec}_y\big(B_i^y; C_i^R(\ell)\big) = y_i. \tag{4}$$

The first equality holds because the factorized side-information decoder inverts its encoder using only bytes inside the unit; the second because both endpoints apply the same map to the same s_i; the third because rANS decoding inverts encoding under identical CDFs. SEAL therefore satisfies Equation 2 for every emitted unit, and a retained bound unit yields exact residual symbols even when its hyperprior unit B_i^z is erased. Unlike the context replay of Equation 1, the receiver recovers the CDFs the sender actually used from packet-local information. The guarantee concerns the entropy interpretation of received symbols only; it guarantees neither identical reconstructions, DPBs, or pixels nor the recovery of erased units. Determinism of g_s and $\mathcal{I}$ across endpoints is an implementation requirement, which the audit of Section 4.2 checks empirically. The side information has a rate cost, which is included in both the training objective and transmitted-rate evaluation. Numerical entropy-model and container details are given in Appendix D.

3.3 SENDER-ONLY STAGED REFINEMENT

Packet-local entropy decoding can still benefit from context at the sender. We partition the latent into channel groups and process them in stages. Hyperprior and temporal context provide initial conditions; later stages refine their scales using already reconstructed sender groups (Figure 2, top left). Each group's final scale $\sigma_i^\star$ is represented by the side latent from Section 3.2. The receiver obtains the scale used for coding directly from s_i, so it does not need to reproduce these cross-group dependencies.

Better conditional modeling can reduce residual rate, while the scale representation adds side-information rate. The net effect is measured by the total-rate and quality ablation in Table 1. Group dimensions and the stage layout are implementation settings reported in Appendix D.

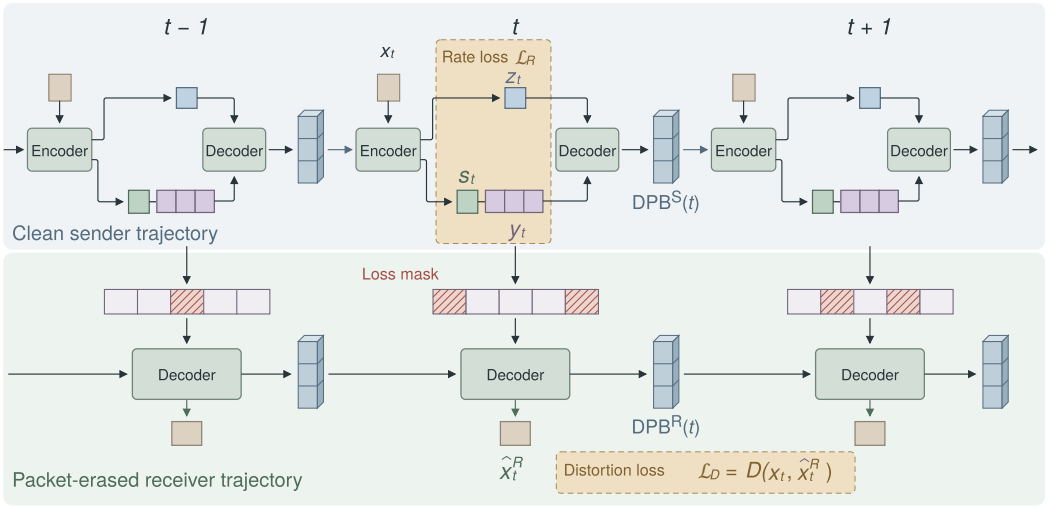

Figure 4: Dual-trajectory loss-aware training. A clean sender (top) and a packet-erased receiver (bottom) maintain separate reference states. Receiver-side distortion is accumulated across frames to train shared weights without feedback.

3.4 Dual-trajectory training

Locality fixes entropy interpretation, but reconstruction still depends on missing content and the receiver's diverging reference state. Dual-trajectory training targets this remaining dependence without feedback or endpoint resynchronization.

Concretely, the receiver fills missing hyperprior groups with zeros before predicting its reconstruction parameters. Received residuals are decoded using s_i and combined with the receiver's predicted mean and quantization parameters. Missing residuals take zero quantized residual values, so their latent estimates are determined by the receiver's mean and quantization parameters. The neural decoder uses these latents and receiver context to reconstruct the frame and update the receiver DPB.

We train this path by unfolding two trajectories (Figure 4). Let H_t^S and H_t^R be the sender and receiver DPBs after frame t, with a common initial state $H_0^S = H_0^R$ established by the reliably delivered I-frame, and let $u_t = (z_t, s_t, y_t)$ collect the quantized transmitted symbols of all channel groups of frame t. Then

$$u_t = \mathcal{E}_\theta(x_t, H_{t-1}^S), \quad (\hat{x}_t^S, H_t^S) = \mathcal{R}_\theta(u_t, H_{t-1}^S), \quad (\hat{x}_t^R, H_t^R) = \mathcal{R}_\theta\big(\mathcal{A}_{m_t}(u_t), H_{t-1}^R\big), \tag{5}$$

where $\mathcal{E}_\theta$ is the sender's encoder and $\mathcal{R}_\theta$ summarizes the missing-content handling, use of context, reconstruction, and DPB update described above; it is not an additional network. The operator $\mathcal{A}_{m_t}$ applies the frame's logical-unit erasures: $z_{t,i}$ can be erased separately, whereas $(s_{t,i}, y_{t,i})$ are retained or erased together. The two paths share the parameters θ. A **clean sender** thus updates its state from its unerased reconstruction, while a **packet-erased receiver** reconstructs from the surviving symbols and its own DPB and carries H_t^R into subsequent frames. Applying erasures at the symbol level is faithful here because Proposition 1 guarantees that retained units decode exactly (Section 2.2). Distortion is computed along the receiver trajectory, so training includes both current packet loss and its recursive effects through the DPB and context, without requiring the endpoint states to coincide. The objective is

$$\mathcal{L} = \frac{1}{T}\sum_{t=1}^{T}\left[R_{z,t} + R_{s,t} + R_{y,t} + \lambda(\text{baseQP})\, D_t(x_t, \hat{x}_t^R)\right], \tag{6}$$

where T is the number of predicted frames. The rate terms are differentiable sender-side estimates in bpp, not actual bitstream lengths, and D_t is the receiver reconstruction distortion, a frame-weighted mixture of YUV and RGB squared error defined with $\lambda(\text{baseQP})$ in Appendix D.2. In particular, R_s makes the cost of the entropy side information part of the optimization; its differentiable form is a surrogate for the coding cost, not the coding distribution itself (Appendix D). The rate terms are not multiplied by receiver erasure masks, and no separately weighted clean-reconstruction loss or DPB-alignment loss is added; robustness is optimized through distortion along the receiver trajectory. The $-$dual control, which trains on a single shared damaged DPB, is formalized in Appendix D.3; packet-mask sampling and training-rate implementation details are also given there.

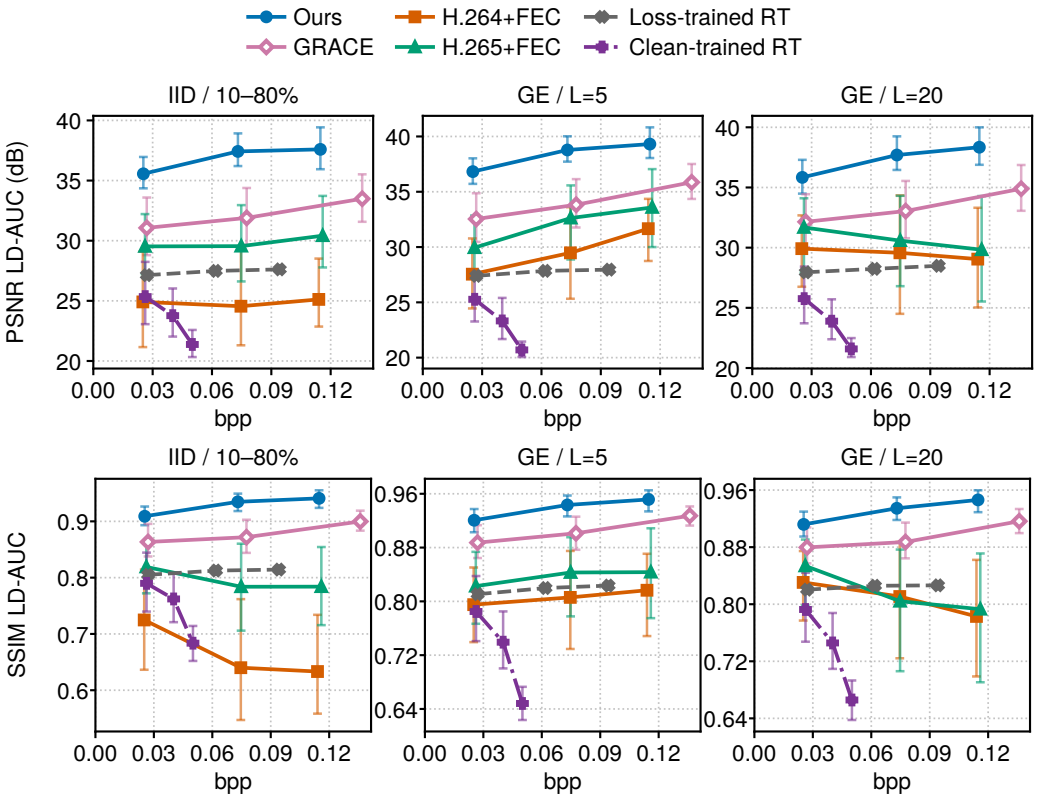

Figure 5: Rate-robustness frontiers: PSNR (top) and SSIM (bottom) loss-distortion AUC versus rate. GRACE uses reported rate; DCVC-RT restart-wrapper overlays (loss- and clean-trained) are diagnostic and excluded from the formal envelope. Whiskers are 95% marginal AUC intervals, not paired-difference intervals (Appendix A.2).

4 Experiments

4.1 Experimental setup

Protocol. We denote the complete SEAL configuration by *Full*. We evaluate Full at QP39/55/63 against GRACE and H.264/H.265+FEC on the seven UVG sequences (32 frames, 1 I + 31 P; channel seeds 0–4), with cross-dataset tests on MCL-JCV and HEVC classes B–E (Appendices D.8.1 and A.6). The I-frame, frame headers, and length tables are reliably delivered and rate-counted; the sender receives no feedback.

Loss models. Each Full P-frame contains 32 hyperprior units z_i and 32 bound (s_i, y_i) units, each erased atomically. "i.i.d." erases a fixed number of units uniformly at random rather than by Bernoulli sampling; Gilbert–Elliott (GE) bursts run across P-frame boundaries in the fixed native-unit order. Methods share videos, seeds, target loss rates p, and mean burst lengths L, matching native-unit statistics but not erased bytes, outage durations, or exact masks (Appendix D.5, including GRACE's assumed-reliable hyperlatent).

Rate and baselines. Full and its controls are charged actual transmitted bytes (I-frame, z, s, y, and framing), including erased bytes. Traditional rates include source, repair, padding, and headers, and GRACE uses its reported rate (estimated P-frame entropy plus actual BPG I/I-patch bytes). We report PSNR, SSIM, and loss-distortion AUC, weighting UVG videos equally (statistics in Appendices A.2 and D.8.3). Traditional operating points are frozen, selected by transmitted rate alone, and approximately rate-matched; in the logical-unit protocol they use ideal block-erasure recovery with erasable, rate-counted repair and decoder-side concealment disabled (Appendix D.6).

Table 1: Matched single-mechanism ablations at QP55. All arms are loss-trained and reported at their own clean-channel actual transmitted rate; PSNR averages the 32-frame GOP. Checks indicate enabled components.

Model	Side info.	Staged	Dual	bpp	Clean	IID 20%	GE 20%, L=20
Full	✓	✓	✓	0.073	40.28	**39.30**	**38.32**
$-s$	×	✓	✓	0.063	40.05	30.47	30.55
−staged	✓	×	✓	0.081	39.87	38.95	37.85
−dual	✓	✓	×	0.074	39.99	38.52	36.68

Table 2: Cross-dataset BD-PSNR (dB) relative to H.265+FEC over overlapping rate intervals without extrapolation. Dataset means use defined cells; Avg. weights the six dataset groups equally. GRACE uses reported rate, and its valid-cell set differs from Full's, so rows are compared with the anchor, not differenced directly (Appendix A.6).

Method	UVG	MCL-JCV	HEVC-B	HEVC-C	HEVC-D	HEVC-E	Avg.
SEAL (Full)	**+5.00**	**+4.37**	**+5.15**	**+4.95**	**+4.79**	**+5.00**	**+4.87**
GRACE	+0.29	−1.38	−0.25	−0.43	+1.32	−1.14	−0.26
H.264+FEC	−1.74	−1.48	−1.90	−0.09	+0.16	−0.07	−0.85
H.265+FEC (anchor)	0.00	0.00	0.00	0.00	0.00	0.00	0.00

4.2 Diagnosis and entropy-decoding verification

Diagnosis. The QP21/QP42 held-payload intervention of Section 2.3 recovers +5.64 dB pooled-QP PSNR at 2% loss (confidence interval above zero; positive on all seven videos), with sender indexes supplied at no assigned communication cost and retained bytes and loss masks unchanged (Appendices E.4 and E.3).

Verification. SEAL obtains this condition by transmission rather than injection. Across QP39/55/63, with 23 independent erasure conditions per QP over hyperprior (Z) and bound (YS) units, its rANS decoder recovers all 453,120 retained CDF indexes and residual symbols with zero mismatch and zero scale difference. This finite audit complements the construction argument of Section 3.2.

4.3 Component ablations

Table 1 removes one mechanism at a time; all arms are loss-trained, use no feedback, and are reported at their own actual transmitted rate. In −dual, both training passes use the receiver DPB instead of separate clean-sender and packet-erased receiver states, isolating explicit state divergence rather than loss-aware training (Appendix A.5).

