Early Sampling: Sampling with Incomplete Queue State Information to Minimize the Age

Sahan Liyanaarachchi Sennur Ulukus
Department of Electrical and Computer Engineering
University of Maryland, College Park, MD 20742
sahanl@umd.edu *ulukus@umd.edu*

***Abstract*—In many existing communication models, the channel state (i.e busy or idle) is conveyed to the sampler via ACKs which are often assumed to be instantaneous. Previous literature shows that in this ideal feedback setting, an optimal sampling policy that minimizes the age of information (AoI), should not sample when the channel is busy, and therefore, must always wait for the ACK of the previous sample before taking the next sample. However, this may not be optimal when the feedback channel (backward channel) has a random delay. In this work, we study the structure of the optimal sampling policy to minimize the AoI when the channel state (i.e., the forward queue state) is not immediately perceived by the sampler due to random delays in the feedback channel. In this setting, we show that it is not always optimal to wait for ACKs before sampling, and thus, early sampling with the available incomplete channel state information may be better. We show that, under certain conditions on the distribution of the ACK delays, the (asymptotically) optimal policy reduces to a mixture of two threshold policies.**

I. INTRODUCTION

Sampling for data freshness has been an increasingly important problem due to its wide use cases in the wireless domain. Data freshness is often measured through a non-decreasing function of the age of information (AoI), simplest being the instantaneous age of the process itself given by $\Delta_t = t - u(t)$, where $u(t)$ is the generation time of the freshest sample obtained from the observed process [1]. Many of the previous work in this area involves modelling the communication system as an *enqueue-and-forward* model [2]–[4], where the updates are generated randomly and enqueued before being transmitted to the receiver. However, recent works involve the *generate-at-will* model introduced in [5] where the sampler has the ability to generate a sample when needed. In [6], the *generate-at-will* model has been studied for general age penalty functions, where it is shown that the *zero-wait* policy is not always optimal.

Most of the existing communication models consist of a single channel with a transmission delay or erasures, and assume instantaneous feedback about channel state [7]–[9]. However, in a practical communication system, the channel carrying the feedback/ACK is non-ideal. The work in [10], [11] introduces a two-channel model, with a forward channel and a backward channel, to address this problem. This has been further extended in [12] by introducing an unreliable communication channel with packet drops. In all these models, it is assumed that the next sample should always be taken after receiving the ACK of the previous sample. Our paper

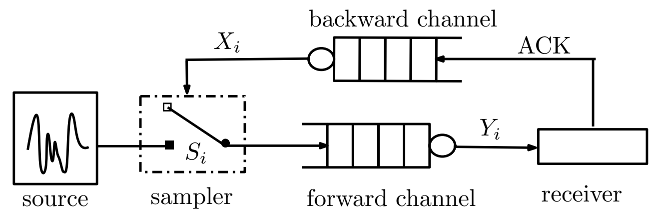


Fig. 1: System model.

extends this line of work by considering the possibility of *early sampling*, where new samples may be generated before ACKs of previous samples are received, as needed. The closest to our work is [13] where a set of rules to be followed by a good sampling policy is first assumed and the performance of the constructed policy is compared to a lower bound and upper bound generated by policies which have the complete channel state information when taking a sample. We note that some of the characteristics that have been assumed for a good policy in [13] can be proven to be true in our setting.

Consider a two-channel communication model as shown in Fig. 1, where a transmitter observes a stochastic source and transmits the samples over a channel with a random delay (forward channel). Once a sample arrives at the receiver, it generates an acknowledgement message (ACK) which is sent to the transmitter again via a channel with a random delay (backward channel). The transmitter perceives the channel state through these ACKs. If these ACKs arrived at the transmitter instantaneously as they were generated, then the transmitter would always know the exact channel state of the forward channel at any given time. Under such circumstances, an optimal sampling policy should not generate a new sample when the channel is busy [6]. If there is a delay in ACKs, the forward channel could become free at a time much earlier than the time at which the transmitter perceives it to be free. In this scenario, a naive approach would be to always wait for the ACK of the previous sample before sampling the next. In this work, we explore how to exploit the time window between knowing that the channel is free and the time at which the channel is actually free, by allowing the transmitter to sample before the arrival of ACKs of the previous samples.

