Beyond Freshness and Semantics: A Coupon-Collector Framework for Effective Status Updates

Youssef Ahmed[1], Arnob Ghosh[2], Chih-Chun Wang[3], and Ness B. Shroff[1,4]

[1]Department of ECE, The Ohio State University, Columbus, OH, USA
[2]Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ, USA
[3]Elmore Family School of ECE, Purdue University, West Lafayette, IN, USA
[4]Department of CSE, The Ohio State University, Columbus, OH, USA
Emails: ahmed.943@osu.edu, arnob.ghosh@njit.edu, chihw@purdue.edu, shroff.11@osu.edu

***Abstract*—For status update systems operating over unreliable energy-constrained wireless channels, we address Weaver's long-standing *Level-C* question [1]: *do my packets actually improve the plant's behaviour?*. Each fresh sample carries a stochastic expiration time—governed by the plant's instability dynamics—after which the information becomes useless for control. Casting the problem as a coupon-collector variant with expiring coupons, we (i) formulate a two-dimensional average-reward MDP, (ii) prove that the optimal schedule is *doubly thresholded* in the receiver's freshness timer and the sender's stored lifetime, (iii) derive a closed-form policy for deterministic lifetimes, and (iv) design a Structure-Aware Q-learning algorithm (SAQ) that learns the optimal policy without knowing the channel success probability or lifetime distribution. Simulations validate our theoretical predictions: SAQ matches optimal Value Iteration performance while converging significantly faster than baseline Q-learning, and expiration-aware scheduling achieves up to 50% higher reward than age-based baselines by adapting transmissions to state-dependent urgency—thereby delivering Level-C effectiveness under tight resource constraints.**

I. Introduction

Modern cyber-physical systems, such as mobile robots navigating busy spaces or learning agents steering autonomous vehicles, rely on timely state information that must traverse unreliable, energy-constrained wireless links [2], [3]. Sending every update guarantees the controller acts on the freshest data, but quickly drains battery life and clogs shared channels [4]. Skipping updates, however, conserves energy while forcing the controller to rely on stale information, increasing the risk of suboptimal or unsafe decisions. Existing frameworks frequently waste energy transmitting fresh but useless" updates because they cannot identify when local knowledge suffices. They lack a measure of Weaver's Level-C" effectiveness [1], i.e., the ability to determine when a specific sample truly improves decision quality.

In contrast, computation costs have plummeted as edge devices benefit from continual advances in transistor scaling and specialized machine-learning accelerators [5]. Modern hardware has shifted the energy bottleneck decisively: while performing a full neural-network inference on TinyML hardware now consumes as little as $1.9\,\mu$J [6], transmitting a single data packet over a standard radio protocol can consume orders of magnitude more energy. For instance, transmitting just one bit requires roughly $0.015\,\mu$J. This means that sending a standard packet (approx. 1000 bits including overhead) consumes roughly $15\,\mu$J [4]. Hence, a device can execute an entire complex inference algorithm locally for significantly less energy than it takes to wirelessly report a single raw measurement. This disparity establishes communication, (not computation) as the primary bottleneck on system performance and lifetime. Using this computation power, one might do effective inference without relying on fresh information. Therefore, we need an intelligent policy that leverages this abundant computational capacity to filter data locally, transmitting updates only when they measurably improve control performance.

Given this asymmetry, the natural question is: how should a sensor decide when to transmit? Classically, this need for local filtering has been addressed by *Event-Triggered Control (ETC)* [7], which trigger transmissions only when stability is threatened or error thresholds are crossed. However, these methods typically require rigid analytical models of the plant and channel. To enable more flexible, data-driven scheduling, the community has moved beyond the classic network–level objectives of end-to-end delay, throughput, and jitter. Instead, we now optimize *freshness-centric performance metrics*, we refer to as **Information Timeliness Metrics (ITM)**. The flagship member is the *Age of Information (AoI)* [8], which measures, at any instant, the time elapsed since the newest successfully received sample was generated. While AoI captures data staleness directly, it is content-agnostic; simply minimizing age does not always guarantee control accuracy or efficient resource usage [9]. This limitation has spurred extensions that refine timeliness by folding in task relevance. The *Value-of-Information (VoI)*, for instance, lacks a single canonical definition; it is modeled either as a non-linear function of age (e.g., via exponential decay) [10], [10] or based on the intrinsic system dynamics to reflect how an update impacts state estimation [11], [12]. Similarly, the *Age of Incorrect Information (AoII)* accumulates staleness only when the receiver's estimate deviates effectively from

TABLE I: Key update metrics mapped to Shannon-Weaver layers. *Symbols:* n: current slot; $U(n)$: generation time of latest received update; $\Delta(n) = n - U(n)$; $x_n, \hat{x}_n$: true/estimated states; $A(n)$: last action time; $\hat{t}_j$: freshest command time; t_i: status update time causing latest action; $S(n)$: last sync time; $w(n)$: urgency weight; $C(\cdot)$: cost function; r_k: reward at age k; ϵ: tolerance.

Metric	**Definition**	**Level**	**Insight**
AoI	$\Delta(n) = n - U(n)$	A	Data freshness
MSE	$\mathbb{E}[\|x_n - \hat{x}_n\|^2]$	A	Estimation accuracy
AoSync	$n - S(n)$	A	Synchronization gap
AoA	$n - A(n)$	A	Actuation staleness
AoL	$t - \hat{t}_j; t - t_i$	A	Round-trip freshness
VoI	$f(\Delta(n), x_n)$	A/B[1]	Context-aware utility
UoI	$w(n) \cdot C(x_n, \hat{x}_n)$	B	Weighted inaccuracy cost
AoII	$\sum_{k=U(n)}^{n} \mathbb{1}\{x_k \neq \hat{x}_k\}$	B	Error-aware staleness
Expiration (Ours)	$\max\{k : r_0 - r_k \leq \epsilon\}$	C	Just-in-time scheduling

[1] Depends on VoI definition used.

the true state [13], while the *Age of Synchronization (AoSync)* tracks the time since the controller and plant last shared a fully synchronized state [14].

However, timeliness metrics are often merely proxies for true operational goals. Consequently, researchers frequently optimize Control-Oriented Metrics directly, such as Mean Squared Error (MSE) in networked estimation [15]. Bridging timeliness and error, the *Urgency of Information (UoI)* captures the weighted cost of estimation inaccuracy based on context-dependent factors [16]. Freshness concepts have also been extended to the control link via the *Age of Actuation (AoA)* [17], while the *Age of Loop (AoL)* [18] unifies both directions by capturing end-to-end freshness across the closed loop—with downlink-initiated (DL-AoL) and uplink-initiated (UL-AoL) variants. Table I summarizes these metrics alongside their corresponding Shannon–Weaver communication layers. Collectively, these approaches underscore that effective policies must balance raw freshness with actual estimation and decision-making quality.

This shift from signal fidelity to functional performance places our work within the emerging paradigm of **Goal-Oriented Communication** (Level-C) [19]–[21]. While classical information theory addresses *Level A* (accurate symbol transmission) and Semantic Communication targets *Level B* (meaning preservation), Goal-Oriented strategies focus on *Level C*: ensuring that the received information successfully steers the system toward a specific objective. The defining feature of this level is that "all information not strictly relevant to the fulfillment of the goal can be neglected." Our framework implements precisely this principle: by filtering updates based on their expiration time (a proxy for their relevance to the control objective) we discard technically correct but functionally irrelevant data, prioritizing the *pragmatic value* of information over its mere existence.

Building on this principle, we develop a status-update framework that explicitly models how each transmission (or silence) influences the decision making process. In particular, we consider a sender-receiver pair in which a fresh sample arrives every time slot, but each sample is only *useful* to the controller for a finite, random *expiration time*. The sender must decide, under an unreliable channel and a non-negligible transmission cost, whether the prospective benefit of informing the controller outweighs the cost of sending the packet. The unreliable channel compounds this decision: stochastic packet losses create uncertainty about successful delivery, forcing the sender to weigh retransmitting a potentially stale packet against transmitting a fresher sample, while each attempt consumes resources regardless of outcome.

We develop the expiration framework in two phases. First, we characterize *when observations expire*: **Section II-A** derives expiration times analytically for linear systems, while **Section II-B** estimates them via Monte Carlo rollouts for nonlinear systems (CartPole), demonstrating generalization beyond tractable dynamics. Both assume reliable channels to isolate the expiration phenomenon. Second, we address *how to schedule under unreliable channels*: **Sections III–IV** formulate the problem as a coupon-collector MDP, prove optimal policies are threshold-based, and develop structure-aware Q-learning for the model-free case. Our main contributions are:

Main Contributions

C1. We close Weaver's *Level-C* gap by attaching a finite, sample-specific *expiration time* to every status update and asking when a transmission truly improves the controller's action quality. Two representative case studies (a linear Kalman-controlled plant with an MSE cap and a data-driven Remotely controlled Cartpole task) show how such expiration horizons arise analytically or via learning.

C2. We recast scheduling as a coupon-collector problem with expiring coupons and formulate the decision process as a two-dimensional average-cost MDP. Using lattice monotonicity we prove the bias value is non-decreasing in both state coordinates; the action gap is super-modular, yielding a *double-threshold* optimal policy. For deterministic lifetimes we derive a closed-form threshold rule.

C3. For unknown, random lifetimes we design a model-free Q-learning algorithm that hard-codes the "never-send-obsolete" rule, confines exploration to the undecided band, and converges far faster than vanilla Q-learning.