Side information matters most: removing it ($-s$) lowers rate but costs 8.8 dB at i.i.d. 20% and 7.8 dB at GE 20%, L=20. At QP55 the s payload is 4.87% of Full's transmitted bytes, distinct from the approximately 14% total-rate gap, relative to Full, between the separately trained Full and $-s$. **Staged refinement** pays for its side-information cost: −staged raises rate and lowers clean and lossy quality. **Dual-trajectory training** helps most under bursty loss: −dual uses more rate but loses 0.3 dB clean and 0.8–1.6 dB under loss. Full's gains over each control hold in every i.i.d. and GE cell (positive hierarchical-bootstrap lower bounds; 35 paired video-seed observations per cell).

4.4 Rate-robustness performance and generalization

Against GRACE, Full attains higher PSNR and SSIM in every evaluated UVG cell (all hierarchical-bootstrap lower bounds above zero). GRACE's reported rate is also 6.76%/6.05%/18.42% higher than Full's actual transmitted rate in the three paired regions. Full also leads the H.264/H.265+FEC configurations under independent and bursty loss and forms the measured non-dominated frontier under both PSNR and SSIM (Figure 5; complete curves and qualitative results in Appendix A).

Generalization across datasets. Relative to H.265+FEC, Full's BD-PSNR ranges from +4.37 to +5.15 dB across six dataset groups (average +4.87 dB), over overlapping rate intervals without extrapolation (Table 2; Appendix A.6).

4.5 CONTINUOUS 128-FRAME EVALUATION

Reference states drift further over long sequences. We run Full and the loss-trained −dual on one continuous 128-frame UVG chain at base QP55, with an intact initial I-frame, no further I-frames, state resets, or feedback, and identical erasure masks. Referenced to its own clean run, Full reduces the growth of the loss-induced PSNR deficit relative to −dual from frames 1–31 to 96–127 by 3.31 dB under i.i.d. 20% and 3.22 dB under GE 20%, L=20 (confidence intervals above zero; consistent on all seven videos). Full slows rather than eliminates deterioration: its absolute PSNR still drops by 3.11 and 4.28 dB (Appendix B).

4.6 EVALUATION UNDER A UNIFIED MTU TRANSPORT

Because the logical-unit protocol abstracts away network packetization, we also run the QP55 Full codec over a shared IPv4/UDP transport with a 1500-byte MTU, erasing network packets and charging all overhead. The frozen H.265+FEC comparator uses validation-selected redundancy, clean-rate QP matching, real Reed–Solomon repair, and tolerant decoding with concealment. The I-frame stays reliable and rate-counted, P-frame metadata is erasable, and actual rates are 0.0784 (Full) and 0.0773 bpp. Across eleven conditions, the codecs tie in PSNR at clean and i.i.d. 5/10/20% (paired intervals include zero), while Full leads in every GE condition and at i.i.d. 40% by +5.54 to +15.46 dB; SSIM point estimates favor Full throughout. These results are reported separately from the logical-unit frontier (Appendices C.5 and C.6).

5 RELATED WORK

Conditional image and video compression. Hyperpriors and autoregressive contexts improve entropy modeling (Ballé et al., 2018; Minnen et al., 2018), with channel-wise (Minnen & Singh, 2020) and checkerboard (He et al., 2021) designs supporting efficient context use. The DCVC family develops temporal context and richer spatial-temporal entropy models (Li et al., 2021; Sheng et al., 2022; Li et al., 2022; 2023). We build on DCVC-RT (Jia et al., 2025); SEAL retains sender-side conditional modeling while making the scale needed to interpret a residual available within its packet. Appendix F gives the full background.

Loss-resilient neural coding. GRACE (Cheng et al., 2024) already carries per-packet distribution information; NeuralMDC (Hu et al., 2024) independently codes descriptions and infers missing tokens. For images, ResiComp (Wang et al., 2025) uses masked tokens for entropy modeling and concealment, while satellite-oriented coding (Sha et al., 2025) trains under channel erasures. Carrying entropy information is therefore not itself new. GRACE satisfies locality without the cross-packet conditioning studied here and is not an instance of our diagnosed failure. Our transmitted scale comes from sender-side staged refinement, and training explicitly maintains diverging endpoint states. Differences from GRACE also reflect backbone generation, packetization, and training; their end-to-end rate gap is not attributed to one mechanism.

6 LIMITATIONS AND CONCLUSION

Scope. GRACE is compared at its reported rather than measured packetized rate (Appendix D.5), and the traditional baselines are frozen configurations rather than an exhaustive source-coding, protection, and concealment design space (Appendices D.6 and C). Most evidence uses 32-frame logical-unit evaluation with an intact I-frame; the unified-MTU comparison tests network-packet erasures on UVG. Both extend to 128 frames at base QP55, where Full slows but does not eliminate deterioration. Independent backbones, feedback/resynchronization, wall-clock latency, peak memory, and end-to-end latency or reordering remain outside the measured scope.

Conclusion. SEAL retains context modeling under packet loss by serializing the sender's entropy decision rather than its dependencies. Packet-local side information restores entropy-condition locality, sender-only staged refinement preserves conditional modeling, and dual-trajectory training addresses reconstruction under diverging states. SEAL lies on the measured rate-robustness frontier over the evaluated baselines.

REPRODUCIBILITY STATEMENT

Section 4 summarizes the evaluation and comparison assumptions. Appendices D.5–D.8 specify erasure topologies, baseline selection, packet ordering, evaluation grids, rate definitions, and statistical procedures; Appendices A.2 and A.6 define the rate-robustness and cross-dataset metrics. Appendix D also records grouping, entropy implementation, training-mask sampling, and container-byte accounting. The matched controls are defined in Section 4.3 and Appendix A.5. Appendix E records the distinct diagnoses and entropy-decoding audit; Appendices B and C specify the long-sequence and unified-MTU evaluations. Model configurations, evaluation scripts, and per-cell statistics will be provided as anonymized supplementary material.

AI USE STATEMENT

Generative AI tools were used to assist with drafting and copy-editing the manuscript text and with LaTeX formatting, working from an author-specified, evidence-bound outline. All technical content, including the method, experiments, numerical results, and their boundaries, was produced and verified by the authors from their own experimental assets; AI-assisted text was reviewed by the authors, who take full responsibility for the final content of this work. We did not use generative AI to generate experimental data or results.

REFERENCES

Eirikur Agustsson, David Minnen, Nick Johnston, Johannes Ballé, Sung Jin Hwang, and George Toderici. Scale-space flow for end-to-end optimized video compression. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. Variational image compression with a scale hyperprior. In *International Conference on Learning Representations (ICLR)*, 2018. arXiv:1802.01436.

Yihua Cheng, Ziyi Zhang, Hanchen Li, Anton Arapin, Yue Zhang, Qizheng Zhang, Yuhan Liu, Kuntai Du, Xu Zhang, Francis Y. Yan, Amrita Mazumdar, Nick Feamster, and Junchen Jiang. GRACE: Loss-resilient real-time video through neural codecs. In *USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, 2024. arXiv:2305.12333.

Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto. Learned image compression with discretized gaussian mixture likelihoods and attention modules. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. arXiv:2001.01568.

Vivek K. Goyal. Multiple description coding: Compression meets the network. *IEEE Signal Processing Magazine*, 18(5):74–93, 2001.

Dailan He, Yaoyan Zheng, Baocheng Sun, Yan Wang, and Hongwei Qin. Checkerboard context model for efficient learned image compression. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. arXiv:2103.15306.

Xinyue Hu, Wei Ye, Jiaxiang Tang, Eman Ramadan, and Zhi-Li Zhang. Robust multiple description neural video codec with masked transformer for dynamic and noisy networks. *arXiv preprint arXiv:2412.07922*, 2024.

Zhaoyang Jia, Bin Li, Jiahao Li, Wenxuan Xie, Linfeng Qi, Houqiang Li, and Yan Lu. Towards practical real-time neural video compression. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. arXiv:2502.20762.

Mohammad Kazemi, Mohammad Ghanbari, and Shervin Shirmohammadi. A review of temporal video error concealment techniques and their suitability for HEVC and VVC. *Multimedia Tools and Applications*, 80:12685–12730, 2021.

Jiahao Li, Bin Li, and Yan Lu. Deep contextual video compression. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. arXiv:2109.15047.

Jiahao Li, Bin Li, and Yan Lu. Hybrid spatial-temporal entropy modelling for neural video compression. In *ACM International Conference on Multimedia (ACM MM)*, 2022. arXiv:2207.05894.

Jiahao Li, Bin Li, and Yan Lu. Neural video compression with diverse contexts. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. arXiv:2302.14402.

Jiahao Li, Bin Li, and Yan Lu. Neural video compression with feature modulation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. arXiv:2402.17414.

Guo Lu, Wanli Ouyang, Dong Xu, Xiaoyun Zhang, Chunlei Cai, and Zhiyong Gao. DVC: An end-to-end deep video compression framework. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. arXiv:1812.00101.

David Minnen and Saurabh Singh. Channel-wise autoregressive entropy models for learned image compression. In *IEEE International Conference on Image Processing (ICIP)*, 2020. arXiv:2007.08739.

David Minnen, Johannes Ballé, and George Toderici. Joint autoregressive and hierarchical priors for learned image compression. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018. arXiv:1809.02736.

Hongwei Sha, Muchen Dong, Quanyou Luo, Ming Lu, Hao Chen, and Zhan Ma. Towards loss-resilient image coding for unstable satellite networks. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2025. arXiv:2501.11263.

Xihua Sheng, Jiahao Li, Bin Li, Li Li, Dong Liu, and Yan Lu. Temporal context mining for learned video compression. *IEEE Transactions on Multimedia*, 2022. arXiv:2111.13850.

Amin Shokrollahi. Raptor codes. *IEEE Transactions on Information Theory*, 52(6):2551–2567, 2006.

Sixian Wang, Jincheng Dai, Xiaoqi Qin, Ke Yang, Kai Niu, and Ping Zhang. ResiComp: Loss-resilient image compression via dual-functional masked visual token modeling. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 2025. arXiv:2502.10812.

Yao Wang, Amy R. Reibman, and Shunan Lin. Multiple description coding for video delivery. *Proceedings of the IEEE*, 93(1):57–70, 2005.

Stephan Wenger. H.264/AVC over IP. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(7):645–656, 2003.

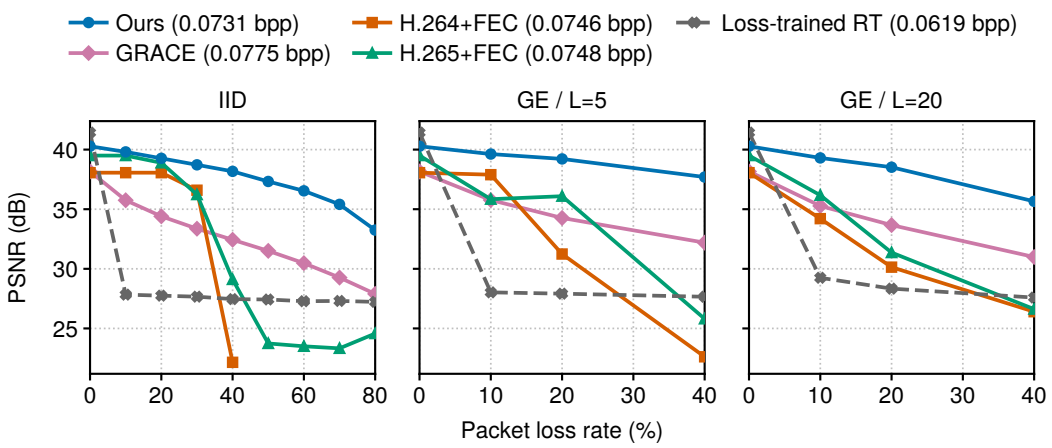

Figure 6: Representative loss-distortion curves at the middle rate region (QP55): weighted PSNR versus target packet-loss rate for i.i.d. loss and Gilbert–Elliott loss (mean burst $L{=}5, 20$; genuine GE runs, no $L{=}1$ proxy panel). Full (ours) leads across the full loss range against GRACE and H.264/H.265+FEC. The *loss-trained RT* overlay (dashed) is an independently restarted non-native operational wrapper shown as a diagnostic; it collapses to a low plateau beyond $\sim$10% loss.

A Supplementary loss-resilience results

This appendix reports complete loss-distortion curves, rate-robustness metric decompositions, complexity accounting, qualitative comparisons, and detailed ablation and cross-dataset reporting. Section 4 summarizes the evaluation; Appendix D specifies its implementation and protocol details. The mechanism diagnoses and their distinct settings are collected in Appendix E.

A.1 Complete loss-distortion curves

Figure 6 shows the compact representative middle-rate (QP55) curves, and Figures 7–8 give the full per-rate-region i.i.d. and Gilbert–Elliott loss-distortion curves for both PSNR and SSIM, at QP39/55/63, for Full, GRACE, and H.264/H.265+FEC. Each grid also overlays the loss-trained RT restart-wrapper diagnostic (dashed), which is excluded from the formal comparison.

The PSNR and SSIM frontiers in Figure 5 retain loss-trained and clean-trained RT diagnostic overlays, which use a non-native, independently restarted per-packet wrapper and are excluded from the formal frontier. Both plateau far below Full beyond approximately 10% loss. These overlays provide a training-recipe comparison; the matched component ablations isolate the effects of our design.

A.2 Rate-robustness metrics

Let $Q(p)$ be a quality measure (PSNR in dB, or SSIM) at target loss rate p, at a fixed rate region. We summarize a loss sweep by two quantities,

$$\text{LD-AUC} = \frac{\int_{p_{\min}}^{p_{\max}} Q(p)\,dp}{p_{\max} - p_{\min}}, \qquad \text{Deg-AUC} = \frac{\int_{p_{\min}}^{p_{\max}} \big[Q(0) - Q(p)\big]\,dp}{p_{\max} - p_{\min}}, \tag{7}$$

both evaluated by trapezoidal integration over the target loss grid. For i.i.d. loss the grid is 10% to 80% in 10% steps; for GE loss it is $10/20/40\%$, computed separately for $L{=}5$ and $L{=}20$. The clean point $Q(0)$ supplies the reference in Deg-AUC but is not part of either integration range, so it does not enter the absolute AUC. We call Deg-AUC "clean-normalized" because it is referenced to the clean point by subtraction; nothing is divided by clean quality.