Here, we consider a *generate-at-will* model with preemptive transmissions [14]–[16], which enables the transmitter to take a sample and transmit it at any time. Since we allow sampling

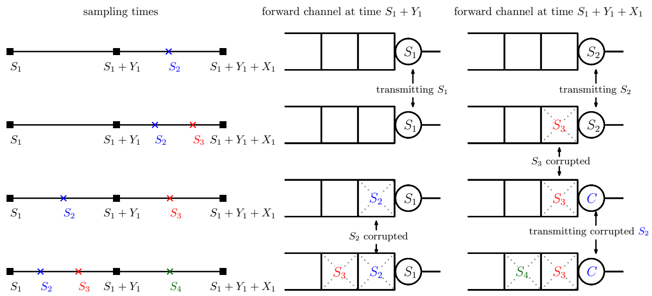


Fig. 2: Transmitting before arrival of ACKs.

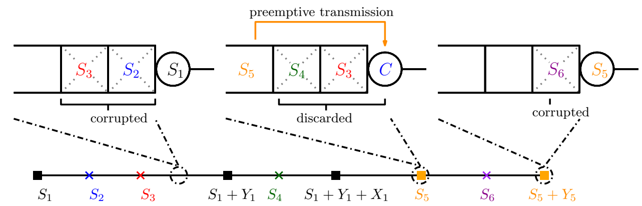
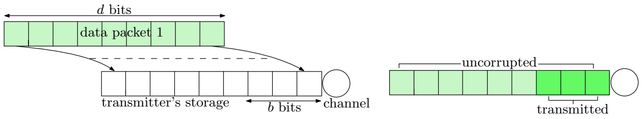


Fig. 3: Preemptive transmission.

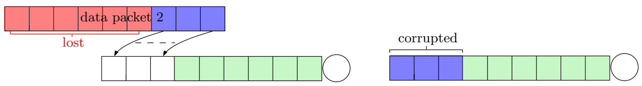
before ACKs, the following questions must be addressed first:

- ***What does it mean to transmit when the channel is busy?*** We model the forward channel as a queue with possible preemption. If a sample is generated and attempted to be transmitted when the channel is busy, we assume that this new sample gets corrupted during its storage into the queue. However, this corrupted sample does not affect the transmission of the sample that is being transmitted. As the queue passively serves what is stored, once the current sample has finished its transmission, the next sample (corrupted) in the queue will be served unless a preemptive transmission is initiated by the transmitter (see Fig. 2).
- ***How are the ACKs generated at the receiver?*** When a new sample is received, an ACK is generated which contains the delivery time of the sample. If the new sample is received while sending back the ACK of an old sample, then as was in the transmitter side, we consider that the newly generated ACK will be corrupted. Under certain conditions on the distribution of the transmission delay and the ACK delay, the collision in ACKs can be eliminated. These conditions will be discussed in the next section and are assumed to hold throughout the paper.
- ***When are preemptive transmissions initiated?*** If a corrupted sample gets transmitted by the forward channel, it would not reduce the age of the process and therefore would have wasted valuable transmission time on a corrupted sample. If the transmitter knows that a corrupted sample is being transmitted, then it is always better to cancel the current transmission and transmit an uncorrupted sample if possible. On the other hand, it is not always ideal to cancel the transmission of an uncorrupted sample. Therefore, we assume that the transmitter would only initiate a preemptive transmission if the transmitter is certain that a corrupted sample is being transmitted when we take the new sample. We further assume that, if a preemptive transmission is initiated, then all samples (corrupted) in the queue would be dropped (see Fig. 3).
- ***Why are enqueued samples assumed to be corrupted?*** In the actual physical model (see Fig. 4), we assume that the data packets involved are of a fixed size (d bits) and the transmitter is only capable of accommodating (storing) d bits at any given time. Suppose these d bits are initially empty. When a sample is taken, it will be written on to these d bits and these d bits will be sequentially transmitted across the channel (say b bits at a time) where the total time to transmit these d bits would correspond to the channel service time. Once b bits have been transmitted, b bits from the transmitter's storage will be relieved. Once a new sample is taken, the bits of the new data packet is written onto the next available bits in the storage space of the transmitter. In doing so, some of the bits of this new sample would be lost and therefore we assume that the current sample is corrupted. Once the channel has finished serving the last b bits of the initial data packet, since the next b bits in the transmitter's storage is non-empty, it will start another cycle of transmission and therefore will start serving the corrupted sample until a total of d bits have been transmitted. If the transmitter has the knowledge that a corrupted sample is being served, it can initiate a preemptive transmission by clearing the d storage bits and storing a new sample in them. If a new data packet arrives when the storage is full, then that data packet is completely lost. This type of a physical model is common in small IoT devices which are often used in remote estimation settings. Therefore, we abstract this physical model with a queue where the queued up samples are considered to be corrupted with probability 1 if the queue is currently serving a sample. This queuing model is a variant of the erasure-queue channel which is commonly used in quantum communication models where stored qubits suffer from a waiting time dependent decoherence [17]. The version of the problem where multiple samples may be saved in the queue and served sequentially over time is an interesting extension of the simpler model studied in this paper.