C4. Simulations demonstrate that expiration-aware scheduling achieves up to $\mathbf{50\%}$ higher cumulative reward compared to periodic baselines at moderate communication costs, while SAQ matches optimal Value Iteration performance and converges significantly faster than baseline Q-learning.

II. Characterizing Observation Expiration

Before addressing the scheduling problem under unreliable channels (Sections III–IV), we must first answer a fundamental question: *how long does an observation remain useful?* This section develops two complementary approaches to characterizing expiration time T: analytical derivation for linear systems

(Section II-A) and data-driven estimation for nonlinear systems (Section II-B). Both assume a reliable channel ($p_s = 1$), isolating the expiration phenomenon from transmission uncertainty.

A. Linear control system with MSE-based expiration

a) Plant and sensor model.: We study the discrete-time linear system

$$\begin{aligned} \mathbf{x}_{n+1} &= A\,\mathbf{x}_n + B\,u_n + \mathbf{w}_n, \\ \mathbf{y}_{n+1} &= H\,\mathbf{x}_{n+1} + \mathbf{v}_{n+1}, \end{aligned} \tag{1}$$

where the state $\mathbf{x}_n \in \mathbb{R}^d$, control $u_n \in \mathcal{U} \subseteq \mathbb{R}^m$, and measurement $\mathbf{y}_{n+1} \in \mathbb{R}^p$. System matrices satisfy $A \in \mathbb{R}^{d\times d}$, $B \in \mathbb{R}^{d\times m}$, $H \in \mathbb{R}^{p\times d}$. The process noise is $\mathbf{w}_n \sim \mathcal{N}(0, Q)$ with $Q \succ 0$. The measurement noise covariances $\{R_n\}_{n\geq 1}$ are drawn i.i.d. from a distribution $\mathcal{R}$ supported on positive-definite matrices, with R_n revealed at the start of slot n.

The sensor runs a standard Kalman filter... The sensor runs a standard Kalman filter, producing $(\hat{\mathbf{x}}_{n+1|n+1}, P_{n+1|n+1})$ at the end of slot $n+1$. If the estimate is *transmitted* ($a_{n+1} = 1$) it is copied verbatim to the controller; otherwise ($a_{n+1} = 0$) the controller propagates its last estimate through the open-loop prediction $\hat{\mathbf{x}}^{\mathrm{c}}_{n+1} = A\hat{\mathbf{x}}^{\mathrm{c}}_n + B\,u_n$, $P^{\mathrm{c}}_{n+1} = AP^{\mathrm{c}}_n A^\top + Q$. The control law is a fixed map $u_n = \mu\big(\hat{\mathbf{x}}^{\mathrm{c}}_n, P^{\mathrm{c}}_n\big)$.

b) Time-varying safety criterion.: The controller cannot access the *true* state; it only sees an *estimate* corrupted by error covariance P^{c}_n. We consider a time-varying tolerance sequence $\{\tau_n\}_{n\geq 0}$ with $\tau_n > 0$ and require

$$\mathrm{tr}\big(P^{\mathrm{c}}_n\big) \;\leq\; \tau_n, \qquad \forall\, n, \tag{2}$$

which upper-bounds the instantaneous mean-squared error the controller is willing to bear.

Assumption 1 (Predictable thresholds)**.** *The threshold sequence $\{\tau_n\}_{n\geq 0}$ is fully predictable: at any time m, the sensor knows τ_n for all $n \geq m$.*[1]

Assumption 2 (Feasibility)**.** *For all n, $\mathrm{tr}(P_{n|n}) \leq \tau_n$ almost surely. That is, transmitting always satisfies the instantaneous constraint.*

At the start of slot n, the noise covariance R_n is revealed to the sensor. The sensor computes $(P_{n|n-1}, P_{n|n})$ and selects $a_n \in \{0, 1\}$. The sensor seeks a causal policy $\pi = \{a_n\}$ maximising the long-run expected reward:

$$\max_{\{a_n\}} \liminf_{N\to\infty} \frac{1}{N}\mathbb{E}\left[\sum_{n=0}^{N-1}\Big[\mathbf{1}\{\mathrm{tr}(P^{\mathrm{c}}_n) \leq \tau_n\} - \beta c\, a_n\Big] \,\Big|\, P_{0|0}\right]. \tag{3}$$

Our formulation departs from the standard remote estimation paradigm in three fundamental respects. *First*, we model measurement noise as time-varying (R_n rather than constant R), capturing realistic scenarios where sensor fidelity depends on environmental conditions, operating modes, or adaptive power management. This generalization is absent from the canonical works [22]–[26]. *Second*, we allow the MSE tolerance τ_n to vary with mission phase or risk context, rather than enforcing a static constraint throughout operation. *Third*, we optimize the *constraint-satisfaction indicator* $\mathbb{1}\{\mathrm{tr}(P^{\mathrm{c}}_n) \leq \tau_n\}$ rather than the MSE itself. While prior work minimizes aggregate estimation cost $\mathbb{E}[\sum_n \mathrm{tr}(P^{\mathrm{c}}_n)]$ [27], [28] or penalizes large errors through convex surrogates [29], our objective maximizes the *fraction of time* the controller operates within its acceptable-accuracy regime, subject to communication costs. This binary formulation directly reflects safety-critical requirements (e.g., "maintain localization accuracy 95% of the time") and naturally gives rise to the *expiration time* abstraction: each sample remains *valid* precisely until the constraint would be violated, after which it contributes zero reward regardless of its residual estimation quality.

Definition 1 (MSE-based expiration time)**.** *Given a transmission at slot m with post-update covariance $P_{m|m}$, the* expiration time *is*

$$T^{\mathrm{MSE}}_m \triangleq \min\{k \geq 1 : \mathrm{tr}(P_{m+k|m}) > \tau_{m+k}\}, \tag{4}$$

with the convention $\min \varnothing = +\infty$.

At slot n, let $m = \max\{k < n : a_k = 1\}$ denote the most recent transmission epoch. The *residual validity* $\Delta_n \triangleq T^{\mathrm{MSE}}_m - (n - m)$ measures remaining slots before the current estimate expires. A causal policy π is *admissible* if it maintains $\Delta_n \geq 1$ for all n, ensuring the constraint (2) holds almost surely.

The *just-in-time (JIT) policy* π^{JIT} transmits exactly when residual validity reaches one: $a^{\mathrm{JIT}}_n = \mathbb{1}\{\Delta_n = 1\}$. The following theorem establishes its optimality.

The following theorem establishes that this elementary policy is optimal among all causal policies.

Theorem 1 (Optimality of JIT policy)**.** *Under Assumptions 1–2, if the transmission cost satisfies $\beta c < \mathbb{E}[T^{\mathrm{MSE}} \mid P_{0|0}]$, then the just-in-time policy is optimal among all causal policies (both admissible and non-admissible), achieving*

$$J(\pi^{\mathrm{JIT}}) = 1 - \frac{\beta c}{\bar{T}}, \tag{5}$$

where $\bar{T} \triangleq \lim_{K\to\infty} \frac{1}{K}\sum_{k=1}^{K} T^{\mathrm{MSE}}_{m_k}$ is the limiting average expiration time along the JIT transmission epochs $\{m_k\}$.

Proof. See Appendix A. □

Remark 1. *The JIT policy extracts maximum value from each transmission by consuming its entire validity window. Any policy that transmits earlier "wastes" validity, while any policy that transmits later violates the constraint.*

B. Case Study 2: Data-Driven Expiration (Deep RL Cartpole)

While the linear control case allows for analytical derivation of expiration times, many real-world systems are highly nonlinear and difficult to model explicitly. To demonstrate the generality of our framework, we study the CARTPOLE system [30], a canonical balancing problem with inherently unstable dynamics. Though simple, CartPole shares key structural properties with safety-critical applications such as propulsive rocket landing [31], and self-balancing platforms [32]. These properties make information freshness critical as stale observations lead to delayed corrective actions and system failure.

[1] This holds for pre-announced mission profiles, periodic task structures, and context-aware systems where environmental cues determine requirements.

Consequently, it is frequently used to evaluate the strengths and limitations of the metrics [16], [18]. Unlike the linear-Gaussian setting where expiration times are computed analytically, here we estimate them empirically.

a) Plant dynamics.: We adopt the classic CartPole environment implemented in the Gymnasium library [33]. The state is $\mathbf{x}_n = \left(x_n, \dot{x}_n, \theta_n, \dot{\theta}_n\right)^\top \in \mathbb{R}^4$, comprising cart position x, cart velocity $\dot{x}$, pole angle θ, and angular velocity $\dot{\theta}$. The controller chooses a discrete action $u_n \in \{0, 1\}$ (push left/right) at each time step.

b) Target-tracking objective.: Rather than the default survival reward, we impose a moving-target tracking task. This reflects operational reality: in practice, the reference trajectory is rarely static. For example, mobile robots receive dynamically updated waypoints from path planners responding to obstacles or user commands [34].

Critically, tracking a time-varying reference introduces a *second channel* through which staleness degrades performance: outdated observations misrepresent not only the current system state, but also its deviation from a target that has since moved. This compounds the expiration problem: the controller optimizes toward an obsolete objective, accelerating divergence during aggressive maneuvers.

