Two-way Application–Radio Cross-Layer FEC and HARQ Control for Deadline-Aware Frame Recovery

Masayuki Kurata†, Akihiro Nakao†
†*The University of Tokyo*, Tokyo, Japan
ma-kurata@g.ecc.u-tokyo.ac.jp, nakao@nakao-lab.org

***Abstract*—For reliable real-time video communication, application servers rely on forward error correction (FEC) to recover frames when packets are lost. In radio access networks (RANs), packets are scheduled at the transport block (TB) level, with unique retransmission mechanisms such as hybrid automatic repeat request (HARQ). Since FEC and HARQ are controlled independently, behavior mismatches arise in loss structure (packet-level loss assumptions vs. TB-level loss correlation) and optimization objectives (deadline-constrained frame recovery vs. TB-level decoding reliability). To bridge these gaps, we propose a two-way cross-layer FEC–HARQ control mechanism. In the radio-to-application direction, the server infers the TB-level loss structure from user equipment (UE) radio-layer feedback and jointly determines pacing, inter-frame spreading, and redundancy to improve frame recovery while reducing FEC overhead. In the application-to-radio direction, the UE signals the RAN to early terminate HARQ retransmissions for packets that are already recovered or past their deadlines, freeing resources for packets still likely to meet the deadline. The ns-3 simulations show up to a 47% mean relative reduction in frame loss rate over adaptive FEC baselines, while matching high-redundancy baselines with up to a 30% mean relative reduction in redundancy.**

***Index Terms*—Cross-layer design, Forward error correction, Hybrid automatic repeat request, Deadline-aware communication**

I. Introduction

Reliable real-time video communication is a key enabler for emerging services such as cloud gaming and mission-critical industrial applications, especially under highly dynamic radio conditions. To satisfy these requirements, forward error correction (FEC) is widely adopted as a proactive reliability mechanism that introduces controlled redundancy into the transmitted frame stream [1]. In FEC-based transmission, video frames are segmented into multiple data packets, and additional FEC packets are generated from them. By enabling loss recovery without retransmissions, FEC prevents retransmission-induced delays and helps avoid deadline violations.

Despite its advantages, the effectiveness of application-layer FEC can significantly degrade in mobile networks. This is because FEC is configured at the packet level while the underlying radio access network (RAN) transmits packets through transport blocks (TBs) and mitigates channel errors using hybrid automatic repeat request (HARQ) retransmissions [2]. The resulting difficulty is not merely a layering issue: *application-layer FEC typically assumes packet-level loss behavior and aims for deadline-constrained frame recovery, whereas radio-layer HARQ operates on TB-level decoding outcomes and aims to improve TB reliability through retransmissions.* Coordinating FEC with these radio-layer mechanisms, therefore, poses three challenges:

ISBN 978-3-903176-82-9 © 2026 IFIP

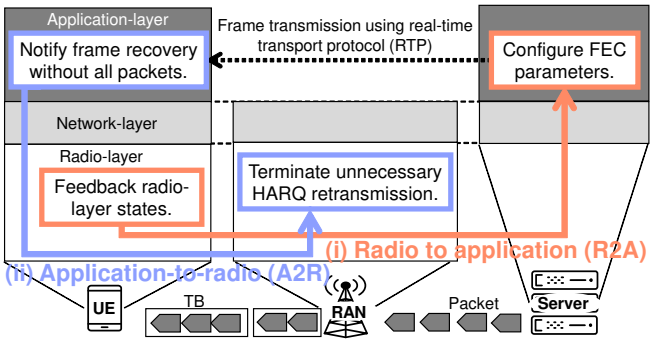


Fig. 1: Overview of the proposed two-way cross-layer mechanism: (i) Radio-to-application (R2A) control optimizes FEC parameters, while (ii) application-to-radio (A2R) control terminates inefficient HARQ retransmissions to conserve resources.

- **Loss structure mismatch:** Post-HARQ residual losses are correlated within individual TBs and occur in bursts across consecutive TBs owing to non-line-of-sight (NLOS) propagation and fading [3]; such temporal and intra-TB correlations cannot be captured by packet-level FEC assumptions or pre-HARQ indicators such as the signal-to-interference-plus-noise ratio (SINR).
- **Objective mismatch:** HARQ maximizes TB decoding probability, whereas the application requires frame recovery within a deadline. This mismatch causes unnecessary retransmissions for frames already recovered or expired.
- **Independent control:** Although these structural and objective constraints are inherently coupled, FEC redundancy and HARQ retransmissions are typically controlled independently at their respective layers.

Existing efforts have studied adaptive FEC [4]–[7] and timeliness-aware HARQ [8]–[10], but they optimize each in isolation and thus fail to capture the coupling between loss structure and deadline-driven recovery objectives.

To jointly resolve these coupled mismatches, we propose a two-way cross-layer FEC–HARQ mechanism, shown in Fig. 1, consisting of: (i) radio-to-application cross-layer FEC (**R2AX-FEC**) and (ii) application-to-radio cross-layer HARQ (**A2RX-HARQ**). *Through this bidirectional coordination, FEC adapts its redundancy to the radio-layer loss structure, and HARQ retransmissions are terminated according to application-layer frame-recovery status and deadline viability. As a result, the*

mechanism improves the deadline-constrained recovery rate while reducing redundancy overhead.

For R2AX-FEC, we determine packet pacing, inter-frame spreading span, and redundancy online from UE-observed TB/HARQ feedback. First, to reduce the co-location of successive packets from the same frame within a TB, we set the pacing interval based on the UE-observed TB delivery interval so that packets are more likely to be scheduled in distinct TBs. Next, to mitigate bursty post-HARQ TB losses, we estimate burst characteristics using a two-state loss model [11] and spread each frame's FEC packets across subsequent frames over a span exceeding the typical burst length while remaining within the deadline budget. Finally, given the pacing interval and span, we estimate the per-frame failure probability using Bayesian updates [12]. Specifically, we treat losses on TBs that carry data packets as correlated, whereas losses on dispersed TBs that carry FEC packets are assumed to be approximately independent. We then set the redundancy to the minimum value that satisfies the target frame-recovery constraint.

For A2RX-HARQ, the UE extends the HARQ feedback alphabet from $\{\mathrm{ACK}, \mathrm{NACK}\}$ to $\{\mathrm{ACK}, \mathrm{NACK}, \mathrm{SKIP}\}$, where SKIP terminates retransmissions for TBs carrying packets of frames that have already been recovered or whose deadlines have expired. This extension is compatible with the 3GPP HARQ framework [2] using cross-layer interfaces [13]. By enabling early termination, A2RX-HARQ frees radio resources for subsequent TBs and, in turn, relaxes the deadline constraint on R2AX-FEC.

Using the ns-3 simulator with the 5G-LENA module [14], we evaluate R2AX-FEC and A2RX-HARQ in a dedicated single-UE scenario representing a network slice [15]. R2AX-FEC reduces the frame loss rate by up to 47% relative to Bolot [16] and USF [17] under high packet loss rates, while achieving comparable recovery to WebRTC [18] with up to 23% less FEC redundancy. Adding A2RX-HARQ further lowers both the frame loss rate and FEC overhead, matching WebRTC with up to 30% less redundancy, with the largest gains under severe channel conditions.

Our contributions to the problem statement (§ III) are below:

- We formulate the parameter selection problem in R2AX-FEC as a nonconvex mixed-integer nonlinear program (MINLP), which is computationally intractable in general. To solve this online, we propose a statistical decomposition algorithm using TB/HARQ feedback (§ IV).
- We present a system model for A2RX-HARQ and discuss its implementation feasibility. We further provide a theoretical analysis showing that A2RX-HARQ relaxes the deadline constraint on R2AX-FEC (§ V).
- We evaluate the proposed R2AX-FEC and A2RX-HARQ mechanisms using ns-3 simulations and demonstrate their effectiveness under lossy mobile network conditions. We also discuss limitations and future directions (§ VI).

II. Related Work

This section covers FEC and HARQ principles, reviews existing efforts, and states the need for cross-layer coordination.

A. Forward Error Correction (FEC)

FEC is a widely adopted technique for real-time frame recovery in video transmission because it enables proactive loss recovery without retransmissions [1]. In particular, maximum distance separable (MDS) coding provides strong robustness against packet loss by allowing a frame consisting of N data packets to be recovered from any N packets out of the total $N + K$ transmitted packets, where K denotes the number of FEC packets. Recent work has demonstrated that MDS-based FEC can achieve high packet loss robustness while still meeting strict real-time deadline constraints [19].

