

Article

Predictive Semantic-Aware Hybrid Automatic Repeat Request for Reliable and Efficient 6G Communications

Osman Kaya ¹, Muhammet Ali Karabulut ¹, Can Eyüpoğlu ^{2,3} and Oktay Karakuş ^{2,*}

¹ Department of Electronics Engineering, Turkish Air Force Academy, National Defence University, 34149 Istanbul, Türkiye; osman.kaya2@msu.edu.tr; muhammetali.karabulut@msu.edu.tr

² Department of Computer Science and Informatics, Cardiff University, Cardiff CF24 4AG, UK; eyupogluc@cardiff.ac.uk or can.eyupoglu@msu.edu.tr

³ Department of Computer Engineering, Turkish Air Force Academy, National Defence University, 34149 Istanbul, Türkiye

* Correspondence: karakuso@cardiff.ac.uk

Abstract

Hybrid automatic repeat request (HARQ) remains the backbone of link-layer reliability, yet it was designed for a setting in which every bit matters equally and the channel of the next slot is unknown. Neither assumption holds in semantic communication over time-varying channels: some parts of a message are decisive for the receiver's task while others are not, and the deep fade that destroyed a first transmission is usually still present when an immediate retransmission arrives. This article proposes predictive semantic-aware HARQ (PSA-HARQ), a retransmission framework that makes its decisions at the level of individual semantic units and bases them jointly on how much each unit matters and on how the channel is expected to evolve. Semantic importance is extracted from a Transformer semantic encoder, a lightweight recurrent predictor forecasts the channel a few slots ahead, and a finite-horizon optimal-stopping controller decides, for every unit and every slot, whether to transmit, wait for a better slot, or abandon the unit. The forecast enters the controller through a confidence-weighted estimator whose weight is tracked at run time from the observed prediction errors, so that the policy falls back to the statistics-only optimum whenever the predictor becomes uninformative. Simulations over a vehicular situational-awareness task and over a real public record-classification dataset, with finite-blocklength coding and time-correlated Rayleigh fading, show that PSA-HARQ matches or exceeds the task accuracy of conventional HARQ in the low-SNR regime where retransmissions are most frequent, while transmitting far fewer blocks and consuming far less energy at the same channel condition. A cost-matched ablation shows that semantic importance and channel prediction contribute complementary gains. The results indicate that anticipating the channel, rather than merely reacting to it, is a practical and largely untapped lever for reliable and efficient 6G semantic communication.



Academic Editor: Yan Zhang

Received: 28 August 2026

Revised: 26 September 2026

Accepted: 28 September 2026

Published: 29 September 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Keywords: semantic communication; hybrid automatic repeat request; channel prediction; long short-term memory; optimal stopping; finite blocklength; 6G; ultra-reliable low-latency communication

1. Introduction

The sixth generation (6G) of mobile networks is expected to serve applications whose value lies not in the faithful delivery of bits but in the timely delivery of meaning: cooperative driving, remote control of machinery, extended reality, and distributed machine

inference [1–4]. This shift has revived Weaver’s semantic and effectiveness levels of communication and has produced a large body of work on semantic communication, in which a learned encoder transmits a task-relevant representation of the source rather than its literal description [5–8].

Reliability mechanisms, however, have largely not kept pace with this shift. Hybrid automatic repeat request (HARQ) remains the workhorse of link-layer reliability in 5G New Radio and is retained in essentially unchanged form in semantic systems: a cyclic redundancy check (CRC) failure produces a negative acknowledgment, and the transmitter resends the packet, possibly with incremental redundancy, until the packet is decoded or a retransmission budget is exhausted [9–11]. Two properties of this rule are difficult to justify once the objective is semantic rather than syntactic.

First, HARQ is semantically blind. Consider the situational-awareness message “an emergency vehicle is approaching the intersection at 63 km/h”. If the field carrying the exact speed is corrupted, the receiver’s ability to classify the situation as an emergency is essentially unaffected. If the field carrying the object class is corrupted, that ability collapses. Conventional HARQ spends an identical retransmission budget on both. Recent semantic HARQ designs have begun to address this by making the retransmission trigger depend on a learned semantic-similarity detector rather than on a CRC [12–14], and by ranking features by importance before retransmission [15]. What these schemes decide is whether a retransmission is semantically worthwhile.

Second, and much less studied, HARQ is temporally blind. A retransmission triggered at slot t is transmitted at slot $t + 1$, but in a time-correlated fading channel the deep fade that destroyed the first copy is very likely still present one slot later. The retransmission is therefore delivered into precisely the channel state least able to carry it. Classical link adaptation faces the same problem in the form of outdated channel state information (CSI), and a mature literature shows that recurrent and attention-based predictors can forecast fading channels several slots ahead with useful accuracy [16–19]. To the best of our knowledge, this predictive capability has not been brought inside the semantic retransmission decision.

The system is built from exactly three components, namely semantic importance estimation, channel prediction with a long short-term memory (LSTM) network, and adaptive HARQ decision-making, and deliberately excludes reconfigurable intelligent surfaces, digital twins, multi-agent learning, and other orthogonal enablers, so that the contribution of each component can be isolated and measured. What distinguishes the proposal from the semantic HARQ schemes cited above is not any one of these components, each of which has a mature literature of its own, but the way they are joined: to our knowledge, this is the first retransmission controller that treats when to resend a semantic unit as a decision in its own right, informed by a forecast of the channel rather than by its present state, and the first to do so inside a single decision-theoretic framework with an explicit guarantee on what happens when the forecast is wrong.

The main contributions are the following.

- Each semantic unit is mapped to an individually decodable short codeword, and retransmission decisions are made per unit on the basis of its estimated semantic importance rather than on packet-level error alone. Importance is obtained from a Transformer semantic encoder as a convex combination of normalized attention mass and leave-one-unit-out task relevance. This combination is validated against a source whose ground-truth task relevance is known by construction and then confirmed on a real public dataset in which relevance is not imposed.
- The retransmission problem is cast as a finite-horizon optimal-stopping problem over the actions (transmit, wait, skip), whose Bellman recursion is solved offline under the

fading statistics. An LSTM forecast of the next $H_{\max}$ slots then refines the continuation values of both the transmit and the wait branch through a confidence-weighted shrinkage estimator whose weight is updated at run time from the prediction errors actually observed on the link. The resulting policy reduces exactly to the statistics-only optimum when the predictor is uninformative and to the exact backward induction over the known channel trajectory when the prediction is exact. We prove both limits rather than assume them, and we quantify how much of the gain over the statistics-only controller survives when the latter is made aware of the channel correlation.

- A single Lagrangian objective balances importance-weighted semantic fidelity against transmission cost and delay. Every scheme is then compared along its whole achievable quality-cost frontier at a fixed channel condition, traced by sweeping whichever control knob that the scheme family exposes (the retransmission limit, the importance threshold, or the Lagrange multiplier), which makes the comparison independent of an arbitrary operating point. The frontiers are reported both in importance-weighted semantic quality and in task accuracy, the latter being independent of the importance scores that drive the controller.
- Monte Carlo experiments over 2×10^4 messages, Clarke time-varying Rayleigh channels, finite-blocklength coding, and Chase combining compare the schemes over a 4–24 dB signal-to-noise ratio (SNR) range and six mobility conditions. A cost-matched ablation separates the contribution of semantic importance from that of channel prediction. Paired confidence intervals, repeated training seeds, sensitivity studies with respect to every heuristic parameter, a predictor failure study, and measured execution times complete the picture.

The remainder of the article is organized as follows. Section 2 reviews related work. Section 3 defines the channel, semantic, and HARQ models. Section 4 develops predictive semantic-aware HARQ (PSA-HARQ). Section 5 describes the simulation setup, Section 6 reports and discusses the results, and Section 7 concludes.

2. Related Work

Deep learning-based joint source–channel coding replaced the separated design of Shannon’s architecture with an autoencoder trained end-to-end over the channel. DeepJSCC demonstrated graceful degradation of image quality with SNR and the absence of the cliff effect [20], and DeepSC extended the idea to text by maximizing sentence similarity instead of bit accuracy [5]. Subsequent surveys formalized the semantic and effectiveness levels, the notion of task-oriented communication, and the associated performance metrics [6,8,21], while Calvanese Strinati and Barbarossa [7] argued that 6G must move beyond Shannon toward goal-oriented design. A recurring theme is that not all transmitted semantic content is equally valuable, which has motivated importance-aware quantization [22], importance-aware power and subchannel allocation [23], importance-aware rate control [24], importance-driven reordering for OFDM transmission [25], feature-driven image transmission [26], lightweight semantic-aware packet transmission [27], and robust importance-aware transmission over satellite links [28]. These works allocate physical resources by importance, whereas the present work allocates retransmission opportunities by importance.

The field is also broadening from point-to-point links toward semantic networking among intelligent agents. Zhang et al. [29] set out a vision of semantic-based agent communication networks, in which agents exchange intent and task-level semantics rather than payloads, and identify the enabling technologies and open challenges of such networks. In the same direction, Meng et al. [30] describe an “intellicise” wireless network in which agentic artificial intelligence performs coverless semantic steganography, so that security is

provided at the semantic level rather than by protecting bits. Both lines of work presuppose a link layer that delivers the semantically decisive parts of a message reliably and on time, which is exactly what the retransmission mechanism proposed here is designed to provide. PSA-HARQ can therefore be read as a link-layer component of such semantic networks.

HARQ combines forward error correction with retransmission. Chase combining coherently accumulates the received copies so that the effective SNR is the sum of the per-attempt SNRs [9], whereas incremental redundancy sends additional parity. The finite-blocklength regime, characterized by the normal approximation of [31], governs the short packets typical of ultra-reliable low-latency communication (URLLC). Makki et al. [10] analyzed incremental-redundancy HARQ in this regime, and Ghanami et al. [32] extended the analysis to non-orthogonal access. Because every retransmission consumes both energy and delay, URLLC design is dominated by an energy–latency–reliability trade-off [11,33]. All of these treatments are semantically neutral: the value of a successfully delivered packet does not depend on its content.

A related body of work studies HARQ through the lens of information freshness. The age of information (AoI) measures the time elapsed since the generation of the most recent update received, and HARQ affects it in two opposite ways: retransmissions make an update more likely to arrive, but every retransmission delays the next update. Ceran et al. derived age-optimal scheduling policies under HARQ with a resource constraint [34] and extended them to multi-user networks with reinforcement learning [35]. Li et al. obtained a unified explicit expression of the average AoI under HARQ [36], and the age-optimal blocklength of HARQ-protected updates was characterized in [37]. These results share with the present work the insight that when to transmit is a decision in its own right, and that waiting can be preferable to transmitting. They differ in what is being optimized: AoI is a property of an update stream and is minimized over the long run, whereas PSA-HARQ optimizes the delivery of one deadline-constrained message whose parts have unequal value. The per-unit delay term of the objective introduced in Section 4.3 plays, within a single message, the role that age plays across a stream, and the two viewpoints are complementary: an AoI-aware scheduler decides which message to send next, and PSA-HARQ decides how the units of that message are protected within its deadline.

The closest line of work replaces the CRC trigger by a semantic criterion. Jiang et al. combined a Transformer semantic coder with Reed–Solomon-coded incremental retransmission and a similarity-detection network that decides whether a sentence needs to be resent [12] and extended semantic transmission to video conferencing [38]. Zhou et al. proposed incremental knowledge-based HARQ, in which retransmissions carry additional semantic knowledge rather than raw parity, together with adaptive bit-rate control [13], and Ni et al. added a topological-data-analysis error detector to the same framework [39]. Gao et al. jointly selected modulation order and retransmission according to semantic rather than bit-level performance [14]. Most recently, SemHARQ introduced feature-importance ranking and a feature-distortion evaluation network for multi-task semantic communication [15], generative semantic HARQ exploited latent-space diversity across retransmissions [40], and semantic HARQ has been applied to intelligent transportation systems [41]. Every one of these schemes bases the retransmission decision on present information: the semantic importance or distortion of the content and, at most, the current channel measurement. None of them anticipates the channel state in which the retransmission will actually be delivered, which is the specific gap addressed here.

Channel aging degrades any decision that relies on outdated CSI. Model-based long-range prediction of fading signals, in which a linear minimum mean square error (MMSE) predictor is built from the second-order statistics of the fading process, was established by Duel-Hallen et al. [42,43]. Recurrent networks were shown to outperform linear predic-

tors for flat-fading prediction over a wide range of normalized Doppler frequencies [16]. Machine-learning predictors recovered a large part of the spectral efficiency lost to channel aging in massive MIMO [17], and spatio-temporal autoregression and deep learning were unified for high-mobility systems [18]. Attention-based predictors were reported to make mobility almost negligible [19], with recent extensions to delay–Doppler domain systems [44]. The LSTM cell [45] remains the standard low-complexity choice for this task and is the one adopted here. These predictors have been used for precoding, scheduling, and link adaptation, but, to our knowledge, not to decide when a semantic retransmission should be released.

Table 1 positions PSA-HARQ against representative schemes. The upper part of the table lists retransmission mechanisms. The lower part lists importance-aware transmission schemes that allocate physical resources rather than retransmissions and are included because they share the semantic-importance ingredient. The distinguishing feature of PSA-HARQ is the joint use of semantic importance and of a multi-slot channel forecast inside a single decision-theoretic controller, together with a formulation in which the predictor’s measured accuracy determines how much the forecast is trusted.

Table 1. Positioning of the proposed scheme with respect to representative semantic transmission and retransmission strategies. “Current CSI” denotes use of the channel measurement of the current slot, and “predicted CSI” denotes use of a forecast of future slots.

Scheme	Semantic Importance	Current CSI	Predicted CSI	Per-Unit Decision
Retransmission mechanisms				
Conventional HARQ [9,10]	–	–	–	–
Semantic HARQ [12,13]	yes	–	–	–
Adaptive modulation + retransmission [14]	yes	yes	–	–
SemHARQ [15]	yes	–	–	yes
Generative semantic HARQ [40]	yes	–	–	–
AoI-aware HARQ [34,36]	–	–	–	–
PSA-HARQ (this work)	yes	yes	yes	yes
Importance-aware resource allocation (no retransmission)				
Importance-aware quantization [22]	yes	yes	–	yes
Importance-aware MIMO allocation [23]	yes	yes	–	yes

3. System Model

Figure 1 shows the end-to-end architecture. A transmitter encodes a source message into N semantic units, estimates the importance of each unit, and maps each unit onto an individually decodable short codeword. The receiver decodes the units, detects decoding errors with the per-unit CRC, estimates and predicts the channel, and reconstructs the message. A controller closes the loop by selecting, for every undelivered unit and every slot, one of three actions: transmit, wait, or skip.

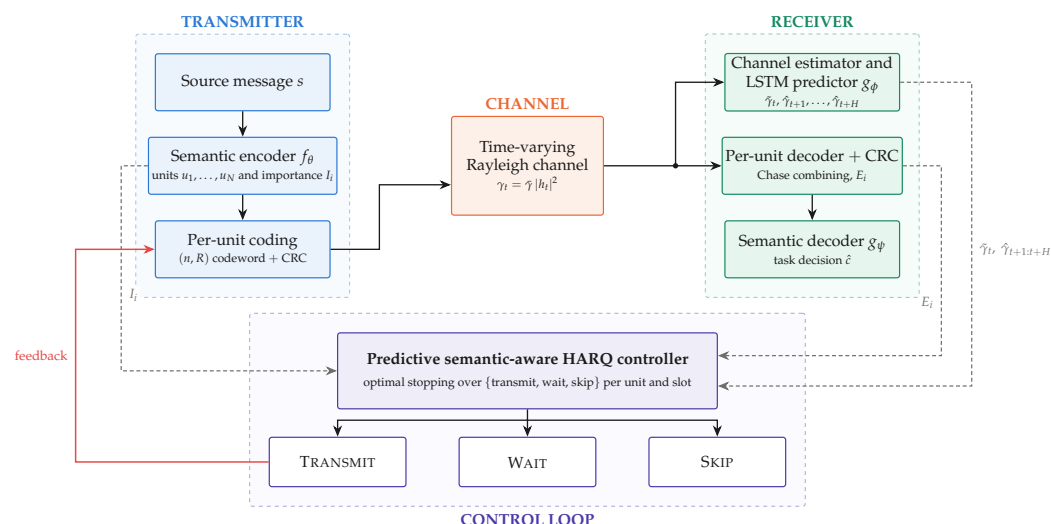


Figure 1. Architecture of the proposed predictive semantic-aware HARQ framework. Solid arrows denote the data path, dashed gray arrows the side information consumed by the controller, and the red arrow the low-rate feedback link carrying the retransmission command.