The reference position t_n evolves as a mean-reverting random walk: $t_n = (1 - \kappa)t_{n-1} + w_n, \quad w_n \sim \mathcal{N}(0, \sigma_T^2)$, with $\kappa = 0.03$ (ensuring the target remains within the feasible operating region) and $\sigma_T = 0.02$ (capturing unpredictable reference variations). The instantaneous reward penalizes deviations from the target: $r(\mathbf{x}, t) = 1 - |x - t|$, with a large penalty $r_{\text{crash}} = -50$ upon episode termination.

c) Sensor and channel model.: A remote sensor observes the true state $\mathbf{x}_n$ and produces a noisy measurement

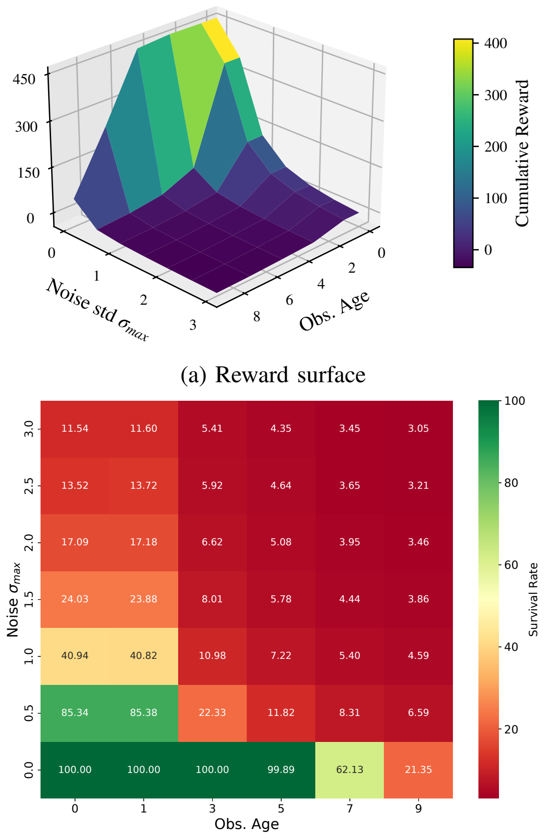
$$\mathbf{y}_n = \mathbf{x}_n + \mathbf{v}_n, \qquad \mathbf{v}_n \sim \mathcal{N}\big(\mathbf{0},\, \sigma_n^2 I_4\big), \tag{6}$$

where σ_n is sampled uniformly from a discrete set of levels from 0 to $\sigma_{\max}$. The sensor may transmit at cost $c \geq 0$ per transmission. Transmissions are assumed reliable ($p_s = 1$) to isolate the effect of the scheduling policy.

d) Controller architecture.: The controller is a recurrent actor-critic network pre-trained via Proximal Policy Optimization (PPO). At deployment, it receives potentially stale noisy observations. The network maintains a hidden state updated by a GRU to partially compensate for missing updates (input features are detailed in Table II), but this tolerance is not unlimited: Fig. 1 reveals a critical coupling between signal quality and permissible latency. In the noise-free regime ($\sigma_{\max} = 0$), the controller maintains 100% survival for observation ages up to 5 steps (Fig. 1b). However, this robustness degrades rapidly with measurement noise: at $\sigma_{\max} = 0.5$, survival drops to 85% after just one step of latency and to 22% by age 3. The reward surface (Fig. 1a) shows that performance remains stable along the axes (low noise *or* low age) but degrades sharply when high latency compounds with high noise, triggering rapid tracking loss and termination. This asymmetric structure motivates state-dependent scheduling: transmissions can be safely deferred during benign conditions but become urgent when the system operates near the degradation boundary.

TABLE II: Input Features for Controller, Predictor, and Periodic Agents

Category	Feature	Ctrl.	Pred.	Per.
State	Current observation ($\mathbf{y}_n$)	–	✓	–
	Stale observation ($\hat{\mathbf{y}}_n$)	✓	–	–
Context	Observation age (δ_n)	✓	–	✓
	Target position (t_n)	✓	✓	–
	Last action (u_{n-1})	✓	✓	–
Error	Tracking error ($x_n - t_n$)	✓	✓	–
Dynamics	Observation drift ($y_n - y_{n-1}$)	–	✓	–



(a) Reward surface

(b) Survival rate (%)

Fig. 1: **Impact of Staleness and Noise.** Performance as observation age and noise $\sigma_{\max}$ vary.

To formalize these observations, consider the trajectory induced when the controller continuously reuses observation $\mathbf{y}_n$. Let $\{\mathbf{x}^{(k)}\}_{k=0}^{K}$ denote the *stale-observation trajectory* starting from $\mathbf{x}^{(0)} = \mathbf{x}_n$:

$$\mathbf{x}^{(k+1)} = f(\mathbf{x}^{(k)}, u^{(k)}, w^{(k)}), \tag{7}$$

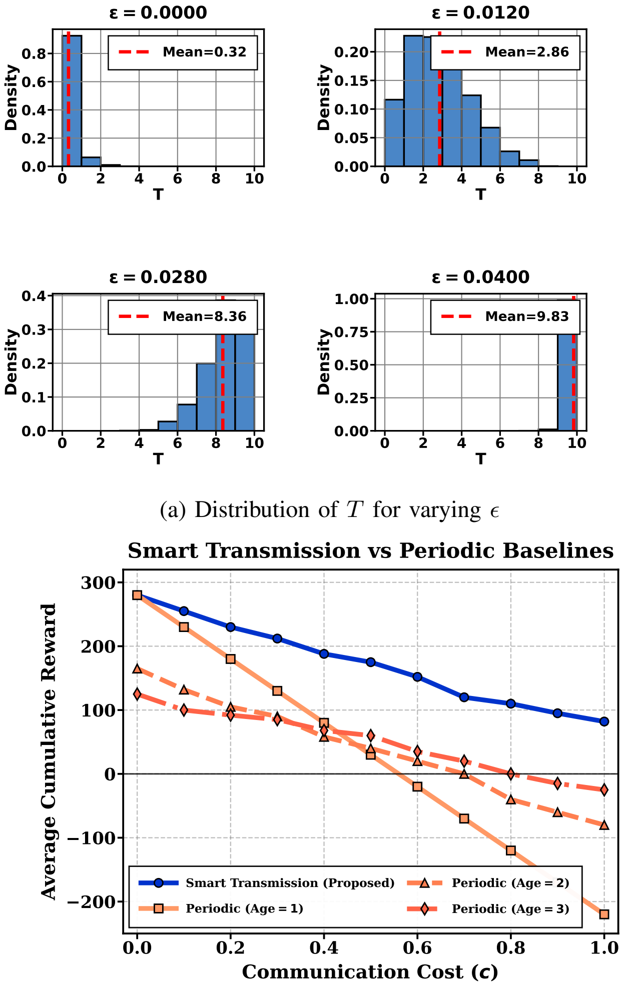
where f is the stochastic transition dynamics. We define the *expiration time* as the maximum reuse horizon before the instantaneous reward degrades beyond tolerance ε:

$$T(\mathbf{y}_n; \epsilon) = \max\Big\{k \in \{0, \ldots, K\} : r_0 - r_k \leq \epsilon\Big\}, \tag{8}$$

where $r_k = r(\mathbf{x}^{(k+1)}, t_{n+k})$ is the reward at step k along the stale trajectory.

The expiration time depends on two primary factors: the system state (which determines proximity to instability boundaries and the aggressiveness of required maneuvers) and the measurement noise intensity (which governs observation reliability). To capture this dependence, we sample the noise level σ_n uniformly from 11 discrete levels spanning $[0, 1]$, i.e., $\sigma_n \in \{0, 0.1, \ldots, 1\}$.

For a given ϵ, we measure the expiration time of noisy samples across training episodes to construct a supervised dataset $\mathcal{D}_\epsilon$, where each sample pairs features (Table II) with the measured expiration time T_n as the label. A feedforward neural network regressor is then trained on $\mathcal{D}_\epsilon$ to predict the expiration time of an incoming noisy observation $\mathbf{y}_n$, given a tolerance level of ϵ, enabling real-time expiration estimation at deployment without requiring costly rollout simulations. Code is available at []



(a) Distribution of T for varying ϵ

(b) Reward vs. communication cost

Fig. 2: **Expiration Analysis and Scheduling Performance.**

Fig. 2a confirms that the useful lifespan of an observation is a dynamic, state-dependent quantity rather than a static system constant. The distributions of predicted expiration time T exhibit significant variance—even for a fixed tolerance ϵ, the valid reuse horizon fluctuates widely, ranging from immediate expiration ($T = 1$) to extended stability ($T > 10$), depending on the instantaneous state and noise realization. As expected, relaxing the tolerance parameter shifts the distribution rightward, providing a tunable mechanism for trading tracking precision against communication frequency.

At each step, the scheduler compares the predicted expiration time $\hat{T}_n$ of the incoming observation against the *remaining* lifetime T^{rem}_{n-1} of the currently held sample, retaining whichever offers the longer valid horizon. Transmission occurs just-in-time, precisely when the selected sample's remaining lifetime expires. For each communication cost c, we select the tolerance ϵ that maximizes cumulative reward, enabling adaptation to resource constraints.

Fig. 2b shows the proposed method (solid blue) achieves nearly 50% higher reward than periodic baselines at moderate costs ($c \approx 0.4$). Following Fig. 1b, the scheduler defers transmission when lifetimes are long and transmits promptly when noise or maneuvering shortens validity.

This performance gap stems directly from the robustness structure revealed in Fig. 1b: the dynamic scheduler avoids redundant transmissions during benign operating conditions, reserving communication budget for critical moments when noise or maneuvering dynamics threaten system stability. By contrast, periodic policies waste resources during safe intervals while potentially under-communicating during high-risk phases.

We have characterized observation expiration both analytically and numerically via learned predictors. Under reliable communication, the optimal policy is simple: transmit *just-in-time*, precisely when the current observation expires. This holds regardless of the incoming sample's expiration time.