Since additional FEC packets reduce the effective data rate, a key objective is to scale redundancy for sufficient robustness with minimal overhead. Recent studies have investigated learning-based adaptive schemes to maintain a high recovery probability while minimizing redundancy, demonstrating that such adaptive control significantly improves real-time video quality compared with static redundancy configurations [4]–[6]. More recently, measurement and control frameworks have been studied to tighten cross-layer adaptation loops for real-time video, indicating that practical redundancy control benefits from explicit coordination with radio-layer dynamics [20]. In practice, cross-layer designs that incorporate radio-layer channel quality indicators (e.g., SINR) have also been proposed for application-aware video scheduling [7].

B. Hybrid Automatic Repeat Request (HARQ)

HARQ [2] is a radio-layer reliability mechanism that combines error-detection feedback $\{\mathrm{ACK}, \mathrm{NACK}\}$ with channel coding to improve decoding reliability over radio channels. Upon receiving TBs from the RAN, the UE attempts decoding; failed decodes lead to NACK-driven retransmissions. To limit radio resource consumption, HARQ operates under a bounded retransmission budget and uses redundancy versions (RVs) across retransmissions to provide complementary coded bits.

Maximizing radio-layer reliability via HARQ can be suboptimal for real-time delivery: extra retransmissions improve decoding but may delay newer updates and make delivered information stale. Timeliness-aware HARQ therefore optimizes information freshness rather than radio-layer reliability alone. Prior work has proposed partial and non-orthogonal retransmissions to limit latency and improve remote estimation [8], analyzed deadline-oriented multi-user scheduling under practical HARQ constraints [9], and used reinforcement learning to adapt scheduling and retransmission decisions when success probabilities are unknown [10]. More recent studies extend freshness-aware objectives to incorporate correctness and distortion, further emphasizing the need to coordinate retransmission control with application-level freshness [21].

III. Problem Statement

From reviews in § II calling for coordination of application-layer FEC with radio-layer HARQ, three challenges emerge: *(1) post-HARQ residual losses are correlated at the TB level and bursty over time, violating packet-level FEC assumptions; (2) HARQ maximizes TB decoding probability, whereas FEC*

requires deadline-bounded frame recovery; and (3) loss patterns and recovery objectives are coupled, yet redundancy and retransmissions are optimized independently across layers.

First, post-HARQ residual losses exhibit TB-level correlation and temporal bursts due to fading [3], which packet-level loss models assumed in existing FEC designs fail to capture [4]–[6]. Although widely used as a cross-layer indicator, SINR reflects pre-HARQ conditions and therefore fails to capture this residual-loss structure [7]. In contrast, UE-observed TB delivery timing and HARQ outcomes directly reveal these post-HARQ properties, but their systematic use for FEC parameterization has not been sufficiently explored.

Second, HARQ increases the probability of TB decoding, whereas the application requires frames to be recovered before playback deadlines. This mismatch triggers retransmissions for TBs belonging to frames that have already been recovered by FEC or whose deadlines have elapsed, wasting radio resources and reducing the time budget available for FEC [21].

Third, the TB-loss pattern determines how redundancy should be placed, and conversely, frame recovery status determines when HARQ should stop. Yet existing designs optimize FEC [4]–[6] and HARQ [8]–[10] in isolation, without exploiting this bidirectional dependency.

To address these issues, we propose a two-way cross-layer mechanism: (i) R2AX-FEC (§ IV), which infers TB-level loss structure from UE feedback and jointly determines pacing, inter-frame spreading, and redundancy; and (ii) A2RX-HARQ (§ V), which feeds back recovery status to suppress redundant retransmissions and effectively expand the time budget for R2AX-FEC. These mechanisms are feasible because the UE can expose radio-layer observations to the application layer [22], [23] and relay application decisions to the radio layer via cross-layer interfaces viable in 3GPP systems [13].

IV. Radio-to-Application Cross-Layer FEC

This section presents R2AX-FEC. We formulate per-epoch FEC parameter selection (pacing, spreading span, redundancy) as MINLP, and solve it sequentially with radio-layer feedback.

A. Overview of UE-Feedback-Assisted FEC

We target RTP-based real-time video communication employing Transport-Wide Congestion Control (TWCC) feedback [24]. TWCC reports UE-side reception timing to the server, and enables rate adaptation at the application layer.

As shown in Fig. 2, R2AX-FEC augments TWCC with TB (TB size and TB delivery time) and HARQ (RV and ACK/NACK) feedback. TB size allows the sender to estimate how many RTP packets are multiplexed within a TB. TB delivery time approximates the per-TB service time and is used to set packet pacing so that transmissions align with TB-level scheduling. HARQ feedback indicates whether link-layer errors are likely to be recovered within the latency budget and is therefore used to scale FEC redundancy.

The server updates the FEC configuration in three steps per scheduling epoch, i.e., per frame period. First, it sets the pacing interval to match the effective TB delivery cadence under the

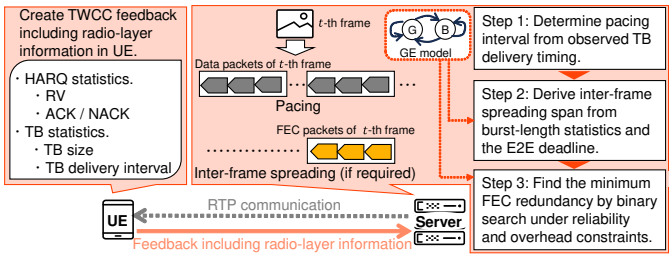


Fig. 2: Overview of R2AX-FEC mechanism.

scheduler time granularity. Second, it selects the inter-frame spreading span to decorrelate FEC packets from loss bursts within the end-to-end deadline. Third, it chooses the minimum redundancy that meets the target reliability within the coding-rate bounds. When inter-frame spreading is enabled, FEC packets from older frames are first assigned to evenly spaced slots for temporal dispersion; the current frame's data packets then fill the remaining slots. To capture bursty residual TB losses, we model the channel as a two-state Gilbert–Elliott (GE) process [11] and estimate its parameters online via Bayesian updating with a forgetting factor [12].

B. System Model

1) Packet and TB Model: We consider a downlink video stream with frame rate V fps and frame period $T_{\mathrm{fr}} = 1/V$. Let R bps denote the transmission rate budget, including both data and FEC packets. With fixed packet size L bytes, the number of outgoing packets in one frame period is

$$N_{\mathrm{total}} = \left\lceil \frac{R \cdot T_{\mathrm{fr}}}{8 \cdot L} \right\rceil . \tag{1}$$

Each frame is independently encoded with an MDS code. Let K_t denote the number of FEC packets allocated to protect frame t, with $0 \le K_t \le N_{\mathrm{total}}$. Under the uniform steady-state allocation assumed throughout (K_t identical across all t), these K_t FEC packets are evenly dispersed into the transmission slots of the subsequent $(F-1)$ frames within an F-frame spreading window. Let $K = FK_t$ be the total FEC count in the window. Accordingly, the number of data packets of frame t is $N_{\mathrm{total}} - K_t$. Over the window, a total of $N_F = FN_{\mathrm{total}}$ packets are transmitted, among which K are FEC packets and $N_F - K$ are data packets. The redundancy ratio (fraction of FEC packets) satisfies

$$\rho_{\min} \le \rho = \frac{K}{N_F} \le \rho_{\max}. \tag{2}$$

At the RAN, packets are scheduled into TBs with effective size Z bytes ($Z \ge L$). We adopt a no-splitting approximation: a packet is placed entirely into one TB (otherwise deferred to a later TB). This assumption is consistent with prior studies that map each packet to the smallest TB size capable of carrying the packet plus fixed overhead bits prior to radio scheduling [25]. Then each TB is approximately modeled to carry

$$m_{\mathrm{TB}} = \max\left(1, \left\lfloor \frac{Z}{L} \right\rfloor\right) \tag{3}$$

packets, and the N_{total} packets of a frame are mapped onto

$$G_{\mathrm{TB}} = \left\lceil \frac{N_{\mathrm{total}}}{m_{\mathrm{TB}}} \right\rceil \tag{4}$$

TBs. Thus, if a TB fails to be decoded by the deadline, all packets carried in that TB are lost.