Under the above model assumptions, we show that the system model oscillates between two distinct states, where in state 1 we sample knowing that the channel is busy, and in state 2 we sample knowing that the channel is free (or can be made free via preemption). We show that the structure of the optimal stationary deterministic sampling policy that minimizes the average age of information is a mix of two threshold policies, one for each state. Due to space limitations here, proof details, extra remarks and figures are presented in the longer version of the paper in [18].



(a) Arrival of a data packet when the storage is empty.



(b) Arrival of a data packet when the storage is non-empty.

Fig. 4: Physical model.

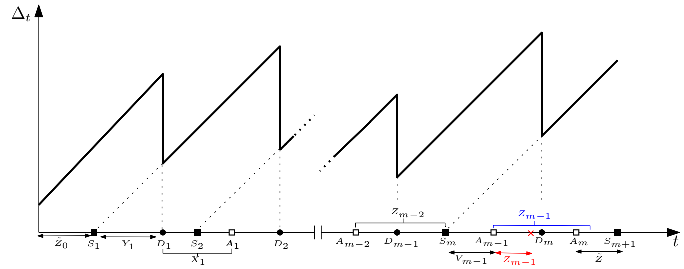


Fig. 5: A typical transition cycle.

II. Problem Formulation

We say that a sample was correctly received, if it was not corrupted before the transmission by the forward channel. Let $S_0, S_1, \ldots$ denote the sequence of sampling times of correctly received samples where $S_i \leq S_{i+1}$. Let the sequence of the forward channel service times (transmission delays) and ACK delays be represented by $\{Y_i \overset{\text{iid}}{\sim} Y\}_{i=0}^{\infty}$ and $\{X_i \overset{\text{iid}}{\sim} X\}_{i=0}^{\infty}$, respectively. We assume that Y and X have finite first and second moments. Denote by D_i the delivery time of the ith correctly received sample and by N_i the total number of samples taken by the time S_i. Let π be a causal stationary deterministic policy, f_{max} be the maximum allowable sampling rate, and Δ_t be the instantaneous AoI of the samples at the receiver. Then, the problem of minimizing the average AoI can be expressed as follows,

$$\begin{aligned} \min_{\pi} \quad & \limsup_{n\to\infty} \frac{\mathbb{E}\left[\int_0^{D_n} \Delta_t \, dt\right]}{\mathbb{E}[D_n]} \\ \text{s.t.} \quad & \liminf_{n\to\infty} \mathbb{E}\left[\frac{S_n}{N_n}\right] \geq \frac{1}{f_{max}} \end{aligned} \tag{1}$$

Solving for the optimal solution in problem (1) can deem difficult for a general distribution of X and Y due to complications such as ACK collisions. Therefore, to simplify the problem, we assume that the distributions of ACK delays and forward channel service times satisfy the condition $X \leq Y$ almost surely (a.s.). This can be argued to be a reasonable assumption in many practical scenarios since in a general communication protocol, the packet size of ACKs is much smaller than data packets, and hence, would almost surely be received faster than the data packets.