Figure 1: summary coordinates. The horizontal coordinate is clean-channel BD-rate (%) relative to H.265+FEC, with the axis decreasing rightward; negative values indicate rate savings. BD-rate

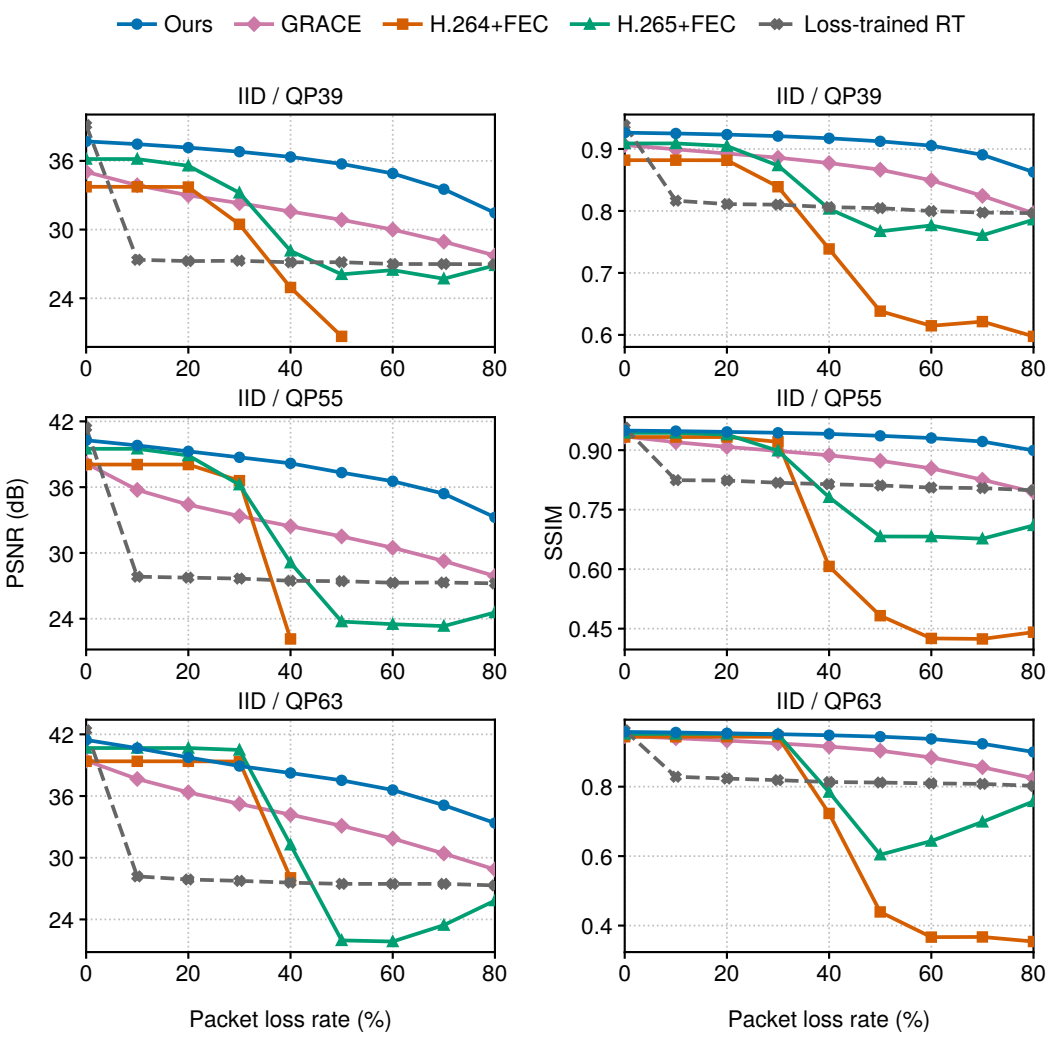

Figure 7: Complete i.i.d. loss-distortion curves by rate region: PSNR (left) and SSIM (right), sweeping i.i.d. loss 0–80%. Dashed curve: loss-trained RT restart-wrapper diagnostic (excluded from the formal comparison).

is computed by PCHIP integration of log rate against PSNR over the shared quality interval 37.70–39.38 dB, using aggregate clean RD points with H.265+FEC as the anchor. The vertical coordinate is the equally weighted mean of the middle-rate normalized degradation AUCs for i.i.d. loss, GE $L{=}5$, and GE $L{=}20$; lower degradation is displayed upward. GRACE uses reported rate. Under this evaluation, GRACE is at +68.45% BD-rate relative to H.265+FEC and Full at -33.38%. These end-to-end figures reflect backbone generation, packetization, and training together, rather than isolating a single mechanism.

Two resampling procedures appear in this paper, and they are not interchangeable. Formal paired comparisons (the ablations, the per-cell Full-versus-control gains, and the long-sequence results) use 10,000 nested hierarchical paired bootstrap replicates over seven videos $\times$ five seeds, resampling videos and then seeds within each selected video. The rate-robustness frontiers (Figure 5) come from an earlier evaluation pass and use 5,000 crossed bootstrap replicates over seven videos $\times$ three nominal seeds, with video and seed weights drawn separately and one replicate's weights applied across all loss points. Their whiskers are marginal intervals on each method's own AUC rather than intervals on a paired difference, so the frontier is read as an ordering of methods and the paired tests are reported separately.

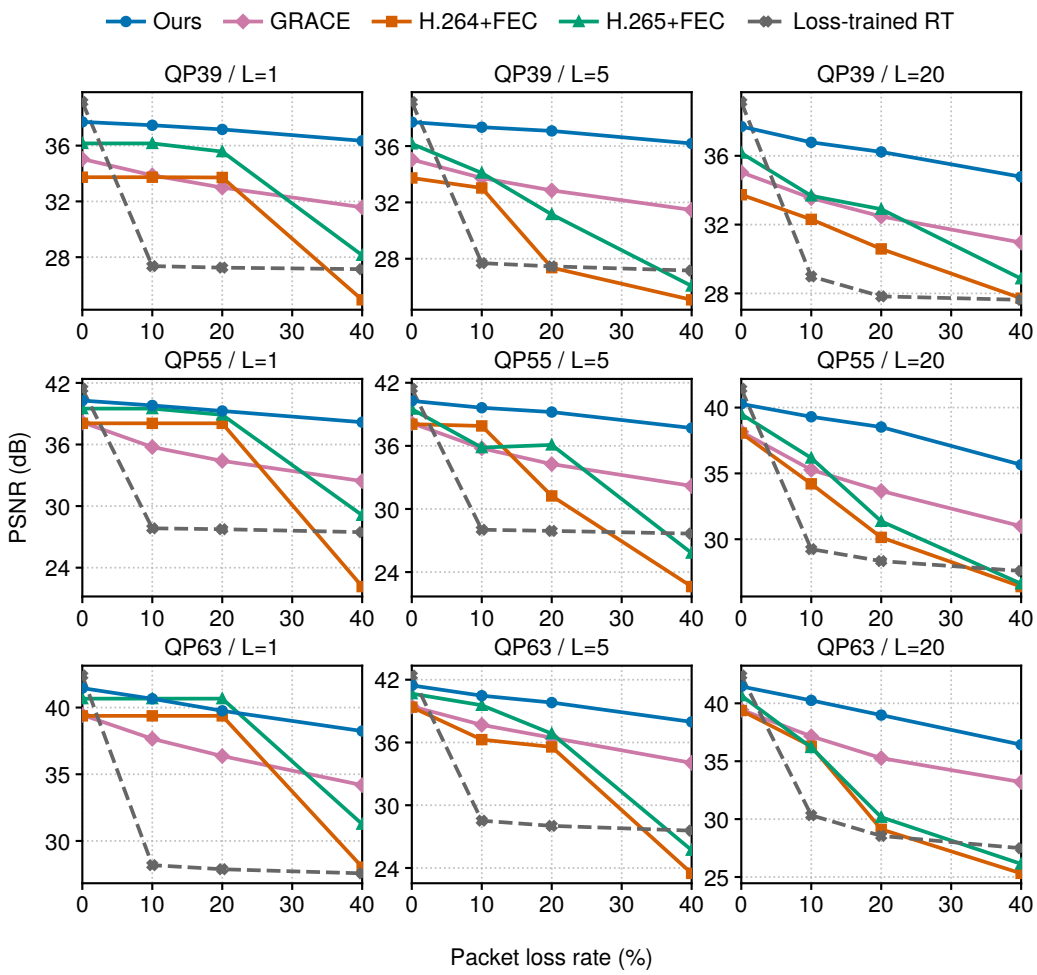

Figure 8: Complete Gilbert–Elliott loss-distortion PSNR curves at mean burst lengths $1/5/20$, with the loss-trained RT restart-wrapper diagnostic overlay (dashed, excluded from the formal comparison). The $L{=}1$ panel reuses the i.i.d. observations as a memoryless proxy; it is not an independently measured GE run. The SSIM counterpart is Figure 9.

Figure 10 decomposes loss-distortion AUC into absolute and clean-normalized degradation terms by rate region. The PSNR and SSIM rate-robustness frontiers are in Figure 5.

A.3 Model complexity

Table 3 reports parameter count and per-pixel compute. Full adds packet-loss resilience with fewer parameters and lower combined encode/decode compute than the real-time backbone, while its encoder compute is slightly higher. GRACE uses fewer parameters but more encode/decode compute at the listed configurations. Compute is normalized to kMAC/pixel (GOP32-average, per frame at 1920×1080; multiply-accumulates taken as half the counted floating-point operations) so it is directly comparable across resolutions and codecs. Wall-clock latency and peak memory are not measured in this work.

A.4 Additional qualitative results

Figure 11 shows a two-row severe-loss (70%) comparison, and Figure 12 shows additional reconstructions across four UVG sequences.

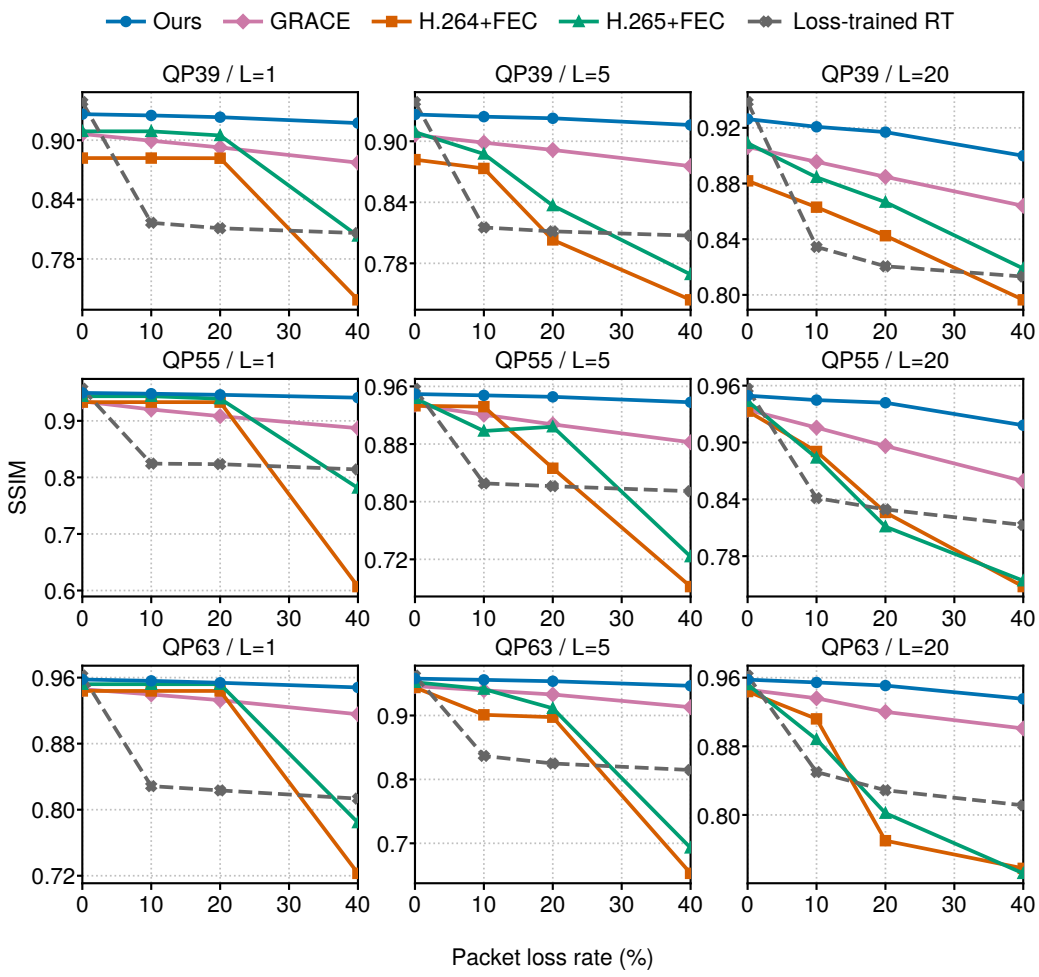

Figure 9: Complete Gilbert–Elliott loss-distortion SSIM curves at mean burst lengths $1/5/20$, with the loss-trained RT restart-wrapper diagnostic overlay (dashed, excluded from the formal comparison). As in Figure 8, the $L{=}1$ panel reuses the i.i.d. observations as a memoryless proxy.

Table 3: Model complexity, normalized per pixel. Params in millions; encode/decode compute in kMAC/pixel (GOP32-average, per frame at 1920×1080). GRACE is listed across its official configurations.

Method [config]	**Params (M)**	**Enc. kMAC/px**	**Dec. kMAC/px**
Original RT backbone	20.69	142.6	167.0
Full (ours)	19.14	146.7	157.4
GRACE [64/1024/2048]	4.75	552–704	217–311

A.5 Component-ablation details and side-information accounting

This section expands the control definition and rate accounting behind Table 1; all controls remain trained under packet loss and are reported at their own actual transmitted rate.