2) HARQ Process Model: Each failed TB is retransmitted by HARQ with up to four attempts (RV_0–RV_3) and round-trip time T_{HARQ}. Let $q_i = \Pr(\mathrm{NACK} \mid \mathrm{RV}=i, \mathrm{reached})$ denote the conditional NACK probability at RV i given that RV i is reached. A TB is regarded as lost if decoding is not completed within the radio-channel deadline budget D_{link}. The number of feasible HARQ attempts within D_{link} is

$$m_D = \min\left(4,\ 1 + \left\lfloor \frac{D_{\mathrm{link}}}{T_{\mathrm{HARQ}}} \right\rfloor\right), \tag{5}$$

yielding the deadline-aware residual TB loss probability

$$p_{\mathrm{TB}}(D_{\mathrm{link}}) = \prod_{i=0}^{m_D-1} q_i. \tag{6}$$

3) Gilbert–Elliott Channel Model: Due to the RAN dynamics, residual TB losses are temporally correlated; we model them over pacing opportunities spaced by τ (aligned with TB scheduling) using a two-state GE hidden Markov model (HMM) with $S_n \in \{G, B\}$. Conditioned on $S_n{=}s$, post-HARQ TB loss is Bernoulli with probability $p_s(D_{\mathrm{link}})$ ($p_B{>}p_G$), yielding $p_{\mathrm{TB}}(D_{\mathrm{link}}){=}\pi_G p_G(D_{\mathrm{link}}){+}\pi_B p_B(D_{\mathrm{link}})$.

Its state transitions follow a continuous-time Markov chain with rates λ_{GB} and λ_{BG} [11]. Sampling every τ yields

$$a(\tau) = \Pr(G{\to}B) = \frac{\lambda_{GB}}{\lambda_{GB}+\lambda_{BG}}\left(1 - e^{-(\lambda_{GB}+\lambda_{BG})\tau}\right), \tag{7}$$

$$b(\tau) = \Pr(B{\to}G) = \frac{\lambda_{BG}}{\lambda_{GB}+\lambda_{BG}}\left(1 - e^{-(\lambda_{GB}+\lambda_{BG})\tau}\right). \tag{8}$$

For small τ, $a(\tau){\approx}\lambda_{GB}\tau$ and $b(\tau){\approx}\lambda_{BG}\tau$, making the reliability constraint nonlinear in τ through $a(\tau)$ and $b(\tau)$.

The parameters $\boldsymbol{\theta} = (\lambda_{GB}, \lambda_{BG}, p_G, p_B)$ are updated online via Bayesian power-prior updating [12] with conjugate Gamma priors on $(\lambda_{GB}, \lambda_{BG})$ and Beta priors on (p_G, p_B); latent states are inferred by HMM filtering/smoothing, enforcing $p_B{>}p_G$ to prevent label switching [26].

C. Problem Formulation

We determine the pacing interval τ, the inter-frame spreading span F (in frames), and the FEC redundancy K for each scheduling epoch T_{fr}. A pacing opportunity corresponds to one scheduled TB transmission at the RAN, and the TB-level transmission timeline is modeled as a sequence of such opportunities spaced by τ. The minimum feasible spacing is bounded by the RAN scheduling granularity Δ_{sched}.

The number of pacing opportunities per frame is

$$S(\tau) = \left\lceil \frac{T_{\mathrm{fr}}}{\tau} \right\rceil, \tag{9}$$

so the total number of opportunities over an F-frame window is $S_{\mathrm{tot}}{=}F\cdot S(\tau)$. We focus on the backlogged regime; when inter-frame spreading is active ($F{>}1$), up to $F{-}1$ opportunities are reserved for FEC packets from other frames, and the remaining opportunities carry the frame's data packets.

Recall $N_F{=}F\cdot N_{\mathrm{total}}$ and $\rho{=}K/N_F$. We solve

$$\min_{K,\,\tau,\,F} \quad \rho = \frac{K}{N_F} \tag{10}$$

$$\text{s.t.} \quad \tau \geq \Delta_{\mathrm{sched}} \tag{C1}$$

Algorithm 1: R2AX-FEC Parameter Selection

Input: $\hat{\Delta}_{\mathrm{TB}}$, $\hat{Z}$, D_{guard}, $\boldsymbol{\theta}$
Output: τ, F, K_t

/* Step 1: Fix pacing interval */
$m_{\mathrm{TB}} \leftarrow \max(1, \lfloor \hat{Z}/L \rfloor)$; $G_{\mathrm{TB}} \leftarrow \lceil N_{\mathrm{total}}/m_{\mathrm{TB}} \rceil$;
$\tau \leftarrow \max\big(\Delta_{\mathrm{sched}},\ \hat{\Delta}_{\mathrm{TB}}\big)$;
$S(\tau) \leftarrow \lceil T_{\mathrm{fr}}/\tau \rceil$;
if $G_{\mathrm{TB}} > S(\tau)$ **then return** $(\tau,\ 1,\ 0)$;
/* Step 2: Determine spreading span */
$b \leftarrow b(\tau)$ via Eq. (8);
$l_{\mathrm{burst}} \leftarrow \big\lceil \log(1{-}\xi_{\mathrm{burst}}) / \log(1{-}b) \big\rceil$;
$F_{\mathrm{burst}} \leftarrow 1 + \lceil l_{\mathrm{burst}} \cdot \tau \,/\, T_{\mathrm{fr}} \rceil$;
$F_{\mathrm{ddl}} \leftarrow 1 + \lfloor (D_{\max} - D_{\mathrm{guard}} - (S(\tau)-1)\tau) \,/\, T_{\mathrm{fr}} \rfloor$;
if $F_{\mathrm{ddl}} < 1$ **then return** $(\tau,\ 1,\ 0)$;
$F \leftarrow \min(F_{\mathrm{burst}},\ F_{\mathrm{ddl}})$;
/* Step 3: Minimize FEC redundancy */
$K_{t,\min} \leftarrow \lceil \rho_{\min} \cdot N_{\mathrm{total}} \rceil$; $K_{t,\max} \leftarrow \lfloor \rho_{\max} \cdot N_{\mathrm{total}} \rfloor$;
$K_t^* \leftarrow K_{t,\max}$;
binary-search min. $K_t \in [K_{t,\min}, K_{t,\max}]$ s.t. $\hat{\bar{P}}_{\mathrm{fail}}(K_t) \leq \varepsilon_{\mathrm{fail}}$;
if *feasible* K_t *found* **then** $K_t^* \leftarrow K_t$;
return $(\tau,\ F,\ K_t^*)$;

$$N_F \leq m_{\mathrm{TB}} \cdot S_{\mathrm{tot}},\ S_{\mathrm{tot}} = F \cdot \left\lceil \frac{T_{\mathrm{fr}}}{\tau} \right\rceil \tag{C2}$$

$$(F{-}1)\,T_{\mathrm{fr}} + (S(\tau){-}1)\,\tau + D_{\mathrm{guard}} \leq D_{\max} \tag{C3}$$

$$\bar{P}_{\mathrm{fail}}\big(K,\ F;\ \tau\big) \leq \varepsilon_{\mathrm{fail}} \tag{C4}$$

$$\rho_{\min} \leq \frac{K}{N_F} \leq \rho_{\max} \tag{C5}$$

$$K \in \mathbb{Z}_{\geq 0}, \quad F \in \mathbb{Z}_{\geq 1}, \quad \tau \in \mathbb{R}_{>0}. \tag{C6}$$

(C1)–(C2) enforce scheduling granularity and packing feasibility. (C3) enforces the end-to-end deadline: because FEC packets are spread across F frames, the last FEC packet protecting a given frame is placed in a slot up to $(F{-}1)T_{\mathrm{fr}}$ later; within that epoch, it may occupy any of the $S(\tau)$ evenly spaced pacing slots, so its worst-case position is $(S(\tau){-}1)\tau$ from the epoch start; together with the guard margin, the total must not exceed $D_{\max}$. (C4) imposes the reliability target (see § IV-D2, Appendix A); (C5)–(C6) bound redundancy and variable domains.

The guard margin in (C3) is the remaining non-radio one-way delay and the worst-case HARQ retransmission time:

$$D_{\mathrm{guard}} = \mathrm{OWD} + (m_D - 1)\,T_{\mathrm{HARQ}}, \tag{11}$$

where OWD is the non-HARQ one-way delay and $(m_D - 1)\,T_{\mathrm{HARQ}}$ is the maximum additional time when delivery occurs at the last feasible HARQ attempt.

The formulated problem is regarded as a nonconvex MINLP [27] with integer variables (K, F) and a continuous variable τ because (C4) evaluates GE/HMM-based reliability with transition probabilities $a(\tau)$ and $b(\tau)$, and (C2) introduces a ceiling term via $S_{\mathrm{tot}} = F\lceil T_{\mathrm{fr}}/\tau \rceil$. Consequently, global optimization is computationally intractable in general [27].

D. Decomposition Algorithm with Radio-layer Observations

We decompose the problem into three sequential steps in Algorithm 1 by fixing (τ, F, K_t). The resulting parameters then drive the per-epoch transmission schedule described below.