Unreliable channels break this structure. When transmissions may fail, the scheduler faces a gambling problem: should it send now, risking packet loss, or wait for a potentially better sample? The *distribution* of expiration times and the *value* of the current sample now enter the decision. In the next section, we formalize this as a coupon collector problem, where a sender holds samples of heterogeneous value and must decide when to transmit across an unreliable channel to a receiver who requires timely information.

III. Coupon-Collector View and MDP Formulation

Having characterized expiration times, we now formulate the unreliable-channel scheduling problem as an MDP. We adopt a coupon-collector perspective where samples are expiring coupons shipped over a lossy channel, then derive the Bellman equation governing optimal transmission decisions.

A. A Coupon-Collector Variant with Expiry

Consider a shopkeeper receiving one coupon per day, each valid for a random number of days $T \in \{1, \ldots, K\}$ drawn i.i.d. from p_T. The shopkeeper may ship the coupon to a customer incurring a fee c with delivery failure probability $1 - p_s$, or may choose to keep it (possibly expiring unused). The customer earns reward r for each day she holds an unexpired coupon. The goal is to maximize average reward minus shipping fees.

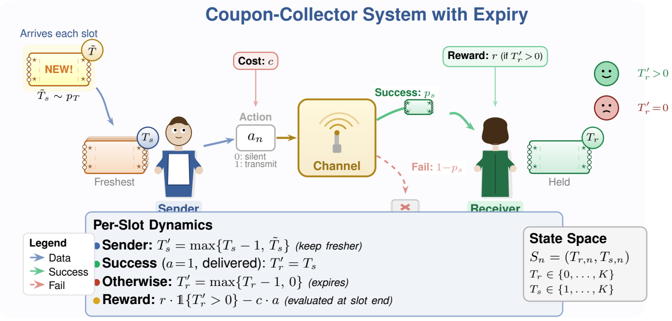

Coupon-Collector System with Expiry
Arrives each slot
NEW!
$\tilde{T}$
$\tilde{T}_s \sim p_T$
T_s
Freshest
Sender
Cost: c
Action
a_n
0: silent
1: transmit
Channel
Success: p_s
Fail: $1-p_s$
Reward: r (if $T_r' > 0$)
Receiver
T_r
Held
$T_r' > 0$
$T_r' = 0$
Legend
Data
Success
Fail
Per-Slot Dynamics
Sender: $T_s' = \max\{T_s - 1, \tilde{T}_s\}$ (keep fresher)
Success ($a=1$, delivered): $T_r' = T_s$
Otherwise: $T_r' = \max\{T_r - 1, 0\}$ (expires)
Reward: $r \cdot \mathbb{1}\{T_r' > 0\} - c \cdot a$ (evaluated at slot end)
State Space
$S_n = (T_{r,n}, T_{s,n})$
$T_r \in \{0, \ldots, K\}$
$T_s \in \{1, \ldots, K\}$

Fig. 3: Coupon-collector model with expiring samples.

Unlike classical coupon-collector problems that minimize collection time with non-expiring coupons [35], our variant features strict expiration times and a steady-state reward–cost trade-off, requiring new analytical tools.

B. MDP Components

We now formalize the coupon-collector analogy as a Markov decision process, with components illustrated in Figure 3. The shopkeeper maps to the sender, the customer to the receiver, and coupon validity to sample expiration time.

a) State and Action.: Let $T_{r,n} \in \{0, \ldots, K\}$ denote the receiver's sample remaining lifetime ($T_{r,n} = 0$ means no valid sample), and $T_{s,n} \in \{1, \ldots, K\}$ the sender's freshest sample lifetime. The state is $S_n = (T_{r,n}, T_{s,n})$. At each slot, the sender chooses $a_n \in \{0, 1\}$ (transmit or remain silent).

b) Transitions.: The sender keeps the fresher of the aged sample and new arrival: $T_{s,n+1} = \max\{T_{s,n} - 1, \tilde{T}_{s,n+1}\}$, where $\tilde{T}_{s,n+1} \sim p_T$. The receiver's timer evolves as:

$$T_{r,n+1} = \begin{cases} T_{s,n}, & \text{if } a_n = 1 \text{ and} \\ & \text{delivery succeeds,} \\ \max\{T_{r,n} - 1, 0\}, & \text{otherwise.} \end{cases} \tag{9}$$

c) Reward and Objective.: The instantaneous reward is $R(S_n, a_n, S_{n+1}) = r \cdot \mathbb{1}\{T_{r,n+1} > 0\} - c \cdot a_n$. We seek a stationary policy $\pi : \mathcal{S} \to \{0, 1\}$ maximizing the long-run average reward:

$$\max_{\pi} \liminf_{N \to \infty} \frac{1}{N} \sum_{n=1}^{N} \mathbb{E}^{\pi}\left[R(S_n, a_n, S_{n+1})\right].$$

d) Bellman Equation.: The optimal policy satisfies the average-reward Bellman equation [36]:

$$\rho^* + V(T_r, T_s) = \max_{a \in \{0,1\}} \left\{\bar{R}(S, a) + \mathbb{E}[V(T_r', T_s') \mid S, a]\right\}, \tag{10}$$

where ρ^* is the optimal average reward (gain), $V(\cdot)$ is the bias function, and $\bar{R}(S, a) = \mathbb{E}[R(S, a, S')]$. Value iteration computes π^* via:

$$\begin{aligned} Q^{(k+1)}(T_r, T_s, a) = -c \cdot a + \sum_{T_r', T_s'} P_a(S \to S') \\ \times \left[r \cdot \mathbb{1}\{T_r' > 0\} + V^{(k)}(T_r', T_s')\right], \end{aligned} \tag{11}$$

but requires exact knowledge of p_s and p_T. The MDP admits a compact state space of $(K + 1) \times K$, making value iteration tractable when the p_s and p_T are known. Two questions remain: (i) does the optimal policy possess exploitable structure, and (ii) can we learn effective policies when these parameters are unknown? The next section addresses both.

IV. Optimal Scheduling Policy

We now study the optimal policy from three angles. We start by establishing structural properties that hold for any expiration time distribution: the value function is monotone and the optimal policy follows a threshold rule. When expiration times are constant and parameters are known, these results yield a closed-form optimal threshold. When parameters are unknown, we show how to embed the same structural insights into a Q-learning algorithm that learns efficiently.

Lemma 1. *Let $V(T_r, T_s)$ satisfy* (10)*. Then $V(T_r, T_s) \leq V(T_r{+}1, T_s)$ and $V(T_r, T_s) \leq V(T_r, T_s{+}1)$ for all valid states.*

Proof. See Appendix B. □

The monotonicity reflects natural intuitions: a receiver with more remaining validity ($T_r{+}1$ vs. T_r) can wait longer before requiring an update, while a sender with a fresher sample ($T_s{+}1$ vs. T_s) offers greater potential value upon successful transmission. Both advantages propagate through the Bellman recursion, yielding higher long-run reward.

For binary actions, the optimal policy satisfies $\pi^*(T_r, T_s) = \mathbb{1}\{\Delta Q(T_r, T_s) > 0\}$, where $\Delta Q = Q(\cdot, 1) - Q(\cdot, 0)$ is the advantage of transmitting. The following theorem shows this advantage has sufficient structure to yield a threshold policy.

Theorem 2. *For each $T_s \in \{1, \ldots, K\}$, there exists a threshold $\theta(T_s) \in \{0, \ldots, K\}$ such that $\pi^*(T_r, T_s) = 1$ iff $T_r \leq \theta(T_s)$. Moreover, $\theta(T_s) \leq \theta(T_s + 1)$.*

Proof. See Appendix C. □

If the receiver still has plenty of validity left, there is no rush to transmit. As T_s increases, the potential gain from successful delivery grows, so the sender becomes more willing to attempt transmission early. The next result identifies three properties that hold across the entire state space.

Theorem 3. *The optimal policy π^* satisfies:*

(i) *If $T_r = 0$ and $p_s r - c > 0$, then $\pi^*(0, T_s) = 1$ for all T_s.*
(ii) *If $T_r > T_s$, then $\pi^*(T_r, T_s) = 0$.*
(iii) *If transmitting is optimal at (T_r, T_s) for $T_r > 0$, it is optimal at $(T_r - 1, T_s)$.*

Proof. See Appendix D. □

Intuitively, an expired receiver should always be refreshed when transmission pays off on average. Sending data staler than what the receiver already holds is clearly wasteful since cost is incurred with no benefit. And as the receiver's validity diminishes, the urgency to transmit naturally grows.

The preceding results hold for general lifetime distributions. When lifetimes are constant, the threshold admits a closed-form characterization.