Control definition. The control is more specific than "no dual-trajectory training" suggests. Full unfolds two reference trajectories during training: a clean sender DPB that never sees erasures, and a packet-erased receiver DPB that does. In −dual both the encoding and decoding passes operate

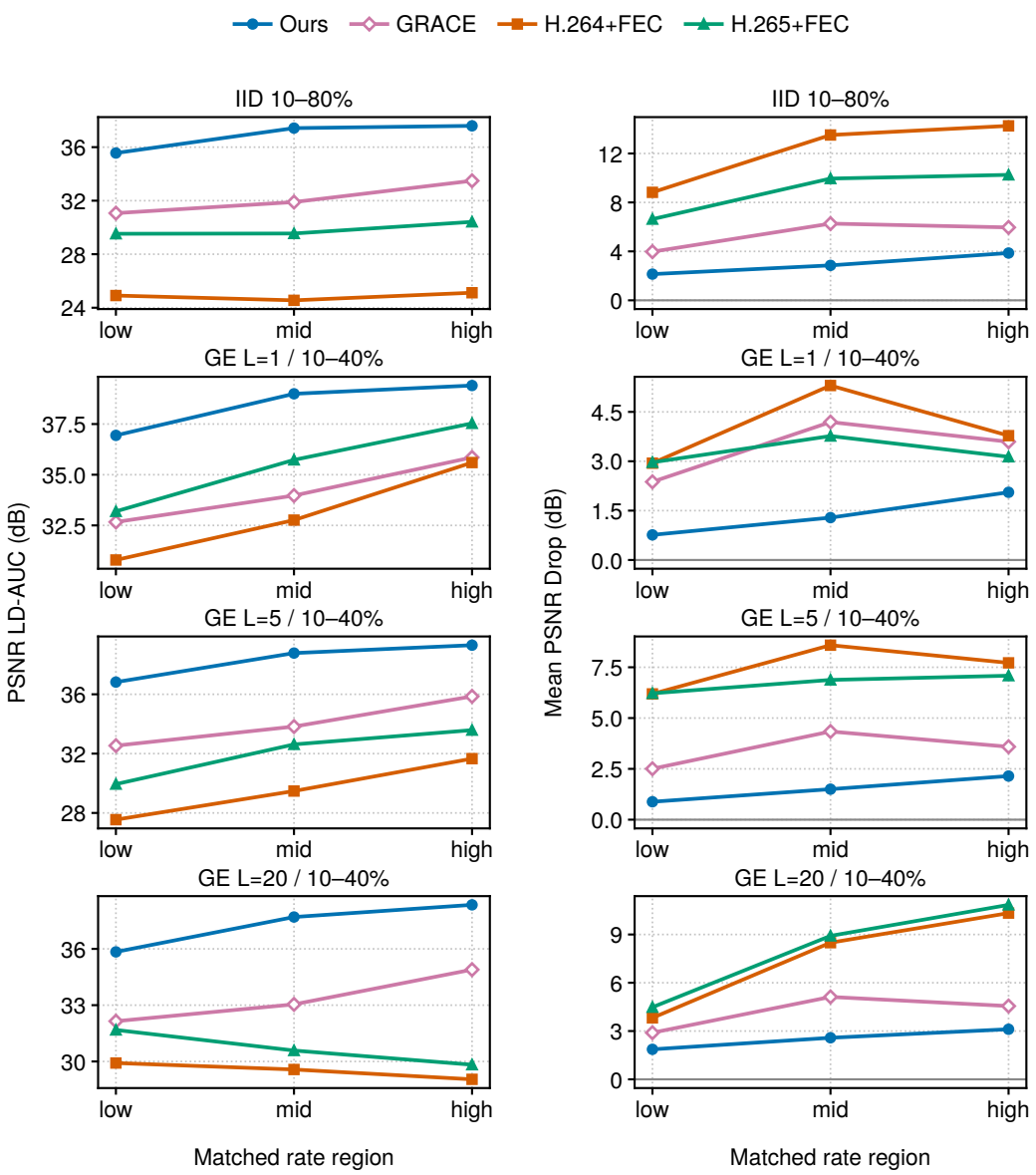

Figure 10: Loss-distortion AUC by rate region (PSNR): absolute AUC (left) and clean-normalized degradation AUC (right), both defined in Appendix A.2. "Clean-normalized" means the clean point supplies the reference $Q(0)$ that each loss point is measured against; it is not a division by clean quality, and the clean point itself is excluded from the absolute AUC.

on the *receiver-side* DPB, sharing one damaged reference state. The control therefore still trains under packet loss and still learns from damaged references; what it lacks is the explicit *separation* between a clean sender state and a diverging receiver state. Full against −dual thus isolates the value of modeling that divergence, with loss-aware training held fixed on both sides. The shared DPB is a training construct: neither variant uses feedback or shares state between endpoints at test time.

Direct side payload versus total-rate gap. Two different quantities are easily confused here. Measured directly on Full's own bitstream at QP55, the entropy-coded s payload is 4.87% of transmitted bytes, averaging the per-sequence fractions equally over the seven UVG sequences. That denominator is the whole 32-frame sequence including the reliable I-frame, packet tables, and outer headers, and the numerator counts only the s entropy payload; shared framing is not attributed to



Figure 11: Qualitative payoff under severe (70%) packet loss. For representative UVG frames, Full (ours) preserves structure and texture that GRACE and the traditional FEC baselines lose to visible artifacts or blanking.

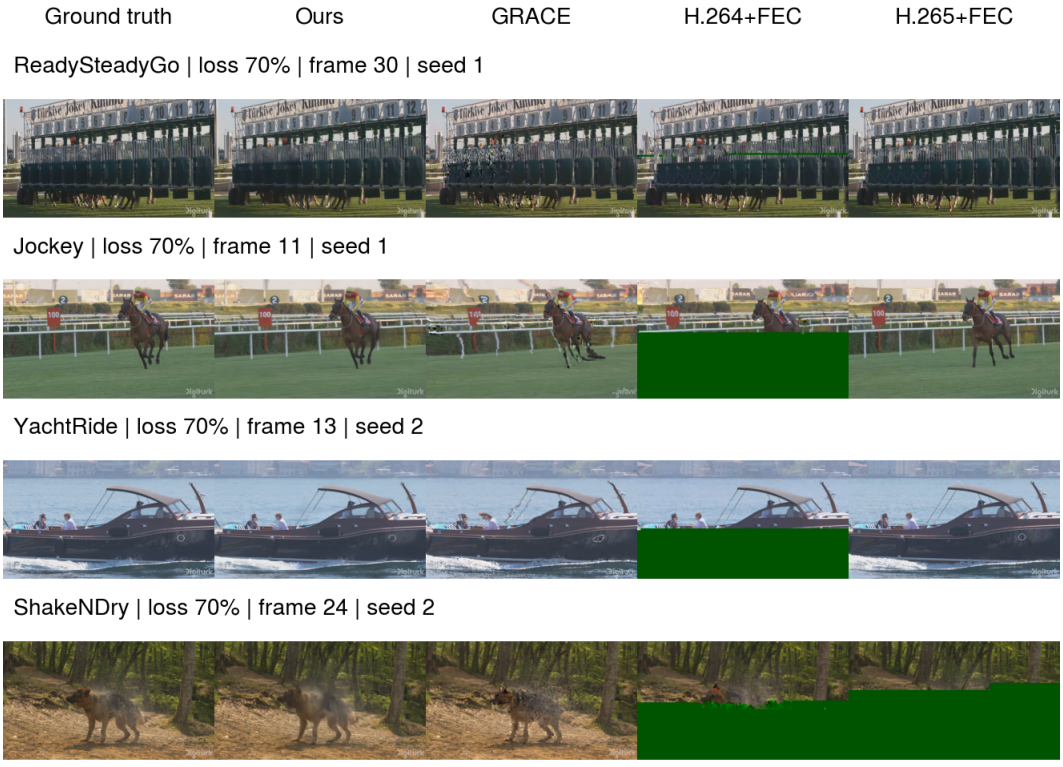

Figure 12: Additional qualitative results under 70% packet loss across four UVG sequences, comparing Full (ours) with GRACE and the traditional FEC baselines.

s. Separately, the total-rate gap between Full and the $-s$ arm is about 14% of Full's total. The two do not measure the same thing: the second is a difference between two separately trained configurations, in which dropping s also changes what the residual coder spends.

Interpretation of the ablation. The clean-channel column therefore reflects the side information's rate cost rather than a quality deficit, and no BD-rate is quoted across the loss cells, where the two curves share no common quality range. What the ablation shows is the trade the mechanism makes:

a few percent of the bitstream buys back roughly 8 dB wherever packets are lost, which the symbol-decoding audit of Section 4.2 confirms is the entropy interpretation being preserved.

A.6 Cross-dataset protocol and aggregation

Protocol. Cross-dataset evaluation covers UVG, MCL-JCV, and HEVC classes B–E over the first 32 frames, at QP39/55/63 with five channel seeds. The ten non-clean conditions are i.i.d. 5/10/20/40% and GE 10/20/40% at $L{=}5$ and $L{=}20$. The six dataset groups are weighted equally in the final average.

Metric construction and valid cells. BD-PSNR is computed separately for each video, seed, and channel condition by PCHIP interpolation of PSNR against $\log_{10}$ rate, over the rate interval that the method shares with H.265+FEC and without extrapolation. A cell with no overlapping rate interval is undefined rather than zero. Dataset means average the defined cells, and the final average weights the six dataset groups equally. Aggregation uses unrounded values, so the final average need not equal the mean of the printed entries. Rates are actual transmitted rates, except that GRACE retains the reported-rate definition of Section 4.1 and is evaluated at its own operating rates rather than being rate-matched to Full.

Results against the anchor. **Full exceeds the anchor on every dataset**, with dataset means from $+4.37$ to $+5.15$ dB and a $+4.87$ dB average. H.264+FEC averages below the anchor and is positive only on HEVC-D; GRACE also averages below it, and is positive on UVG and HEVC-D. Full and GRACE each contribute 2550 of 2650 possible cells over 51 of 53 videos; because their valid sets differ, the two rows are read against the anchor rather than differenced against each other.

Against H.265+FEC the two curves frequently do not overlap in the higher-loss regime: Full attains a quality that H.265+FEC never reaches at any rate, leaving BD-rate undefined there. We therefore report equal-rate BD-PSNR as the primary indicator. GRACE is scored at its own *reported* operating rates (it is not rate-matched to Full); its cells use the same Bjøntegaard construction over the overlapping rate range.

SSIM at the selected operating points. Mean SSIM differences at the selected operating points are positive in every dataset group against both frozen FEC baselines. These are averages over the ten non-clean conditions and the three operating regions at each method's own rate, rather than equal-rate SSIM gains or improvements in every individual cell. Against H.265+FEC the dataset means are $+0.084$ on UVG, $+0.083$ on MCL-JCV, $+0.123$ on HEVC-B, $+0.119$ on HEVC-C, $+0.162$ on HEVC-D, and $+0.048$ on HEVC-E; the corresponding means against H.264+FEC are positive throughout.

B Continuous 128-frame evaluation

This appendix details the evaluation summarized in Section 4.5, which extends Full and the $-$dual control from the 32-frame span used everywhere else to a 128-frame predictive chain.

B.1 Protocol

The evaluation covers the seven canonical UVG sequences at 1920×1080 over their first 128 consecutive frames. Frame 0 is an intact I-frame and frames 1–127 form one continuous predictive chain. No additional I-frames are inserted, and no receiver-state reset, clean-reference injection, feature refresh, or feedback occurs at any point, so prediction continues through frames 32, 64, and 96 using only the state accumulated to that point. Existing checkpoints are evaluated with no retraining for long sequences, so the chain measures how the codec behaves well past the 32-frame span used everywhere else in this work. The operating point is base QP55, retaining the hierarchical QP schedule used elsewhere.

Two channels are evaluated. The first is the "IID20%" condition, which follows the same fixed-count uniform rule as the main sweeps (Section 4.1): a fixed number of logical units is erased per trajectory, chosen uniformly without replacement, so positions are exchangeable rather than independent Bernoulli trials. The realized fraction is fixed per trajectory rather than 20% in expectation,

Table 4: Long-sequence PSNR (dB) in the first and last 32-frame windows, Full against the −dual control, with 95% hierarchical paired bootstrap intervals. Frames 0–31 *include* the protected I-frame; frames 96–127 are purely predictive. "IID20%" is the fixed-count uniform erasure condition of Appendix B.1. The paired advantage is computed per video, seed, and channel from unrounded values, so it is not the difference of the rounded columns.

Channel	Frames	Full	−dual	Paired advantage
IID20%	0–31	39.30 [38.31, 40.38]	38.53 [37.53, 39.76]	+0.77 [0.44, 1.11]
IID20%	96–127	36.10 [35.04, 37.02]	30.20 [27.49, 32.57]	+5.90 [4.02, 8.13]
GE20%, L=20	0–31	38.32 [36.92, 39.83]	36.68 [34.97, 38.59]	+1.63 [0.89, 2.43]
GE20%, L=20	96–127	33.91 [32.84, 35.04]	27.20 [24.43, 29.66]	+6.70 [5.09, 8.59]

Table 5: Mean paired PSNR advantage of Full over the −dual control (dB) across four consecutive predictive windows, with the protected I-frame excluded. Each entry is positive for all seven videos individually.

Frames	IID20%	GE20%, L=20
1–31	+0.79	+1.69
32–63	+2.37	+3.62
64–95	+4.29	+5.57
96–127	+5.90	+6.70

and it equals 20% only up to rounding: the 128-frame Full trajectory erases 1,626 of 8,128 units, or approximately 20.00%. The second channel is Gilbert–Elliott loss at 20% with mean burst length L=20 logical packets, whose chain state continues across frame boundaries as described in Appendix D.7.

For each video, seed, and channel, Full and −dual are driven by identical realized erasure masks, so the two arms differ only in trained weights. The design uses five seeds per video and channel together with one complete clean trajectory per model and video. All 154 trajectories are valid (140 lossy and 14 clean), totalling 19,712 decoded frames.