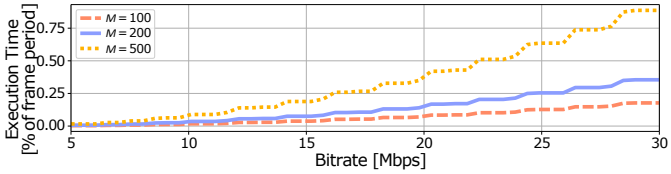


Fig. 3: Estimated execution time vs. bitrate across M in Algorithm 1.

1) Algorithm Details: **Step 1: Fix τ (lines 1–4).** The per-TB packet count m_{TB} is obtained from the posterior mean TB size $\hat{Z}$, yielding the per-frame TB count G_{TB}. The pacing interval is set to $\tau{=}\max\{\Delta_{\mathrm{sched}}, \hat{\Delta}_{\mathrm{TB}}\}$, where Δ_{sched} is the scheduling granularity (C1) and $\hat{\Delta}_{\mathrm{TB}}$ is the posterior mean TB delivery interval. Fixing τ eliminates the continuous variable and reduces the problem to an integer search over (F, K_t). If $G_{\mathrm{TB}}{>}S(\tau)$, (C2) is violated and K_t is set to 0.

Step 2: Determine F from burst analysis (lines 5–10). Under the continuous-time GE model, $b(\tau)$ (B$\to$G) governs burst duration while $a(\tau)$ (G$\to$B) governs burst frequency. Since the spreading span must outlast a single burst, this step uses only $b(\tau)$: a Bad-state run is geometric with parameter $b(\tau)$ from Eq. (8), and the algorithm computes l_{burst}, its ξ_{burst}-quantile, to obtain F_{burst}. The effect of $a(\tau)$ on burst frequency is captured in Step 3 through the full GE-based reliability evaluation. Meanwhile, F_{ddl} is the largest span that satisfies the end-to-end deadline (C3). The final span is $F{=}\min(F_{\mathrm{burst}}, F_{\mathrm{ddl}})$; if $F_{\mathrm{ddl}}{<}1$, return $F{=}1$, $K_t{=}0$. If the feasibility checks in Steps 1 or 2 fail, the algorithm falls back to a no-FEC mode outside the constrained problem.

Step 3: Find the minimum feasible K_t (lines 11–15). With (τ, F) fixed and uniform allocation $K = F \cdot K_t$, the algorithm seeks the minimum per-frame FEC count K_t within the redundancy bounds $\rho_{\min} \le K_t/N_{\mathrm{total}} \le \rho_{\max}$ (equivalent to (C5)) such that the tail-averaged failure probability $\hat{\bar{P}}_{\mathrm{fail}}(K_t)$ (§ IV-D2) meets the reliability target (C4). Each candidate K_t determines the per-frame loss-tolerance threshold $Y_t = \lfloor K_t/m_{\mathrm{TB}} \rfloor$; frame failure is decomposed into correlated data-TB losses (GE Markov–binomial DP) and dispersed FEC-TB losses (binomial), and the window failure probability is upper-bounded by $F{\cdot}P_{\mathrm{fail}}^{\mathrm{frame}}$ via the union bound (Appendix A). Because $\hat{\bar{P}}_{\mathrm{fail}}(K_t)$ is non-increasing in K_t, K_t^* is found by binary search. If even $K_{t,\max} = \lfloor \rho_{\max} N_{\mathrm{total}} \rfloor$ is infeasible, the algorithm sets $K_t^* {=} K_{t,\max}$.

Complexity. Steps 1–2 are $O(1)$; Step 3 runs $O(\log(K_{t,\max} {-} K_{t,\min}{+}1))$ binary-search iterations. Each iteration evaluates $\hat{\bar{P}}_{\mathrm{fail}}(K_t)$ from M posterior samples, each requiring a per-frame Markov–binomial DP over G_{d} data TBs with threshold Y_t (Appendix A), yielding the per-epoch complexity

$$O(M \cdot G_{\mathrm{d}} \cdot Y_t \cdot \log(K_{t,\max} - K_{t,\min} + 1)). \qquad (12)$$

Figure 3 shows a baseline execution time $T_{\mathrm{base}} = N_{\mathrm{flop}} / P_{\mathrm{scalar}}$ following the roofline model [28], where N_{flop} counts 14 FLOPs per DP cell update and $P_{\mathrm{scalar}} {=} 12$ GFLOPS is the scalar per-core throughput [29], using the same parameters as in § VI-A. Because the per-frame DP operates on G_{d} TBs of a single frame, T_{base} is independent of the spreading span F and scales linearly with M. Even with $M = 500$ posterior samples, the ratio $T_{\mathrm{base}}/T_{\mathrm{fr}}$ stays below 1% over 5–30 Mbps at 60 fps, confirming the feasibility of per-frame invocation.

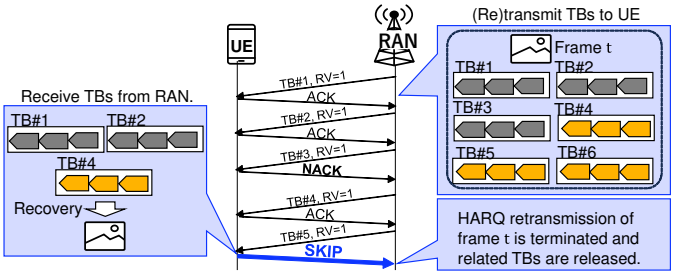


Fig. 4: Overview of A2RX-HARQ mechanism.

Transmission policy. With (τ, F, K_t) determined, packets are transmitted at pacing interval τ. When inter-frame spreading is enabled ($F{>}1$), FEC packets from older frames are first assigned to evenly spaced pacing slots for temporal dispersion; the current frame's data packets then fill the remaining slots.

2) Robust Bayesian Reliability Constraint: To robustly account for posterior uncertainty, we evaluate the reliability constraint via a tail-averaged failure probability—the mean of $P_{\mathrm{fail}}(\boldsymbol{\theta})$ over its pessimistic upper-tail posterior realizations $\mathcal{T}$, analogous to conditional value at risk [30]:

$$\hat{\bar{P}}_{\mathrm{fail}}(K_t) = \frac{1}{|\mathcal{T}|} \sum_{s \in \mathcal{T}} P_{\mathrm{fail}}^{(s)}(K_t), \qquad (13)$$

where each $P_{\mathrm{fail}}^{(s)}(K_t)$ is evaluated via the DP in Appendix A for the s-th posterior sample $\boldsymbol{\theta}^{(s)}$.

V. Application-to-Radio Cross-Layer HARQ

This section introduces A2RX-HARQ, which enables UE-initiated early termination of HARQ retransmissions at the application layer. This indirectly relaxes the deadline constraint imposed on R2AX-FEC. We also discuss its implementation compatibility within the 3GPP architecture.

A. Ternary Feedback and Decision Logic

As illustrated in Fig. 4, A2RX-HARQ extends the HARQ feedback decision from the binary set $\{\mathrm{ACK}, \mathrm{NACK}\}$ to the ternary set $\{\mathrm{ACK}, \mathrm{NACK}, \mathrm{SKIP}\}$, allowing UE-initiated early termination when additional retransmissions are redundant or useless from the application-layer perspective. The UE may issue SKIP on a NACK candidate when either of the following triggering conditions holds:

- **Recovery Complete**: The UE has already recovered the entire frame to which the unsuccessfully decoded TB payload belongs. Since each frame is independently MDS-coded, frame t becomes decodable once any $N_{\mathrm{total}} - K_t$ out of its N_{total} packets are obtained, where K_t denotes the number of FEC packets allocated to frame t. After recovery, retransmission of the remaining packets of that frame is unnecessary.
- **Deadline Expired**: The feedback time exceeds the deadline $t_f{+}D_{\max}$, where t_f is the frame generation time. Packets arriving after $D_{\max}$ provide no utility for live rendering.

Algorithm 2: Channel-Aware UE-side SKIP Gating

Input: $\mathcal{H}_W$, τ_{ch}
Output: UE feedback $\in \{\mathrm{SKIP}, \mathrm{NACK}\}$
decide $\leftarrow$ RECOVERYCOMPLETE$(\cdot) \vee$ DEADLINEEXPIRED$(\cdot)$;
if $\neg$ *decide* **then return** NACK;
$\hat{p}_{\mathrm{fail}} \leftarrow \frac{1}{|\mathcal{H}_W|} \sum_{c \in \mathcal{H}_W} \mathbb{1}[c = \mathrm{corrupt}]$;
if $\hat{p}_{\mathrm{fail}} \geq \tau_{\mathrm{ch}}$ **then return** NACK;
return SKIP;

A SKIP candidate is actually emitted only when neither of the following suppression rules overrides it:

- **Multi-frame TB rule**: R2AX-FEC pacing places each frame's packets into separate TBs whenever feasible. If a TB contains packets from multiple frames, SKIP is emitted only when all those frames are unnecessary; otherwise, it is withheld to avoid stopping retransmissions still needed by other frames.
- **Channel-aware rule**: The freed process is often reassigned to a new TB, whose initial transmission is also likely to fail, causing further HARQ retransmissions and consuming the resources that SKIP was meant to save. Therefore, the UE tracks corruption indicators over a sliding window $\mathcal{H}_W$ of the last W TB receptions and uses the empirical failure rate $\hat{p}_{\mathrm{fail}}$ as a coarse measure of channel degradation. If $\hat{p}_{\mathrm{fail}} \geq \tau_{\mathrm{ch}}$, SKIP is deferred and NACK is sent instead, keeping the process active until the channel recovers (see Algorithm 2).