3.1. Wireless Channel Model

Time is slotted with slot duration T_s . The link is modeled as frequency-flat Rayleigh fading that is constant within a slot and correlated across slots according to Clarke's isotropic scattering model [46,47]. This is the flat-fading specialization of the standardized 3GPP model [48], and the complex gain h_t has autocorrelation

$$\mathbb{E}[h_t h_{t+k}^*] = J_0(2\pi f_D T_s k), \quad (1)$$

where $J_0(\cdot)$ is the zeroth-order Bessel function of the first kind and f_D is the maximum Doppler frequency. With $\mathbb{E}[|h_t|^2] = 1$, the instantaneous receive SNR of slot t is

$$\gamma_t = \bar{\gamma} |h_t|^2, \quad \gamma_t \sim \text{Exp}(1/\bar{\gamma}), \quad (2)$$

with $\bar{\gamma}$ the average SNR. Equation (2) describes the marginal law of a single slot, and the joint law of successive slots is fixed by (1). The two are used for different purposes in Section 4. The marginal law is the basis of the offline value function, and the correlation is what the predictor exploits online. The receiver obtains an imperfect pilot-based estimate

$$\tilde{\gamma}_t = \gamma_t 10^{\epsilon_t/10}, \quad \epsilon_t \sim \mathcal{N}(0, \sigma_e^2), \quad (3)$$

where σ_e is the estimation-error standard deviation in dB. Only $\tilde{\gamma}_t$ and quantities derived from it are available to the controller.

3.2. Semantic Communication Model

A source message s is mapped by a semantic encoder f_θ into an ordered tuple of N semantic units,

$$(u_1, \dots, u_N) = f_\theta(s), \quad \mathcal{S} = \{u_1, \dots, u_N\}, \quad (4)$$

each carrying one atomic element of meaning, where $\mathcal{S}$ denotes the set of units of the message. A separately parameterized semantic decoder g_ψ operates on the subset of units that were delivered and produces both a reconstruction $\hat{s}$ and a task decision $\hat{c}$. Every unit is protected by its own CRC and mapped to a codeword of n channel uses at a rate of R bits per channel use, so that unit-level error detection and unit-level retransmission are possible. This semantic packetization is what allows the retransmission granularity to match the

semantic granularity. It costs a modest CRC overhead per unit and is the direct analog of the feature-level treatment used in [15].

Every unit carries a semantic importance $I_i \in [0, 1]$, defined in Section 4.1. Writing $\delta_i \in \{0, 1\}$ for the indicator that unit i was eventually delivered, the semantic quality of a message is the importance-weighted delivery ratio

$$Q_s = \frac{\sum_{i=1}^N I_i \delta_i}{\sum_{i=1}^N I_i} \in [0, 1]. \quad (5)$$

Because (5) is weighted by the same importance scores that the controller optimizes, we additionally report the task-inference accuracy, i.e., the probability that $\hat{c}$ equals the ground-truth label of s when the undelivered units are masked. The latter metric is independent of I_i and therefore constitutes an unbiased, effectiveness-level measure of performance. Every principal comparison in Section 6 is made on this metric as well.

3.3. Finite-Blocklength Error and HARQ Model

Let $a_i(t)$ denote the post-combining accumulated SNR of unit i at the beginning of slot t , with $a_i(0) = 0$. Under Chase combining the copies are coherently accumulated [9], so that transmitting unit i in slot t raises the accumulated SNR to $a_i(t) + \gamma_t$. For a codeword of n channel uses at rate R , the block error probability is given by the finite-blocklength normal approximation [31]

$$\varepsilon(\gamma) = \mathcal{Q}\left(\frac{C(\gamma) - R}{\sqrt{V(\gamma)/n}}\right), \quad (6)$$

$$C(\gamma) = \log_2(1 + \gamma), \quad V(\gamma) = \left(1 - \frac{1}{(1 + \gamma)^2}\right) \log_2^2 e, \quad (7)$$

where $\mathcal{Q}(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-v^2/2} dv$. In (6), n is the blocklength of the coded data part of a block, that is, the number of channel uses that carry the codeword, and R is the coding rate of this part. With $n = 128$ and $R = 2.5$, a codeword carries $nR = 320$ bits. These bits include the 16-bit CRC of the unit, so the rate R counts the CRC bits and each unit carries 304 payload bits. In addition to the codeword, every block contains a pilot field of $n_p = 16$ channel uses. The pilots are used only to obtain the SNR estimate (3). They are not part of the codeword and therefore do not enter (6). A block thus occupies $n + n_p = 144$ channel uses, and its payload rate is $304/144 \approx 2.1$ bits per channel use. We assume that the CRC detects every decoding error, since the probability of an undetected error of a 16-bit CRC is about 2^{-16} . The success probability of an attempt made in slot t is therefore

$$p_i(t) = 1 - \varepsilon(a_i(t) + \gamma_t), \quad (8)$$

and the decoding-error indicator observed at the receiver is $E_i(t) = 1$ if the CRC of unit i still fails after slot t and $E_i(t) = 0$ otherwise. The controller acts on the units with $E_i(t) = 1$. This indicator reports a failure of the physical-layer decoder. It is not a measure of semantic distortion, which is captured separately by the quality and accuracy metrics defined above.

Multiplexing and Overhead Assumptions

The N codewords released in one slot occupy N orthogonal resource blocks of that slot, for instance N groups of subcarriers of a New Radio mini-slot, so that any subset of the units can be transmitted in parallel. All blocks of a slot see the same flat-fading gain h_t , and each block is transmitted with the same power P_{tx} and consumes the energy $P_{tx}T_s$. The bandwidth of a slot is therefore proportional to the number of blocks it carries, whereas its duration is not. This is why the transmission cost counts blocks and the latency counts

slots. The pilot field and the CRC are the same for every block and every scheme. They therefore scale the transmission cost and the energy of all schemes by the same factor and do not change any comparison. Feedback consists of an N -bit acknowledgment bitmap and an N -bit release command per slot, carried on a low-rate control channel. The processing of the receiver and the controller must finish within the slot in which the corresponding observation is made. Section 6 shows that a compiled implementation of this processing meets this requirement on a single core of a general-purpose processor.

A message must be completed within a deadline of K slots, which for URLLC traffic is dictated by the application. A unit is open at a slot if it has not yet been delivered and has not been permanently terminated, where termination means that the controller has skipped the unit or its retransmission budget has been exhausted. Let m_i denote the total number of blocks transmitted for unit i , let ℓ_i denote the number of slots after the first one in which unit i was transmitted or deliberately held back, and let L denote the index of the slot at the end of which no unit of the message remains open, i.e., the slot at which every unit has either been delivered or terminated. The transmission cost, message latency, and transmit energy of a message are

$$C_{\text{tx}} = \frac{1}{N} \sum_{i=1}^N m_i, \quad L_{\text{ms}} = (L + 1) T_s, \quad \mathcal{E} = P_{\text{tx}} T_s \sum_{i=1}^N m_i, \quad (9)$$

where P_{tx} is the transmit power. Note that $C_{\text{tx}} \geq 1$ because every unit is transmitted at least once, and that slots spent waiting are counted in the latency exactly like slots spent transmitting: a controller that holds a unit back and later abandons it is charged for the time it kept the message open. Because a unit that is abandoned at slot t was held back through slot $t - 1$, the message latency and the per-unit open times satisfy $\max_i \ell_i \leq L \leq \max_i \ell_i + 1$, with the upper value attained only when the last open unit is abandoned rather than delivered.

4. Proposed Predictive Semantic-Aware HARQ

4.1. Semantic Importance Estimation

The semantic encoder is a Transformer encoder [49] operating on the N semantic units preceded by a classification token, in the manner of masked language models [50], trained end-to-end on the downstream task with random unit masking so that the decoder is robust to missing units, which is exactly the condition it faces when low-importance units are deliberately not retransmitted. Two complementary quantities are extracted from the trained network.

The first is the attention mass that the classification token places on unit i , averaged over the M heads and Λ layers,

$$A_i = \frac{1}{\Lambda M} \sum_{\ell=1}^{\Lambda} \sum_{m=1}^M \omega_{0,i}^{(\ell,m)}, \quad \omega_{0,i}^{(\ell,m)} = \frac{\exp(q_0^{(\ell,m)\top} \mathbf{k}_i^{(\ell,m)} / \sqrt{d})}{\sum_{j=0}^N \exp(q_0^{(\ell,m)\top} \mathbf{k}_j^{(\ell,m)} / \sqrt{d})}, \quad (10)$$

with $q, \mathbf{k}$ the query and key vectors and d the head dimension. The second is the task relevance, measured by the drop in the probability assigned to the predicted class when unit i is masked out,

$$T_i = \left[\mathbb{P}(\hat{c} | \mathcal{S}) - \mathbb{P}(\hat{c} | \mathcal{S} \setminus \{u_i\}) \right]_+, \quad (11)$$

where $[\cdot]_+ = \max(\cdot, 0)$. Both quantities are min–max normalized within each message, $\bar{A}_i = (A_i - \min_j A_j) / (\max_j A_j - \min_j A_j)$ and likewise for $\bar{T}_i$, and combined convexly:

$$I_i = \mathcal{N}(\alpha \bar{A}_i + (1 - \alpha) \bar{T}_i), \quad \alpha \in [0, 1], \quad (12)$$

where $\mathcal{N}(\cdot)$ denotes the same within-message min–max normalization, applied once more so that every message contains at least one unit of importance 1 and at least one of importance 0. The purpose of the within-message normalization is to make the importance scale comparable across messages, so that the fixed thresholds introduced below have the same meaning for every message. Its degenerate case is handled explicitly: if all A_i (or all T_i) of a message are equal, the normalization of that quantity returns 1 for every unit, i.e., a message in which no unit can be singled out is treated as one in which every unit matters. Because T_i is a probability drop, all T_i of a message are equal only when masking any single unit leaves the decision probability unchanged, which happens almost exclusively for messages whose decision is redundantly supported. On the corpora used in Section 6 this occurs in a negligible fraction of messages, and Section 6 shows that the results are insensitive to the normalization variant, to α , and to the thresholds. The attention term is obtained in a single forward pass, whereas the task-relevance term requires N additional masked evaluations, which are batched into a single forward pass of N inputs at the transmitter before the first transmission (Section 6). The convex combination lets the designer trade estimation fidelity against complexity through α .

Units are grouped into three tiers by two thresholds $\tau_L < \tau_H$, which determine the maximum number of retransmissions each unit may request:

$$B(I_i) = \begin{cases} B_H, & I_i \geq \tau_H \quad (\text{high importance}), \\ B_M, & \tau_L < I_i < \tau_H \quad (\text{medium importance}), \\ 0, & I_i \leq \tau_L \quad (\text{low importance}). \end{cases} \quad (13)$$

4.2. LSTM-Based Channel Prediction

At every slot the receiver forms the four-dimensional observation

$$\mathbf{x}_t = \left[10 \log_{10} \tilde{\gamma}_t, \quad P_{r,t}^{\text{dB}}, \quad \sqrt{\tilde{\gamma}_t / \bar{\gamma}}, \quad \log_{10} \text{BER}_t \right]^\top, \quad (14)$$

comprising the estimated SNR in dB, a received signal strength indicator (RSSI) proxy, a channel magnitude proxy, and an uncoded bit error rate proxy. All four are deterministic functions of $\tilde{\gamma}_t$ alone and therefore carry no additional channel information. They are fixed nonlinear re-expressions of the same measurement (linear, square-root, and exponential scales) that relieve the network of learning these transformations, and Section 6 shows that they lower the prediction error relative to feeding the estimated SNR only. A single-layer LSTM [45] consumes a sliding window of the last L_w observations, $\mathbf{X}_t = [\mathbf{x}_{t-L_w+1}, \dots, \mathbf{x}_t]$, through the standard recursion

$$\begin{aligned} \mathbf{f}_\tau &= \sigma(\mathbf{W}_f[\mathbf{c}_{\tau-1}^h, \mathbf{x}_\tau] + \mathbf{b}_f), \quad \mathbf{i}_\tau = \sigma(\mathbf{W}_i[\mathbf{c}_{\tau-1}^h, \mathbf{x}_\tau] + \mathbf{b}_i), \\ \mathbf{o}_\tau &= \sigma(\mathbf{W}_o[\mathbf{c}_{\tau-1}^h, \mathbf{x}_\tau] + \mathbf{b}_o), \quad \mathbf{c}_\tau = \mathbf{f}_\tau \odot \mathbf{c}_{\tau-1} + \mathbf{i}_\tau \odot \tanh(\mathbf{W}_c[\mathbf{c}_{\tau-1}^h, \mathbf{x}_\tau] + \mathbf{b}_c), \end{aligned} \quad (15)$$

where $\mathbf{f}_\tau, \mathbf{i}_\tau, \mathbf{o}_\tau$ are the forget, input, and output gates, $\mathbf{c}_\tau$ the cell state, $\mathbf{c}_\tau^h = \mathbf{o}_\tau \odot \tanh(\mathbf{c}_\tau)$ the hidden state, $\sigma(\cdot)$ the logistic function, and $\odot$ the elementwise product. A multi-layer perceptron head maps the final hidden state to the $H_{\max}$ -step forecast

$$[\hat{\gamma}_{t+1}^{\text{dB}}, \dots, \hat{\gamma}_{t+H_{\max}}^{\text{dB}}] = g_\phi(\mathbf{X}_t). \quad (16)$$

The network is trained offline by minimizing the Huber loss over channel realizations drawn from a range of Doppler frequencies and average SNRs, so that a single predictor generalizes across mobility conditions. Its accuracy at horizon h is summarized by the root mean square error σ_h (in dB), which is measured on a held-out set at design time and

tracked at run time on the link itself, as described in Section 4.6. The error σ_h is the only property of the predictor that the controller uses.

4.3. Optimization Objective

Let π denote a retransmission policy, i.e., a mapping from the information available at slot t to an action for every open unit. We seek the policy that maximizes the expected semantic value delivered per message, penalized by the resources it consumes:

$$\max_{\pi} J(\pi) = \mathbb{E}_{\pi} \left[\sum_{i=1}^N I_i \delta_i - \lambda_c \sum_{i=1}^N m_i - \lambda_{\ell} \sum_{i=1}^N \ell_i \right], \quad (17)$$

subject to $m_i \leq 1 + B(I_i)$ and $L < K$. Equation (17) is the Lagrangian of the constrained problem “maximize semantic quality subject to a transmission-cost and a delay budget”, and $\lambda_c \geq 0$ and $\lambda_{\ell} \geq 0$ are the corresponding multipliers. Sweeping λ_c traces the achievable (C_{tx}, Q_s) frontier, which is how the schemes are compared in Section 6: comparing two policies at a single arbitrary operating point would confound “spends more” with “decides better”.