Aggregation follows Section 4.1: the five seeds are averaged within each video, and the seven videos are then weighted equally. Confidence intervals are 95% intervals from 10,000 paired hierarchical bootstrap replicates that resample videos and then seeds, resampling entire temporal trajectories rather than individual frames so that within-trajectory correlation is preserved. Paired quantities are formed per video, seed, and channel before aggregation, and deficits pair each lossy trajectory with the clean trajectory of the same model and video. Paired differences are computed from unrounded per-trajectory values and therefore need not equal the difference of the rounded entries printed in the tables.

B.2 First and last window

Table 4 contrasts the first 32-frame window, which corresponds to the span used for the main results, with the final one. Frames 0–31 include the protected I-frame, which raises both models' averages in that row; the window comparisons in Section 4.5 and in Table 5 exclude it.

Excluding the I-frame and splitting the chain into four consecutive predictive windows gives the progression in Table 5. The advantage increases monotonically across windows in both channels, and each video-level window average is positive for all seven videos in every channel-window combination.

B.3 Clean-relative analysis

Absolute lossy PSNR conflates two things over this horizon, because the models' *clean* trajectories also deteriorate and do so at different rates. Between frames 1–31 and 96–127, Full's clean PSNR

Table 6: Clean-relative analysis over 128 frames, with 95% hierarchical paired bootstrap intervals. The deficit is Equation 8, computed for each model against its own clean trajectory, so lower is better. *Top*: deficit in the late window. *Bottom*: growth in deficit from the early predictive window (frames 1–31) to the late one (frames 96–127). Paired columns are computed from unrounded per-trajectory values.

Channel	Full	−dual	Paired reduction
Late-window deficit, frames 96–127 (dB)			
IID20%	2.66 [1.85, 3.58]	6.47 [3.92, 9.40]	3.81 [1.77, 6.32]
GE20%, L=20	4.85 [3.79, 5.92]	9.46 [6.60, 12.60]	4.61 [2.63, 6.86]
Growth in deficit, frames 1–31 → 96–127 (dB)			
IID20%	1.65 [1.22, 2.18]	4.96 [2.77, 7.61]	3.31 [1.36, 5.77]
GE20%, L=20	2.82 [1.96, 3.73]	6.05 [3.81, 8.38]	3.22 [1.62, 5.16]

falls from 40.21 to 38.75 dB and −dual's from 39.92 to 36.66 dB, leaving a late-window clean-quality gap of 2.09 dB that has nothing to do with packet loss. The late-window lossy advantage in Table 4 should therefore not be attributed in full to loss robustness.

We separate the two by referencing each model to its own clean run. For a model, the loss-induced deficit at frame t is

$$\Delta(t) = \mathrm{PSNR}^{\mathrm{clean}}(t) - \mathrm{PSNR}^{\mathrm{lossy}}(t), \tag{8}$$

so a model is charged only for the quality its own channel losses cost it. Table 6 reports the late-window deficit and, more informatively, how much the deficit grows between the early and late predictive windows. The growth quantity is the strongest evidence available here that Full mitigates later deterioration relative to the evaluated control: it is clean-relative, paired, positive in both channels with intervals above zero, and consistent in direction across all seven videos.

Full slows deterioration relative to −dual without reaching a plateau: its own deficit still grows by 1.65 dB under the i.i.d. condition and 2.82 dB under GE, and its absolute PSNR falls by 3.11 and 4.28 dB over the same span. The per-frame curves of Appendix B.4 resolve the shape of that growth between the two windows.

B.4 Per-frame trajectories

The window averages above compress each 128-frame trajectory into four numbers, which is the wrong resolution for judging whether deterioration is gradual, stepwise, or concentrated in a few frames. Figures 13 and 14 therefore show the trajectories at per-frame resolution: absolute PSNR against frame index, and each model's deficit relative to its own clean trajectory. Both retain every P-frame from 1 to 127 without smoothing; frame 0 is the reliable I-frame. They are drawn from saved per-frame statistics, not reconstructed from the window averages above. The shaded regions are *pointwise* 95% hierarchical-bootstrap intervals, a distinction that matters over 128 frames because pointwise intervals do not control the probability that a true curve leaves the shaded region somewhere along its length. The −dual curves are the matched control of Section 4.3, evaluated on these trajectories at its own actual transmitted rate.

B.5 Reference-state discrepancy diagnostic

As a supplementary diagnostic we measure how far the receiver's reference state drifts from the sender's over the chain. For each frame we take the actual reference feature retained after decoding and compare it with the sender's corresponding feature from its own clean reconstruction trajectory, normalizing by the sender's feature energy:

$$\delta(t) = \frac{\mathrm{mean}\big[(f_t^R - f_t^S)^2\big]}{\mathrm{mean}\big[(f_t^S)^2\big] + 10^{-12}}. \tag{9}$$

Table 7 reports early- and late-window means together with the interval for the late-minus-early change. Both models drift, and both drift further under bursty loss than under uniform erasure, but

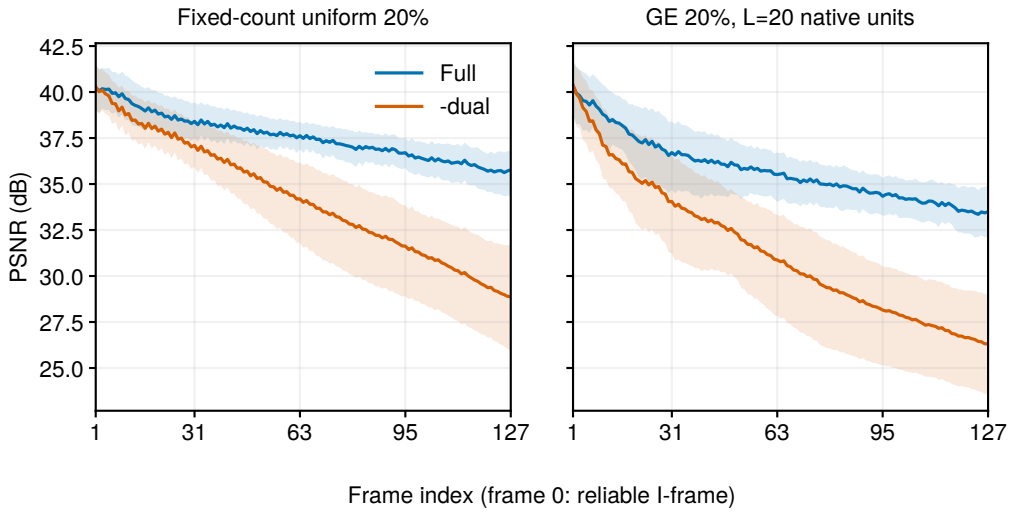

Figure 13: Per-frame PSNR over the continuous 128-frame chain for Full and the −dual control under the fixed-count uniform 20% condition and GE 20%, L=20. Unsmoothed, every P-frame retained; frame 0 is the reliable I-frame. Shaded regions are pointwise 95% hierarchical-bootstrap intervals.

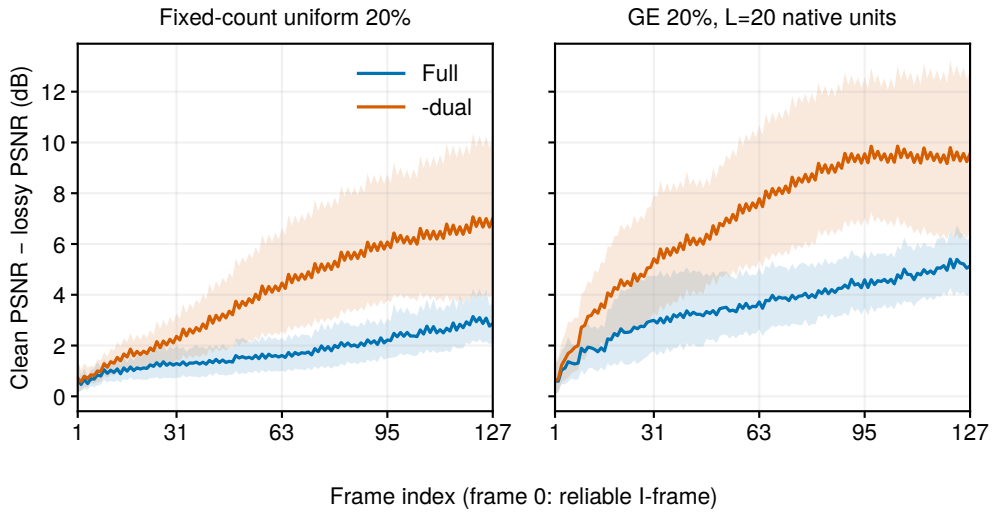

Figure 14: Per-frame loss-induced deficit (Equation 8) for each model against its own clean trajectory, same protocol and channels as Figure 13. Unsmoothed; shaded regions are pointwise 95% hierarchical-bootstrap intervals.

the increase for −dual is several times larger than Full's in each channel. Measured on the same trajectories as the quality results, the smaller state discrepancy tracks the slower deficit growth of Table 6, which is the mechanism dual-trajectory training is meant to act on.

C UNIFIED MTU TRANSPORT COMPARISON

This appendix specifies the transport comparison of Section 4.6, in which both codecs are packetized onto a common IPv4/UDP transport and network packets, rather than logical units, are erased. It is a

Table 7: Normalized reference-state discrepancy (Equation 9), window means and the 95% hierarchical paired bootstrap interval for the late-minus-early change. Lower is closer to the sender's state. Dimensionless.

Channel	Model	Early, 1–31	Late, 96–127	95% CI, late − early
IID20%	Full	0.0050	0.0164	[0.0061, 0.0187]
IID20%	−dual	0.0074	0.0869	[0.0361, 0.1546]
GE20%, L=20	Full	0.0140	0.0341	[0.0102, 0.0321]
GE20%, L=20	−dual	0.0285	0.1725	[0.0527, 0.2653]

self-contained comparator: its configurations are frozen on data independent of the test metric, and its results are reported separately from the logical-unit results of Section 4.1.

This comparison matches erasure statistics over network packets rather than outage durations in milliseconds; the mapping changes transport behavior and loss correlation while leaving Full's logical units and model architecture unchanged.

C.1 Coding settings

All seven UVG sequences are evaluated at their native 1920×1080, over the first 32 frames for the main experiment and the first 128 for the supplement, with seeds 0–4. Full uses its existing weights at base QP55 under the original hierarchical QP schedule, without retraining. Each sequence is one initial I-frame followed exclusively by P-frames, with no intermediate I-frame insertion and no reference-state reset.

C.2 Unified packetization

Both methods use IPv4/UDP at a 1500-byte MTU. Every datagram carries 28 bytes of IPv4/UDP header, a 24-byte application header, and a 12-byte descriptor per fragment. Small units are coalesced and large units fragmented in a fixed order, with no coalescing across frame boundaries.

Full retains its logical structure underneath this mapping: each P-frame emits 32 z units and 32 bound (s, y) units in the order $z_0, \ldots, z_{31}$ then $(s_0, y_0), \ldots, (s_{31}, y_{31})$, and a logical unit counts as received only if all of its fragments arrive. What changes is the accounting for framing metadata. The 388-byte length table of Appendix D.4 no longer exists as a separately delivered structure. Its content is carried by the transport descriptors and by length fields inside the (s, y) units, so it is charged in the transmitted rate and exposed to erasure like any other content, with no reliable-delivery assumption.

C.3 H.265+FEC configuration

H.265 uses libx265 at the `veryfast` preset with eight slices per frame and no B-frames. Actual NAL bytes pass through the same packetizer and are protected by systematic Reed–Solomon FEC within each frame: a block of D source packets generates $R = \lceil 0.75D \rceil$ repair packets, source packets are transmitted before repair packets, and all source padding and repair traffic are charged to the rate. The receiver runs with tolerant decoding and error concealment enabled; a missing display frame repeats the previous displayed frame, with timestamps preserving alignment to the source frames.

Both free parameters are fixed without reference to the quantity being compared. The FEC redundancy is selected on an independent three-sequence MCL-JCV validation set. The per-sequence H.265 QP is then matched using only the clean transmitted byte counts of UVG, so no UVG loss-quality measurement enters the selection. The frozen QPs are 33 for Beauty, 31 for Bosphorus, 30 for HoneyBee, 28 for Jockey, 32 for ReadySteadyGo, 33 for ShakeNDry, and 33 for YachtRide. Each sequence keeps that QP and FEC configuration across every channel condition, every seed, and the 128-frame supplement. Over the 32-frame experiment the mean actual transmitted rates are

Table 8: Unified 1500-byte-MTU transport on UVG7, 32 frames, seeds 0–4. Network packets are erased and all transmission overhead is charged. Δ is the paired Full $-$ H.265+FEC PSNR difference; brackets give per-condition 95% hierarchical paired-bootstrap intervals, not simultaneous bands. The initial I-frame is reliable and rate-counted; rates and protocol are in Section 4.6 and Appendix C.

	PSNR (dB)			SSIM	
Channel	**Full**	**H.265+FEC**	**Δ [95% CI]**	**Full**	**H.265+FEC**
Clean	40.28	39.68	$+0.60$ [-0.15, $+1.13$]	0.9494	0.9437
IID 5%	39.84	39.68	$+0.16$ [-0.75, $+0.76$]	0.9479	0.9437
IID 10%	39.27	39.11	$+0.16$ [-0.90, $+1.40$]	0.9451	0.9388
IID 20%	38.12	34.94	$+3.18$ [-1.16, $+8.07$]	0.9398	0.8979
IID 40%	35.96	20.50	$\mathbf{+15.46}$ [$+13.79$, $+17.30$]	0.9217	0.7145
GE 10%, L=5	38.69	29.56	$\mathbf{+9.13}$ [$+6.32$, $+11.94$]	0.9399	0.8412
GE 10%, L=20	38.07	32.53	$\mathbf{+5.54}$ [$+2.87$, $+8.30$]	0.9325	0.8731
GE 20%, L=5	37.20	25.76	$\mathbf{+11.44}$ [$+8.85$, $+13.96$]	0.9285	0.7914
GE 20%, L=20	34.87	25.81	$\mathbf{+9.06}$ [$+6.90$, $+11.22$]	0.9031	0.7925
GE 40%, L=5	34.44	20.85	$\mathbf{+13.59}$ [$+11.15$, $+15.94$]	0.9034	0.7337
GE 40%, L=20	32.57	21.22	$\mathbf{+11.35}$ [$+8.90$, $+13.84$]	0.8830	0.7443

0.0784 bpp for Full and 0.0773 bpp for H.265+FEC, a -1.38% difference; the two are matched on average rather than sequence by sequence.