B. Deadline Constraint Relaxation for R2AX-FEC

In the baseline model of Eq. (11), the margin is set for the worst case where the last TB succeeds only on the final HARQ attempt. With MDS coding, however, a frame is decodable once any $N_{\mathrm{total}} - K_t$ of its N_{total} packets are received, so the completion time is determined by the $(N_{\mathrm{total}} - K_t)$-th earliest delivery rather than the worst case. Let $T_i = \mathrm{OWD} + (H_i - 1)T_{\mathrm{HARQ}}$ denote the delivery time of packet i, where $H_i \in \{1, \ldots, m_D\}$ is its HARQ attempt count ($H_i = \infty$ if lost). The per-frame FEC recovery time is the $(N_{\mathrm{total}} - K_t)$-th order statistic

$$T_{\mathrm{FEC}} = T_{(N_{\mathrm{total}} - K_t)} = \min_{\substack{\mathcal{S} \subseteq \{1, \ldots, N_{\mathrm{total}}\} \\ |\mathcal{S}| = N_{\mathrm{total}} - K_t}} \max_{i \in \mathcal{S}} T_i. \tag{14}$$

Accordingly, the guard margin can be tightened to

$$D'_{\mathrm{guard}} = \mathrm{OWD} + (H_{(N_{\mathrm{total}} - K_t)} - 1)\, T_{\mathrm{HARQ}}, \tag{15}$$

where $H_{(N_{\mathrm{total}} - K_t)}$ is the HARQ attempt count of the packet delivered at $T_{(N_{\mathrm{total}} - K_t)}$. Assuming, for tractability, an i.i.d. first-attempt success probability p_{ack},

$$\mathbb{E}[H_{(N_{\mathrm{total}} - K_t)}] \lesssim 1 + \frac{(1 - p_{\mathrm{ack}})(K_t + 1)}{N_{\mathrm{total}} \cdot p_{\mathrm{ack}}}, \tag{16}$$

yielding, e.g., $\mathbb{E}[H_{(N_{\mathrm{total}} - K_t)}] \approx 1.04$ for $p_{\mathrm{ack}} \approx 0.9$ and $K_t/(N_{\mathrm{total}} - K_t) \approx 0.3$ in the typical case observed in § VI-D. The expected deadline reduction is

$$\Delta_D = D_{\mathrm{guard}} - D'_{\mathrm{guard}} \approx \left(m_D - \mathbb{E}[H_{(N_{\mathrm{total}} - K_t)}]\right) T_{\mathrm{HARQ}}, \tag{17}$$

which relaxes (C3) and increases the maximum admissible inter-frame spreading span by $\Delta_F = \lfloor \Delta_D / T_{\mathrm{fr}} \rfloor$ frames. In practice, K_t in (15) is taken from the previous epoch to avoid circularity with Step 3 in Algorithm 1.

However, exploiting the tighter margin requires that HARQ processes still retransmitting TBs of the recovered frame are released promptly. Without A2RX-HARQ, these processes continue retransmitting until m_D attempts, competing with new TBs, including the additional FEC packets enabled by the expanded F, for HARQ processes and radio resources. A2RX-HARQ resolves this contention by terminating these processes at T_{FEC} via SKIP, making the relaxed margin D'_{guard} practically achievable.

While A2RX-HARQ does not modify the form of (C4), $\bar{P}_{\mathrm{fail}}(K, F; \tau) \leq \varepsilon_{\mathrm{fail}}$, the expanded feasible range of F provides greater temporal dispersion of FEC packets across frames, enabling the same $\varepsilon_{\mathrm{fail}}$ to be met with smaller redundancy K_t and hence lower FEC overhead.

C. Discussion on Implementation Feasibility

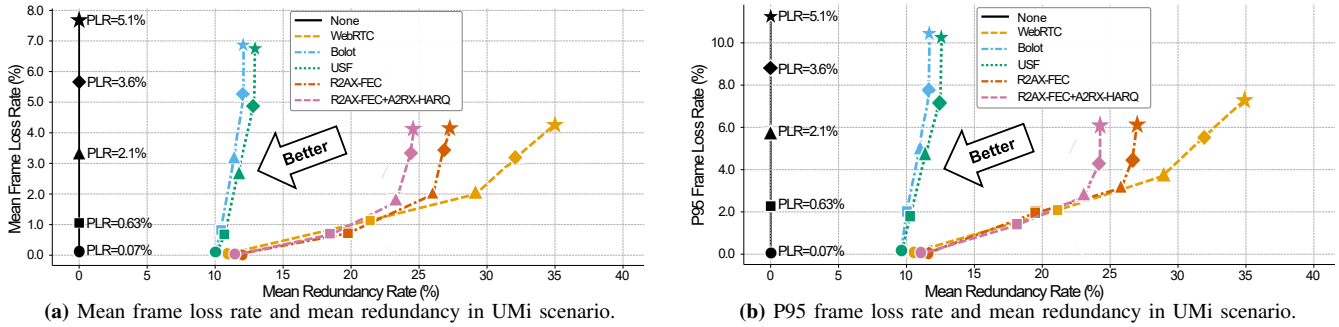
The XR-related 3GPP specification [31] provides per-frame metadata, including identifiers and boundary indicators, which the RAN uses to associate HARQ processes with frames. When SKIP is received for a TB, the RAN uses this mapping to release the relevant HARQ processes. The UE also relies on the same metadata to determine whether a TB contains packets from multiple frames when applying the multi-frame TB rule in § V-A.

SKIP has the same loss semantics as NACK: the RAN treats the TB as not received, ensuring that link adaptation observes accurate error statistics. If SKIP were encoded as ACK, it would inflate success counts, drive link adaptation toward overly aggressive settings, and increase the actual TB error rate in later transmissions. Unlike NACK, SKIP also signals "do not retransmit," which the HARQ scheduler must honor while still counting the TB as lost. Under the existing 3GPP HARQ feedback framework [2], this is achieved by sending NACK on the feedback channel and conveying retransmission cancellation through the cross-layer interface described below. This separation may cause one redundant retransmission before cancellation takes effect. Defining SKIP as a distinct feedback value would combine both signals into one message and remove this overhead.

The cross-layer interface between the UE radio and application layers enables the execution of Algorithm 2. The radio layer provides per-TB corruption indicators for $\mathcal{H}_W$ and HARQ feedback context to the application layer, which evaluates the SKIP conditions and suppression rules in § V-A. The resulting decision is then passed back to the radio layer for inclusion in the next HARQ feedback occasion. Although this interface is implementation-specific and outside the scope of 3GPP standardization, prior work [13] has demonstrated application–radio signaling in a 3GPP-compliant system.

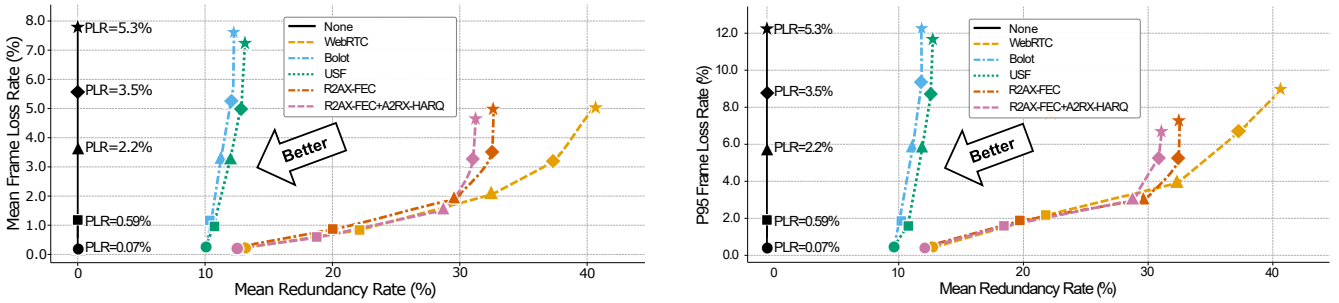
VI. EVALUATION

This section evaluates R2AX-FEC and A2RX-HARQ (with SKIP feedback) using ns-3 simulations with the 5G-LENA



(a) Mean frame loss rate and mean redundancy in UMi scenario. **(b)** P95 frame loss rate and mean redundancy in UMi scenario.