The delay term of (17) deserves a comment. The message latency L of (9) is a maximum over units and is not additive, so a Lagrangian built on $\lambda_{\ell} L$ would couple the units and could not be solved one unit at a time. The objective therefore penalizes the per-unit open times ℓ_i instead. This choice is exact rather than approximate: the controller optimizes (17) as written, and the message latency, which is reported separately in Section 6, is controlled through the bound $L \leq \max_i \ell_i + 1 \leq \sum_i \ell_i + 1$. Each open slot of each unit is charged λ_{ℓ} , so a unit that is held back pays for the slots it waits, which is the mechanism that keeps the wait action from being abused. Because the units are separately encoded and separately CRC-protected, and because the three terms of (17) are all sums over units, the objective decomposes into N subproblems that are coupled only through the shared channel realization, and each subproblem is a finite-horizon optimal-stopping problem.

4.4. Finite-Horizon Optimal-Stopping Formulation

Consider a single unit of importance I with k slots still available and accumulated SNR a . The controller may transmit in the current slot, wait one slot, or abandon the unit permanently. Write $p(\gamma) = 1 - \varepsilon(\gamma)$, with $\varepsilon(\cdot)$ as in (6), and define the transmit value

$$\Phi_j(a, \gamma, I) = I p(a + \gamma) - \lambda_c + (1 - p(a + \gamma)) W_j(a + \gamma, I) - \lambda_{\ell}, \quad (18)$$

i.e., the value of transmitting into a slot of SNR γ when j slots remain afterwards.

The offline value function is computed under the independent-marginal approximation, in which the SNRs of successive slots are treated as independent draws from the marginal law (2). This is an approximation of the correlated process (1): under the true joint law the conditional distribution of a future SNR depends on the current channel state, and the state (k, a) is then not sufficient. The approximation is adopted deliberately, for two reasons. First, it makes the value function scalar in a and independent of the Doppler frequency, so that a single offline table serves every mobility condition, whereas a correlation-aware table must be recomputed whenever the Doppler changes. Second, the correlation is exploited elsewhere in the design, namely by the predictor of Section 4.2, whose forecast is conditioned on the observed channel history through a sixteen-slot window and therefore captures the temporal structure of (1) more completely than any low-order Markov model of it. Section 4.5 gives the correlation-aware counterpart of the recursion and Section 6 quantifies, on the same channel realizations, how much of the predictive gain is attributable to the approximation.

Proposition 1 (Statistics-only value function under the independent-marginal approximation). Define $W_0(a, I) = 0$ and, for $k \geq 1$,

$$W_k(a, I) = \mathbb{E}_\gamma \left[\max \left\{ \underbrace{\Phi_{k-1}(a, \gamma, I)}_{\text{transmit}}, \underbrace{W_{k-1}(a, I) - \lambda_\ell}_{\text{wait}}, \underbrace{0}_{\text{skip}} \right\} \right], \quad (19)$$

with $\gamma \sim \text{Exp}(1/\bar{\gamma})$. Then $W_k(a, I)$ is the optimal expected contribution of the unit to (17) when successive slot SNRs are independent draws from the marginal fading law and only that law, not the future realizations, is known. The optimal action is the argument attaining the maximum.

Under the independent-marginal law, the state (k, a) is Markov: the accumulated SNR summarizes the whole transmission history through (8), the residual horizon k is deterministic, and the SNR of the coming slot is independent of the past. The three actions exhaust the action space, and abandoning yields exactly zero further reward, so W_k is the value function of a finite-horizon Markov decision process with terminal value $W_0 \equiv 0$. Equation (19) is its dynamic-programming recursion, and the optimality of the greedy argument follows from Bellman's principle of optimality. Note that on the "wait" branch no transmission occurs, hence a is unchanged, which is what makes the recursion scalar in a .

Proposition 2 (Monotonicity and threshold structure). For every $I \in [0, 1]$, $W_k(a, I)$ is non-decreasing in k , in a , and in I . Moreover, for fixed (k, a, I) the transmit action is optimal on an upper interval of γ , i.e., the optimal statistics-only policy is a threshold policy in the instantaneous SNR.

Monotonicity in k follows by induction: $W_1 \geq W_0 = 0$ because the maximum in (19) includes 0. If $W_k \geq W_{k-1}$ pointwise, then both the transmit and the wait arguments at stage $k+1$ dominate their counterparts at stage k , and the maximum and the expectation preserve the order. Monotonicity in a follows because $\varepsilon(\cdot)$ in (6) is strictly decreasing, so $p(a + \gamma)$ is non-decreasing in a , and the induction hypothesis makes $W_{k-1}(a + \gamma, I)$ non-decreasing in a . Monotonicity in I is immediate since I enters only through the non-negative term $I p(\cdot)$. For the threshold property, fix (k, a, I) and let $\Delta(\gamma) = \Phi_{k-1}(a, \gamma, I) - \max\{W_{k-1}(a, I) - \lambda_\ell, 0\}$. The subtracted term does not depend on γ , while $p(a + \gamma)$ is non-decreasing in γ and, by the induction hypothesis, so is $W_{k-1}(a + \gamma, I)$. Moreover $W_k \leq I$ for every k : this holds for $W_0 = 0$, and if $W_{k-1} \leq I$ then $\Phi_{k-1} \leq I p - \lambda_c + (1 - p)I - \lambda_\ell \leq I$, so $W_k \leq I$ as well. Hence, the coefficient $I - W_{k-1}(a + \gamma, I)$ multiplying $p(a + \gamma)$ in Φ_{k-1} is non-negative, Δ is non-decreasing in γ , and $\{\gamma : \Delta(\gamma) \geq 0\}$ is an upper interval.

Because the recursion (19) depends on the channel only through the marginal law of γ , W_k needs to be computed only once per average SNR, offline, on a grid of (a, I) . The monotonicity established in Proposition 2 guarantees that this tabulation is well behaved under the bilinear interpolation used at run time.

4.5. Correlation-Aware Statistics-Only Recursion

The independence assumption of Proposition 1 can be relaxed by conditioning on the current channel state. The Clarke process is not Markov of any finite order, but its first-order Gauss–Markov approximation, $h_{t+1} = r h_t + \sqrt{1 - r^2} w_t$ with $r = J_0(2\pi f_D T_s)$ and $w_t \sim \mathcal{CN}(0, 1)$, matches the one-slot correlation of (1) exactly. Under this model the SNR of the next slot given the current one is non-central chi-square distributed with two degrees of freedom, i.e., Rician with factor $r^2 \gamma_t / ((1 - r^2) \bar{\gamma})$, and the state must be enlarged

to (k, a, γ) , where γ is the SNR of the current slot. Writing $M_k(a, \gamma, I) = \mathbb{E}[W_k(a, \gamma', I) | \gamma]$ for the expected value one slot ahead, the recursion becomes

$$W_k(a, \gamma, I) = \max \left\{ I p(a + \gamma) - \lambda_c - \lambda_\ell + (1 - p(a + \gamma)) M_{k-1}(a + \gamma, \gamma, I), \right. \\ \left. M_{k-1}(a, \gamma, I) - \lambda_\ell, 0 \right\}, \quad (20)$$

with $W_0 \equiv M_0 \equiv 0$. The proof of Proposition 1 carries over verbatim to (20) with the enlarged state, and the resulting controller, denoted B4c in Section 6, is optimal for the first-order Markov channel given the current SNR estimate. Its cost is a three-dimensional table that depends on $f_D T_s$ and must therefore be recomputed when the mobility changes. We use it as a second statistics-only reference: comparing the proposed predictive controller against both B4 and B4c separates the part of its gain that comes from anticipating the channel from the part that the independent-marginal approximation concedes.

4.6. Refining the Continuation Values with the Predicted Trajectory

The continuation values in (19) are expectations over the unknown future channel. When a forecast $\hat{\gamma}_{t+1}, \dots, \hat{\gamma}_{t+H}$ is available, with $H = \min(k-1, H_{\max})$ the usable part of the forecast, these expectations can be replaced by planning along the predicted trajectory. The plan must respect two facts. First, the future SNRs are taken as known, but the outcome of each transmission remains random, because a block transmitted into a slot of SNR γ still fails with probability $\varepsilon(a + \gamma)$. Second, a transmission that fails changes the accumulated SNR, so the plan after a failed transmission continues from $a + \hat{\gamma}$ and not from a . The trajectory-conditioned value of entering slot $t + 1$ with accumulated SNR a' is therefore obtained by backward induction over the predicted slots,

$$V_{H+1}(a') = W_{k-1-H}(a', I), \quad (21)$$

$$V_h(a') = \max \left\{ I p(a' + \hat{\gamma}_{t+h}) - \lambda_c - \lambda_\ell + (1 - p(a' + \hat{\gamma}_{t+h})) V_{h+1}(a' + \hat{\gamma}_{t+h}), \right. \\ \left. V_{h+1}(a') - \lambda_\ell, 0 \right\}, \quad (22)$$

for $h = H, H-1, \dots, 1$, and the quantity of interest is $Z(a') = V_1(a')$. Equation (21) closes the recursion with the statistics-only value beyond the prediction window. Unlike a recursion that follows the waiting branch only, (22) is evaluated on the binary tree of accumulated SNRs reachable from a' : at every predicted slot the unit either transmits, which moves it to $a' + \hat{\gamma}_{t+h}$ if the attempt fails, or waits, which leaves it at a' . The tree has at most 2^H leaves, and Section 4.7 shows that its evaluation is inexpensive. Throughout, the predicted quantities are converted from the decibel scale of (16) to the linear scale used by $p(\cdot)$, $\hat{\gamma}_{t+h} = 10^{\hat{\gamma}_{t+h}^{\text{dB}}/10}$.

Trusting the trajectory-conditioned value unconditionally would be unwise, because the forecast carries an error that grows with the horizon. We therefore combine the two estimates of every continuation value by a confidence-weighted shrinkage estimator,

$$\hat{W}_{k-1}(a') = \rho Z(a') + (1 - \rho) W_{k-1}(a', I), \quad \rho = \exp(-\kappa \bar{\sigma}_H), \quad \bar{\sigma}_H = \left(\frac{1}{H} \sum_{h=1}^H \sigma_h^2 \right)^{1/2}, \quad (23)$$

where $\kappa > 0$ calibrates how quickly confidence decays with the predictor's root mean square error and $\bar{\sigma}_H$ is the aggregate accuracy over the horizons actually used in the plan. The same shrunk continuation value is used on both branches of the decision: on the wait

branch at $a' = a$, and on the transmit branch at $a' = a + \tilde{\gamma}_t$, i.e., for the state reached if the transmission made now fails. The three action values at slot t are

$$\begin{aligned} \mathcal{U}^{\text{tx}}(t) &= I p(a + \tilde{\gamma}_t) - \lambda_c - \lambda_\ell + (1 - p(a + \tilde{\gamma}_t)) \hat{W}_{k-1}(a + \tilde{\gamma}_t), \\ \mathcal{U}^{\text{wt}}(t) &= \hat{W}_{k-1}(a) - \lambda_\ell, \quad \mathcal{U}^{\text{sk}} = 0, \end{aligned} \quad (24)$$

and the decision taken for unit i at slot t is

$$D_i(t) = \arg \max \left\{ \mathcal{U}_i^{\text{tx}}(t), \mathcal{U}_i^{\text{wt}}(t), \mathcal{U}_i^{\text{sk}} \right\} \in \{\text{TRANSMIT, WAIT, SKIP}\}, \quad (25)$$

subject to the importance-dependent budget (13) and to the deadline, with $\mathcal{U}^{\text{wt}}(t) = -\infty$ when $k = 1$.

A confidence weight computed once from a held-out set cannot react to a change in the operating conditions. The controller therefore maintains its own estimate of σ_h on the link. A forecast $\hat{\gamma}_{t+h}^{\text{dB}}$ made at slot t can be compared, h slots later, with the estimate $10 \log_{10} \tilde{\gamma}_{t+h}$ that the receiver obtains in any case, and the squared error is folded into an exponentially weighted average,

$$\hat{\sigma}_h^2(t) = (1 - \beta) \hat{\sigma}_h^2(t-1) + \beta (\hat{\gamma}_t^{\text{dB}}|_{t-h} - 10 \log_{10} \tilde{\gamma}_t)^2, \quad (26)$$

where $\hat{\gamma}_t^{\text{dB}}|_{t-h}$ is the forecast of slot t made at slot $t-h$, $\beta \in (0, 1)$ is the forgetting factor, and the average is initialized with the design-time value σ_h^2 . The estimate (26) uses only quantities the receiver already has, costs one multiply-add per horizon and slot and includes the SNR-estimation noise of (3), so it slightly overstates the true forecast error and errs on the side of trusting the forecast less. The weight ρ of (23) is computed from $\hat{\sigma}_h(t)$ at every decision.

The construction has two limiting behaviors.

Proposition 3 (Consistency of the predictive policy). *Let π_{PSA} be the policy defined by (25).*

(i) *If the predictor is uninformative, in the sense that $\bar{\sigma}_H \rightarrow \infty$ and hence $\rho \rightarrow 0$, then π_{PSA} coincides with the optimal statistics-only policy of Proposition 1.* (ii) *If the prediction is exact, $\hat{\gamma}_{t+h} = \gamma_{t+h}$ for $h \leq H$, if $\tilde{\gamma}_t = \gamma_t$, and $H \geq k-1$, then $\rho \rightarrow 1$ and π_{PSA} is the optimal policy of the remaining-horizon problem in which the channel trajectory is known and only the decoding outcomes are random.*

(i) With $\rho = 0$, (23) gives $\hat{W}_{k-1}(a') = W_{k-1}(a', I)$ for every a' , so (24) reproduces, term by term, the three arguments of the maximum in (19), whose argmax is optimal by Proposition 1. (ii) Since $H = \min(k-1, H_{\text{max}}) \leq k-1$, the hypothesis $H \geq k-1$ forces $H = k-1$, so (21) is evaluated at $W_{k-1-H} = W_0 = 0$ and the tail contributes nothing. Consider the finite-horizon Markov decision process with state a' (the accumulated SNR), known slot SNRs $\gamma_{t+1}, \dots, \gamma_{t+k-1}$, actions {transmit, wait, skip}, transition $a' \rightarrow a' + \gamma_{t+h}$ with probability $\varepsilon(a' + \gamma_{t+h})$ and absorption with reward I otherwise on a transmission, and per-slot costs λ_c and λ_ℓ as in (17). Equation (22) is exactly the dynamic-programming recursion of this process, evaluated at every state reachable from a' , so $Z(a') = V_1(a')$ is its optimal value from slot $t+1$ onward. With $\rho = 1$ the action values (24) become $\mathcal{U}^{\text{tx}} = I p(a + \gamma_t) - \lambda_c - \lambda_\ell + (1 - p(a + \gamma_t)) V_1(a + \gamma_t)$, $\mathcal{U}^{\text{wt}} = V_1(a) - \lambda_\ell$, and $\mathcal{U}^{\text{sk}} = 0$, which are the three arguments of the same recursion at slot t . Hence (25) is the optimal decision at slot t , and since the same argument applies at every later slot with the then-remaining trajectory, the policy is optimal for the whole remaining horizon.

Two remarks clarify what Proposition 3 does and does not say. First, “optimal for the known trajectory” does not mean that the plan is deterministic: the controller does not know whether a transmission will succeed, and (22) averages over that outcome at

every predicted slot. What the forecast removes is only the uncertainty about the channel. This is why the recursion branches after a transmission, continuing from $a' + \hat{\gamma}_{t+h}$ if the attempt fails, and a recursion that continued with the statistics-only value after a failed transmission would not satisfy part (ii). Second, part (i) describes what the controller becomes when the forecast is worthless, but it is the run-time estimator (26) that makes the fallback operational: if the predictor is deployed outside the conditions it was trained for, the observed errors grow, $\hat{\sigma}_h$ follows them within a few tens of slots, and ρ decays toward the statistics-only policy. Section 6 demonstrates this behavior with a deliberately mismatched predictor. Between the two limits, ρ interpolates between the two continuation values, and the sensitivity of the results to the decay rate κ is reported in Section 6.