C.4 Channels and reliability

The main experiment spans eleven conditions: clean transmission, Bernoulli i.i.d. loss at $5/10/20/40\%$, and Gilbert–Elliott loss at $10/20/40\%$ mean rate with mean burst lengths of five or twenty *network packets*. Unlike the fixed-count uniform conditions of Section 4.1, the i.i.d. loss here is genuine Bernoulli sampling. GE is initialized in its stationary distribution and its state evolves continuously across P-frames. The 128-frame supplement evaluates i.i.d. 20% and GE 20% at L=20.

The initial I-frame is assumed to arrive reliably, but its transmission cost is included. P-frame metadata is repeated in transmitted packets and fully charged. Data, metadata, and FEC repair packets are all subject to erasure. If every packet of a Full frame is lost, the receiver repeats the previous displayed frame and preserves its current reference state.

C.5 Metrics, statistics, and 32-frame results

We report actual transmitted bpp, PSNR, and single-scale SSIM at Y:U:V weights of 6:1:1. Quality is the per-frame mean including the initial I-frame. Transmitted rate counts payloads, network and application headers, fragment descriptors, padding, and repair traffic, including bytes subsequently lost in the channel. Differences are Full minus H.265+FEC, with 95% intervals from $10{,}000$ paired hierarchical bootstrap replicates that resample videos and then seeds within videos while preserving complete temporal trajectories; intervals are computed per condition and are not simultaneous bands.

Table 8 reports all eleven 32-frame conditions. Full transmits 0.0784 bpp in 13.34 network packets per P-frame; H.265+FEC transmits 0.0773 bpp in 12.24 packets. The paired PSNR intervals at clean and i.i.d. 5/10/20% include zero. Every GE condition and i.i.d. 40% has an interval above zero, with gains from +5.54 to +15.46 dB; SSIM point estimates favor Full in all eleven conditions. Thus, the measured advantage is clearest under bursty and heavy loss, not uniformly separated from zero across all conditions. This comparison changes transport mapping and loss correlation without changing Full's logical units or model architecture.

Table 9: Unified MTU transport, 128-frame supplement, same frozen configurations as Table 8 and no intermediate I-frames. Full transmits 0.0753 bpp in 13.32 network packets per P-frame; H.265+FEC transmits 0.0744 bpp in 12.58 packets. Δ is the paired Full $-$ H.265+FEC difference with a 95% hierarchical paired bootstrap interval. The lower block restricts the same trajectories to frames 96–127.

	PSNR (dB)			SSIM	
Channel	**Full**	**H.265+FEC**	**Δ [95% CI]**	**Full**	**H.265+FEC**
All 128 frames					
IID 20%	35.68	29.37	$+6.31$ [$+0.69$, $+12.39$]	0.9243	0.8508
GE 20%, L=20	32.06	19.23	$+12.83$ [$+10.25$, $+15.19$]	0.8878	0.7289
Final window, frames 96–127					
IID 20%	33.60	25.25	$+8.35$ [$+1.37$, $+15.32$]	n/a	n/a
GE 20%, L=20	30.38	16.84	$+13.54$ [$+11.12$, $+15.89$]	n/a	n/a

C.6 Long-sequence supplement

Table 9 repeats the comparison over 128 frames under the same frozen configurations, with no additional I-frames inserted. Full transmits 0.0753 bpp in 13.32 network packets per P-frame against H.265+FEC's 0.0744 bpp in 12.58 packets, so the same modest rate and packet disadvantage carries over. The margin over the whole sequence is $+6.31$ dB at i.i.d. 20% and $+12.83$ dB at GE 20%, L=20; restricted to the final 32-frame window it grows to $+8.35$ and $+13.54$ dB. Both window intervals lie above zero, so the advantage over a frozen conventional baseline widens with distance from the I-frame rather than saturating.

D Method implementation and protocol details

D.1 Grouping and entropy models

The 128-channel continuous latent $\tilde{y}$ is partitioned into 32 groups of four channels and processed in four stages of eight groups. Later stages refine their scales from already reconstructed sender groups. This is the implemented configuration; the study does not optimize the number of groups or stages.

Scale representation. The refined scale of a group is compressed by a small analysis/synthesis pair. The scale encoder applies a DepthConvBlock mapping $4 \to 24$ channels followed by stride-2 residual blocks $24 \to 24$ and $24 \to 4$; the scale decoder applies $2\times$ upsampling residual blocks $4 \to 24$ and $24 \to 24$ followed by a DepthConvBlock $24 \to 4$. The side latent s_i therefore has four channels at one quarter of the spatial resolution in each dimension. Both endpoints apply the same QP-indexed learned scaling and the same side-latent synthesis, and the sender entropy-codes its residuals against the scale recovered from its *own quantized* side latent, which is what makes the two endpoints agree. Scale-to-CDF selection quantizes the recovered scale onto 128 logarithmically spaced levels spanning 0.11 to 16.

Entropy models. The side latent s_i is entropy-coded using a four-channel fully factorized model whose parameters are selected by the sequence-level QP and shared across groups. Its decoding therefore uses the packet's own bytes under the shared codec configuration, independently of z_i, other groups, temporal context, and receiver DPB values. The residual CDF is a function of decoded s_i and QP. The hyperprior/temporal scale tensor supplies its spatial shape, rather than its CDF values. Both sender and receiver apply the same scale decoder and discrete CDF-index selection.

Residual reconstruction. Agreement on the CDF fixes the transmitted symbols, not the reconstruction. The sender forms the quantized residual symbol

$$y = Q\big(\tilde{y}/q^S - \mu^S\big), \qquad \hat{\tilde{y}}^R = \big(y + \mu^R\big)\, q^R, \tag{10}$$

where q and μ are the quantization step and predicted mean at the endpoint indicated, and the implementation clips y to the coder's integer range. A receiver that decodes y exactly still reconstructs through its *own* μ^R and q^R, so identical residual symbols do not imply identical reconstruction parameters or identical pixels. For a bound (s_i, y_i) unit that did not arrive, $y_i = 0$, and the latent estimate is determined entirely by the receiver's mean and quantization step.

Rate surrogate. Training and coding use different probability models. The differentiable residual-rate term is a Laplace model integrated over unit-width quantization intervals, serving as a smooth surrogate for optimization, while the actual entropy coder uses discrete Normal CDFs with rANS; s and z use their learned factorized models in both settings. Evaluation is independent of the surrogate, since reported rates are actual transmitted bytes, and the exact symbol recovery of Section 4.2 concerns agreement between the two endpoints' coding distributions.

D.2 Training objective

For a normalized YUV444 frame, let e_Y, e_U, e_V be the per-plane pixel-mean squared errors of the receiver reconstruction, and let d_{RGB} be the RGB squared error summed over the three channels, without division by three, and divided by the pixel count. The YUV-to-RGB conversion is left unclamped. The per-frame distortion is

$$d_{\mathrm{YUV}} = 3\, e_Y^{5/6}\, e_U^{1/12}\, e_V^{1/12}, \qquad D_t = w_{[t \bmod 8]}\left(0.8\, d_{\mathrm{YUV}} + 0.2\, d_{\mathrm{RGB}}\right), \tag{11}$$

with frame weights $w = [0.52, 1.33, 0.52, 0.83, 0.52, 0.83, 0.52, 0.83]$ indexed by $t \bmod 8$ for residues 0 through 7. The first P-frame has $t = 1$ and therefore weight 1.33. These weights scale distortion only; they do not scale any rate term. The Lagrange multiplier is $\lambda(\mathrm{baseQP}) = 768^{\mathrm{baseQP}/63}$, and the objective averages $\lambda D_t + R_{y,t} + R_{z,t} + R_{s,t}$ over training samples and predicted frames, as in Equation 6. Distortion is taken on the receiver reconstruction; there is no separately weighted clean-reconstruction loss term.

Base QP is sampled uniformly over the integers 0 through 63. Temporal QP offsets $[0, 8, 0, 4, 0, 4, 0, 4]$ are applied across the frame cycle, while λ remains a function of the base QP alone. This training distortion is not the evaluation metric: reported quality is YUV420 PSNR and SSIM with $Y{:}U{:}V$ weights of 6:1:1, so training weights and evaluation weights should not be compared directly. Block sizes, padding, clipping bounds, and index-construction details follow the released code rather than being restated here.

D.3 Training trajectories and rate accounting

Training unfolds the clean sender and packet-erased receiver over a sequence of one I-frame and 31 P-frames. A single loss probability is drawn from $\mathrm{Uniform}(0, 0.5)$ per training sample and shared across that sample's sequence, so the network sees channels ranging from clean to severe. Hyperprior and bound residual-plus-side erasure masks are resampled for each P-frame. The sender updates its DPB without network erasures; the receiver updates its own DPB from the packet-erased reconstruction. Weights are shared, and receiver distortion is accumulated across frames as in Equation 6. This simulates reconstruction under state divergence without feedback. The −dual control of Section 4.3 keeps this packet-loss training but drops the separate clean sender trajectory, running both training passes on the receiver DPB. Formally, with a shared training state $\bar{H}_t$ and separately trained parameters ϕ,

$$\bar{u}_t = \mathcal{E}_\phi(x_t, \bar{H}_{t-1}), \qquad (\hat{x}_t^R, \bar{H}_t) = \mathcal{R}_\phi\big(\mathcal{A}_{m_t}(\bar{u}_t), \bar{H}_{t-1}\big), \tag{12}$$

so encoding and reconstruction use the same damaged DPB, in contrast to the separate states of Equation 5. The shared DPB is a training construct only; neither configuration uses feedback or endpoint state sharing at deployment.

The distortion term in Equation 6 is defined in Appendix D.2. The rate estimates cover y, z, and s, and use the training surrogates described above; they differ from evaluation rate, which counts actual transmitted bytes.

D.4 Logical packets and transmitted bytes

Each channel-group index emits one hyperprior unit z_i and one bound (s_i, y_i) unit. The internal frame container stores 32 stream triples, while the loss model acts on 64 logical units: 32 hyperprior units and 32 bound residual-plus-side units. The container's entry count is therefore distinct from the number of erasure units. The side and residual streams of a bound unit are retained or erased together.

Transmitted rate counts the I-frame, all z, s, and y payloads, a 388-byte length table per P-frame, and the outer frame header. Erased bytes are counted because they were transmitted. Length tables and frame headers are assumed reliably delivered. The simulation uses logical loss units within these frame containers rather than modeling datagram fragmentation, reassembly, or integrity checks. The unified MTU comparison of Appendix C supplies that mapping and charges its headers, descriptors, and padding. The transmission order and the burst process acting on it are specified in Appendix D.7. This byte accounting does not measure transport latency or memory overhead.

D.5 Erasure topologies and cross-method channel matching

Table 10 records what a native erasure unit is for each evaluated method, how many such units a P-frame carries, and what is exempt from erasure. Comparisons share the video, the seed, and the target (p, L) of Section 4.1, where p is the stationary erasure probability of a native unit and L the expected number of consecutively erased native units. This matching is over native units. It does not equalize erased bytes across methods, because unit sizes differ, and it does not equalize outage duration, because a burst of L units covers a different amount of coded content in each topology. Rankings should therefore be read as holding under a common unit-level channel, not under a common byte-level or time-level one.

Two kinds of exemption have to be kept apart, because they differ in what they cost. Information that is *assumed reliably delivered* is never erased and carries no repair bytes, so the assumption buys reliability without a modeled cost, although the information's own payload is still counted in the transmitted rate. Information that is *explicitly protected* is defended by repair symbols that are themselves transmitted, counted in the rate, and subject to erasure. Full uses only the first kind: its frame headers and length tables are assumed reliable and no repair bytes are modeled anywhere. The H.264/H.265+FEC baselines use the second, and their repair datagrams are part of their reported rate.

Two entries in Table 10 need qualification. GRACE's hyperlatent is treated as assumed reliable rather than as explicitly protected: it is not erased, its estimated entropy bits are already included in the reported rate, and no additional repair bytes are modeled for it. This is the configuration evaluated here; GRACE admits other protection settings.

For the traditional baselines the number of datagrams a P-frame carries is not a constant, and the repair count is not even a per-frame quantity. Repair datagrams belong to FEC blocks spanning three P-frames, so any per-frame figure is amortized across the block and does not measure repair dedicated to an individual frame. Table 11 reports, for each of the six evaluated configurations, the mean source datagrams per P-frame with its observed range and the amortized mean repair datagrams per P-frame, measured over all 217 UVG P-frames (seven sequences of 31 P-frames each). The protection parameter α governs how many repair datagrams are generated, through $R = \min(D, \lceil \alpha D \rceil)$ for a block of D source datagrams; α is not the FEC code rate, which for such a block is $D/(D + R)$.

Table 12 supplements the topology definitions with the sizes of the units the channel actually acts on and with the loss fractions those units realize, which is what makes the gap between unit-level and byte-level matching concrete. The *realized unit-loss fraction* is the number of erased native units divided by the number of native units of that same family transmitted over the 31 P-frames of the GOP; the intact I-frame contributes to neither term. The *realized byte-loss fraction* cannot be given a single cross-method definition, because the methods declare their transmitted bytes differently; it is computed under each method's own accounting, itemized below. What is common is the denominator's coverage: bytes that are assumed reliably delivered enter the denominator but never the numerator. Every row of Table 12 shows a byte-loss fraction below its unit-loss fraction; with variable-size units this is an observed property of these traces rather than a general bound.