Fig. 5: Performance comparison illustrating the trade-off between the redundancy ratio and the frame loss rate in the UMi scenario. The orange and pink plots represent R2AX-FEC and R2AX-FEC combined with A2RX-HARQ, respectively. PLR labels on the None curve denote the mean packet loss rate.



(a) Mean frame loss rate and mean redundancy in UMa scenario. **(b)** P95 frame loss rate and mean redundancy in UMa scenario.

Fig. 6: Performance comparison illustrating the trade-off between the redundancy ratio and the frame loss rate in the UMa scenario. The orange and pink plots represent R2AX-FEC and R2AX-FEC combined with A2RX-HARQ, respectively. PLR labels on the None curve denote the mean packet loss rate.

TABLE I: Key simulation parameters

Item	Value
Topology	1 gNB; 1 UE with a random walk
Runs / Seeds	30 runs with different seed values
Duration	120 s
Carrier / Bandwidth	3.5 GHz, 100 MHz
Numerology	μ=0 (15 kHz)
Beamforming	Quasi-omni direct path
Base station antenna	4×8
UE antenna	2×4
Tx power	gNB 42 dBm; UE 30 dBm
Error model	NR-EESM-IR (Table 1) [32]
Radio-layer retrans.	DL: HARQ only, UL: HARQ & ARQ
Backhaul delay	10 ms one-way delay

module [14]. We also assess the contribution of each component in Algorithms 1 and 2, and discuss their limitations. Our key findings are:

- Both mechanisms significantly reduce the frame loss rate, including deadline violations, compared with conventional FEC schemes such as Bolot [16] and USF [17], by adapting the FEC configuration to radio conditions.
- Both mechanisms achieve frame recovery with substantially lower redundancy, while matching the performance of the high-redundancy configuration in WebRTC [18].

A. Evaluation Setup

We use cloud gaming as a representative RTP application, as it imposes strict requirements on both latency and throughput. The target configuration is a downlink rate of $R = 20$ Mbps, a frame rate of $V = 60$ fps, and an end-to-end deadline of $D_{\max} = 100$ ms [33]. The server transmits RTP packets of $L = 1400$ bytes and applies FEC with redundancy-ratio bounds of $\rho_{\min} = 0.1$ and $\rho_{\max} = 0.5$. We set the reliability target to $\varepsilon_{\text{fail}} = 0.1$, the burst quantile to $\xi_{\text{burst}} = 0.99$ to exclude only extreme tail bursts, and the number of Monte Carlo posterior samples in Algorithm 1 to $M = 200$. The channel-quality threshold used in Algorithm 2 is set to $\tau_{\text{ch}} = 0.25$.

Table I summarizes the RAN parameters. We simulate a single UE connected to a single base station. This setting represents a preferentially managed UE, for example, through network slicing, and allows us to isolate the effects of channel impairments and redundancy adaptation. The UE follows a random-walk mobility model within an annular region centered at the base station, moving at a constant speed of 3 m/s with a pause time of 1 s between direction changes. For the low-loss operating point, where the PLR is 0.63%, the inner and outer radii are set to 80 m and 120 m, respectively. For the middle-loss operating point, where the PLR is 2.1%, they are set to 130 m and 170 m, respectively.

At the radio layer, HARQ is used as the downlink retransmission mechanism, while uplink automatic retransmission request (ARQ) is enabled to ensure reliable feedback delivery. We evaluate two channel scenarios, Urban Micro (UMi) and Urban Macro (UMa) [14]. Degraded radio conditions are modeled using a wireless error model in which the packet loss rate (PLR) increases with the distance between the UE and the base station. More specifically, downlink transport blocks are decoded using the 5G radio link-to-system error model with

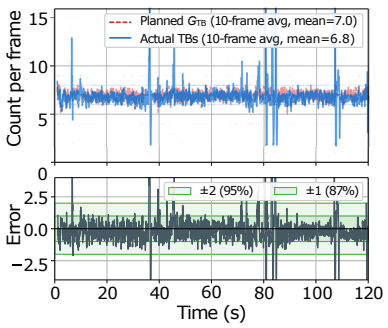


Fig. 7: Packet separation into TB group by packet pacing.

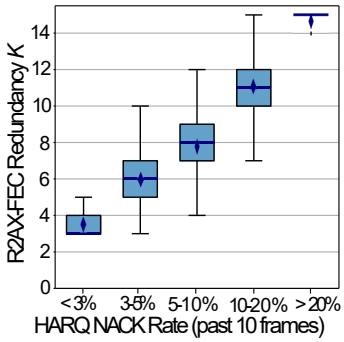


Fig. 8: FEC redundancy across recent HARQ NACK rates.

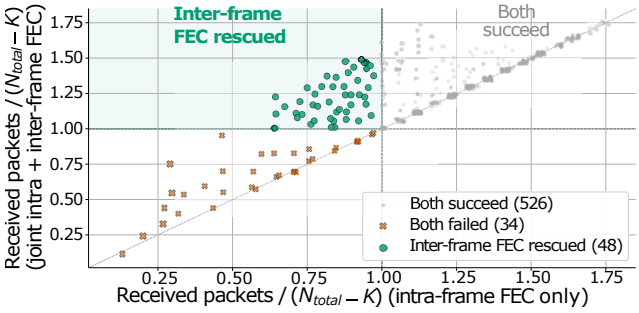


Fig. 9: Counterfactual analysis of inter-frame spreading effectiveness targeting the frames requiring recovery.

incremental-redundancy combining [32]. LOS/NLOS transitions follow a two-state Markov channel-condition model [34], with mean LOS and NLOS durations of 2.0 s and 0.05–0.5 s, respectively. This setting produces realistic burst-loss patterns that the proposed FEC adaptation scheme must handle.

B. Trade-off between Frame Recovery and Redundancy

Figures 5 and 6 show the mean and P95 frame loss rates (FLRs), including deadline violations, against the average FEC redundancy ratio in the UMi and UMa scenarios. The None policy disables FEC. For each PLR setting, all policies are evaluated at the same UE–base-station distance, calibrated from the PLR observed under None.

Bolot and USF maintain nearly constant redundancy (10–13%) regardless of channel conditions, so their FLR remains close to None in both scenarios. In UMi at PLR $= 5.1\%$, both reduce the mean FLR by less than one percentage point from None (7.7%), and their P95 FLR stays near 10%; the same pattern holds in UMa. These results confirm that loss-rate-only adaptation cannot track burst losses effectively.

In contrast, WebRTC and R2AX-FEC scale redundancy with PLR, but R2AX-FEC achieves comparable or better recovery with less overhead. In UMi at PLR $= 5.1\%$, R2AX-FEC matches WebRTC's mean FLR (4.1%) using roughly 8 percentage points less redundancy (27% vs. 35%), and its P95 FLR (6.1%) undercuts WebRTC's (7.3%). In UMa at PLR $= 5.3\%$, R2AX-FEC reduces WebRTC's P95 FLR from 9% to 7.3% while cutting redundancy from 41% to 32%, confirming that channel- and deadline-aware allocation handles burst-induced tail losses more efficiently.

Adding A2RX-HARQ further shifts the trade-off in both scenarios. In UMi, it reduces redundancy from 27% to 24.5% at PLR $= 5.1\%$ while keeping the P95 FLR near 6.1%, and achieves a mean FLR of 0.7% with 17% redundancy at PLR $= 0.63\%$ (versus 21% for WebRTC). In UMa at PLR $= 5.3\%$, the combination achieves the lowest P95 FLR among all schemes (6.65%), roughly halving None's 12.2%, while using far less redundancy than WebRTC. These gains reflect the SKIP mechanism's ability to reclaim HARQ deadline budget for inter-frame spreading.

Note that, since Algorithm 1 returns $K_t = 0$ when pacing is infeasible, or the remaining deadline margin is insufficient, the reliability target $\varepsilon_{\text{fail}}$ should be interpreted as a design target under feasible transmission conditions.

C. Effectiveness of Proposed Algorithm Steps

We assess the impact of Steps 1–3 in Algorithm 1 in UMi scenario for R2AX-FEC and A2RX-HARQ at PLR $= 2.1\%$.

1) TB Separation by Pacing: Figure 7 shows the time-series error between the planned number of TB groups G_{TB} and the observed TB count per frame over a 120-second simulation. Both signals are smoothed using a 10-frame centered moving average. The shaded bands indicate that 87% of the averaged error lies within ± 1 and 95% within ± 2. Overall, the proposed algorithm produces TB groups that closely track the planned G_{TB}.