Under the above assumption, Lemma 1 below uncovers an important structural property of the optimal policy.

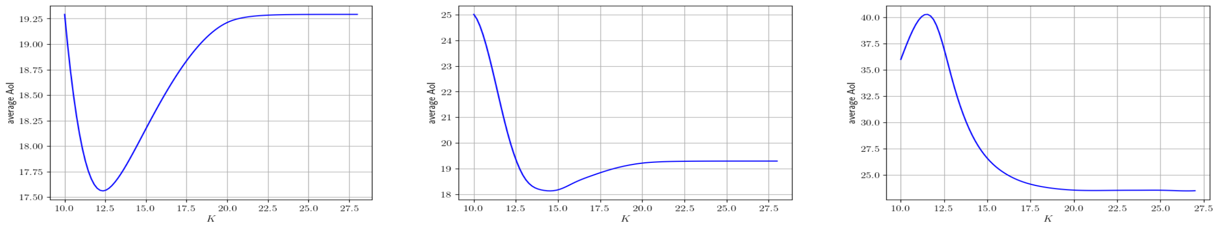
Lemma 1 *If $X \leq Y$ a.s., then under an optimal sampling policy, one should not take more than 1 sample before receiving the ACK of the previous sample.*

Under the assumption that $X \leq Y$ a.s., Lemma 1 shows that only at most one sample may be taken before receiving the ACK of the previous sample. Moreover, the delivery time of this sample (either corrupted or not) would fall after the time of reception of the ACK of the previous sample. Therefore, there will be no collision in ACKs.

Let S_i, D_i and A_i be the sampling time, delivery time and the acknowledgement time of a correctly received sample. Let $\tilde{S}_{i+1}$ be the sampling time of the next sample. Since we send back the delivery time D_i along with the ACK, if the next sample was taken at a time $\tilde{S}_{i+1} < A_i$, then at time A_i we exactly know if the new sample was corrupted or not. When we receive the ACK at time A_i, if $\tilde{S}_{i+1} < D_i$, we know the new sample got corrupted, and therefore, the channel is serving a corrupted sample. If we know the channel is serving a corrupted sample, we can free up the channel through a preemptive transmission of the next sample. At A_i, if $D_i \leq \tilde{S}_{i+1} < A_i$, we know the new sample will be successfully transmitted, and therefore, the channel is busy serving an uncorrupted sample. If $A_i < \tilde{S}_{i+1}$, then we know the channel is definitely free.

Therefore, we can characterize the system into two states based on the information available to the transmitter when an ACK arrives. In state 1, we have the knowledge that the channel is busy serving an uncorrupted sample and in state 2 we have the knowledge that the channel is free (or can be made free via preemption). If the system is in state 1, when we receive an ACK if we had already taken the next sample and know it was corrupted (depicted by red Z_{m-1} in Fig. 5) or if we have not taken the next sample by the time we received the ACK (depicted by blue Z_{m-1} in Fig. 5), the system would make a transition from state 1 to state 2. Otherwise it would stay in state 1. If the system is in state 2 and we take a sample then it would directly revert back to state 1. Thus, the system model would consist of cycles of multiple state 1 to state 1 transitions, followed by a state 1 to state 2 to state 1 transition. Fig. 5 shows one such transition cycle.

After receiving an ACK, let Z be the waiting time before taking the next sample when in state 1 and $\tilde{Z}$ be the waiting time before taking the next sample when in state 2. Any sampling policy under consideration can be characterized using the waiting times in these two system states. Therefore, the goal of this paper is to find these waiting times based on the system state, previous transmission times and delivery times available to the system when an ACK arrives. Since we are only considering the stationary deterministic policies, the problem (1) reduces to determining the optimal waiting times for one transition cycle. A stationary policy in this setting



(a) $1/f_{max} = 8$ (b) $1/f_{max} = 14$ (c) $1/f_{max} = 25$

Fig. 6: Variation of average AoI with K for $Y \sim (10 + Exp(1))$ and $X \sim Uniform[0, 10]$.

is defined as a policy which induces a stationary distribution among these two system states.