4.7. Algorithm and Complexity

Algorithm 1 summarizes the online operation. The offline stage solves (19) on a grid of N_a accumulated-SNR values, N_I importance values, and K horizons using N_q Gauss–Laguerre quadrature nodes, at a cost of $\mathcal{O}(KN_a N_q N_I)$ operations. With the values of Table 2 this is below 10^6 operations, and it is performed once per average SNR. The online cost per slot is dominated by the trajectory recursion (22), which evaluates (6) at every node of the binary tree of depth H and interpolates the tabulated W at its 2^H leaves, for two starting states (a and $a + \tilde{\gamma}_t$) and N units: $\mathcal{O}(N 2^{H+1})$ operations, which with $N = 10$ and $H_{\max} = 6$ is about 1.3×10^3 evaluations of (6) per slot. The depth of the tree can be limited further by the remaining retransmission budget of the unit, since at most $B(I_i) - r_i$ transmissions can be planned. Section 6 reports the measured execution times of every component. The LSTM forward pass costs $\mathcal{O}(L_w d_h^2)$ with d_h the hidden size and is shared by all units.

Table 2. Simulation parameters.

Parameter	Symbol	Value
Semantic units per message	N	10 (V2X corpus); 14 (Adult dataset)
Codeword length (coded data part)	n	128 channel uses
Code rate (including the CRC bits)	R	2.5 bits/channel use
CRC length	–	16 bits
Pilot field per block	n_p	16 channel uses
Slot duration	T_s	0.125 ms
Transmit power	P_{tx}	0.2 W
Deadline	K	8 slots (1 ms)
Retransmission budgets	B_H, B_M	3, 1
Importance thresholds	τ_H, τ_L	0.75, 0.40
Importance mixing weight	α	0.5
Cost and delay weights	λ_c, λ_ℓ	0.20, 0.010
Confidence decay	κ	0.1054 ($\rho = 0.9$ at $\sigma = 1$ dB)
Error-tracker forgetting factor	β	0.1
Average SNR range	$\tilde{\gamma}$	4–24 dB, 2 dB steps
Normalized Doppler	$f_D T_s$	0.05 (nominal); 0.01–0.15 (sweep)
SNR estimation error	σ_e	1.0 dB
Observation window/horizon	$L_w, H_{\max}$	16, 6 slots
Value-function grid	N_a, N_I, N_q	49, 21, 48
Correlation-aware grid	N_γ , quadrature	24 current-SNR points, 12×12 Gauss–Hermite nodes
Monte Carlo messages per point	–	20,000
Training seeds	–	3

Algorithm 1 Predictive semantic-aware HARQ (per message)

Input: semantic units $\{u_i\}_{i=1}^N$, importance $\{I_i\}$, deadline K , tabulated $W_k(\cdot, \cdot)$, predictor g_ϕ , error estimates $\{\hat{\sigma}_h\}$

Output: delivery indicators $\{\delta_i\}$, transmission counts $\{m_i\}$

- 1: $a_i \leftarrow 0, r_i \leftarrow 0$ (attempts used), $\delta_i \leftarrow 0, m_i \leftarrow 0, \text{alive}_i \leftarrow \text{TRUE}$ for all i
- 2: transmit all N units in slot $t = 0$; update a_i, m_i, δ_i from the CRC outcomes
- 3: **for** $t = 1$ to $K - 1$ **do**
- 4: $k \leftarrow K - t$ ▷ slots still available
- 5: obtain $\tilde{\gamma}_t$ and the forecast $\hat{\gamma}_{t+1:t+H} \leftarrow g_\phi(\mathbf{X}_t), H \leftarrow \min(k - 1, H_{\max})$
- 6: update $\hat{\sigma}_h$ from the forecasts that matured at slot t , Equation (26)
- 7: $\rho \leftarrow \exp(-\kappa \bar{\sigma}_H)$ ▷ Equation (23)
- 8: **for all** i with $\delta_i = 0, \text{alive}_i, r_i < B(I_i)$ **do**
- 9: $Z(a_i), Z(a_i + \tilde{\gamma}_t) \leftarrow$ backward recursion (21)–(22)
- 10: $\hat{W}_{k-1}(a') \leftarrow \rho Z(a') + (1 - \rho)W_{k-1}(a', I_i)$ for $a' \in \{a_i, a_i + \tilde{\gamma}_t\}$
- 11: $\mathcal{U}_i^{\text{tx}}, \mathcal{U}_i^{\text{wt}} \leftarrow$ Equation (24); $D_i(t) \leftarrow \arg \max\{\mathcal{U}_i^{\text{tx}}, \mathcal{U}_i^{\text{wt}}, 0\}$
- 12: **if** $D_i(t) = \text{SKIP}$ **then** $\text{alive}_i \leftarrow \text{FALSE}$
- 13: **end if**
- 14: **end for**
- 15: transmit every unit with $D_i(t) = \text{TRANSMIT}$; $m_i += 1, r_i += 1$
- 16: $a_i += \gamma_t$ for those units; set $\delta_i \leftarrow 1$ where the CRC now passes
- 17: **if** no unit remains open **then break**
- 18: **end for**
- 19: **return** $\{\delta_i\}, \{m_i\}$

5. Simulation Setup

Verifying an importance estimator requires a source whose ground-truth task relevance is known. We therefore generate a synthetic vehicle-to-everything (V2X) situational-awareness corpus in which each message consists of $N = 10$ semantic units: one unit for each of seven single-instance slot types (object class, action, location, coarse dynamics, coarse distance, exact speed, and sequence identifier) plus three ambient context units. A deterministic rule maps the first five slots to one of four criticality classes (informational, advisory, warning, emergency), so that the exact speed, sequence identifier, and ambient context units are irrelevant to the task by construction. This design makes it possible to verify that the estimated importance (12) recovers the true task relevance rather than merely being asserted. The corpus contains 24,000 training, 4000 validation, and 20,000 test messages.

A source with known relevance is, at the same time, a favorable setting for an importance-aware method, because a part of every message can be dropped at no cost to the task. The second source is therefore a real, public dataset in which nothing of the kind is imposed: the Adult census-record dataset of the UCI Machine Learning Repository, taken from the PMLB benchmark collection, with 48,842 records each containing 14 attribute fields (age, work class, education, marital status, occupation, relationship, capital gain and loss, hours per week, native country, and others) and a binary label indicating whether the annual income exceeds a threshold. Each record is treated as a message of $N = 14$ semantic units, one per field. The six continuous fields are quantized into eight quantile bins, so that every unit is a categorical token drawn from a vocabulary of 135 symbols. The relevance of the fields to the label is unknown, uneven, and graded, several fields are partly redundant with one another, and no field is irrelevant by design. The same 24,000/4000/20,000 split is used. Every model is trained from scratch on this source, and nothing else in the framework changes.

The semantic encoder is a two-layer, four-head Transformer with model dimension 64, feed-forward dimension 128, and dropout 0.1, trained for 18 epochs with the AdamW

optimizer (learning rate 2×10^{-3} , weight decay 10^{-2} , batch size 256), a cosine learning rate schedule, gradient-norm clipping at 1, the cross-entropy loss on the task label, and random unit masking at rate 0.15. The channel predictor is a single-layer LSTM with 64 hidden units followed by a two-layer perceptron head with 64 hidden units producing $H_{\max} = 6$ outputs, trained for 15 epochs with AdamW (learning rate 2×10^{-3} , weight decay 10^{-4} , batch size 512), a cosine schedule, gradient-norm clipping at 1, and the Huber loss with unit transition point, on 2.44×10^5 windows generated at six normalized Doppler frequencies and four average SNRs. The input features are standardized with the training-set mean and standard deviation. Every learned component is trained with three different random seeds, and the seed-to-seed spread of the principal results is reported in Section 6.

Four reference predictors are used. The outdated-CSI predictor is $\hat{\gamma}_{t+h} = \tilde{\gamma}_t$. The linear predictor of order 8, “AR(8)”, is a direct multi-horizon least-squares filter fitted on the training windows. The Wiener predictor is the linear MMSE predictor of γ_{t+h} from the L_w past noisy observations, built analytically from the second-order statistics of the process: for Rayleigh fading the SNR autocovariance is $\bar{\gamma}^2 J_0^2(2\pi f_D T_s k)$, and the multiplicative estimation error of (3) enters through the moments of the log-normal factor. This predictor is given the exact Doppler frequency and average SNR of the trace, which the LSTM is not, so it is a genie-aided model-based reference in the spirit of [42]. Finally, an LSTM with the estimated SNR as its only input, but otherwise identical, isolates the contribution of the feature transformations in (14).

All parameters are listed in Table 2. The slot duration $T_s = 0.125$ ms corresponds to a New Radio mini-slot at 120 kHz subcarrier spacing. At a carrier frequency of 3.5 GHz, the nominal normalized Doppler $f_D T_s = 0.05$ corresponds to a relative speed of about 123 km/h, and the Doppler sweep spans roughly 25–370 km/h. The deadline $K = 8$ slots equals 1 ms, a representative V2X URLLC budget. Each simulated trace is 55 slots long: the first 16 slots fill the observation window of the predictor, the next 24 slots let the run-time error tracker (26) accumulate matured forecasts before the message arrives, and the message occupies the following $K = 8$ slots.

All schemes are evaluated on the same channel realizations and, through common random numbers, on the same decoding outcomes: the CRC of a block succeeds if a uniform variate drawn once per unit and slot exceeds the block error probability (6), and the same variates are used for every scheme. Differences between schemes are therefore paired, and the confidence intervals reported for a difference are computed from the per-message differences rather than from the two marginal distributions. All confidence intervals are 95% normal intervals over the 2×10^4 test messages.

Five schemes form the main comparison:

- **B1 (No HARQ):** a single transmission per unit, no retransmission.
- **B2 (Conventional HARQ):** every erroneous unit is retransmitted immediately, up to B_H times, irrespective of importance.
- **B3 (Semantic HARQ):** immediate retransmission, but with the importance-tiered budget (13). This is the behavior of importance-ranked semantic HARQ [15].
- **B4 (Semantic + channel-aware HARQ):** the optimal-stopping controller of Proposition 1, using the current channel estimate and the marginal fading statistics but no forecast. It is optimal under the independent-marginal approximation and is the natural statistics-only reference for the proposed scheme.
- **B5 (Proposed PSA-HARQ):** as B4, with the continuation values refined by the predicted trajectory through (23), using the design-time σ_h unless stated otherwise.

Three further variants appear where they are informative. **B4c** is the correlation-aware statistics-only controller of Section 4.5, which knows the Doppler frequency. **B5c** is the proposed controller with the tail of the trajectory recursion (21) taken from the

correlation-aware table. **B5 (run-time $\hat{\sigma}$)** is the proposed controller with the confidence weight computed from (26).

6. Results and Discussion

Figure 2a reports the estimated importance of each slot type of the V2X corpus. The five task-critical slots receive mean importance between 0.50 and 0.73, whereas the exact speed, sequence identifier, and ambient context slots all receive below 0.05, a group-mean ratio of roughly 19 that was never supplied to the estimator. The attention and task-relevance components are strongly but not perfectly correlated (Pearson $r = 0.78$), which justifies combining them rather than using either alone. With the thresholds of Table 2, 20.8% of the units fall in the high tier, 19.1% in the medium tier, and 60.2% in the low tier (Figure 2b). The degenerate case of the normalization, in which all task-relevance scores of a message coincide, occurs in 0.01% of the test messages. The task decoder attains 98.6% accuracy when all units are delivered, which upper-bounds every accuracy reported below for this corpus.

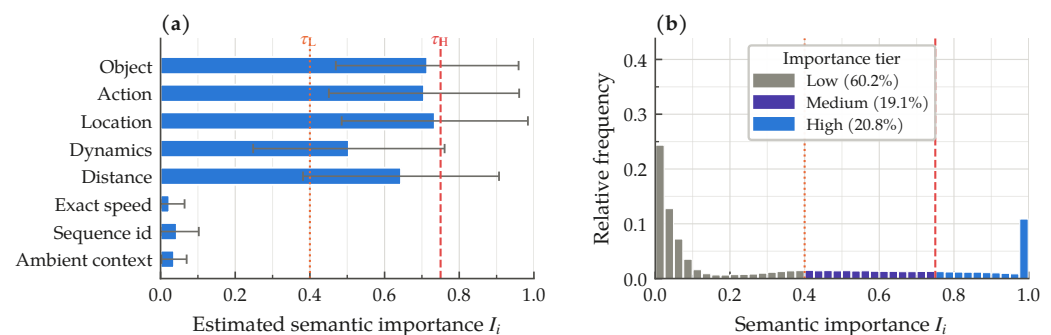


Figure 2. Semantic importance estimation on the V2X corpus. (a) Mean estimated importance per slot type, with one standard deviation. The dashed and dotted lines mark the thresholds τ_H and τ_L . (b) Distribution of I_i over the test set and the induced importance tiers.

Figure 3a superimposes the predicted and true SNR over a representative trace at $f_D T_s = 0.05$: the LSTM tracks the fade structure closely, whereas the outdated-CSI predictor is visibly lagged, which is precisely the behavior that makes immediate retransmission wasteful. Over the mobility sweep (Figure 3b), the LSTM achieves a normalized mean square error (NMSE) of 0.019 at $f_D T_s = 0.01$ and 0.225 at 0.15, improving on the AR(8) predictor by a factor of 1.3–2.8 and on outdated CSI by a factor of 1.7–3.1 across the sweep. The genie-aided Wiener predictor, which is given the exact Doppler frequency and average SNR, is the strongest of the model-based references: it is marginally better than the LSTM at the two lowest Doppler values (0.016 against 0.019 at 0.01) and marginally worse from $f_D T_s = 0.05$ upward (0.232 against 0.225 at 0.15), so a single LSTM that is never told the Doppler frequency matches, over the whole sweep, a linear MMSE predictor. The LSTM fed with the estimated SNR only is consistently worse than the four-feature LSTM (0.070 against 0.060 at $f_D T_s = 0.05$, 0.119 against 0.093 at 0.075), which confirms that the fixed nonlinear re-expressions in (14), although they add no information, do help the network. Figure 3c shows the degradation of the prediction NMSE with the horizon. The ranking of the predictors is preserved at every horizon, and the Wiener predictor, whose linear-scale MMSE criterion favors long horizons, overtakes the LSTM in NMSE from three slots ahead while remaining worse in the decibel-scale error that the controller actually uses. On the held-out validation set the per-horizon root mean square error σ_h that the controller consumes rises from 2.68 dB at one slot to 4.96 dB at six slots, against a fading spread of 5.57 dB. The forecast is therefore highly informative at short horizons and only mildly

informative at the end of the window, which is the situation for which the shrinkage estimator (23) was introduced and which it handles by reducing ρ from 0.75 at $H = 1$ to 0.64 at $H = 6$.

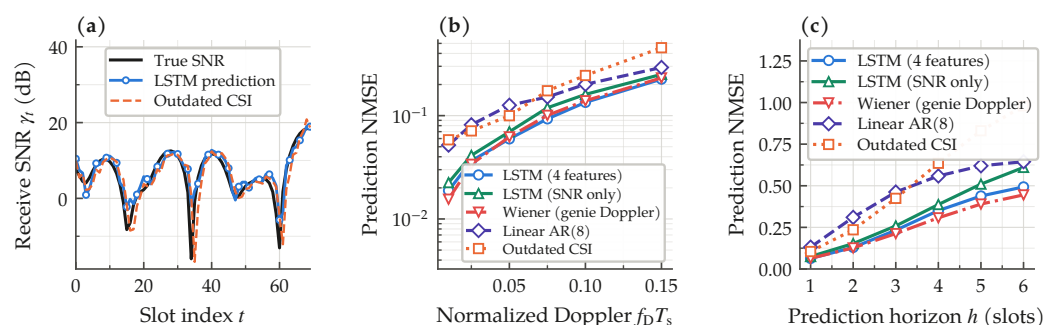


Figure 3. Channel prediction. (a) True and predicted receive SNR over a representative trace at $f_D T_s = 0.05$, $\bar{\gamma} = 12$ dB. (b) One-step prediction NMSE versus normalized Doppler frequency for the four-feature LSTM, the SNR-only LSTM, the genie-aided Wiener predictor, the linear AR(8) predictor, and outdated CSI. (c) Prediction NMSE versus horizon at $f_D T_s = 0.05$.