The remaining deviation is mainly due to scheduler-dependent merging/splitting of TBs under time-varying channel and resource conditions, and per-frame updates of G_{TB} versus slot-level TB allocation. Nevertheless, the strong concentration around zero indicates that the packet-level pacing plan is largely preserved at the radio layer, supporting the feasibility of the proposed TB group partitioning.

2) FEC Redundancy across HARQ NACK rates: Figure 8 presents the distribution of the selected FEC redundancy K_t with respect to the recent HARQ NACK rate, defined as the 10-frame rolling mean of the per-frame TB failure rate observed at the FEC-decision time. The samples are divided into five NACK-rate bins: $< 3\%$, 3–5%, 5–10%, 10–20%, and $> 20\%$.

These results demonstrate that K_t increases monotonically with the recent NACK rate across the full dynamic range of the algorithm, $K_t \in [3, 15]$. In the low-NACK regime, K_t remains at its lower bound, yielding a redundancy ratio of 10% and avoiding over-provisioning under stable channel conditions. As the NACK rate increases, K_t grows approximately linearly until the $\rho_{\max}$ ceiling becomes active for NACK rates above 20%. The widening interquartile range observed in the moderate-NACK bins, i.e., 3–20%, reflects variations in burst severity and the confidence-aware margin within each bin.

3) Inter-Frame Spreading: Figure 9 shows the contribution of inter-frame spreading through a counterfactual analysis of the 608 frames requiring packet recovery. For each frame, we compare the recovery outcome with and without inter-frame FEC packets. Inter-frame spreading improves the recovery rate

TABLE II: A2RX-HARQ outcome breakdown per HARQ process.

Outcome	Percentage
Recovery Complete	5.6%
Deadline Expired	94.4%

TABLE III: HARQ RV at which SKIP feedback is issued.

RV version	Percentage
RV0	58.2%
RV1	40.8%
RV2	1.0%

from 86.5% to 94.4%, thereby reducing residual frame loss. The rescued frames are concentrated at intra-only recovery ratios between 0.55 and 0.95, indicating moderate-to-severe within-frame losses that are compensated for by FEC packets arriving in later frames. Overall, inter-frame FEC spreading provides temporal diversity that complements intra-frame redundancy and improves robustness against burst losses.

D. Analysis of A2RX-HARQ Effectiveness

Tables II and III show how the SKIP feedback in A2RX-HARQ terminates HARQ retransmissions, based on 30 UMi runs at PLR $= 5.1\%$. As shown in Table II, most terminations are deadline-driven: 94.4% occur because the deadline of the corresponding frame has already expired. SKIP feedback is also issued early in the HARQ process, with 58.2% of terminations at RV0, 40.8% at RV1, and only 1.0% at RV2, as shown in Table III.

A2RX-HARQ can be viewed as a UE-triggered active queue management mechanism for the HARQ buffer. Rather than allowing retransmissions to continue after they can no longer meet the deadline, the UE explicitly signals the network to stop them, in the same spirit as CoDel [35]. Given the tight end-to-end constraint of $D_{\max} = 100$ ms, once a TB has consumed too much of the deadline budget, further HARQ retransmissions are guaranteed to arrive too late. The channel-aware predicate in Algorithm 2 therefore terminates such attempts before they waste additional radio resources.

The fact that 99.0% of SKIP events occur at RV0 or RV1 shows that the threshold $\tau_{\mathrm{ch}} = 0.25$ identifies unpromising retransmissions early. This frees both radio resources and deadline budget, allowing R2AX-FEC to spread FEC packets more effectively across subsequent frames. By reducing the HARQ-induced guard margin in Eq. (11), A2RX-HARQ increases the feasible inter-frame spreading span F, which helps later FEC packets arrive before $D_{\max}$. This mechanism explains the P95 FLR gains observed in Figs. 5 and 6.

E. Discussion

Rate–redundancy coupling. Our evaluation fixes the sending rate at 20 Mbps and optimizes only (τ, F, K_t). In practice, congestion control adjusts R; for example, Google Congestion Control (GCC) [36] adapts $R \approx$ 10–30 Mbps in response to feedback and reduces frame loss, as shown in Table IV. Moreover, existing congestion-control algorithms may overthrottle R and still fail to act proactively before losses occur; joint optimization of R and ρ is a promising direction to further improve the redundancy–recovery trade-off.

TABLE IV: Comparison of UMi scenario in Fig. 5: without congestion control (w/o CC) vs. GCC.

	Frame Loss (%)		Throughput (Mbps)	
PLR (%)	w/o CC	GCC	w/o CC	GCC
0.07	0.05	0.04	20.0	22.1
0.63	0.70	0.63	20.0	16.4
2.1	1.70	1.25	20.0	13.2
3.6	3.30	2.38	20.0	11.7
5.1	4.10	3.15	20.0	10.1

Non-dedicated scheduling. The TB separation by pacing assumes a dedicated-resource slice, as provisioned by a network slice with resource isolation [15]. In non-dedicated deployments, e.g., multi-UE scheduling, contention can increase variability in TB delivery timing, degrading the accuracy of the pacing interval τ estimated in Step 1 and, in turn, the GE-based reliability evaluation. Since such scheduling dynamics are hard to capture in closed form, reinforcement-learning-based approaches that learn a mapping from radio-layer observations to near-optimal FEC decisions can be a promising extension.

Frame-importance awareness. Our design uses a uniform reliability target $\varepsilon_{\mathrm{fail}}$ for all frames, although frame importance varies (e.g., losing an I-frame propagates errors to subsequent P/B-frames). Adapting $\varepsilon_{\mathrm{fail}}$ and the packet budget, including data packets, by frame type is a natural extension within our formulation to improve the redundancy–quality trade-off.

VII. Conclusion

We identified three challenges in coordinating application-layer FEC with radio-layer HARQ—loss structure mismatch, objective mismatch, and uncoordinated control—and proposed R2AX-FEC and A2RX-HARQ, a two-way cross-layer mechanism that jointly addresses them. R2AX-FEC determines pacing, inter-frame spreading, and FEC redundancy via a decomposition guided by Bayesian-estimated Gilbert–Elliott channel parameters based on UE radio-layer feedback. A2RX-HARQ enables UE-initiated early HARQ termination, relaxing deadline constraints and expanding the feasible FEC parameter space. ns-3 simulations show that the mechanism reduces frame loss relative to Bolot and USF while matching WebRTC recovery with less redundancy.

Ongoing work extends the mechanism to non-dedicated scheduling with shared radio resources among multiple UEs.

Acknowledgment

This work was partly supported by NICT, grant number 09101 “R&D for 6G Mobile System Optimization with Cross-Layer/Multi-Domain AI Integration”, Social Cooperation Program with KYOCERA, and JST ASPIRE Grant Number JPMJAP2323, Japan.

Appendix

Each frame is independently MDS encoded with a per-frame allocation of K_t FEC packets ($K = F \cdot K_t$). The per-frame loss-tolerance threshold in TB units is $Y_t = \lfloor K_t / m_{\mathrm{TB}} \rfloor$. A frame

fails if the number of lost TBs carrying its data and FEC packets exceeds Y_t.

Within each frame, $G_{\mathrm{d}}{=}\lceil (N_{\mathrm{total}} - K_t)/m_{\mathrm{TB}} \rceil$ TBs carry data packets (data TBs). We model these G_{d} TBs as consecutive under the GE process; this conservatively neglects the few interleaved FEC slots, overestimating loss correlation. Their loss count can then be evaluated via a Markov–binomial DP:

$$\Pr(d_t = d) = f_{G_{\mathrm{d}}}(G, d) + f_{G_{\mathrm{d}}}(B, d), \tag{18}$$

where $f_n(s, k)$ denotes the probability that after n transmissions the GE state is $s{\in}\{G, B\}$ and exactly k losses have occurred; the recursion follows the standard Markov–binomial forward algorithm [37].

The $\lceil K_t/m_{\mathrm{TB}} \rceil$ FEC TBs are spread over transmission slots of other frames within the spreading window. Because these slots are separated by epoch T_{fr}, their losses are approximately independent under the GE stationary distribution, yielding

$$e_t \sim \mathrm{Binomial}\big(\lceil K_t/m_{\mathrm{TB}} \rceil,\ \bar{p}\big), \quad \bar{p} = \pi_G p_G + \pi_B p_B. \tag{19}$$

The per-frame failure probability is then

$$P_{\mathrm{fail}}^{\mathrm{frame}} = \sum_{d=0}^{G_{\mathrm{d}}} \Pr(d_t = d) \cdot \Pr(e_t > Y_t - d), \tag{20}$$

and the window failure probability is upper-bounded by the union bound: $P_{\mathrm{fail}} \le F \cdot P_{\mathrm{fail}}^{\mathrm{frame}}$.