Theorem 4. *For constant lifetime $T_{s,n} = K$, the optimal policy is a threshold rule: transmit iff $T_r \leq \theta^*$. The average reward is $\rho(\theta) = r - \frac{c + r(1-p_s)^{\theta}}{p_s(K-\theta)+1}$, and θ^* is the largest integer satisfying $(K - \theta + 1)(1 - p_s)^{\theta - 1} > c/(p_s r)$.*

Algorithm 1 Structure-Aware Q-Learning (SAQ)

```
Init: $Q_0(T_r, T_s, a) \leftarrow 0$ for all states/actions; $\hat{\rho}_0 \leftarrow 0$
for $n = 0, 1, 2, \ldots$ do
    Sample $T_s \sim p_T$, $Z \sim \text{Bern}(p_s)$, $\tilde{T}_s \sim p_T$
    $R_{\text{sum}} \leftarrow 0$                                  ▷ Initialize accumulator
    for $T_r = 0$ to $K$ do                                  ▷ Sweep all receiver states
        if $T_r > T_s$ then
            $a \leftarrow 0$                                  ▷ Never send obsolete
        else
            $a \leftarrow \epsilon\text{-greedy}\big(\arg\max_{a'} Q_n(T_r, T_s, a')\big)$
        $T_r' \leftarrow T_s$ if $(a = 1 \wedge Z = 1)$ else $\max\{T_r - 1, 0\}$
        $T_s' \leftarrow \max\{T_s - 1, \tilde{T}_s\}$
        $R \leftarrow r \cdot \mathbb{1}_{\{T_r' > 0\}} - c \cdot a$
        $R_{\text{sum}} \leftarrow R_{\text{sum}} + R$                                  ▷ Accumulate reward
        $\delta \leftarrow R - \hat{\rho}_n + \max_{a'} Q_n(T_r', T_s', a') - Q_n(T_r, T_s, a)$
        $Q_{n+1}(T_r, T_s, a) \leftarrow Q_n(T_r, T_s, a) + \alpha_n \delta$
    $\bar{R}_n \leftarrow R_{\text{sum}}/(K + 1)$
    $\hat{\rho}_{n+1} \leftarrow \hat{\rho}_n + \beta_n(\bar{R}_n - \hat{\rho}_n)$
```

Proof. See Appendix E. □

The threshold increases with K and r, and decreases with p_s (reliable channels require fewer transmission attempts) and c, reflecting the tradeoff between transmission cost and expiration risk. For random lifetimes, closed-form solutions are generally unavailable, motivating the learning approach below.

A. Structure-Aware Q-Learning

When the channel success probability p_s and lifetime distribution p_T are unknown, we turn to model-free reinforcement learning. Our structural results from Theorems 2–3 enable significant acceleration over naive Q-learning through two mechanisms. First, we **hard-code suboptimal actions**: when $T_r > T_s$, the receiver holds fresher data, so transmitting incurs cost c for inferior information and we enforce $a = 0$. Second, we use **batch updates**: we sample T_s and the channel once, then sweep all $T_r \in \{0, \ldots, K\}$, exploiting shared T_s dynamics for $O(K)$ updates per sample.

Algorithm 1 summarizes our approach. At each iteration, we sample a sender lifetime $T_s \sim p_T$, a channel realization $Z \sim \text{Bern}(p_s)$, and a next-slot arrival $\tilde{T}_s \sim p_T$. We then loop over all receiver states: for $T_r > T_s$, we force $a = 0$; for $T_r \leq T_s$, we apply ϵ-greedy exploration. Each (T_r, T_s, a) triple receives a Q-update based on the resulting transition and reward.

The algorithm uses two-timescale stochastic approximation [36]. Q-values evolve on step-size α_n; the average reward estimate $\hat{\rho}$ evolves on $\beta_n = o(\alpha_n)$. With $\sum_n \alpha_n = \infty$, $\sum_n \alpha_n^2 < \infty$ (and likewise for β_n), convergence to the optimal policy is guaranteed. The structural constraints from Theorem 3 cut the exploration space roughly in half, and the batch sweep yields $O(K)$ updates per sample. Both modifications speed up learning considerably in practice.

V. Simulation Results

We validate our structure-aware Q-learning (SAQ) on a system with maximum expiration time $K = 20$, reward $r = 1$, lifetime distribution p_T uniform on $\{1, \ldots, K\}$, and varying p_s and cost ratio c/r. Figure 4 reveals how channel reliability p_s shapes the optimal policy structure. At $p_s = 1.0$ (perfect channel), the send region is minimal, the scheduler waits until the receiver's sample nears expiration, confident that a single transmission will succeed. In other words, because of reliability, the optimal policy is JIT as shown by Theorem 1. the As p_s decreases, the send region progressively expands: at $p_s = 0.25$, the scheduler transmits earlier to hedge against repeated failures. Quantitatively, the threshold $\theta(T_s)$ increases as p_s drops, confirming that unreliable channels demand more aggressive scheduling. Across all settings, the double-threshold structure from Theorem 2 holds: thresholds increase monotonically with T_s, reflecting the value of transmitting fresher samples. The diagonal boundary $T_r = T_s$ remains inviolate as transmitting obsolete data is never optimal regardless of channel conditions.

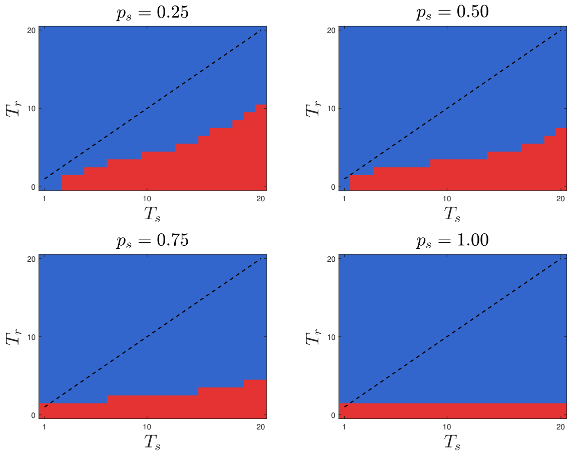
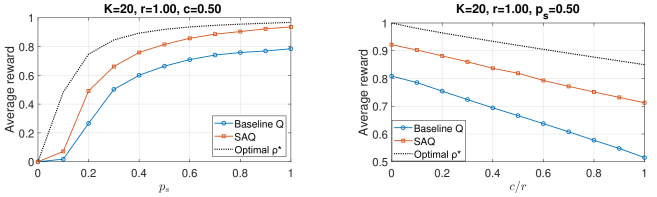


Fig. 4: Optimal policy regions for varying p_s values ($K = 20$, $c/r = 0.5$).



(a) Varying p_s ($c/r = 0.5$)

(b) Varying c/r ($p_s = 0.5$)

Fig. 5: Average reward vs. system parameters for Baseline Q, SAQ, and optimal VI.

Figure 5 evaluates performance of SAQ by sweeping channel reliability p_s and cost ratio c/r. As expected, average reward increases with p_s (more successful deliveries) and decreases with c/r (higher transmission penalty). Across all parameter ranges, SAQ consistently closer to the optimal ρ^* from Value Iteration, while baseline Q-learning exhibits a persistent gap due to slower convergence.

VI. Conclusion

We studied status updating when samples expire after a finite horizon. A key step is characterizing when a sample loses its value. The scheduling problem reduces to a coupon-collector

MDP, and we showed the optimal policy follows a monotone threshold rule, yielding closed-form solutions for constant lifetimes. For random lifetimes with unknown statistics, our structure-aware Q-learning converges faster than standard Q-learning. A natural next step is to identify expiration times for more complex applications and to develop new effectiveness metrics beyond freshness and semantics.

References

[1] W. Weaver, "Recent contributions to the mathematical theory of communication," *ETC: a review of general semantics*, pp. 261–281, 1953.

[2] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, 2016.

[3] Z. Zhou, X. Chen, E. Li, L. Zeng, K. Luo, and J. Zhang, "Edge intelligence: Paving the last mile of artificial intelligence with edge computing," *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1738–1762, 2019.

[4] V. Raghunathan, C. Schurgers, S. Park, and M. B. Srivastava, "Energy-aware wireless microsensor networks," *IEEE Signal processing magazine*, vol. 19, no. 2, pp. 40–50, 2002.

[5] V. Sze, Y.-H. Chen, T.-J. Yang, and J. S. Emer, "Efficient processing of deep neural networks: A tutorial and survey," *Proceedings of the IEEE*, vol. 105, no. 12, pp. 2295–2329, 2017.

[6] Y. Wu, Q. Wang, Z. Wang, X. Wang, B. Ayyagari, S. Krishnan, M. Chudzik, and W. D. Lu, "Bulk-switching memristor-based compute-in-memory module for deep neural network training," *Advanced Materials*, vol. 35, no. 46, p. 2305465, 2023.

[7] P. Tabuada, "Event-triggered real-time scheduling of stabilizing control tasks," *IEEE Transactions on Automatic Control*, vol. 52, no. 9, pp. 1680–1685, 2007.

[8] S. Kaul, R. Yates, and M. Gruteser, "Real-time status: How often should one update?" in *2012 Proceedings IEEE INFOCOM*. IEEE, 2012, pp. 2731–2735.

[9] Y. Sun, E. Uysal-Biyikoglu, R. D. Yates, C. E. Koksal, and N. B. Shroff, "Update or wait: How to keep your data fresh," *IEEE Transactions on Information Theory*, vol. 63, no. 11, pp. 7492–7508, 2017.

[10] Y. Sun and B. Cyr, "Sampling for data freshness optimization: Non-linear age functions," *Journal of Communications and Networks*, vol. 21, no. 3, pp. 204–219, 2019.

[11] O. Ayan, M. Vilgelm, M. Klügel, S. Hirche, and W. Kellerer, "Age-of-information vs. value-of-information scheduling for cellular networked control systems," in *Proceedings of the 10th ACM/IEEE International Conference on Cyber-Physical Systems*, 2019, pp. 109–117.

[12] A. Arafa and R. D. Yates, "Age and value of information optimization for systems with multi-class updates," in *ICC 2024-IEEE International Conference on Communications*. IEEE, 2024, pp. 195–200.

[13] A. Maatouk, S. Kriouile, M. Assaad, and A. Ephremides, "The age of incorrect information: A new performance metric for status updates," *IEEE/ACM Transactions on Networking*, vol. 28, no. 5, pp. 2215–2228, 2020.

[14] J. Zhong, R. D. Yates, and E. Soljanin, "Two freshness metrics for local cache refresh," in *2018 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2018, pp. 1924–1928.