Figure 4 compares the five schemes over the SNR sweep, and Table 3 lists the values at three representative operating points. The single-transmission baseline B1 is included as a reference floor: it fixes $C_{tx} = 1$ and shows how much semantic content the channel destroys when no reliability mechanism is present at all, which at 8 dB is 48% of the importance-weighted content. Beyond that, three observations stand out.

First, semantic quality alone is a misleading yardstick. Conventional HARQ (B2) attains the highest Q_s at every SNR, which is unsurprising because it retransmits everything. However, its advantage over PSA-HARQ shrinks from 7.7 to 2.9 percentage points between 4 and 8 dB and falls below one percentage point at and above 18 dB, and it is bought at up to 2.8 blocks per unit.

Second, on the metric that actually measures effectiveness, task-inference accuracy, which does not depend on I_i , the proposed scheme overtakes conventional HARQ throughout the 6–12 dB range in which retransmission is both frequent and decisive. At $\bar{\gamma} = 8$ dB, PSA-HARQ reaches 93.0% against 91.7% for conventional HARQ, a paired difference of 1.3 percentage points with a 95% confidence interval of ± 0.4 percentage points. At 6 dB the difference is 0.9 ± 0.5 percentage points, and at 10 dB it is 0.5 ± 0.3 percentage points. Outside that window the two coincide to within a few tenths of a percentage point, conventional HARQ being marginally ahead at 4 dB and at 16 dB and above, where it simply spends far more. The mechanism is that anticipated in Section 1: conventional HARQ exhausts its budget by retransmitting immediately into the fade that caused the failure, so that its attempts on the units that matter are frequently wasted, whereas PSA-HARQ postpones those attempts until the predicted channel can carry them.

Third, the benefit of prediction is concentrated where the channel is the binding constraint. Between B4 and B5 the paired accuracy difference is 2.4 ± 0.3 , 2.0 ± 0.2 , 1.0 ± 0.2 , 0.4 ± 0.1 , and 0.2 ± 0.1 percentage points at 4, 6, 8, 10, and 12 dB, statistically significant at every one of these points, and below 0.1 percentage points above 14 dB. This is a physically sensible outcome: at high SNR almost every first attempt succeeds and there is nothing left to schedule. It is also the operationally relevant regime rather than a corner case: a URLLC link is dimensioned so that the target reliability is met at the edge of its coverage and at the deepest fades it must tolerate, which is precisely where retransmissions are frequent and where Figure 4 places the gain.

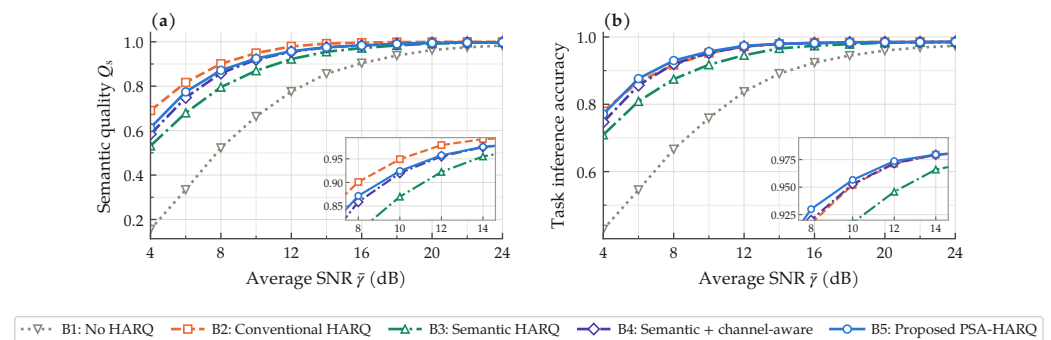


Figure 4. Delivered quality versus average SNR at $f_D T_s = 0.05$. (a) Importance-weighted semantic quality Q_s . (b) Task-inference accuracy. Insets magnify the 8–14 dB region.

Table 3. Performance at three representative average SNRs ($f_D T_s = 0.05$, 2×10^4 messages per point, common random numbers). Q_s : semantic quality; Acc: task-inference accuracy; C_{tx} : transmitted blocks per semantic unit; $\mathcal{E}$: transmit energy per message in mJ. Bold marks the best value among the retransmitting schemes B2–B5; the no-HARQ floor B1 attains $C_{tx} = 1$ by definition. Paired differences with confidence intervals are given in the text and in Table 4.

Scheme	$\bar{\gamma} = 6$ dB				$\bar{\gamma} = 8$ dB				$\bar{\gamma} = 12$ dB			
	Q_s	Acc	C_{tx}	$\mathcal{E}$	Q_s	Acc	C_{tx}	$\mathcal{E}$	Q_s	Acc	C_{tx}	$\mathcal{E}$
B1	0.334	0.545	1.000	0.250	0.522	0.666	1.000	0.250	0.776	0.837	1.000	0.250
B2	0.817	0.867	2.335	0.584	0.901	0.917	1.907	0.477	0.979	0.971	1.387	0.347
B3	0.680	0.809	1.404	0.351	0.796	0.875	1.280	0.320	0.922	0.946	1.123	0.281
B4	0.749	0.857	1.236	0.309	0.859	0.920	1.175	0.294	0.955	0.972	1.087	0.272
B5	0.774	0.876	1.267	0.317	0.872	0.930	1.193	0.298	0.957	0.974	1.089	0.272

Table 4 isolates the part of the B5–B4 gain that the independent-marginal approximation of Proposition 1 concedes. The correlation-aware controller B4c of Section 4.5, which is given the exact Doppler frequency, improves on B4 by 0.4 ± 0.4 percentage points of accuracy at 4 and 6 dB and by a statistically insignificant amount at 8 dB and above, while spending slightly more ($C_{tx} = 1.288$ against 1.236 at 6 dB). On the quality–cost frontier at 8 dB the frontier of B4c lies below that of B4 over most of the cost range, its best point being $Q_s = 0.861$ at $C_{tx} = 1.217$ against 0.865 at 1.195 for B4. The reason is that the first-order Markov model, which matches the one-slot correlation $J_0(2\pi f_D T_s) = 0.976$ exactly, overstates the correlation at longer lags: it predicts a six-slot correlation of $0.976^6 = 0.86$ where the Clarke process has $J_0(2\pi \cdot 0.05 \cdot 6) = 0.29$. A controller that plans several slots ahead with this model is therefore too confident that a good slot will persist and too pessimistic that a deep fade will end, and the two errors do not cancel. The independent-marginal table, which makes no claim about persistence at all, turns out to be the more robust statistics-only choice for a controller that plans over a deadline of eight slots.

The predictive gain survives the change of reference. Against B4c, the proposed controller with the correlation-aware tail (B5c) gains 2.1 ± 0.3 , 1.3 ± 0.3 , 0.8 ± 0.2 , 0.4 ± 0.1 , and 0.2 ± 0.1 percentage points at 4 to 12 dB, i.e., between 65% and 87% of the gain measured against B4 at 4 to 8 dB and all of it at 10 and 12 dB, and over the central part of the 8 dB frontier the paired difference B5c–B4c is positive and significant at every λ_c (up to 2.2 ± 0.3 percentage points). B5c is never better than B5 and stays within 0.3 percentage points of it at every SNR, which shows that the forecast, not the tail table, is what the controller relies on inside the deadline. In summary, the optimality claim of Proposition 1 is restricted to the independent-marginal approximation, but the approximation costs the statistics-only controller nothing measurable in the regime that matters, and the advantage of anticipating the channel with a learned predictor is not an artifact of it.

Table 4. Effect of the independent-marginal approximation ($f_D T_s = 0.05$, common random numbers). B4 and B5 use the marginal-law table of Proposition 1; B4c and B5c use the correlation-aware table of Section 4.5, which is given the exact Doppler frequency. Acc: task-inference accuracy; C_{tx} : blocks per unit. The right-hand columns give paired accuracy differences in percentage points with 95% confidence intervals.

$\bar{\gamma}$ (dB)	B4		B4c		B5		B5c		Paired Difference (pp)		
	Acc	C_{tx}	Acc	C_{tx}	Acc	C_{tx}	Acc	C_{tx}	B4c—B4	B5—B4	B5c—B4c
4	0.747	1.29	0.751	1.32	0.771	1.32	0.771	1.31	$+0.4 \pm 0.4$	$+2.4 \pm 0.3$	$+2.1 \pm 0.3$
6	0.857	1.24	0.860	1.29	0.876	1.27	0.873	1.27	$+0.4 \pm 0.3$	$+2.0 \pm 0.2$	$+1.3 \pm 0.3$
8	0.920	1.17	0.921	1.21	0.930	1.19	0.928	1.20	$+0.1 \pm 0.2$	$+1.0 \pm 0.2$	$+0.8 \pm 0.2$
10	0.953	1.13	0.953	1.15	0.957	1.13	0.957	1.14	-0.0 ± 0.2	$+0.4 \pm 0.1$	$+0.4 \pm 0.1$
12	0.972	1.09	0.971	1.09	0.974	1.09	0.973	1.09	-0.1 ± 0.1	$+0.2 \pm 0.1$	$+0.2 \pm 0.1$

Figure 5 reports the resource side of the same experiments. At 8 dB, conventional HARQ transmits 1.91 blocks per semantic unit while PSA-HARQ transmits 1.19, a reduction of 37%. At 6 dB the reduction is 46%, and at 4 dB it reaches 53%. Because energy is proportional to the number of transmitted blocks, the same factors apply to transmit energy: 0.298 mJ against 0.477 mJ per message at 8 dB.

Figure 6 presents the latency and energy trade-offs at a fixed average SNR of 8 dB, so that every point of every curve corresponds to the same channel condition and the curves differ only through the policy parameter that is swept: the retransmission limit for B2, the low-importance threshold for B3, and the delay weight λ_ℓ (panel a) or the cost weight λ_c (panel b) for B4 and B5. Panel (b) is the clearest summary of the contribution: to reach a semantic quality of 0.87 at 8 dB, conventional HARQ needs 1.83 blocks per unit and about 0.46 mJ per message, whereas PSA-HARQ needs 1.19 blocks per unit and about 0.30 mJ, a 35% energy saving at equal delivered quality and equal channel. Measured in task accuracy the picture is the same: an accuracy of 92% costs conventional HARQ 0.48 mJ and PSA-HARQ 0.31 mJ. Panel (a) shows the price that is paid for this. The message latency is now defined by the slot at which the last unit is delivered or terminated (Section 3), so that slots spent waiting are counted in full. Under this definition, the average latency of PSA-HARQ at its nominal operating point is 0.33 ms at 8 dB against 0.25 ms for conventional HARQ, and 0.62 ms against 0.38 ms at 4 dB. Sweeping λ_ℓ moves PSA-HARQ along its own curve between 0.35 ms and 0.31 ms with a loss of quality only at the largest weight, so the delay is a design choice that can be traded without retraining anything. Two remarks are in order. First, all schemes respect the 1 ms deadline by construction, so the additional latency is consumed inside the budget the application has already granted. Second, the latency is a property of the waiting strategy and not of the predictor: B4 and B5 have nearly identical latencies at every SNR, and what the forecast changes is which slots are chosen, not how many.

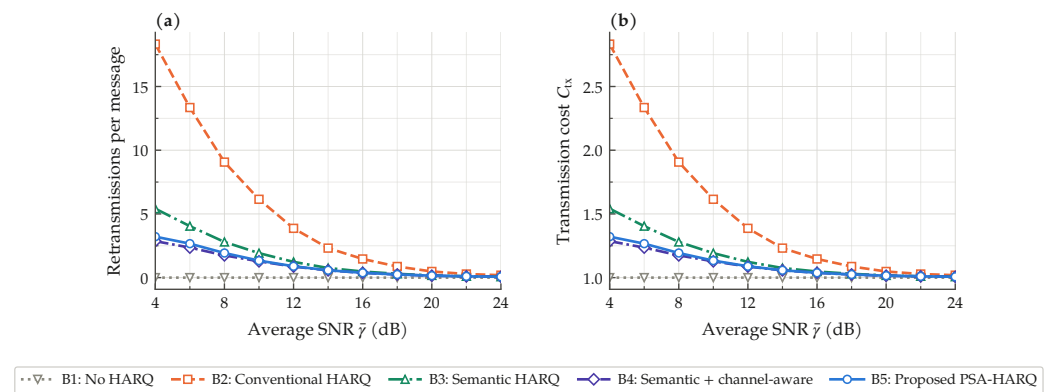


Figure 5. Retransmission efficiency versus average SNR at $f_D T_s = 0.05$. (a) Number of retransmitted blocks per message. (b) Transmission cost C_{tx} in transmitted blocks per semantic unit. The value 1.0 corresponds to a single transmission of every unit.

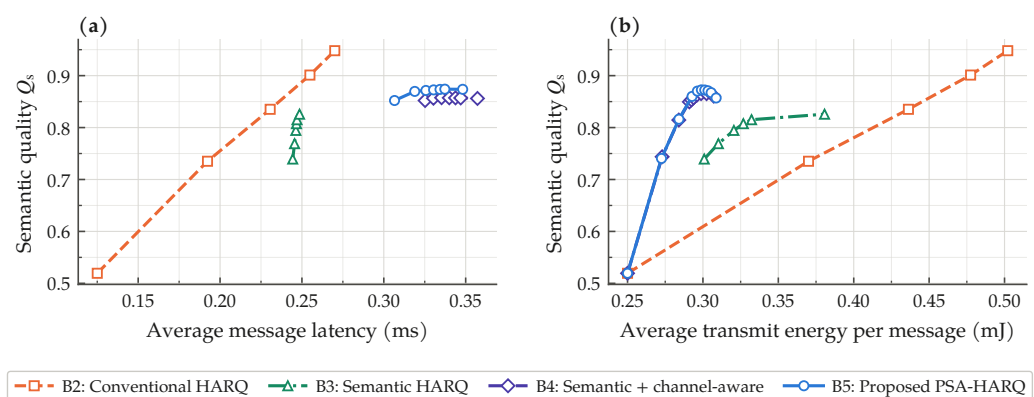


Figure 6. Resource–quality trade-offs at a fixed average SNR of 8 dB and $f_D T_s = 0.05$, traced by sweeping a policy parameter of each family: the retransmission limit (B2), the low-importance threshold (B3), the delay weight λ_ℓ (B4 and B5, panel a), and the cost weight λ_c (B4 and B5, panel b). (a) Semantic quality versus average message latency. (b) Semantic quality versus transmit energy per message.

Comparing schemes at one operating point cannot distinguish a policy that decides better from one that simply spends more. Figure 7 therefore traces the achievable region of each family at a fixed SNR by sweeping its own control knob: the retransmission limit for conventional HARQ, the low-importance threshold τ_L for semantic HARQ, and the cost weight λ_c for B4 and B5. The upper row reports the region in semantic quality and the lower row in task accuracy. Task accuracy does not depend on the importance scores that drive the controllers, so it is the metric on which an importance-aware controller and a conventional one can be compared without bias.