Since each frame tolerates up to K_t packet losses under MDS coding, the exact failure probability is non-increasing in K_t. The TB-level evaluation (20) conservatively upper-bounds this by quantizing losses at TB granularity ($Y_t{=}\lfloor K_t/m_{\mathrm{TB}} \rfloor$); hence, binary search on $\hat{\hat{P}}_{\mathrm{fail}}(K_t)$ in Step 3 yields a feasible, near-minimal K_t. The DP runs in $O(G_{\mathrm{d}} \cdot Y_t)$ per sample.

REFERENCES

[1] L. Fanari, E. Iradier, I. Bilbao, R. Cabrera, J. Montalban, P. Angueira *et al.*, "A survey on fec techniques for industrial wireless communications," *IEEE Open J. Ind. Electron. Soc.*, vol. 3, pp. 674–699, 2022.

[2] 3GPP, "NR; Medium Access Control (MAC) protocol specification," 3GPP, TS 38.321, 01 2026, version 19.1.0.

[3] J. J. Nielsen, I. Leyva-Mayorga, and P. Popovski, "Reliability and error burst length analysis of wireless multi-connectivity," in *Proc. IEEE Int. Symp. Wireless Commun. Syst. (ISWCS)*, 2019, pp. 107–111.

[4] H. Hu, S. Cheng, X. Zhang, and Z. Guo, "LightFEC: Network adaptive fec with a lightweight deep-learning approach," in *Proc. ACM MM*, 2021, pp. 3592–3600.

[5] S. Baghaee and E. Uysal, "A3 l-fec-fsfb: Age-aware application layer forward error correction with fixed sampling rate and fixed block-length," in *Proc. IEEE Signal Process. Commun. Appl. Conf. (SIU)*, 2024, pp. 1–4.

[6] K. Chen, H. Wang, S. Fang, X. Li, M. Ye, and H. J. Chao, "Rl-afec: adaptive forward error correction for real-time video communication based on reinforcement learning," in *Proc. ACM MMSys*, 2022, pp. 96–108.

[7] M. B. Imtiaz and R. Kamran, "Mitigating transmission errors: A forward error correction-based framework for enhancing objective video quality," *Sensors*, vol. 25, no. 11, p. 3503, 2025.

[8] F. Nadeem, Y. Li, B. Vucetic, and M. Shirvanimoghaddam, "Harq optimization for real-time remote estimation in wireless networked control," *arXiv preprint arXiv:2201.05838*, 2022.

[9] Z. Jiang, Y. Huang, S. Zhang, and S. Xu, "Analysis on asymptotic optimality of round-robin scheduling for minimizing age of information with harq," *IEICE Trans. Commun.*, vol. 104, no. 12, pp. 1465–1478, 2021.

[10] E. T. Ceran, D. Gündüz, and A. György, "A reinforcement learning approach to age of information in multi-user networks with harq," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 5, pp. 1412–1426, 2021.

[11] H. Al-Zubaidy, J. Liebeherr, and A. Burchard, "A (min,×) network calculus for multi-hop fading channels," in *Proc. IEEE INFOCOM*, 2013, pp. 1833–1841.

[12] J. G. Ibrahim and M.-H. Chen, "Power prior distributions for regression models," *Stat. Sci.*, pp. 46–60, 2000.

[13] A. Nakao, "In-band context signaling for cross-layer qos in software-defined local 5g," in *Proc. ACM Workshop Open Res. Infrastruct. Toolkits 6G*, 2025, pp. 1–6.

[14] B. Bojovic, K. Koutlia, S. Lagen, N. Patriciello, Z. Ali, L. Giupponi *et al.*, "5g-lena ns-3 nr module," *Zenodo*, 2023.

[15] 3GPP, "System architecture for the 5G System (5GS)," 3GPP, TS 23.501, 12 2025, version 20.0.0.

[16] J.-C. Bolot, S. Fosse-Parisis, and D. Towsley, "Adaptive fec-based error control for internet telephony," in *Proc. IEEE INFOCOM*, vol. 3, 1999, pp. 1453–1460.

[17] C. Padhye, K. J. Christensen, and W. Moreno, "A new adaptive fec loss control algorithm for voice over ip applications," in *Proc. IEEE Int. Perform. Comput. Commun. Conf. (IPCCC)*, 2000, pp. 307–313.

[18] S. Holmer, M. Shemer, and M. Paniconi, "Handling packet loss in webrtc," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, 2013, pp. 1860–1864.

[19] R. Wang, L. Si, and B. He, "Sliding-window forward error correction based on reference order for real-time video streaming," *IEEE Access*, vol. 10, pp. 34 288–34 295, 2022.

[20] A. Casparsen, V.-P. Bui, S. R. Pandey, J. J. Nielsen, and P. Popovski, "Experimental study of low-latency video streaming in an oran setup with generative ai," *IEEE Netw. Lett.*, 2026.

[21] K. Bountrogiannis, A. Ephremides, P. Tsakalides, and G. Tzagkarakis, "Age of incorrect information with hybrid arq under a resource constraint for n-ary symmetric markov sources," *IEEE/ACM Trans. Netw.*, 2024.

[22] Y. Li, C. Peng, Z. Yuan, J. Li, H. Deng, and T. Wang, "Mobileinsight: Extracting and analyzing cellular network information on smartphones," in *Proc. ACM MobiCom*, 2016, pp. 202–215.

[23] H. Wan, X. Cao, A. Marder, and K. Jamieson, "Nr-scope: A practical 5g standalone telemetry tool," in *Proc. ACM CoNEXT*, 2024, pp. 73–80.

[24] S. Holmer, M. Flodman, and E. Språng, "RTP Extensions for Transport-wide Congestion Control," Internet Engineering Task Force, Internet-Draft draft-holmer-rmcat-transport-wide-cc-extensions-00, Mar. 2015.

[25] A. Larrañaga, M. C. Lucas-Estañ, S. Lagén, Z. Ali, I. Martinez, and J. Gozalvez, "An open-source implementation and validation of 5g nr configured grant for urllc in ns-3 5g lena: A scheduling case study in industry 4.0 scenarios," *J. Netw. Comput. Appl.*, vol. 215, p. 103638, 2023.

[26] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[27] P. Belotti, C. Kirches, S. Leyffer, J. Linderoth, J. Luedtke, and A. Mahajan, "Mixed-integer nonlinear optimization," *Acta Numer.*, vol. 22, pp. 1–131, 2013.

[28] S. Williams, A. Waterman, and D. Patterson, "Roofline: an insightful visual performance model for multicore architectures," *Commun. ACM*, vol. 52, no. 4, pp. 65–76, 2009.

[29] R. Dolbeau, "Theoretical peak flops per instruction set: A tutorial," *J. Supercomput.*, vol. 74, no. 3, pp. 1341–1377, 2018.

[30] R. T. Rockafellar and S. Uryasev, "Optimization of conditional value-at-risk," *J. Risk*, vol. 2, pp. 21–42, 2000.

[31] 3GPP, "5G Real-time Media Transport Protocol Configurations," 3GPP, TS 26.522, 01 2026, version 19.3.0.

[32] S. Lagen, K. Wanuga, H. Elkotby, S. Goyal, N. Patriciello, and L. Giupponi, "New radio physical layer abstraction for system-level simulations of 5g networks," in *Proc. IEEE ICC*, 2020, pp. 1–7.

[33] M. Carrascosa and B. Bellalta, "Cloud-gaming: Analysis of google stadia traffic," *Comput. Commun.*, vol. 188, pp. 99–116, 2022.

[34] M. Gapeyenko, A. Samuylov, M. Gerasimenko, D. Moltchanov, S. Singh, M. R. Akdeniz *et al.*, "On the temporal effects of mobile blockers in urban millimeter-wave cellular scenarios," *IEEE Trans. Veh. Technol.*, vol. 66, no. 11, pp. 10 124–10 138, 2017.

[35] K. Nichols and V. Jacobson, "Controlling queue delay," *Communications of the ACM*, vol. 55, no. 7, pp. 42–50, 2012.

[36] G. Carlucci, L. De Cicco, S. Holmer, and S. Mascolo, "Analysis and design of the google congestion control for web real-time communication (webrtc)," in *Proc. ACM MMSys*, 2016, pp. 1–12.

[37] U. N. Bhat and R. Lal, "Number of successes in markov trials," *Adv. Appl. Probab.*, vol. 20, no. 3, pp. 677–680, 1988.