[15] Y. Sun, Y. Polyanskiy, and E. Uysal-Biyikoglu, "Remote estimation of the wiener process over a channel with random delay," in *2017 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2017, pp. 321–325.

[16] X. Zheng, S. Zhou, and Z. Niu, "Urgency of information for context-aware timely status updates in remote control systems," *IEEE Transactions on Wireless Communications*, vol. 19, no. 11, pp. 7237–7250, 2020.

[17] A. Nikkhah, A. Ephremides, and N. Pappas, "Age of actuation in a wireless power transfer system," in *IEEE INFOCOM 2023-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2023, pp. 1–6.

[18] P. M. de Sant Ana, N. Marchenko, P. Popovski, and B. Soret, "Age of loop for wireless networked control systems optimization," in *2021 IEEE 32nd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. IEEE, 2021, pp. 1–7.

[19] E. C. Strinati and S. Barbarossa, "6g networks: Beyond shannon towards semantic and goal-oriented communications," *Computer Networks*, vol. 190, p. 107930, 2021.

[20] E. Uysal, O. Kaya, and A. Ephremides, "Semantic communications: Overview, taxonomy, and research challenges," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 3, pp. 1721–1756, 2023.

[21] A. Li, S. Wu, S. Meng, R. Lu, S. Sun, and Q. Zhang, "Toward goal-oriented semantic communications: New metrics, framework, and open challenges," *IEEE Wireless Communications*, vol. 31, no. 5, pp. 238–245, 2024.

[22] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. I. Jordan, and S. S. Sastry, "Kalman filtering with intermittent observations," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1453–1464, 2004.

[23] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla, and S. S. Sastry, "Foundations of control and estimation over lossy networks," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 163–187, 2007.

[24] L. Shi, P. Cheng, and J. Chen, "Sensor data scheduling for optimal state estimation with communication costs," *Automatica*, vol. 47, no. 8, pp. 1489–1498, 2011.

[25] A. S. Leong, D. E. Quevedo, and S. Dey, "Transmission scheduling for remote state estimation over packet dropping links," *Automatica*, vol. 81, pp. 54–63, 2017.

[26] J. Chakravorty and A. Mahajan, "Remote state estimation with posterior-based update policy," *IEEE Transactions on Automatic Control*, vol. 67, no. 3, pp. 1307–1322, 2022.

[27] Y. Xu and J. P. Hespanha, "Optimal communication logics in networked control systems," in *Proc. IEEE CDC*, 2004, pp. 3527–3532.

[28] G. M. Lipsa and N. C. Martins, "Remote state estimation with communication costs for first-order LTI systems," *IEEE Trans. Autom. Control*, vol. 56, no. 9, pp. 2013–2025, 2011.

[29] A. S. Leong, A. Ramaswamy, D. E. Quevedo, H. Karl, and L. Shi, "Deep reinforcement learning for wireless sensor scheduling in cyber-physical systems," *Automatica*, vol. 113, p. 108759, 2020.

[30] A. G. Barto, R. S. Sutton, and C. W. Anderson, "Neuronlike adaptive elements that can solve difficult learning control problems," *IEEE Transactions on Systems, Man, and Cybernetics*, no. 5, pp. 834–846, 1983.

[31] L. Blackmore, B. Açıkmeşe, and D. P. Scharf, "Minimum-landing-error powered-descent guidance for Mars landing using convex optimization," *Journal of Guidance, Control, and Dynamics*, vol. 33, no. 4, pp. 1161–1171, 2010.

[32] F. Grasser, A. D'Arrigo, S. Colombi, and A. C. Rufer, "JOE: A mobile, inverted pendulum," *IEEE Transactions on Industrial Electronics*, vol. 49, no. 1, pp. 107–114, 2002.

[33] M. Towers *et al.*, "Gymnasium," *arXiv preprint arXiv:2301.03577*, 2023.

[34] K. Pathak, J. Franch, and S. K. Agrawal, "Velocity and position control of a wheeled inverted pendulum by partial feedback linearization," *IEEE Transactions on Robotics*, vol. 21, no. 3, pp. 505–513, 2005.

[35] S. Ross, *Introduction to Probability Models*. Academic Press, 2010. [Online]. Available: https://books.google.com/books?id=7hbLoQEACAAJ

[36] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, 3rd ed. Belmont, MA: Athena Scientific, 2011, vol. II.

Appendix A
Proof of Theorem 1

The proof proceeds in five steps: we first establish key structural properties of the covariance dynamics, then show that non-admissible policies are suboptimal, that JIT transmissions are necessary for admissibility, that early transmissions only increase the transmission rate, and finally compute the JIT value.

Step 1: Covariance dominance. We first establish that fresher information yields uniformly better predictions.

Lemma 2 (Measurement update reduction)**.** *For any $R_n \succ 0$, the Kalman measurement update satisfies $P_{n|n} \preceq P_{n|n-1}$.*

Proof. The measurement update is $P_{n|n} = P_{n|n-1} - P_{n|n-1}H^\top S_n^{-1} H P_{n|n-1}$, where $S_n = HP_{n|n-1}H^\top + R_n \succ 0$. This is a congruence transformation of $S_n^{-1} \succ 0$, hence positive semidefinite. □

Lemma 3 (Covariance dominance). *For any transmission epochs $m < n$ and any prediction horizon $k \geq 0$,*

$$P_{n+k|n} \preceq P_{n+k|m}. \tag{12}$$

Proof. We proceed by induction on $n - m$.

Base case ($n = m+1$): By Lemma 2, $P_{m+1|m+1} \preceq P_{m+1|m}$. For any $k \geq 0$, the Lyapunov recursion gives

$$\begin{aligned} P_{m+1+k|m+1} &= A^k P_{m+1|m+1}(A^k)^\top + \sum_{\ell=0}^{k-1} A^\ell Q (A^\ell)^\top && (13)\\ &\preceq A^k P_{m+1|m}(A^k)^\top + \sum_{\ell=0}^{k-1} A^\ell Q (A^\ell)^\top \\ &= P_{m+1+k|m}. && (14) \end{aligned}$$

where the inequality uses $X \preceq Y \Rightarrow A^k X (A^k)^\top \preceq A^k Y (A^k)^\top$.

Inductive step: Suppose the result holds for gap $n - m$. For gap $(n+1) - m$, apply the base case with n in place of $m+1$: $P_{n+1+k|n+1} \preceq P_{n+1+k|n}$. By the inductive hypothesis with horizon $k+1$: $P_{n+1+k|n} = P_{n+(k+1)|n} \preceq P_{n+(k+1)|m} = P_{n+1+k|m}$. Combining yields $P_{n+1+k|n+1} \preceq P_{n+1+k|m}$. □

Lemma 4 (Absolute expiration ordering). *For any transmission epochs $m < n$, the absolute expiration times satisfy*

$$n + T_n^{\mathrm{MSE}} \geq m + T_m^{\mathrm{MSE}}. \tag{15}$$

Proof. By definition of T_m^{MSE}, we have $\operatorname{tr}(P_{m+j|m}) \leq \tau_{m+j}$ for $j = 1, \ldots, T_m^{\mathrm{MSE}} - 1$.

For any slot $n + j$ with $n + j \leq m + T_m^{\mathrm{MSE}} - 1$, setting $\ell = (n - m) + j$ gives $m + \ell = n + j$ and $\ell \leq T_m^{\mathrm{MSE}} - 1$. By Lemma 3:

$$\operatorname{tr}(P_{n+j|n}) \leq \operatorname{tr}(P_{n+j|m}) = \operatorname{tr}(P_{m+\ell|m}) \leq \tau_{m+\ell} = \tau_{n+j}.$$

Thus the constraint is satisfied at slot $n+j$ under transmission at n for all $j = 1, \ldots, (m + T_m^{\mathrm{MSE}} - 1) - n$.

By definition of T_n^{MSE}: $T_n^{\mathrm{MSE}} \geq (m + T_m^{\mathrm{MSE}}) - n$, which rearranges to (15). □

Step 2: Non-admissible policies are suboptimal.

Consider any policy π that is not admissible, i.e., allows $\Delta_n = 0$ (constraint violation) for some slots. Partition the time axis into renewal cycles, where cycle k begins at transmission epoch m_k. Within cycle k, define:

- V_k: number of valid slots where $\operatorname{tr}(P_n^{\mathrm{c}}) \leq \tau_n$ (each earns reward 1),
- I_k: number of invalid slots where $\operatorname{tr}(P_n^{\mathrm{c}}) > \tau_n$ (each earns reward 0).

The cycle length is $L_k = V_k + I_k$ and the cycle reward is $V_k - \beta c$.

By the renewal-reward theorem:

$$J(\pi) = \frac{\mathbb{E}[V_k - \beta c]}{\mathbb{E}[V_k + I_k]} = \frac{\mathbb{E}[V_k] - \beta c}{\mathbb{E}[V_k] + \mathbb{E}[I_k]}. \tag{16}$$

Since validity cannot extend beyond expiration, $V_k \leq T_{m_k}^{\mathrm{MSE}}$. For a non-admissible policy, $\mathbb{E}[I_k] > 0$ (violations occur with positive probability). Therefore:

$$J(\pi) < \frac{\mathbb{E}[V_k] - \beta c}{\mathbb{E}[V_k]} \leq \frac{\mathbb{E}[T^{\mathrm{MSE}}] - \beta c}{\mathbb{E}[T^{\mathrm{MSE}}]}. \tag{17}$$

The strict inequality follows from $\mathbb{E}[I_k] > 0$ and $\mathbb{E}[V_k] - \beta c > 0$ (which holds when $\beta c < \mathbb{E}[T^{\mathrm{MSE}}]$).