Let τ be the time duration of one transmission cycle and N_τ be the number of samples taken in that transmission cycle. Then, the problem (1) can be expressed as,

$$\min_{\pi} \quad \frac{\mathbb{E}\left[\int_0^\tau \Delta_t \, dt\right]}{\mathbb{E}[\tau]}$$
$$\text{s.t.} \quad \frac{\mathbb{E}[\tau]}{\mathbb{E}[N_\tau]} \geq \frac{1}{f_{max}} \tag{2}$$

We say that a policy is optimal if it solves (2) exactly, and a policy is asymptotically optimal if the policy becomes optimal as f_{max} goes to ∞. Next, we present the main results of our work which gives the structural characteristics of optimal and asymptotically optimal policies.

III. Main Result

Theorem 1 *If $X \leq Y$ a.s. and $\inf X + \inf Y \leq \sup Y$, then the optimal waiting time $\tilde{Z}$ in state 2 must be a function of the previous transmission time and the ACK delay. The asymptotically optimal waiting time Z in state 1 should be a function of V which is the time elapsed from previous sampling time to the time we received the ACK and further $Z(V) + V = K$, where K is some constant.*

Remark 1 *The condition $\inf X + \inf Y \leq \sup Y$ ensures that the expectations computed are finite and this condition is true for many practical transmission delay distributions such as shifted exponential distribution.*

Theorem 1 implies that under an optimal policy, the waiting time in state 2 should be a function of the previous transmission delay and ACK delay. Further, under an asymptotically optimal policy, when in state 1, one should not wait more than a constant time period K to obtain the next sample. Using structural properties obtained in Theorem 1, we can further simplify (2) as given by Lemma 2.

Lemma 2 *If $X \leq Y$ a.s. and $\inf X + \inf Y \leq \sup Y$, then the optimization problem in (2) is equivalent to the following optimization problem,*

$$\min_{K, \tilde{Z}(\tilde{Y}, \tilde{X})} \quad \frac{(1-p)\mathbb{E}[(\tilde{X} + \tilde{Y} + \tilde{Z})^2] + pK^2}{2\Big((1-p)\mathbb{E}[\tilde{X} + \tilde{Y} + \tilde{Z}] + pK\Big)} + \mathbb{E}[Y]$$
$$\text{s.t.} \quad (1-p)\mathbb{E}[\tilde{X} + \tilde{Y} + \tilde{Z}] + pK \geq \frac{1 + \mathbb{P}(Y > K)}{f_{max}} \tag{3}$$

where $p = \mathbb{P}(Y < K < Y + X)$, K is the waiting time in state 1 and $\tilde{Z}$ is the waiting time in state 2 which is a function of the previous transmission time $\tilde{Y}$ and ACK delay $\tilde{X}$ which belongs to the set $\{\tilde{Y} > K\} \cup \{\tilde{Y} + \tilde{X} < K\}$.

Theorem 2 *If $\mathbb{E}[Y^2] < \infty$, then the optimal policy that minimizes (3) achieves a lower average AoI than any optimal policy that always waits for ACKs before taking the next sample.*

Due to the non-convex nature (see Fig. 6) of the optimization problem presented in (3), jointly optimizing with respect K and $\tilde{Z}(\tilde{Y}, \tilde{X})$ is extremely difficult. However, for a fixed K, the problem is convex [18, Appendix A] with respect to the functional $\tilde{Z}(\tilde{Y}, \tilde{X})$. Therefore, we optimally solve $\tilde{Z}(\tilde{Y}, \tilde{X})$ for a fixed K and present a descent type algorithm to find K. The algorithm is presented as Algorithm 1.