The separation between families is large. At $\bar{\gamma} = 8$ dB, conventional HARQ needs $C_{tx} = 1.91$ to reach $Q_s = 0.90$ and an accuracy of 91.9%, while both utility-driven schemes exceed $Q_s = 0.86$ and 92.5% accuracy at $C_{tx} < 1.20$. The semantic-only scheme B3 lies between them. More importantly, the insets show that the frontier of B5 lies above that of B4 over the useful cost range in both metrics, and that its maximum, $Q_s = 0.873$ at $C_{tx} = 1.20$, exceeds the best value attainable at any cost by B4, namely 0.865. Because the two families are swept over the same values of λ_c on the same channel realizations, their points can be paired. The accuracy difference B5–B4 is 1.0 ± 0.2 , 1.2 ± 0.2 , and 0.7 ± 0.2 percentage points at the three central values of λ_c , and the quality difference at the same points is 0.015 ± 0.002 , 0.017 ± 0.001 , and 0.011 ± 0.001 . The differences vanish, as they must, at the two ends of the sweep where both controllers either transmit everything or

nothing. Channel prediction therefore enlarges the achievable region rather than merely repositioning the operating point within it. At $\bar{\gamma} = 14$ dB, the two frontiers nearly coincide, consistent with the vanishing prediction gain observed in Figure 4.

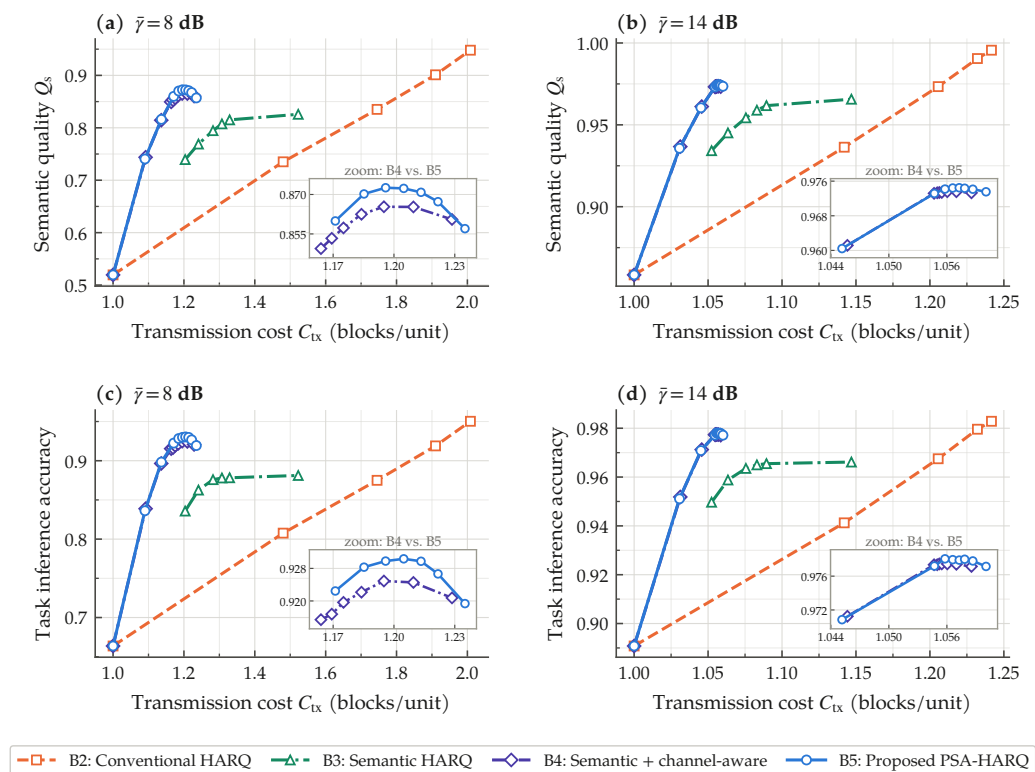


Figure 7. Achievable regions of each scheme family at fixed SNR, obtained by sweeping the retransmission limit (B2), the low-importance threshold (B3), and the cost weight λ_c (B4, B5). (a,b) Semantic quality versus transmission cost at $\bar{\gamma} = 8$ dB and 14 dB. (c,d) Task-inference accuracy versus transmission cost at the same SNRs. The insets magnify the two utility-driven families.

The ablation isolates the two ingredients. Because the four variants naturally settle at different transmission costs, they are compared at a common cost obtained by interpolating along each variant's own frontier. Table 5 reports the result, and Figure 8a visualizes it.

At $\bar{\gamma} = 8$ dB and a matched cost of $C_{tx} = 1.20$ blocks per unit, conventional HARQ attains a task accuracy of 71.9%. Adding semantic importance alone raises this to 92.3%, adding channel prediction alone raises it to 78.5%, and the two together reach 92.9%. Semantic importance is therefore responsible for the larger share of the gain, because it redirects a fixed retransmission budget away from units that cannot influence the decision and toward the units that determine it. Channel prediction contributes a smaller but consistent additional gain, 1.0, 0.6, and 0.1 percentage points over the semantic-only variant at 6, 8, and 12 dB (the two variants are evaluated on the same channel realizations, and the paired confidence intervals of the corresponding B5–B4 differences in Table 4 are about ± 0.2 percentage points at 6 and 8 dB), and the two ingredients are complementary rather than redundant, since neither alone reaches the accuracy of the combined scheme at the matched cost. The prediction-only variant is instructive in its own right: it clearly improves on conventional HARQ, confirming that anticipating the channel is valuable independently of semantics, but it cannot reach low transmission costs because, without importance information, it has no principled reason ever to abandon a unit.

Table 5. Cost-matched ablation study at $f_D T_s = 0.05$. Each variant is evaluated at the same transmission cost C_{tx} , obtained by interpolating along its own quality–cost frontier, so that the comparison isolates decision quality from resource expenditure. The matched operating points are $C_{tx} = 1.28$ at 6 dB, $C_{tx} = 1.20$ at 8 dB, $C_{tx} = 1.09$ at 12 dB. Acc: task-inference accuracy with its 95% confidence interval.

Variant	Sem. I_i	Pred. $\hat{\gamma}$	$\bar{\gamma} = 6 \text{ dB}$		$\bar{\gamma} = 8 \text{ dB}$		$\bar{\gamma} = 12 \text{ dB}$	
			Q_s	Acc	Q_s	Acc	Q_s	Acc
A1: neither	—	—	0.454	0.625 ± 0.007	0.603	0.719 ± 0.006	0.824	0.867 ± 0.005
A2: semantic only	✓	—	0.763	0.865 ± 0.005	0.862	0.923 ± 0.004	0.956	0.972 ± 0.002
A3: prediction only	—	✓	0.600	0.722 ± 0.006	0.698	0.785 ± 0.005	0.866	0.895 ± 0.004
A4: both (proposed)	✓	✓	0.777	0.875 ± 0.005	0.870	0.929 ± 0.004	0.958	0.973 ± 0.002

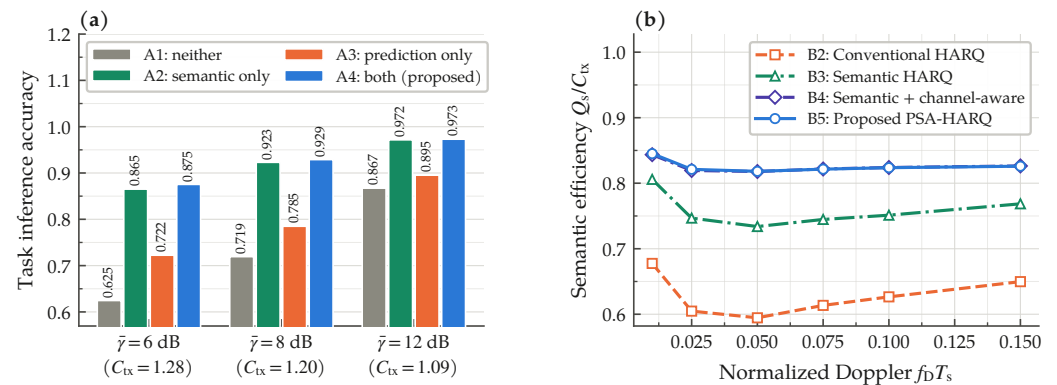


Figure 8. (a) Cost-matched ablation: task-inference accuracy of the four variants, each evaluated at the common transmission cost indicated below the corresponding SNR. (b) Semantic efficiency Q_s/C_{tx} versus normalized Doppler frequency at $\bar{\gamma} = 10$ dB.

Figure 8b sweeps the normalized Doppler frequency from 0.01 to 0.15, i.e., relative speeds from roughly 25 to 370 km/h, at $\bar{\gamma} = 10$ dB. The semantic efficiency Q_s/C_{tx} of every retransmitting scheme falls as mobility increases and then partially recovers, the minimum occurring where the coherence time is comparable to the deadline. The ordering of the four HARQ schemes is preserved throughout, and the advantage of the utility-driven controllers over conventional HARQ grows from 25% at $f_D T_s = 0.01$ to a maximum of 38% at 0.05, and is still 27% at 0.15. B4 and B5 remain close on this aggregate metric because, as Figure 7 showed, they occupy different points on closely spaced frontiers. The separation between them is visible in delivered quality at matched cost, not in a ratio that rewards spending less.

The decay rate κ of (23) is the one parameter of the controller that is not a Lagrange multiplier, and Table 6 and Figure 9 report its influence at 6 and 8 dB. Two features are visible. First, trusting the forecast fully ($\kappa = 0$) is worse than shrinking it: the accuracy drops from 87.6% to 86.4% at 6 dB and from 93.0% to 92.1% at 8 dB, although even the fully trusted forecast still improves on the statistics-only controller (85.7% and 92.0%). Second, the dependence on κ is flat over more than a decade: every value between 0.03 and 0.2 lies within 0.7 percentage points of the best, and the nominal value 0.105, which was chosen a priori so that a 1 dB forecast error retains 90% confidence, is within 0.4 percentage points of the best value found. The run-time error estimate (26) reproduces the design-time curve to within 0.1 percentage points at every κ , which confirms that in the nominal condition the tracker converges to the design-time accuracy and is transparent.

Table 6. Sensitivity to the confidence-decay rate κ at $f_D T_s = 0.05$. Task-inference accuracy (transmission cost in parentheses) of the proposed controller with the design-time σ_h and with the run-time estimate (26); $\kappa = 0$ trusts the forecast fully and $\kappa \rightarrow \infty$ recovers B4, whose values are given in the last row. The nominal value is $\kappa = 0.105$.

κ	$\bar{\gamma} = 6 \text{ dB}$		$\bar{\gamma} = 8 \text{ dB}$	
	Design-Time σ_h	Run-Time $\hat{\sigma}_h$	Design-Time σ_h	Run-Time $\hat{\sigma}_h$
0	0.864 (1.25)	0.864 (1.25)	0.921 (1.19)	0.921 (1.19)
0.03	0.880 (1.28)	0.880 (1.28)	0.931 (1.20)	0.931 (1.20)
0.06	0.878 (1.27)	0.879 (1.27)	0.931 (1.20)	0.931 (1.20)
0.10536 (nominal)	0.876 (1.27)	0.877 (1.27)	0.930 (1.19)	0.930 (1.19)
0.2	0.873 (1.26)	0.874 (1.26)	0.929 (1.19)	0.928 (1.19)
0.4	0.867 (1.25)	0.868 (1.25)	0.926 (1.18)	0.927 (1.18)
1	0.859 (1.24)	0.861 (1.24)	0.922 (1.18)	0.923 (1.18)
∞ (B4)	0.857 (1.24)		0.920 (1.17)	

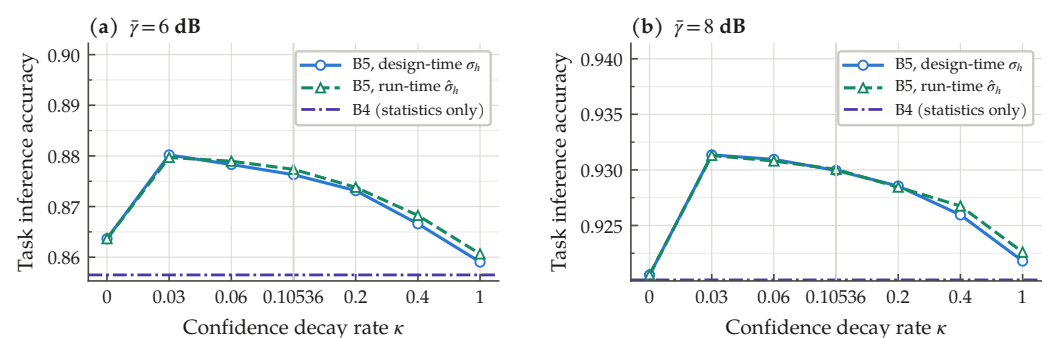


Figure 9. Task-inference accuracy of the proposed controller versus the confidence-decay rate κ with the design-time and the run-time prediction error estimates, at $f_D T_s = 0.05$. The horizontal line is the statistics-only controller B4, and the vertical line marks the nominal value of κ . (a) $\bar{\gamma} = 6 \text{ dB}$. (b) $\bar{\gamma} = 8 \text{ dB}$.

Proposition 3(i) states what the controller becomes when the forecast is worthless, and Table 7 examines whether the deployed system gets there. A second LSTM is trained on traces at $f_D T_s = 0.01$ only, where prediction is easy, so that its design-time error is $\sigma_1 = 0.85 \text{ dB}$ and its design-time confidence is $\rho = 0.91$. Deployed at $f_D T_s = 0.05$ its realized one-slot error is 4.6 dB, and at 0.15 it is 6.0 dB, larger than the fading spread itself. With the design-time σ_h the controller keeps trusting this predictor and pays for it: its accuracy falls to 91.2% at $f_D T_s = 0.05$ and 94.6% at 0.15, below the statistics-only controller (92.0% and 95.8%). With the run-time estimate the tracker measures 4.3 dB and 5.7 dB within the 24-slot history that precedes each message, the confidence falls below 0.64 and 0.55, and the accuracy recovers to 91.9% and 95.4%, within 0.1 and 0.5 percentage points of the fallback. The residual gap is the price of the forgetting factor $\beta = 0.1$, which needs a few tens of slots to converge. A larger β closes it faster at the cost of a noisier estimate. At $f_D T_s = 0.01$, where the mismatched predictor is in fact the better one, both variants exploit it fully. The nominal predictor is unaffected by the tracker at every Doppler, as it should be. The safety property of the design is therefore operational and not only asymptotic: a predictor that fails is detected from the errors it makes on the link, and the controller degrades toward the statistics-only policy rather than toward an arbitrary one.

Table 7. Predictor failure study at $\bar{\gamma} = 8$ dB. The nominal predictor is trained across $f_D T_s \in [0.01, 0.15]$; the mismatched predictor is trained at $f_D T_s = 0.01$ only and deployed at the Doppler of each row. σ_1 : one-slot RMSE in dB, as assumed offline (design-time value), as estimated at run time by (26), and as actually realized on the test traces. Acc: task-inference accuracy; C_{tx} in parentheses.