Step 3: Objective simplification for admissible policies.

For any admissible policy π, we have $\Delta_n \geq 1$ for all n, so $\operatorname{tr}(P_n^{\mathrm{c}}) \leq \tau_n$ always. Every slot earns reward 1, and the objective (3) simplifies to:

$$J(\pi) = \liminf_{N \to \infty} \frac{1}{N} \mathbb{E}\left[\sum_{n=0}^{N-1} (1 - \beta c\, a_n) \,\middle|\, P_{0|0} \right] = 1 - \beta c \cdot \bar{a}(\pi), \tag{18}$$

where $\bar{a}(\pi) \triangleq \limsup_{N\to\infty} \frac{1}{N} \mathbb{E}[\sum_{n=0}^{N-1} a_n \mid P_{0|0}]$ is the long-run transmission rate.

Maximizing $J(\pi)$ over admissible policies is thus equivalent to minimizing $\bar{a}(\pi)$.

Step 4: JIT transmissions are necessary for admissibility.

Lemma 5 (Necessity of JIT transmissions). *Any admissible policy must transmit whenever JIT transmits: $a_n^{\mathrm{JIT}} = 1 \Rightarrow a_n^{\pi} = 1$.*

Proof. Suppose $a_n^{\mathrm{JIT}} = 1$, i.e., $\Delta_n = 1$ under JIT. This means $\operatorname{tr}(P_{n+1|m^{\mathrm{JIT}}}) > \tau_{n+1}$, where m^{JIT} is the last JIT transmission epoch.

Let $m^\pi \leq m^{\mathrm{JIT}}$ be the last transmission epoch under π (the inequality holds because π cannot have transmitted more recently than JIT without JIT also transmitting, by the structure of admissible policies).

By Lemma 3 with $m = m^\pi$ and $n = m^{\mathrm{JIT}}$:

$$P_{n+1|m^{\mathrm{JIT}}} \preceq P_{n+1|m^\pi}.$$

Taking traces:

$$\operatorname{tr}(P_{n+1|m^\pi}) \geq \operatorname{tr}(P_{n+1|m^{\mathrm{JIT}}}) > \tau_{n+1}.$$

If π does not transmit at slot n (i.e., $a_n^\pi = 0$), then $P_{n+1}^{\mathrm{c}} = P_{n+1|m^\pi}$ and $\operatorname{tr}(P_{n+1}^{\mathrm{c}}) > \tau_{n+1}$, violating admissibility at slot $n+1$. Therefore, π must transmit at slot n. □

Step 5: Early transmissions do not reduce the rate.

We now show that JIT achieves the minimum transmission rate among admissible policies.

Let $\{m_k^{\mathrm{JIT}}\}_{k\geq1}$ and $\{m_k^\pi\}_{k\geq1}$ denote the transmission epochs under JIT and any admissible policy π, respectively.

Lemma 6 (Transmission epoch ordering). *For all $k \geq 1$: $m_k^\pi \leq m_k^{\mathrm{JIT}}$.*

Proof. We proceed by induction on k.

Base case ($k = 1$): Both policies must transmit before the initial estimate expires. Policy π may transmit at or before the moment JIT transmits (possibly earlier if π transmits "early" when $\Delta > 1$). Thus $m_1^\pi \leq m_1^{\mathrm{JIT}}$.

Inductive step: Suppose $m_{k-1}^\pi \leq m_{k-1}^{\mathrm{JIT}}$.

By Lemma 4:

$$m_{k-1}^{\mathrm{JIT}} + T_{m_{k-1}^{\mathrm{JIT}}}^{\mathrm{MSE}} \geq m_{k-1}^\pi + T_{m_{k-1}^\pi}^{\mathrm{MSE}}. \tag{19}$$

Under JIT, the next transmission occurs exactly at expiration:

$$m_k^{\mathrm{JIT}} = m_{k-1}^{\mathrm{JIT}} + T_{m_{k-1}^{\mathrm{JIT}}}^{\mathrm{MSE}}.$$

Under π, the next transmission occurs at or before expiration (either at $\Delta = 1$ by necessity, or earlier by choice):

$$m_k^\pi \leq m_{k-1}^\pi + T_{m_{k-1}^\pi}^{\mathrm{MSE}}.$$

Combining with (19):

$$m_k^\pi \le m_{k-1}^\pi + T^{\rm MSE}_{m^\pi_{k-1}} \le m_{k-1}^{\rm JIT} + T^{\rm MSE}_{m^{\rm JIT}_{k-1}} = m_k^{\rm JIT}.$$

□

Lemma 6 implies that for any N:

$$\big|\{k : m_k^\pi \le N\}\big| \ge \big|\{k : m_k^{\rm JIT} \le N\}\big|.$$

That is, π incurs at least as many transmissions as JIT by time N. Taking limits:

$$\bar{a}(\pi) \ge \bar{a}(\pi^{\rm JIT}), \tag{20}$$

with equality if and only if π never transmits early (i.e., $\pi \equiv \pi^{\rm JIT}$ on the event that both are admissible).

Step 6: JIT value computation.

Under $\pi^{\rm JIT}$, transmissions occur at epochs $m_1 < m_2 < \cdots$ with inter-transmission times

$$L_k \triangleq m_{k+1} - m_k = T^{\rm MSE}_{m_k}.$$

Each cycle k has:

- Length: $L_k = T^{\rm MSE}_{m_k}$ slots,
- Reward: $T^{\rm MSE}_{m_k}$ (all slots valid) minus βc (one transmission),
- Net cycle reward: $T^{\rm MSE}_{m_k} - \beta c$.

The transmission process forms a renewal process with cycle lengths $\{L_k\}$. By the renewal-reward theorem:

$$J(\pi^{\rm JIT}) = \frac{\mathbb{E}[\text{cycle reward}]}{\mathbb{E}[\text{cycle length}]} = \frac{\mathbb{E}[T^{\rm MSE}_{m_k}] - \beta c}{\mathbb{E}[T^{\rm MSE}_{m_k}]} = 1 - \frac{\beta c}{\bar{T}}, \tag{21}$$

where $\bar{T} = \lim_{K\to\infty} \frac{1}{K}\sum_{k=1}^K T^{\rm MSE}_{m_k}$ equals $\mathbb{E}[T^{\rm MSE}_{m_k}]$ by the ergodic theorem when the limit exists.

The condition $\beta c < \mathbb{E}[T^{\rm MSE}]$ ensures $J(\pi^{\rm JIT}) > 0$.

Conclusion.

Combining Steps 2–6:

- Non-admissible policies achieve strictly lower value than $1 - \beta c/\bar{T}$ (Step 2).
- Among admissible policies, JIT achieves the minimum transmission rate (Steps 4–5) and hence the maximum value (Step 3).
- The JIT value is $1 - \beta c/\bar{T}$ (Step 6).

Therefore, $\pi^{\rm JIT}$ is optimal among all causal policies. □

Appendix B
Proof of Lemma 1

We prove both inequalities by induction on the value iteration sequence. Define $\{V^{(k)}\}_{k\ge 0}$ by $V^{(0)} \equiv 0$ and

$$V^{(k+1)}(T_r, T_s) = \max_{a\in\{0,1\}} Q^{(k+1)}(T_r, T_s, a),$$

where the Q-function update is

$$\begin{aligned} Q^{(k+1)}(T_r, T_s, a) = \bar{R}(s,a) &+ P^{\rm s}_a\, \mathbb{E}_{T_s'}[V^{(k)}(T_s, T_s')] \\ &+ P^{\rm f}_a\, \mathbb{E}_{T_s'}[V^{(k)}((T_r{-}1)^+, T_s')], \end{aligned}$$

with $T_s' = \max\{T_s - 1, \tilde{T}_s\}$, $\tilde{T}_s \sim p_T$, and $(P_a^{\rm succ}, P_a^{\rm fail}) = (p_s, 1{-}p_s)$ if $a = 1$, and $(0, 1)$ if $a = 0$.

Monotonicity in T_r. Define $\Delta^{(k)}(T_r, T_s, a) := Q^{(k)}(T_r{+}1, T_s, a) - Q^{(k)}(T_r, T_s, a)$.

Induction hypothesis: $\Delta^{(k)}(T_r, T_s, a) \ge 0$ for all (T_r, T_s) with $T_r < K$ and all a. The base case $k = 0$ holds since $Q^{(0)} \equiv 0$.

Inductive step: For action $a = 1$:

$$\begin{aligned} &\Delta^{(k+1)}(T_r, T_s, 1) \\ &\quad = \big[\bar{R}((T_r{+}1, T_s), 1) - \bar{R}((T_r, T_s), 1)\big] \\ &\qquad + (1{-}p_s)\mathbb{E}_{T_s'}\big[V^{(k)}(T_r, T_s') - V^{(k)}((T_r{-}1)^+, T_s')\big]. \end{aligned}$$

The reward difference equals $r(1{-}p_s)\big[\mathbb{1}\{T_r \ge 1\} - \mathbb{1}\{T_r > 1\}\big] \ge 0$. The bracketed expectation is non-negative by the induction hypothesis. Hence $\Delta^{(k+1)}(T_r, T_s, 1) \ge 0$.

For action $a = 0$:

$$\begin{aligned} &\Delta^{(k+1)}(T_r, T_s, 0) \\ &\quad = \big[\bar{R}((T_r{+}1, T_s), 0) - \bar{R}((T_r, T_s), 0)\big] \\ &\qquad + \mathbb{E}_{T_s'}\big[V^{(k)}(T_r, T_s') - V^{(k)}((T_r{-}1)^+, T_s')\big]. \end{aligned}$$

The reward difference is $r\big[\mathbb{1}\{T_r \ge 1\} - \mathbb{1}\{T_r > 1\}\big] \ge 0$. Again, the expectation is non-negative by induction. Hence $\Delta^{(k+1)}(T_r, T_s, 0) \ge 0$.