Theorem 3 *For a fixed K, the optimal functional $\tilde{Z}$ that solves (3) is given by,*

$$\tilde{Z} = \Big(\beta - (\tilde{X} + \tilde{Y})\Big)^+ \tag{4}$$

and $\beta > 0$ and satisfies,

$$(1-p)\mathbb{E}\left[\max(\beta, \tilde{X} + \tilde{Y})\right] + pK = \max\{\phi(K), \psi(K, \beta)\} \tag{5}$$

where $\phi(K)$ and $\psi(K)$ are given by,

$$\phi(K) = \frac{1 + \mathbb{P}(Y > K)}{f_{max}} \tag{6}$$
$$\psi(K) = \frac{(1-p)\mathbb{E}\left[\max(\beta^2, (\tilde{X} + \tilde{Y})^2)\right] + pK^2}{2\beta} \tag{7}$$

For the distributions of X and Y which do not satisfy the condition $\inf X + \inf Y \leq \sup Y$ given in Theorem 1, we present a periodic sampling policy that performs better that a policy that always waits for ACKs to obtain the next sample. Let the set Ω be defined as follows,

$$\Omega = \left\{K : \max\left\{\sup Y, \frac{1}{f_{max}}\right\} < K < \inf Y + \inf X\right\} \tag{8}$$

Algorithm 1 Algorithm for finding optimal K and β

Require: $K = \inf Y$, $l_0 = 0$, $\{u_0, old, new, K_0\}$ sufficiently large, $\{\lambda, \epsilon\}$ sufficiently small $K^* = K_0$, $\beta^* = \beta_{K_0}$ (optimal β for $K = K_0$), $old > new$

while $(old > new$ **and** $K < K_0)$ **do**

  $old = new$, $u = u_0$, $l = l_0$

  while $(u - l) > \epsilon$ **do**

    $B_K = \{Y < K < Y + X\}$

    $\beta_K = \frac{u+l}{2}$, $\quad p = \mathbb{P}(B_K)$

    $Q = (1-p)\mathbb{E}\left[\max\{\beta_K^2, (X+Y)^2\}|\overline{B_K}\right] + pK^2$

    $R = (1-p)\mathbb{E}\left[\max\{\beta_K, (X+Y)\}|\overline{B_K}\right] + pK$

    $T = \frac{1+\mathbb{P}(Y>K)}{f_{max}}$

    $diff = R - \max\{T, \frac{Q}{2\beta_K}\}$

    if $diff \leq 0$ **then**

      $l = \beta_K$

    else

      $u = \beta_K$

    end if

  end while

  $new = \frac{Q}{2R}$

  if $old - new > 0$ **then**

    $K^* = K$, $\beta^* = \beta_K$

  end if

  $K = K + \lambda$

end while

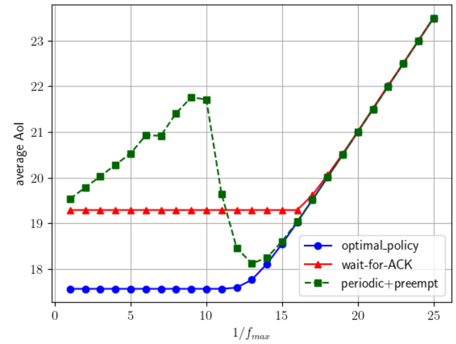
Theorem 4 *If $\Omega \neq \emptyset$, then there exists a periodic sampling policy which always has a lower average AoI than any optimal policy constructed where one always waits for an ACK before sampling the next. The period K_p^* of the optimal periodic sampling policy is given by $K_p* = \inf \Omega$.*

IV. Numerical Results

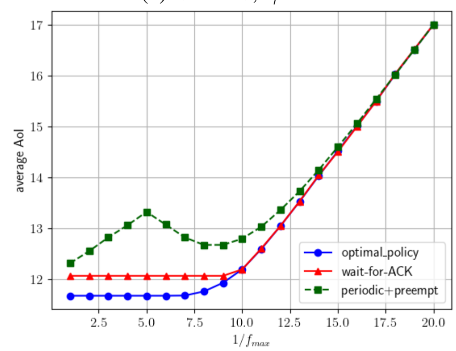
In this section we compare the performance of our *optimal* policy with the following two policies.