Table 10: Native erasure topologies. p is the stationary erasure probability of a native unit and L the expected number of consecutively erased native units; cross-method comparisons share (p, L), the video, and the seed, with each codec erasing its own units. "Units" is the number of native erasure units per P-frame; it is not a constant for the traditional baselines, whose per-configuration counts are in Table 11. The "Assumed reliable" column lists *additional* P-frame content exempt from erasure: the initial I-frame is delivered reliably for *every* method, counted in every method's rate, and is outside every erasure pool, so it is not repeated per row. Matching (p, L) aligns channel statistics over native units, not erased bytes or outage duration. "Assumed reliable" information is never erased and carries no repair bytes; "explicitly protected" information is defended by transmitted repair bytes that are themselves erasable and counted in rate.

Method	**Native erasure unit**	**Units**	**Assumed reliable**	**Protection accounting**
Full (ours)	32 hyperprior units z_i; 32 bound units (s_i, y_i), erased atomically	64	Frame header, length table	No repair bytes modeled; reliable bytes counted in rate
GRACE	64 spatial latent units; 1 I-patch unit	65	Hyperlatent	No repair bytes modeled; hyperlatent entropy bits in reported rate
H.264/ H.265+FEC	Source and repair datagrams; protection block spans 3 P-frames	Varies	None beyond the I-frame	Repair datagrams transmitted, erasable, counted in rate

Table 11: Datagram counts for the traditional baselines, measured over all 217 UVG P-frames. Source counts are per P-frame, with the observed range in parentheses. Repair counts are *amortized*: repair datagrams belong to FEC blocks spanning three P-frames and cannot be attributed to an individual frame. The protection parameter α sets the number of generated repair datagrams through $R = \min(D, \lceil \alpha D \rceil)$ for a block of D source datagrams; it is not the code rate. These six rows are the configurations actually evaluated, not a full QP $\times$ protection grid.

Method	**Codec QP**	α	**Source/P (range)**	**Repair/P (amortized)**
H.264+FEC	42	0.80	2.14 (1–4)	1.84
H.264+FEC	33	0.80	6.67 (1–16)	5.47
H.264+FEC	30	0.85	10.02 (1–25)	8.70
H.265+FEC	40	0.80	2.21 (1–4)	1.93
H.265+FEC	32	0.80	6.69 (1–17)	5.53
H.265+FEC	29	0.85	10.24 (1–25)	8.88

The per-method accounting behind the last column is as follows. A single table note reading "erased payload bytes" would not be accurate enough, because the three methods do not measure the same thing.

- **Full.** The two rows' unit-loss values use their own unit families as denominators, but both byte-loss values use the same denominator: Full's total transmitted bytes over the complete 32-frame sequence. The two entries are therefore *contributions* to one overall byte-loss figure and add to approximately 17.37%; they are not two independent fractions. The (s, y) units dominate that total, being roughly eight times the size of a hyperprior unit on average.
- **GRACE.** Packet sizes are expressed in reported-rate-equivalent bytes, comprising estimated latent entropy together with the actual I-patch bytes. The available artifacts do not provide the actual arithmetic-coded packet lengths, so 15.48% is likewise a loss fraction defined under reported-rate accounting rather than over measured packetized bytes. It is not directly comparable in kind to the other rows.

Table 12: Packet sizes and realized loss under Gilbert–Elliott loss at $p = 20\%$, $L = 20$, over UVG with equal weighting across the three rate regions, the seven videos, and five seeds. Unit loss uses each row's own unit family as the denominator. Byte/rate loss is computed under each method's declared transmitted-byte accounting, itemized in the text: Full's two entries share one denominator and are contributions summing to 17.37%; GRACE's is on a reported-rate-equivalent basis; the FEC entries measure channel erasure before FEC recovery. Sizes are per native unit, in bytes.

Method / unit	Mean (B)	Median (B)	Unit loss	Byte/rate loss
Full (ours), z_i	59.17	31.00	19.81%	3.27%
Full (ours), (s_i, y_i)	472.51	284.00	20.54%	14.10%
GRACE	280.38	217.57	20.11%	15.48%
H.264+FEC	1467.00	1467.00	21.69%	18.58%
H.265+FEC	1467.00	1467.00	19.67%	17.06%

- **H.264/H.265+FEC.** Each 1467-byte datagram is a 1450-byte payload, including padding, plus a 17-byte header, so the erased-byte count includes erased header bytes. These figures measure channel erasure *before* FEC recovery; they are not residual loss after decoding, and they therefore say nothing on their own about what the baselines reconstruct.

The realized unit-loss fractions range from 19.67% to 21.69% against a nominal 20%, which is the ordinary spread of a bursty channel at $L = 20$ over a finite number of units rather than a discrepancy between methods.

D.6 H.264/H.265+FEC BASELINE CONFIGURATION AND OPERATING-POINT SELECTION

Encoding and packetization. Both traditional references encode with FFmpeg 8.1.2 using libx264 0.164.3095 (`baee400`) and libx265 3.5+1-`f0c1022b6` respectively, at the `veryfast` preset, with `tune=psnr` additionally set for H.264. Both use the GOP32 structure used throughout this work: one I-frame followed by 31 P-frames, no B-frames, scene-cut keyframe insertion disabled, and open GOP disabled. The coded bitstream is packetized into datagrams carrying 1450-byte payloads, with a 17-byte header counted for every source and repair datagram; padding within a datagram is included in the transmitted rate. Protection blocks cover three consecutive P-frames. The 31 P-frames of a sequence therefore form ten three-frame blocks and one final single-P-frame block, and that final block uses the same protection rule as the others. No fictitious frames are used to pad it out, and nothing carries over into the next sequence. A block of D source datagrams generates $R = \min(D, \lceil \alpha D \rceil)$ repair datagrams, with α the redundancy parameter of Table 13. The initial I-frame is delivered reliably and counted in rate, and is excluded from both FEC protection and erasure.

Idealized recovery. FEC is modeled as *ideal block-erasure recovery* rather than simulated at the code level. Within a block, all missing source datagrams are recovered whenever the total number of erased source and repair datagrams is at most R; otherwise none are. No Reed–Solomon, Raptor, or Tambur encoder or decoder is executed. Two properties of this model matter for the comparison. Repair traffic is real in the accounting sense: it is transmitted, counted in the rate, and itself subject to erasure, so protection is paid for. And recovery within a block is optimal (no decoding failures below the erasure threshold and no partial recovery above it), so the baselines are credited with everything a realized code could deliver and none of its inefficiency. Any advantage we report is therefore measured against protection that never underperforms its own bound.

Decoding when recovery fails. When a block exceeds its erasure threshold, the evaluation does not simulate the consequence abstractly: it damages the actual bitstream bytes corresponding to the unrecovered source datagrams and passes the resulting stream to FFmpeg. Explicit FFmpeg error concealment is disabled, which keeps the comparison on the axis this paper is about: how transmitted rate is divided between source coding and protection, rather than how well a particular decoder's concealment heuristics happen to interpolate. What remains is the decoder's own internal error handling on a damaged stream, together with a display-level fallback that repeats the previous displayed frame whenever the decoder emits no output at all.

Table 13: Selected operating points for the traditional baselines. Codec QP and the redundancy parameter α are the six configurations of Table 11, whose per-configuration datagram counts they share. "Target region" is the Full rate region the point was selected against, "Target bpp" that region's rate, and "Actual bpp" the point's own transmitted rate; "Error" is the signed relative difference, computed from unrounded values rather than the six-decimal figures shown. Selection minimized absolute relative error in total transmitted rate against a Full target rate, using no quality measurement; total rate includes the I-frame, source and repair payloads, padding, and headers.

Method	Codec QP	α	Target region	Target bpp	Actual bpp	Error
H.264+FEC	42	0.80	QP39, low	0.0253	0.0250	-1.02%
H.264+FEC	33	0.80	QP55, middle	0.0731	0.0746	$+2.09\%$
H.264+FEC	30	0.85	QP63, high	0.1148	0.1140	-0.70%
H.265+FEC	40	0.80	QP39, low	0.0253	0.0261	$+3.28\%$
H.265+FEC	32	0.80	QP55, middle	0.0731	0.0748	$+2.29\%$
H.265+FEC	29	0.85	QP63, high	0.1148	0.1160	$+1.02\%$

Operating-point selection. Operating points were selected by a rate rule alone. For each codec, the codec QP was scanned over 28–51 and the redundancy parameter over $\alpha \in \{0.80, 0.85, 0.90, 0.95, 1.00\}$, giving 120 candidate combinations. For each of Full's three target rates, we selected the combination whose *total transmitted rate* had the smallest absolute relative error against that target, where total rate includes the I-frame, source and repair payloads, padding, and headers. Neither clean PSNR nor lossy PSNR or SSIM entered the selection. The six selected points all took $\alpha \in \{0.80, 0.85\}$; recomputing the selection reproduced all six, which establishes that the rule is reproducible, not that the resulting configurations are quality-optimal.

Across the six selected configurations the signed rate error relative to Full ranges from -1.02% to $+3.28\%$ (Table 13). We therefore describe the baselines as *approximately* rate-matched rather than strictly rate-matched; not every point falls within 3%, and every method is plotted and scored at its own actual transmitted rate rather than at a nominal common rate.

Why the selection is rate-based. Selecting on rate alone keeps the baselines independent of the quantity being compared: no baseline is tuned on the PSNR or SSIM it is later scored by, so the comparison cannot be biased by choosing operating points that flatter or disadvantage either side. It also matches how such a system is deployed. A sender that cannot observe the channel commits to one (QP, α) pair per rate region and holds it across conditions, as these baselines do across the loss sweep. Re-tuning the source/protection split separately at 0% and at 40% loss would grant them information our own codec never receives either. The `veryfast` preset follows the same reasoning: this work targets real-time delivery, and a real-time preset is the operating regime in which a loss-resilient codec is actually used.

A supplementary screening supports the high-redundancy range spanned by the candidate grid. At the tested middle-rate H.265 operating region, over the seven UVG sequences, three seeds, and five equally weighted channel conditions, raising α from 0.5 to 0.8 improves mean PSNR by $+1.67$ dB (CI $[0.78, 2.63]$), and raising it further to 1.0 adds $+1.38$ dB (CI $[0.34, 2.53]$). For the tested channel mixture, these results support $\alpha \in [0.8, 1.0]$ as a reasonable high-protection range. Two qualifications come from the same screening. The α=0.8 against α=0.75 difference is $+0.20$ dB (CI $[-0.10, 0.62]$) and is not separated from zero, so nothing here marks 0.8 as a threshold. The ordering is also not uniform across conditions: at i.i.d. 10%, α=0.5 exceeds α=0.8 by $+0.42$ dB (CI $[0.32, 0.52]$), the lighter setting winning where loss is mild. Rates are approximately matched: α=0.75 and α=0.8 share a source point, with the latter spending about 2.96% more total rate. The screening covers one rate point on one codec, so it supports the redundancy range rather than establishing an optimal policy.

D.7 Packet ordering and interleaving sensitivity

For the logical-unit experiments, frames are transmitted chronologically and each P-frame sends the block $z_0, \ldots, z_{31}$ followed by the block $(s_0, y_0), \ldots, (s_{31}, y_{31})$. The Gilbert–Elliott chain advances

Table 14: Ordering sensitivity for Full. Mean PSNR change (dB) of the interleaved order $z_0, (s_0, y_0), \ldots, z_{31}, (s_{31}, y_{31})$ relative to the main order $z_0, \ldots, z_{31}, (s_0, y_0), \ldots, (s_{31}, y_{31})$; positive means the interleaved order is better, so the main results use the worse of the two. Gilbert–Elliott loss at $p = 20\%$, seven UVG sequences, five seeds, 35 paired video-seed observations per cell (210 instances). Intervals are 95% hierarchical paired bootstrap intervals computed as in Section 4.1, resampling videos and then seeds within each sampled video while preserving the pairing, over $10{,}000$ replicates. Only the QP39, $L{=}5$ interval contains zero.

Burst		**QP39**	**QP55**	**QP63**
$L{=}5$	ΔPSNR (dB)	$+0.05$	$+0.37$	$+0.76$
	95% CI	$[-0.01, +0.12]$	$[+0.16, +0.67]$	$[+0.29, +1.35]$
$L{=}20$	ΔPSNR (dB)	$+0.67$	$+0.81$	$+1.67$
	95% CI	$[+0.30, +1.15]$	$[+0.38, +1.34]$	$[+0.85, +2.68]$

one step per erasure unit and is not reset at frame boundaries, so a burst that begins inside one frame's z block can run through that frame's (s, y) block and into the following P-frame. This order was fixed before evaluation.

The contiguous layout concentrates each family's units, so a single burst can remove a long run from within one family. We therefore re-ran Full under the interleaved order $z_0, (s_0, y_0), z_1, (s_1, y_1), \ldots, z_{31}, (s_{31}, y_{31})$, which alternates the two families so that a burst of a given length spans both. Only Full's schedule changes: payloads, rate accounting, baselines, and channel parameters are unchanged, and reordering moves the same units, so the same bytes are transmitted and this is a quality comparison at fixed rate. The sweep covers the seven UVG sequences, QP 39/55/63, five seeds, and Gilbert–Elliott loss at $p = 20\%$ with $L = 5$ and $L = 20$: 210 evaluation instances, or 35 paired video-seed observations in each of the six cells.

Table 14 reports the mean PSNR change of the interleaved order relative to the main order. All six point estimates are positive, ranging from $+0.05$ to $+1.67$ dB, with the larger differences at the longer burst length and the higher rate. Full's main results are thus produced by the worse of the two orderings tested and do not come from selecting the more favorable schedule.

Five of the six intervals in Table 14 exclude zero, the exception being QP39 at $L{=}5$ ($[-0.01, +0.12]$): interleaving helps at the middle and higher rate regions under short bursts and at all three under long bursts. The experiment bounds Full's own sensitivity over the pair of layouts tested; it varies only Full's schedule, leaving the baselines on their native orders, so it speaks to the robustness of Full's reported numbers rather than to ranking stability under joint rescheduling.