Since $V^{(k)}(T_r, T_s) = \max_a Q^{(k)}(T_r, T_s, a)$ and the max of non-decreasing functions is non-decreasing, we have $V^{(k)}(T_r, T_s) \le V^{(k)}(T_r{+}1, T_s)$. Taking $k \to \infty$ yields $V(T_r, T_s) \le V(T_r{+}1, T_s)$.

Monotonicity in T_s. An analogous argument, using the fact that larger T_s yields $T_s' = \max\{T_s{-}1, \tilde{T}_s\}$ stochastically larger and better next-state values upon success, establishes $V(T_r, T_s) \le V(T_r, T_s{+}1)$. □

Appendix C
Proof of Theorem 2

The proof relies on monotonicity properties of the action gap $\Delta Q(T_r, T_s) = Q(T_r, T_s, 1) - Q(T_r, T_s, 0)$.

Lemma 7 (Action gap monotonicity)**.** *$\Delta Q(T_r, T_s)$ is non-increasing in T_r and non-decreasing in T_s.*

Proof. We first derive an explicit expression for the action gap. Using the transition $T_{r,n+1} = T_{s,n}$ upon successful delivery:

$$\begin{aligned} Q(T_r, T_s, 1) &= -c + r\big[p_s + (1{-}p_s)\mathbb{1}\{T_r > 1\}\big] \\ &\qquad + \mathbb{E}_{T_s'}\big[p_s V(T_s, T_s') + (1{-}p_s)V((T_r{-}1)^+, T_s')\big], \\ Q(T_r, T_s, 0) &= r\,\mathbb{1}\{T_r > 1\} + \mathbb{E}_{T_s'}\big[V((T_r{-}1)^+, T_s')\big], \end{aligned}$$

where $T_s' = \max\{T_s{-}1, \tilde{T}_s\}$ with $\tilde{T}_s \sim p_T$. Subtracting yields

$$\begin{aligned} \Delta Q(T_r, T_s) &= -c + r p_s \mathbb{1}\{T_r \le 1\} \\ &\qquad + p_s \mathbb{E}_{T_s'}\big[V(T_s, T_s') - V((T_r{-}1)^+, T_s')\big]. \end{aligned} \tag{22}$$

Monotonicity in T_r. From (22), we have

$$\begin{aligned} &\Delta Q(T_r{+}1, T_s) - \Delta Q(T_r, T_s) \\ &\quad = r\, p_s \big[\mathbb{1}\{T_r{+}1 \le 1\} - \mathbb{1}\{T_r \le 1\}\big] \\ &\qquad - p_s\, \mathbb{E}_{T_s'}\big[V(T_r, T_s') - V((T_r{-}1)^+, T_s')\big]. \end{aligned}$$

For $T_r = 0$, both the indicator difference and value difference vanish. For $T_r = 1$, the indicator difference is -1, contributing $-rp_s < 0$, while the value term is non-positive by Lemma 1. For $T_r \ge 2$, the indicator difference is zero and the value term remains non-positive. Thus $\Delta Q(T_r{+}1, T_s) \le \Delta Q(T_r, T_s)$ for all T_r.

Monotonicity in T_s. Let $T_s' = \max\{T_s - 1, \tilde{T}_s\}$ and $T_s'' = \max\{T_s, \tilde{T}_s\}$ for the same realization of $\tilde{T}_s$. Note $T_s'' \geq T_s'$ always. A coupling argument on $\tilde{T}_s$ shows that the increase in $V(T_s{+}1, T_s'')$ versus $V(T_s, T_s')$ dominates any increase in $V((T_r{-}1)^+, T_s'')$ versus $V((T_r{-}1)^+, T_s')$, using monotonicity of V from Lemma 1. Hence $\Delta Q(T_r, T_s) \leq \Delta Q(T_r, T_s{+}1)$. □

We now prove the theorem. Since $\Delta Q(\cdot, T_s)$ is non-increasing in T_r, its sign changes at most once as T_r increases from 0 to K. Define

$$\theta(T_s) := \max\big\{T_r : \Delta Q(T_r, T_s) > 0\big\},$$

and set $\pi^*(T_r, T_s) = 1$ for $T_r \leq \theta(T_s)$, and 0 otherwise. The single sign-change property ensures this coincides with the true maximizer, establishing the threshold form.

For the monotonicity of thresholds, fix $T_s < K$ and note that $\Delta Q(\theta(T_s), T_s) \geq 0$ by definition. Since ΔQ is non-decreasing in T_s,

$$\Delta Q(\theta(T_s), T_s + 1) \geq \Delta Q(\theta(T_s), T_s) \geq 0,$$

so state $(\theta(T_s), T_s{+}1)$ lies in the "send" region, implying $\theta(T_s) \leq \theta(T_s + 1)$. □

Appendix D
Proof of Theorem 3

(i) Empty receiver, positive net gain. Using (22) with $T_r = 0$:

$$\begin{aligned}\Delta Q(0, T_s) &= -c + r\, p_s\, \mathbb{1}\{0 \leq 1\}\\ &\quad + p_s\, \mathbb{E}_{T_s'}\big[V(T_s, T_s') - V(0, T_s')\big]\\ &= (p_s\, r - c) + p_s\, \mathbb{E}_{T_s'}\big[V(T_s, T_s') - V(0, T_s')\big].\end{aligned}$$

Since $p_s\, r - c > 0$ by assumption and $V(T_s, T_s') \geq V(0, T_s')$ by Lemma 1 (with $T_s \geq 1 > 0$), we have $\Delta Q(0, T_s) > 0$. Thus $\pi^*(0, T_s) = 1$. □

(ii) Receiver strictly fresher than sender. Fix any state with $T_r > T_s$. Using (22):

$$\begin{aligned}\Delta Q(T_r, T_s) &= -c + r\, p_s\, \mathbb{1}\{T_r \leq 1\}\\ &\quad + p_s\, \mathbb{E}_{T_s'}\big[V(T_s, T_s') - V((T_r{-}1)^+, T_s')\big].\end{aligned} \tag{23}$$

Since $T_r > T_s \geq 1$, we have $T_r \geq 2$, so $\mathbb{1}\{T_r \leq 1\} = 0$ and $(T_r - 1)^+ = T_r - 1 \geq 1$.

Also, $T_r > T_s$ implies $T_r - 1 \geq T_s$, so by Lemma 1:

$$V(T_s, T_s') \leq V(T_r - 1, T_s').$$

Therefore:

$$\begin{aligned}\Delta Q(T_r, T_s) &= -c + p_s\, \mathbb{E}_{T_s'}\big[V(T_s, T_s') - V(T_r{-}1, T_s')\big]\\ &= -c + p_s\, \mathbb{E}_{T_s'}\underbrace{\big[V(T_s, T_s') - V(T_r{-}1, T_s')\big]}_{\leq 0}\\ &\leq -c < 0.\end{aligned}$$

Thus $\pi^*(T_r, T_s) = 0$. □

(iii) Monotonicity of transmission. Suppose it is optimal to transmit at (T_r, T_s) for some $T_r > 0$, i.e., $\Delta Q(T_r, T_s) > 0$. By Lemma **??**(i):

$$\Delta Q(T_r - 1, T_s) \geq \Delta Q(T_r, T_s) > 0.$$

Hence it is also optimal to transmit at $(T_r - 1, T_s)$. □

Appendix E
Proof of Theorem 4

Fix a threshold policy with parameter $\theta \in \{0, 1, \ldots, K\}$. A renewal cycle begins each time a transmission succeeds, at which point the receiver's timer resets to $T_r = K$.

During slots $T_r = K, K{-}1, \ldots, \theta{+}1$, the sender waits; this *waiting phase* lasts $K - \theta$ slots. Once $T_r = \theta$, transmission attempts begin and continue until success. Since each attempt succeeds independently with probability p_s, the *attempting phase* has duration $N_A \sim \mathrm{Geom}(p_s)$.

By the renewal-reward theorem, the average reward equals $\rho(\theta) = \mathbb{E}[R_{\text{cycle}}]/\mathbb{E}[T_{\text{cycle}}]$. The expected cycle length is

$$\mathbb{E}[T_{\text{cycle}}] = (K - \theta) + \frac{1}{p_s}.$$

For the reward, the waiting phase contributes $r(K{-}\theta)$ since the receiver holds a valid sample throughout. During the attempting phase, the sender incurs expected cost c/p_s, while the receiver earns reward r in slot i only if $T_r = \theta{-}(i{-}1) > 0$, i.e., for $i \leq \theta$. Thus the expected happiness reward is $r \cdot \mathbb{E}[\min(N_A, \theta)] = r(1 - (1 - p_s)^\theta)/p_s$. Combining terms,

$$\rho(\theta) = r - \frac{c + r(1 - p_s)^\theta}{p_s(K - \theta) + 1}.$$

To maximize $\rho(\theta)$, define $g(\theta) := (K - \theta + 1)(1 - p_s)^{\theta - 1}$. One verifies that g is strictly decreasing in θ, and that $\rho(\theta) > \rho(\theta - 1)$ if and only if $g(\theta) > c/(p_s r)$. Hence the optimal threshold is

$$\theta^* = \max\big\{\theta \geq 1 : g(\theta) > c/(p_s r)\big\},$$

with the convention $\theta^* = 0$ if the set is empty. □