$f_D T_s$	Predictor	σ_1 Offline	σ_1 Run-Time	σ_1 Realized	B4	B5 (Offline σ)	B5 (Run-Time $\hat{\sigma}$)
0.01	nominal	2.68	1.70	1.32	0.888 (1.10)	0.911 (1.14)	0.914 (1.14)
0.01	mismatched	0.85	1.24	0.88	0.888 (1.10)	0.914 (1.14)	0.913 (1.14)
0.05	nominal	2.68	2.34	2.18	0.920 (1.18)	0.930 (1.20)	0.930 (1.20)
0.05	mismatched	0.85	4.30	4.59	0.920 (1.18)	0.912 (1.21)	0.918 (1.20)
0.15	nominal	2.68	4.12	4.17	0.958 (1.20)	0.958 (1.21)	0.959 (1.21)
0.15	mismatched	0.85	5.72	5.96	0.958 (1.20)	0.946 (1.22)	0.954 (1.22)

Table 8 varies the three heuristic elements of the importance estimator at 8 dB. The mixing weight α has a clear optimum at the nominal value 0.5: using attention alone ($\alpha = 1$) or task relevance alone ($\alpha = 0$) lowers the accuracy of the proposed controller by about three percentage points at practically the same cost, which is the quantitative counterpart of the imperfect correlation between the two components noted above. The within-message normalization is essential: removing only the second normalization changes the results by half a percentage point, but replacing the per-message normalization by a single global scaling collapses the high tier to 2% of the units, because the raw attention and relevance scores of different messages live on different scales, and costs six percentage points of accuracy. The thresholds behave as a monotone trade-off: lowering (τ_L, τ_H) admits more units to the retransmission budget and raises both accuracy and cost, raising them does the opposite, and in every setting the proposed controller stays above semantic HARQ with the same thresholds by four to seven percentage points of accuracy at a lower cost. None of the settings changes the ordering of the schemes.

Table 8. Sensitivity of the importance estimator at $\bar{\gamma} = 8$ dB, $f_D T_s = 0.05$: mixing weight α , normalization variant (“message”: within-message min–max applied to each component and once more to the mixture, as in (12); “single”: no second normalization; “global”: one global scaling over the test set and no within-message normalization), and importance thresholds. Tier shares are the fractions of units in the high/medium/low tiers. Acc: task-inference accuracy; C_{tx} : blocks per unit.

Setting	Tier Shares H/M/L (%)	B3 (Semantic HARQ)		B5 (Proposed)	
		Acc	C_{tx}	Acc	C_{tx}
<i>Mixing weight</i>					
$\alpha = 0$	18/16/66	0.848	1.24	0.896	1.17
$\alpha = 0.25$	19/18/63	0.866	1.26	0.922	1.18
$\alpha = 0.5$ (nominal)	21/19/60	0.875	1.28	0.930	1.19
$\alpha = 0.75$	22/20/59	0.868	1.29	0.924	1.20
$\alpha = 1$	21/20/59	0.843	1.29	0.899	1.20
<i>Normalization</i>					
message (nominal)	21/19/60	0.875	1.28	0.930	1.19
single	18/21/61	0.865	1.26	0.925	1.19
global	2/25/73	0.793	1.14	0.867	1.12

Table 8. Cont.

Setting	Tier Shares H/M/L (%)	B3 (Semantic HARQ)		B5 (Proposed)	
		Acc	C_{tx}	Acc	C_{tx}
<i>Thresholds</i> ($\tau_{\text{L}}, \tau_{\text{H}}$)					
(0.25, 0.60)	—	0.898	1.35	0.945	1.23
(0.40, 0.75) (nominal)	—	0.875	1.28	0.930	1.19
(0.50, 0.85)	—	0.853	1.23	0.911	1.17
(0.30, 0.90)	—	0.856	1.27	0.923	1.21
(0.00, 0.75)	—	0.882	1.52	0.936	1.22

The synthetic corpus makes the true relevance verifiable but also makes part of every message dispensable by design. Figure 10 and Table 9 repeat the main experiment on the real Adult dataset described in Section 5, in which every one of the 14 fields is a real attribute, their relevance is unknown and graded, and the task decoder reaches 85.1% accuracy with every unit delivered. The degenerate case of the normalization occurs in 0.09% of the records. The estimator assigns the highest importance to the fields that are known from the extensive literature on this dataset to carry most of the predictive information (marital status, occupation, relationship, age, hours per week, and education) and the lowest to native country and race. It places 76% of the units in the low tier without any supervision on relevance. The picture on the link is the same as on the V2X corpus, with a smaller absolute spread because the decoder itself is limited to about 85% accuracy: at 4–8 dB the proposed controller matches or exceeds the task accuracy of conventional HARQ (82.9% against 82.6% at 4 dB, 84.0% against 83.7% at 6 dB, and 84.4% against 84.3% at 8 dB) while transmitting 1.12–1.18 blocks per unit instead of 1.91–2.84, and the paired accuracy difference over B4 is positive and significant at every SNR up to 8 dB. On the accuracy–cost frontier at 8 dB (Figure 10c), the proposed controller again reaches the accuracy of conventional HARQ at less than two thirds of its cost. The mechanism therefore does not depend on relevance being imposed by construction. It depends only on the relevance being uneven, which is the normal case.

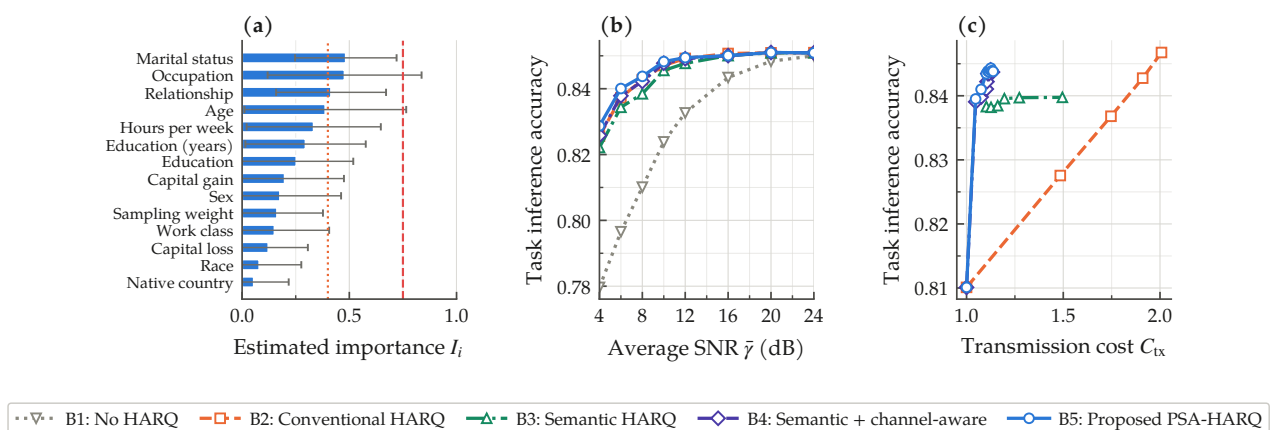


Figure 10. Results on the real adult dataset. (a) Mean estimated importance per field with one standard deviation. (b) Task-inference accuracy versus average SNR at $f_D T_s = 0.05$. (c) Task-inference accuracy versus transmission cost at $\bar{\gamma} = 8$ dB.

Table 9. Results on the real Adult dataset ($N = 14$ units per message, $f_D T_s = 0.05$, 2×10^4 test records, common random numbers). Acc: task-inference accuracy; C_{tx} : blocks per unit; the last two columns give the paired accuracy differences in percentage points with 95% confidence intervals. The task decoder reaches 85.1% accuracy when every unit is delivered.

$\bar{\gamma}$ (dB)	B2		B3		B4		B5		Paired Diff. (pp)	
	Acc	C_{tx}	Acc	C_{tx}	Acc	C_{tx}	Acc	C_{tx}	B5–B4	B5–B2
4	0.826	2.84	0.822	1.31	0.826	1.17	0.829	1.18	$+0.3 \pm 0.2$	$+0.3 \pm 0.3$
6	0.837	2.34	0.834	1.23	0.838	1.14	0.840	1.16	$+0.2 \pm 0.1$	$+0.3 \pm 0.3$
8	0.843	1.91	0.838	1.16	0.842	1.11	0.844	1.12	$+0.2 \pm 0.1$	$+0.1 \pm 0.2$
10	0.847	1.60	0.846	1.11	0.848	1.07	0.848	1.08	$+0.1 \pm 0.1$	$+0.1 \pm 0.2$
12	0.849	1.38	0.848	1.07	0.849	1.05	0.849	1.05	$+0.0 \pm 0.0$	$+0.0 \pm 0.1$
16	0.851	1.14	0.850	1.03	0.850	1.02	0.850	1.02	$+0.0 \pm 0.0$	-0.1 ± 0.1
20	0.851	1.05	0.851	1.01	0.851	1.01	0.851	1.01	$+0.0 \pm 0.0$	$+0.0 \pm 0.0$
24	0.851	1.02	0.851	1.00	0.851	1.00	0.851	1.00	$+0.0 \pm 0.0$	-0.0 ± 0.0

Both the semantic encoder and the LSTM predictor are learned components, so Table 10 repeats the principal comparison with three independently trained pairs of models, each evaluated on its own block of 2×10^4 messages. The seed-to-seed standard deviation of the accuracy of the proposed controller is below 0.5 percentage points at every SNR (and below 0.15 percentage points from 8 dB upward), and the paired B5–B4 difference is positive for every seed at every SNR, ranging from 1.4 to 2.0 percentage points at 6 dB and from 0.7 to 1.2 percentage points at 8 dB. The conclusions drawn above do not depend on the particular training run.

Table 10. Seed-to-seed variability. Task-inference accuracy (mean $\pm$ standard deviation over three independently trained semantic encoders and LSTM predictors, each evaluated on its own 2×10^4 -message channel block) of B2, B4, and B5, and the paired difference B5–B4 in percentage points (mean over seeds, with the range in brackets).

$\bar{\gamma}$ (dB)	B2	B4	B5	B5–B4 (pp)
4	0.797 ± 0.011	0.762 ± 0.006	0.781 ± 0.005	$+1.9 [+1.8, +2.1]$
6	0.875 ± 0.006	0.857 ± 0.002	0.875 ± 0.004	$+1.7 [+1.4, +2.0]$
8	0.926 ± 0.005	0.920 ± 0.002	0.929 ± 0.001	$+0.9 [+0.7, +1.2]$
10	0.957 ± 0.001	0.955 ± 0.001	0.957 ± 0.001	$+0.2 [+0.1, +0.3]$
12	0.974 ± 0.001	0.971 ± 0.001	0.972 ± 0.001	$+0.2 [+0.2, +0.2]$

Table 11 reports the measured execution time of every component on a single core of a general-purpose processor, with batch size one and no accelerator. Two implementations were measured on the same core. The reference implementation is the Python and PyTorch code used for all simulations in this paper. The compiled implementation is a C version of the per-slot receiver processing, built with an optimizing compiler. In every slot it computes the features of the new observation, runs the LSTM, updates the run-time error tracker (26), and takes the transmit, wait, or skip decision for all N units. We checked that the two implementations agree. Over 24,000 randomly drawn controller states, the compiled controller took the same decision as the reference in every case, and its LSTM outputs differ from those of PyTorch by less than 10^{-5} dB.

The transmitter-side computation consists of one encoder pass for the attention mass and one batched pass of N masked inputs for the leave-one-unit-out relevance. It takes about 1.5 ms per message and is performed once, before the first transmission. It therefore does not sit on the retransmission loop, because the importance scores are ready when the message enters the HARQ process and are not recomputed.

On the receiver side, the reference implementation needs 0.19 ms for the LSTM and 0.65 ms for the controller in every slot, about 0.84 ms in total. This is longer than the slot

duration of 0.125 ms, so the reference code does not meet the real-time requirement by itself. Most of this time is interpreter and framework overhead rather than arithmetic. The recursion for $N = 10$ units needs about 1.3×10^3 evaluations of (6) and 1.3×10^3 bilinear table look-ups, and the LSTM is a 64-unit cell unrolled over 16 steps, with about 0.28×10^6 multiply-accumulate operations. The compiled implementation removes this overhead. Its complete per-slot processing takes 35 μ s on average, of which 23 μ s are spent in the LSTM and 10 μ s in the controller. In 10^5 repeated runs of the per-slot processing, 99% of the runs finished within 63 μ s, about half of the slot duration. Fewer than 0.1% of the runs took longer than 0.125 ms, and these delays were caused by preemption by the general-purpose operating system rather than by the computation.

These are measurements on a general-purpose processor. In a modem, the same processing would run on dedicated vector or fixed-function hardware under a real-time operating system, which we expect to be faster still. This expectation is not part of the measured results. The semantic decoder runs once per message and takes 0.42 ms, and the offline value tables take 8 ms and 258 ms and are computed once per average SNR and once per SNR and Doppler value, respectively. The measurements therefore support the assumption of Section 3.3 that the decision of the controller for slot $t + 1$ is available by the end of slot t , provided that the per-slot processing is implemented in compiled code, as it would be in any practical receiver.

Table 11. Measured execution times on a single core of an Intel Xeon 2.1 GHz processor, with batch size one and no accelerator. Reference: the Python and PyTorch implementation used for all simulations. Compiled: a C implementation of the per-slot receiver processing that produces the same decisions as the reference. Transmitter-side items run once per message before the first transmission, and receiver-side items run once per slot.

Component	Where	Reference	Compiled
Semantic encoder forward pass (attention mass A_i)	transmitter, per message	0.53 ms	–
Leave-one-unit-out relevance T_i (N masked inputs, one batched pass)	transmitter, per message	0.96 ms	–
LSTM predictor forward pass	receiver, per slot	0.19 ms	23 μ s
Controller: trajectory recursion (22) for N units, two starting states, $H = 6$	receiver, per slot	0.65 ms	10 μ s
Complete per-slot processing (features, LSTM, error tracker (26), controller)	receiver, per slot	–	35 μ s
Semantic decoder (task decision)	receiver, per message	0.42 ms	–
Offline value table, Proposition 1	once per average SNR	8 ms	–
Offline value table, Section 4.5	once per SNR and Doppler	258 ms	–

7. Conclusions

This article argued that, in semantic communication, the retransmission decision should be made neither at the packet level nor on the basis of the present channel alone, and proposed predictive semantic-aware HARQ as a concrete realization of that argument. Semantic units are individually coded and individually scheduled. Their importance is estimated from a Transformer encoder by combining attention mass with leave-one-unit-out task relevance. An LSTM forecasts the receive SNR over the next $H_{\max}$ slots of the retransmission deadline, and a finite-horizon optimal-stopping controller selects, for every unit and every slot, whether to transmit, wait, or abandon. The predicted trajectory enters the controller through a confidence-weighted shrinkage estimator that is applied to both branches of the decision and whose weight is tracked at run time from the errors the predictor makes on the link. This construction guarantees that the policy degenerates to the statistics-only optimum when the forecast fails and to the exact backward induction over the known trajectory when the forecast is exact.

Across 2×10^4 vehicular situational-awareness messages transmitted over time-varying Rayleigh channels with finite-blocklength coding and Chase combining, the scheme delivered 1.3 percentage points more task-inference accuracy than conventional HARQ at 8 dB while transmitting 37% fewer blocks and reached the same semantic quality at the same channel condition with about 35% less transmit energy. A cost-matched ablation showed that semantic importance and channel prediction contribute separable and complementary gains, with prediction mattering most at low SNR, precisely the regime in which retransmissions are both most frequent and most likely to be wasted. The gain of prediction over the statistics-only controller was confirmed with paired confidence intervals, across three training seeds, against a correlation-aware statistics-only reference, on a real dataset in which the relevance of the message fields is not imposed, and under a deliberately mismatched predictor, where the run-time error tracker returned the controller to within half a percentage point of its fallback. Comparing schemes along their entire quality–cost and accuracy–cost frontiers rather than at a single operating point revealed that channel prediction enlarges the achievable region itself, not merely the position within it.

Taken together, these results make a case that reliability mechanisms for semantic communication should reason about the future channel and not only about the present one, and they provide a controller that does so at a modest computational cost. In a compiled implementation on a single core of a general-purpose processor, the complete per-slot processing of the predictor and the controller took 35 μ s on average, well within the 0.125-ms slot. Several extensions follow naturally from the design. The fading process studied here is frequency-flat and single-antenna, and the framework is agnostic to the source. Frequency-selective and multi-antenna channels enlarge the state of the predictor but leave the temporal structure it exploits intact, and natural traffic such as camera features or sensor streams requires only retraining the semantic encoder. A higher-order or learned model of the channel correlation inside the offline recursion would strengthen the statistics-only baseline further. Semantic packetization can be extended to continuous-valued feature vectors, where importance is graded rather than tiered and partial delivery is meaningful, and the loop can be closed around the semantic encoder itself, so that the encoder learns to produce units whose importance profile is matched to what the HARQ controller can protect. Finally, the per-message delay term of the objective is a natural bridge to age-of-information scheduling across a stream of messages, which would let the scheme decide not only how a message is protected but also when its successor should be released.