- *Wait-for-ACK policy*: In this policy, the sampler always waits for the ACK of the previous sample and then chooses an optimal waiting time based on the total time elapsed from the previous sampling time. The optimal waiting time can be found in [10], [18].
- *Periodic + preempt* policy: This is a naive policy where we periodically sample and transmit. Upon receiving an ACK, if it indicates that the channel is currently serving a corrupted sample, a preemptive transmission will be initiated when transmitting the next sample.

To evaluate the performance, we plot the variation of the average AoI against the minimum allowed sampling period ($1/f_{max}$) for different distributions of X and Y. In the first experiment, we take the distribution of Y to be a shifted exponential (i.e., $Y = C + \bar{Y}$ where $\bar{Y} \sim exp(\gamma)$ and $C > 0$) and we take the distribution of X to be uniform in the interval from zero to $\inf Y$ (i.e., $X \sim Unif[0, C]$). In the second experiment, the distribution of Y is again chosen to be the same shifted exponential as before but here we take X to take



(a) $C = 10$, $\gamma = 1$



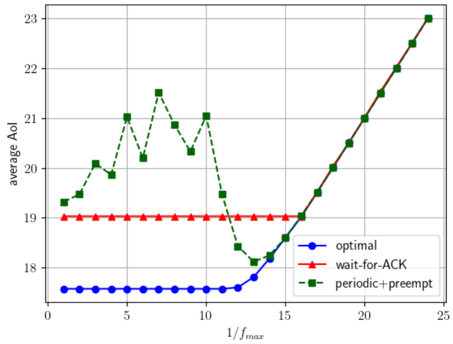
(b) $C = 5$, $\gamma = 0.5$

Fig. 7: Variation of the average AoI with maximum allowable sampling rate for $Y \sim (C + Exp(\gamma))$ and $X \sim Uniform[0, C]$.

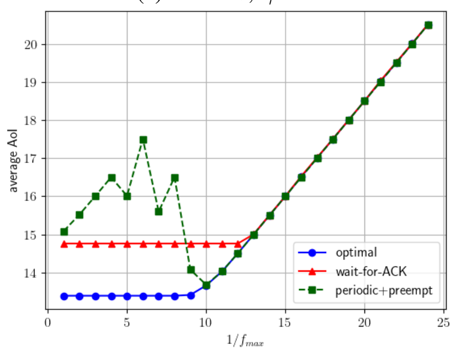
the constant value $\frac{C}{2}$. In the third experiment, we set Y to be a constant C and set X to have the same uniform distribution as in first experiment.

As seen in Figs. 7, 8 and 9, for higher values of f_{max} (lower values of $1/f_{max}$), our policy is significantly better than the other two policies considered. However, for lower values of f_{max}, the average AoI of all three policies tend to be similar. This is because, even though our policy allows sampling at a faster rate when in state 1 as we can compensate by waiting longer in state 2, if the rate of sampling in state 1 is too fast, corrupted samples will arise more frequently and as a result the required waiting time to satisfy the sampling constraint for low values of f_{max} will be much larger. Additionally, long cycles of state 1 to state 1 transitions will be less often in this case. Therefore, the optimal value K to sample in state 1 would generally increase with $\frac{1}{f_{max}}$. As K increases, corrupted samples will be less frequent and ACKs would arrive before K time units have elapsed more often. Hence, the similarity in the three curves for lower values of f_{max}.

Fig. 7 shows that when the variation of Y is greater, then the periodic sampling policy is far from optimal, however when the values of Y become concentrated at its lower bound ($\gamma = 1$ or $Y = C$), the periodic sampling policy closely follows the optimal policy when $1/f_{max} > \inf Y$. However, at any given value of $1/f_{max}$, the periodic sampling policy never goes below the curve of the optimal policy. This indicates that even



(a) $C = 10$, $\gamma = 1$



(b) $C = 8$, $\gamma = 2$

Fig. 8: Variation of the average AoI with maximum allowable sampling rate for $Y \sim (C + Exp(\gamma))$ and $X = C/2$.

in the absence of the sampling constraint, a periodic sampling policy with any period (i.e., sampling at a rate other than f_{max}) will not be better than the optimal policy constructed here.