D.8 Evaluation grids and statistical procedures

D.8.1 Evaluation grids

Main logical-unit comparison. The main UVG evaluation uses the seven canonical sequences over the first 32 frames (1 I + 31 P), QP39/55/63, and channel seeds 0–4. It sweeps i.i.d. target loss from 0 to 80% in 10% steps and GE mean loss of 10/20/40% at mean burst lengths $L{=}5/20$. Panels labelled $L{=}1$ reuse the i.i.d. observations as a memoryless proxy and are not independently measured GE runs.

Ablations. Component ablations report i.i.d. 0/10/20/40% and GE burst lengths 5/20 under the shared protocol. Table 1 summarizes the QP55 operating point; the controls remain trained under packet loss, as detailed in Section 4.3 and Appendix A.5.

Distinct diagnostic settings. The held-payload diagnosis uses QP21/QP42 caches. The separate five-mode diagnosis uses QP42 and seeds 0–2 on the first 32 frames of the seven UVG sequences. These diagnoses are detailed in Appendix E and must not be pooled with the Full-codec verification at QP39/55/63.

Additional evaluations. Appendix A.6 specifies the cross-dataset grid. The continuous 128-frame logical-unit evaluation uses Full and $-$dual at base QP55 (Appendix B). The unified-MTU compar-

ison is a separate packet-domain experiment with its own coding, channel, and accounting settings (Appendix C).

D.8.2 FIXED-COUNT UNIFORM ERASURE

For the logical-unit experiments, "i.i.d." denotes fixed-count uniform erasure. At target rate p over N eligible native units, $K = \mathrm{round}(pN)$ units are erased uniformly without replacement. Prefixes of a single random permutation are reused across target rates. Erasure positions are therefore exchangeable rather than independent Bernoulli trials, and the realized fraction is K/N, equal to p only up to rounding. The initial I-frame lies outside the erasure pool. Burst length L is controlled for GE but is not an independently chosen parameter of the i.i.d. conditions. Cross-method comparisons share the video, seed, and target channel parameters, while each codec erases its own native units; they do not require byte-identical masks across different topologies. The unified-MTU comparison instead uses the sampling specified in Appendix C.4.

D.8.3 AGGREGATION AND RUN HANDLING

UVG point estimates first average the five seeds within each video and then weight the seven videos equally. Formal paired comparisons use 95% intervals from 10,000 nested hierarchical paired bootstrap replicates, resampling videos and then seeds within each selected video. Paired quantities retain their video-seed pairing. The rate-robustness frontier figures use the separate 5,000-replicate crossed procedure over seven videos and three nominal seeds described in Appendix A.2; those intervals are marginal AUC intervals, not paired-difference intervals. Long-sequence and unified-MTU resampling preserve whole trajectories as specified in Appendices B and C. Failed runs follow a single identical-retry policy with valid/planned reporting.

E EXTENDED FAILURE ANALYSIS AND DIAGNOSTIC EVIDENCE

This appendix expands the failure analysis in Section 2 and the diagnostics summarized in Section 4.2. It separates the five-mode QP42 diagnosis, the QP21/QP42 held-payload intervention, and the QP39/55/63 verification of the proposed codec; they have different settings and roles.

E.1 FAILURE MECHANISM AND DESIGN OVERVIEW

The following analysis concerns the context-conditioned backbone before adding the side information of Section 3.

In a conditional neural codec the sender entropy-codes the residual symbols of a logical loss unit i against a discrete CDF sequence selected from its priors and context, producing payload bytes B_i; the receiver recovers those symbols only by selecting the same sequence. Packet loss never interferes with the sender, which continues to encode correctly under its own intact state. What it changes are the inputs from which the *receiver* makes that selection. Figure 3 illustrates the resulting failure.

Two routes to a perturbed condition. Erasing a frame's hyperprior packets z_i leaves the receiver inferring its entropy parameters from an incomplete hyperprior representation, and retaining the residual bytes does not help: the scale the receiver recovers may differ from the one the sender used. Independently of that, past losses without feedback leave the receiver updating its decoded-picture buffer (DPB) from damaged reconstructions while the sender continues from its own unerased ones. The temporal context the two endpoints derive therefore differs, and their entropy conditions can disagree even when every packet relevant to the current frame arrives intact. The second route opens without any prior entropy failure, since missing residual content is enough to start the buffer diverging. Either route on its own is sufficient.

From a perturbed scale to wrong symbols. A perturbed input does not make the receiver's network fail visibly. It produces a scale that is numerically valid, and nothing in the received bytes signals that the scale differs from the sender's. Whether the difference matters is settled at the discrete boundary: coding selects one of a finite set of CDFs, so a scale deviation changes the distribution actually used only when it crosses a selection boundary, and deviations that stay within a cell

leave the coding distribution intact. That threshold is what makes the failure a measurable event. The diagnostic codec exposes the selection as an explicit index sequence, so the receiver's indexes can be compared against the sender's for the very same payload bytes, which is the intervention Section 2.3 reports.

Propagation within the entropy stream. Where the selected distribution does differ, the consequence is not confined to one sample. The residual is coded with rANS, whose decoder state advances with each symbol under the distribution used to decode it; decoding under the wrong distribution therefore risks both a wrong value and a decoder state the sender never occupied. Later symbols in the same stream are then read from a desynchronized state and can be wrong even where their own selection was correct. How far the damage runs depends on where the divergence falls and on how the stream is structured, but its unit is a stretch of the stream, not a single sample. The held-payload audit of Section 2.3 observes exactly that signature: boundary failures on payloads that were fully received.

Propagation across frames. A second and slower path runs through reconstruction. Whatever the receiver decodes yields a frame that differs from the sender's, and that frame updates the receiver's DPB, which perturbs the temporal context of later frames by the second route above. The loop closes, so the effect of an erasure need not be confined to the frame that suffered it. This path is a deduction from the codec's structure; what we measure are its consequences. Restoring a clean sender-compatible receiver DPB recovers substantial quality (Appendix E.3), and reference-state discrepancy grows rather than settles along a long predictive chain (Appendix B.5). Those measurements establish that reference divergence matters and persists under continuing loss. The cost of an erasure is therefore not bounded by the content it removed, and Section 2.3 measures how far past that bound it reaches.

E.2 Locality and post-entropy latent masking

Equation 2 quantifies over admissible loss histories, whereas any experiment covers a finite family of them, so we keep the property and its evidence distinct throughout. Section 4.2 audits a specified set of erasure conditions and reports what was observed over them. The separate argument that the proposed codec satisfies the property *by construction* (the information selecting the distribution travels inside the unit it applies to) is made in Section 3.2.

What this implies for post-entropy latent masking. Packet loss in learned codecs is frequently simulated after entropy decoding, by zeroing the latents carried by erased packets and keeping the rest. That procedure presupposes locality: it assumes every retained payload was already interpreted as the sender intended, and so models loss purely as absence. Where locality holds, as it does for codecs whose packets carry the distribution information needed to decode them, the assumption is sound, and masking faithfully represents erasure at the entropy-symbol layer. Where the relevant cross-packet dependencies are present and locality fails, masking omits the corruption of symbols that were physically received, and the simulated and actual bitstreams part company at that layer. Which case a given codec falls into is settled by checking Equation 2 for it. For the backbone studied here it falls into the second, which is why the diagnoses in Appendices E.3 and E.4 are run on the actual bitstream and paired with a held-payload audit instead of on masked latents.

E.3 Five-mode diagnosis

Figure 15 examines the two routes into the entropy-decoding failure and the reconstruction role of reference-state divergence. It compares five modes of the diagnostic DCVC-RT codec under a fixed 2:1 z:y nominal loss-budget rule. Receiver replay and Oracle-s share sender payloads and loss masks. The y-only curve has higher quality than z-only under this rule, which assigns about twice as many erased packets to z-only. The z-only degradation despite retained residual bytes, together with the gains from supplying sender scale/CDF indexes (Oracle-s), supports the importance of consistent entropy conditions.

Periodically restoring a clean sender-compatible receiver DPB (Replay + reset every 8 frames) also substantially improves quality. This supports the effect of accumulated reference-state divergence on reconstruction. Reset quality remains below Oracle-s at every plotted nonzero loss budget, so

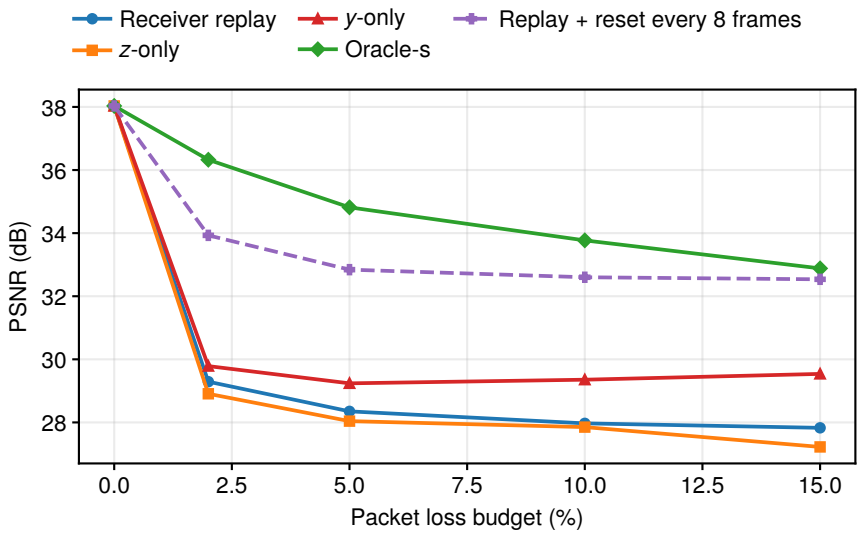

Figure 15: Five-mode diagnosis on DCVC-RT before adding entropy side information (UVG, 7 sequences, first 32 frames, QP 42; seeds 0–2 under loss). The vertical axis is 32-frame mean YUV420-weighted PSNR; the horizontal axis is nominal loss budget under the fixed 2:1 packet rule. Receiver replay and Oracle-s share payloads and loss masks; z-only and y-only redistribute erasures. Oracle-s supplies sender CDF indexes. The dashed curve uses the receiver-replay payloads and masks but restores the receiver's clean sender-compatible state before frames 8, 16, and 24 (zero-based). Parameter and state injections are oracle interventions whose communication cost is not included.

periodic restoration alone is insufficient to recover the benefit of sender-consistent CDFs in this setting. A restored DPB changes the receiver's entropy conditions and its reconstruction context together, so this intervention measures the importance of divergence rather than isolating one causal path. Together, the interventions identify scale/CDF consistency as a key bottleneck and motivate handling reference-state divergence alongside entropy decoding. The injected parameters and states are diagnostic oracles with no assigned transmission cost.

E.4 Held-payload diagnosis and entropy-decoding audit

Held-payload diagnosis. In the separate QP21/QP42 diagnosis on the unmodified backbone, ordinary receiver replay produces index and boundary failures on byte-identical retained payloads. Substituting sender indexes eliminates index, symbol, boundary, and zero-fill failures and improves pooled-QP PSNR by $+5.64/+4.89/+3.69/+2.73$ dB at $2/5/10/20\%$ loss, with all confidence intervals above zero and all seven video directions positive. Because the retained bytes and the erasure pattern are unchanged, this isolates the value of restoring entropy conditions for the same received bytes. The intervention is an oracle: it supplies indexes the receiver has no way to obtain. What it establishes is the size of the prize, and therefore the target a deployable design has to reach by transmitting something instead.

Proposed-codec verification. The side information of Section 3.2 is built to deliver the same condition the oracle imposed, at a rate cost counted in Table 1. Across QP 39/55/63 and 23 independent $Z/Y\,S$ erasure conditions per QP, the real-rANS decoder of the proposed codec recovers **all 453,120 retained CDF indexes and residual symbols with zero mismatch and zero scale difference**. The agreement is exact rather than approximate, so on every retained payload tested the entropy interpretation identified by the diagnosis is restored in full, by transmission rather than by injection.

F ADDITIONAL RELATED-WORK BACKGROUND

This appendix expands the background discussion of Section 5. The main text states the distinctions needed for the paper's positioning; the following paragraphs provide the fuller methodological context and original references.

F.1 LEARNED IMAGE COMPRESSION AND ENTROPY MODELS

Learned image compression established scale hyperpriors (Ballé et al., 2018), joint autoregressive and hierarchical priors (Minnen et al., 2018), Gaussian-mixture likelihoods (Cheng et al., 2020), and channel-wise (Minnen & Singh, 2020) and checkerboard (He et al., 2021) contexts. These models improve compression by predicting latent distributions from available information. We use this conditional-modeling capability while making the scale needed to interpret a received residual available within its packet.

F.2 NEURAL VIDEO COMPRESSION AND THE DCVC FAMILY

Neural video compression progressed from DVC (Lu et al., 2019) and scale-space flow (Agustsson et al., 2020) to the DCVC family: contextual coding (Li et al., 2021), temporal context mining (Sheng et al., 2022), hybrid entropy modeling (Li et al., 2022), diverse contexts (Li et al., 2023), and feature modulation (Li et al., 2024). We build on the real-time DCVC-RT (Jia et al., 2025), retaining sender-side conditional modeling and adapting the entropy and reconstruction paths to packet loss.

F.3 LOSS-RESILIENT NEURAL CODING

GRACE (Cheng et al., 2024) combines independent packet entropy coding and per-packet distribution information with loss-aware reconstruction training and feedback-based state resynchronization. NeuralMDC (Hu et al., 2024) encodes interleaved descriptions independently and uses a masked transformer to infer lost tokens, following the broader multiple-description approach (Goyal, 2001; Wang et al., 2005). For images, ResiComp (Wang et al., 2025) uses masked visual token modeling for entropy modeling and concealment, while satellite-oriented coding (Sha et al., 2025) trains for channel erasures under a Gilbert–Elliott model. These designs address the availability of received representations and the reconstruction of missing content.

F.4 CLASSICAL ERROR-RESILIENT TRANSPORT

FEC (Shokrollahi, 2006), error control (Wenger, 2003), and concealment (Kazemi et al., 2021) protect delivery or reconstruct missing content. Our evaluated logical-unit baselines and unified-MTU comparator use different recovery and concealment configurations, detailed in Appendix D.6 and Appendix C. Latency and reordering are outside the measured scope.