Author Contributions: Conceptualization, methodology and software, O.Kaya; validation, O.Kaya, M.A.K., C.E. and O.Karakuş; formal analysis, O.Kaya; investigation, O.Kaya, M.A.K., C.E. and O.K. (Oktay Karakuş); writing-original draft preparation, O.Kaya; review and editing, M.A.K., C.E. and O.Karakuş; visualization, O.Kaya; supervision, O.Karakuş. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The simulation code, the synthetic V2X corpus, and the trained models can be provided by the corresponding author by e-mail upon reasonable request after publication. The Adult dataset is publicly available from the UCI Machine Learning Repository and the PMLB collection.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

6G	Sixth generation
AoI	Age of information
AR	Autoregressive
CRC	Cyclic redundancy check
CSI	Channel state information
HARQ	Hybrid automatic repeat request
JSCC	Joint source–channel coding
LSTM	Long short-term memory
MMSE	Minimum mean square error
NMSE	Normalized mean square error
PSA-HARQ	Predictive semantic-aware HARQ
RSSI	Received signal strength indicator
SNR	Signal-to-noise ratio
URLLC	Ultra-reliable low-latency communication
V2X	Vehicle-to-everything

References

1. Saad, W.; Bennis, M.; Chen, M. A Vision of 6G Wireless Systems: Applications, Trends, Technologies, and Open Research Problems. *IEEE Netw.* **2020**, *34*, 134–142. <https://doi.org/10.1109/MNET.001.1900287>.
2. Tataria, H.; Shafi, M.; Molisch, A.F.; Dohler, M.; Sjöland, H.; Tufvesson, F. 6G Wireless Systems: Vision, Requirements, Challenges, Insights, and Opportunities. *Proc. IEEE* **2021**, *109*, 1166–1199. <https://doi.org/10.1109/JPROC.2021.3061701>.
3. Jiang, W.; Han, B.; Habibi, M.A.; Schotten, H.D. The Road Towards 6G: A Comprehensive Survey. *IEEE Open J. Commun. Soc.* **2021**, *2*, 334–366. <https://doi.org/10.1109/OJCOMS.2021.3057679>.
4. ITU-R. Framework and Overall Objectives of the Future Development of IMT for 2030 and Beyond. In *Recommendation ITU-R M.2160-0*; International Telecommunication Union, Radiocommunication Sector: Geneva, Switzerland, 2023.
5. Xie, H.; Qin, Z.; Li, G.Y.; Juang, B.H. Deep Learning Enabled Semantic Communication Systems. *IEEE Trans. Signal Process.* **2021**, *69*, 2663–2675. <https://doi.org/10.1109/TSP.2021.3071210>.
6. Gündüz, D.; Qin, Z.; Estella Aguerri, I.; Dhillon, H.S.; Yang, Z.; Yener, A.; Wong, K.K.; Chae, C.B. Beyond Transmitting Bits: Context, Semantics, and Task-Oriented Communications. *IEEE J. Sel. Areas Commun.* **2023**, *41*, 5–41. <https://doi.org/10.1109/JSAC.2022.3223408>.
7. Calvanese Strinati, E.; Barbarossa, S. 6G Networks: Beyond Shannon Towards Semantic and Goal-Oriented Communications. *Comput. Netw.* **2021**, *190*, 107930. <https://doi.org/10.1016/j.comnet.2021.107930>.
8. Yang, W.; Du, H.; Liew, Z.Q.; Lim, W.Y.B.; Xiong, Z.; Niyato, D.; Chi, X.; Shen, X.; Miao, C. Semantic Communications for Future Internet: Fundamentals, Applications, and Challenges. *IEEE Commun. Surv. Tutor.* **2023**, *25*, 213–250. <https://doi.org/10.1109/COMST.2022.3223224>.
9. Chase, D. Code Combining—A Maximum-Likelihood Decoding Approach for Combining an Arbitrary Number of Noisy Packets. *IEEE Trans. Commun.* **1985**, *33*, 385–393. <https://doi.org/10.1109/TCOM.1985.1096314>.
10. Makki, B.; Svensson, T.; Zorzi, M. Finite Block-Length Analysis of the Incremental Redundancy HARQ. *IEEE Wirel. Commun. Lett.* **2014**, *3*, 529–532. <https://doi.org/10.1109/LWC.2014.2353059>.
11. Popovski, P.; Stefanović, Č.; Nielsen, J.J.; de Carvalho, E.; Angjelichinoski, M.; Trillingsgaard, K.F.; Bana, A.S. Wireless Access in Ultra-Reliable Low-Latency Communication (URLLC). *IEEE Trans. Commun.* **2019**, *67*, 5783–5801. <https://doi.org/10.1109/TCOMM.2019.2914652>.
12. Jiang, P.; Wen, C.K.; Jin, S.; Li, G.Y. Deep Source-Channel Coding for Sentence Semantic Transmission With HARQ. *IEEE Trans. Commun.* **2022**, *70*, 5225–5240. <https://doi.org/10.1109/TCOMM.2022.3180997>.
13. Zhou, Q.; Li, R.; Zhao, Z.; Xiao, Y.; Zhang, H. Adaptive Bit Rate Control in Semantic Communication With Incremental Knowledge-Based HARQ. *IEEE Open J. Commun. Soc.* **2022**, *3*, 1076–1089. <https://doi.org/10.1109/OJCOMS.2022.3189023>.
14. Gao, H.; Yu, G.; Cai, Y. Adaptive Modulation and Retransmission Scheme for Semantic Communication Systems. *IEEE Trans. Cogn. Commun. Netw.* **2024**, *10*, 150–163. <https://doi.org/10.1109/TCCN.2023.3315386>.
15. Hu, J.; Wang, F.; Xu, W.; Gao, H.; Zhang, P. SemHARQ: Semantic-Aware Hybrid Automatic Repeat Request for Multi-Task Semantic Communications. *IEEE Trans. Wirel. Commun.* **2026**, *25*, 3170–3185. <https://doi.org/10.1109/TWC.2025.3602146>.
16. Jiang, W.; Schotten, H.D. Deep Learning for Fading Channel Prediction. *IEEE Open J. Commun. Soc.* **2020**, *1*, 320–332. <https://doi.org/10.1109/OJCOMS.2020.2982513>.

17. Yuan, J.; Ngo, H.Q.; Matthaiou, M. Machine Learning-Based Channel Prediction in Massive MIMO With Channel Aging. *IEEE Trans. Wirel. Commun.* **2020**, *19*, 2960–2973. <https://doi.org/10.1109/TWC.2020.2969627>.
18. Wu, C.; Yi, X.; Zhu, Y.; Wang, W.; You, L.; Gao, X. Channel Prediction in High-Mobility Massive MIMO: From Spatio-Temporal Autoregression to Deep Learning. *IEEE J. Sel. Areas Commun.* **2021**, *39*, 1915–1930. <https://doi.org/10.1109/JSAC.2021.3078503>.
19. Jiang, H.; Cui, M.; Ng, D.W.K.; Dai, L. Accurate Channel Prediction Based on Transformer: Making Mobility Negligible. *IEEE J. Sel. Areas Commun.* **2022**, *40*, 2717–2732. <https://doi.org/10.1109/JSAC.2022.3191334>.
20. Bourtsoulatz, E.; Kurka, D.B.; Gündüz, D. Deep Joint Source-Channel Coding for Wireless Image Transmission. *IEEE Trans. Cogn. Commun. Netw.* **2019**, *5*, 567–579. <https://doi.org/10.1109/TCCN.2019.2919300>.
21. Luo, X.; Chen, H.H.; Guo, Q. Semantic Communications: Overview, Open Issues, and Future Research Directions. *IEEE Wirel. Commun.* **2022**, *29*, 210–219. <https://doi.org/10.1109/MWC.101.2100269>.
22. Park, J.; Oh, Y.; Kim, Y.; Jeon, Y.S. Vision Transformer-Based Semantic Communications With Importance-Aware Quantization. *IEEE Internet Things J.* **2025**, *12*, 35662–35677. <https://doi.org/10.1109/JIOT.2025.3580597>.
23. Cao, Y.; Wu, Y.; Lian, L.; Tao, M. Importance-Aware Resource Allocations for MIMO Semantic Communication. *Entropy* **2025**, *27*, 605. <https://doi.org/10.3390/e27060605>.
24. Sun, Z.; Ma, S.; Li, S. Task-Oriented Semantic Communication With Importance-Aware Rate Control. *IEEE Commun. Lett.* **2025**, *29*, 1520–1524. <https://doi.org/10.1109/LCOMM.2025.3566092>.
25. Liu, Y.; Dong, C.; Liang, H.; Li, W.; Bao, Z.; Zheng, Z.; Xu, X.; Zhang, P. Semantic-Importance-Aware Reordering-Enhanced Semantic Communication System With OFDM Transmission. *IEEE Internet Things J.* **2025**, *12*, 7938–7954. <https://doi.org/10.1109/JIOT.2025.3529492>.
26. Zhang, J.; Zhang, Y.; Ji, B.; Chen, A.; Liu, A.; Xu, H. Feature-Driven Semantic Communication for Efficient Image Transmission. *Entropy* **2025**, *27*, 369. <https://doi.org/10.3390/e27040369>.
27. Dong, B.; Cao, B.; Zhang, Q. Lightweight Semantic-Aware Communication With Packet Transmission. *IEEE Commun. Lett.* **2025**, *29*, 1569–1573. <https://doi.org/10.1109/LCOMM.2025.3568427>.
28. Cao, H.; Meng, R.; Xu, X.; Han, S.; Zhang, P. Importance-Aware Robust Semantic Transmission for LEO Satellite–Ground Communication. *IEEE Internet Things J.* **2026**, *13*, 9665–9681. <https://doi.org/10.1109/JIOT.2025.3645361>.
29. Zhang, P.; Meng, R.; Xu, X.; Wang, Y.; Huang, Z.; Liu, Y.; Zhang, R.; Liu, Y.; Tong, H.; Song, H.; et al. Towards Semantic-Based Agent Communication Networks: Vision, Technologies, and Challenges. *arXiv* **2026**, arXiv:2603.24328. <https://doi.org/10.48550/arXiv.2603.24328>.
30. Meng, R.; Gao, S.; Xu, B.; Xu, X.; Chen, J.; Ma, N.; Xiao, P.; Zhang, P.; Tafazolli, R. Secure Intellicise Wireless Network: Agentic AI for Coverless Semantic Steganography Communication. *arXiv* **2026**, arXiv:2601.16472. <https://doi.org/10.48550/arXiv.2601.16472>.
31. Polyanskiy, Y.; Poor, H.V.; Verdú, S. Channel Coding Rate in the Finite Blocklength Regime. *IEEE Trans. Inf. Theory* **2010**, *56*, 2307–2359. <https://doi.org/10.1109/TIT.2010.2043769>.
32. Ghanami, F.; Poursheikhali, S.; Shirvanimoghaddam, M. Dynamic NOMA-HARQ in the Finite Blocklength Regime: Performance Analysis and Optimization. *IEEE Commun. Lett.* **2023**, *27*, 1220–1224. <https://doi.org/10.1109/LCOMM.2023.3244609>.
33. Avranas, A.; Kountouris, M.; Ciblât, P. Energy-Latency Tradeoff in Ultra-Reliable Low-Latency Communication With Retransmissions. *IEEE J. Sel. Areas Commun.* **2018**, *36*, 2475–2485. <https://doi.org/10.1109/JSAC.2018.2874143>.
34. Ceran, E.T.; Gündüz, D.; György, A. Average Age of Information With Hybrid ARQ Under a Resource Constraint. *IEEE Trans. Wirel. Commun.* **2019**, *18*, 1900–1913. <https://doi.org/10.1109/TWC.2019.2899303>.
35. Ceran, E.T.; Gündüz, D.; György, A. A Reinforcement Learning Approach to Age of Information in Multi-User Networks With HARQ. *IEEE J. Sel. Areas Commun.* **2021**, *39*, 1412–1426. <https://doi.org/10.1109/JSAC.2021.3065057>.
36. Li, A.; Wu, S.; Jiao, J.; Zhang, N.; Zhang, Q. Age of Information With Hybrid-ARQ: A Unified Explicit Result. *IEEE Trans. Commun.* **2022**, *70*, 7899–7914. <https://doi.org/10.1109/TCOMM.2022.3217574>.
37. Sac, H.; Bacinoglu, T.; Uysal-Biyikoglu, E.; Durisi, G. Age-Optimal Channel Coding Blocklength for an M/G/1 Queue with HARQ. In Proceedings of the 2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Kalamata, Greece, 25–28 June 2018; pp. 1–5. <https://doi.org/10.1109/SPAWC.2018.8445909>.
38. Jiang, P.; Wen, C.K.; Jin, S.; Li, G.Y. Wireless Semantic Communications for Video Conferencing. *IEEE J. Sel. Areas Commun.* **2023**, *41*, 230–244. <https://doi.org/10.1109/JSAC.2022.3221968>.
39. Ni, F.; Li, R.; Zhao, Z.; Zhang, H. Topology Data Analysis-Based Error Detection for Semantic Image Transmission with Incremental Knowledge-Based HARQ. *China Commun.* **2025**, *22*, 235–255. <https://doi.org/10.23919/JCC.fa.2023-0799.202501>.
40. Han, B.; Hu, Y.; Schotten, H.D. Generative Semantic HARQ: Latent-Space Text Retransmission and Combining. *arXiv* **2026**, arXiv:2603.15068.
41. Li, Y.; Wang, X.; Shi, Z.; Fu, Y. Semantic HARQ for Intelligent Transportation Systems: Joint Source-Channel Coding-Powered Reliable Retransmissions. *arXiv* **2025**, arXiv:2504.14615.
42. Duel-Hallen, A.; Hu, S.; Hallen, H. Long-Range Prediction of Fading Signals. *IEEE Signal Process. Mag.* **2000**, *17*, 62–75. <https://doi.org/10.1109/79.841729>.

43. Duel-Hallen, A. Fading Channel Prediction for Mobile Radio Adaptive Transmission Systems. *Proc. IEEE* **2007**, *95*, 2299–2313. <https://doi.org/10.1109/JPROC.2007.904443>.
44. Guan, Z.; Wen, W.; Wu, P.; Wang, C.; Xia, M. Hybrid CNN-Transformer Based Sparse Channel Prediction for High-Mobility OTFS Systems. *IEEE Wirel. Commun. Lett.* **2026**, *15*, 215–219. <https://doi.org/10.1109/LWC.2025.3624081>.
45. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>.
46. Clarke, R.H. A Statistical Theory of Mobile-Radio Reception. *Bell Syst. Tech. J.* **1968**, *47*, 957–1000. <https://doi.org/10.1002/j.1538-7305.1968.tb00069.x>.
47. Baddour, K.E.; Beaulieu, N.C. Autoregressive Modeling for Fading Channel Simulation. *IEEE Trans. Wirel. Commun.* **2005**, *4*, 1650–1662. <https://doi.org/10.1109/TWC.2005.850327>.
48. 3GPP. *Study on Channel Model for Frequencies from 0.5 to 100 GHz*; Technical Report TR 38.901, V18.0.0; 3rd Generation Partnership Project: Sophia Antipolis, France, 2024.
49. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention Is All You Need. In Proceedings of the Advances in Neural Information Processing Systems 30 (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
50. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186. <https://doi.org/10.18653/v1/N19-1423>.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.