As seen by the presented figures, our simulation results validate our theoretical development of the optimal policy for the given system model.

V. Conclusion

In this work, we have introduced a new system model which facilitates early sampling and transmission before receiving an ACK. We have shown through theoretical results and simulations that it is not always optimal to wait for ACKs before sampling when there is a delay in the feedback channel. The system model introduced here may be an optimistic abstraction of what is really happening in the real world scenarios when collisions in transmissions occur (here we assumed that the already transmitting packet is not affected but the new arriving packet is corrupted; in real world scenarios both packets may be corrupted or none may be corrupted and the new arriving packet may be queued up). Our work could provide a useful first perspective when tackling more complex scenarios. Future directions of work may include considering that both the transmitting and the new sample get corrupted in case of a collision, samples obtained before an ACK are not corrupted but just queued up to be transmitted in the forward channel, and corrupted samples are transmitted without any preemptive transmissions.

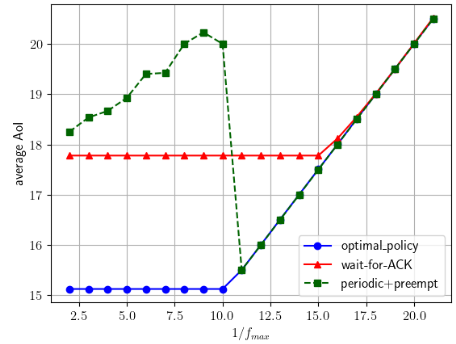


Fig. 9: Variation of the average AoI with maximum allowable sampling rate for $Y = 10$ and $X \sim Uniform[0, 10]$.

References

[1] R. D. Yates, Y. Sun, D. R. Brown, S. K. Kaul, E. Modiano, and S. Ulukus. Age of information: An introduction and survey. *IEEE Journal on Selected Areas in Communication*, 39(5):1183–1210, May 2021.

[2] S. K. Kaul, R. D. Yates, and M. Gruteser. Real-time status: How often should one update? In *IEEE Infocom*, March 2012.

[3] R. D. Yates and S. K. Kaul. Real-time status updating: Multiple sources. In *IEEE ISIT*, July 2012.

[4] R. D. Yates and S. K. Kaul. The age of information: Real-time status updating by multiple sources. *IEEE Transactions in Information Theory*, 65(3):1807–1827, March 2019.

[5] R. D. Yates. Lazy is timely: Status updates by an energy harvesting source. In *IEEE ISIT*, June 2015.

[6] Y. Sun, E. Uysal, R. D. Yates, C. E. Koksal, and N. B. Shroff. Update or wait: How to keep your data fresh. In *IEEE Infocom*, April 2016.

[7] Y. Sun, Y. Polyanskiy, and E. Uysal. Sampling of the Wiener process for remote estimation over a channel with random delay. *IEEE Transactions on Information Theory*, 66(2):1118–1135, Februaury 2020.

[8] K. Banawan, A. Arafa, and K. G. Seddik. Timely multi-process estimation with erasures. In *Asilomar Conference on Signals, Systems, and Computers*, Octomber 2022.

[9] K. Banawan, A. Arafa, and K. G. Seddik. Timely multi-process estimation over erasure channels with and without feedback: Signal-independent policies. *IEEE J. Sel. Areas Inf. Theory*, 4:607–623, November 2023.

[10] C. H. Tsai and C. C. Wang. Age-of-information revisited: Two-way delay and distribution-oblivious online algorithm. In *IEEE ISIT*, June 2020.

[11] C. H. Tsai and C. C. Wang. Jointly minimizing aoi penalty and network cost among coexisting source-destination pairs. In *IEEE ISIT*, July 2021.

[12] J. Pan, A. M. Bedewy, Y. Sun, and N. B. Shroff. Optimizing sampling for data freshness: Unreliable transmission with random two-way delay. In *IEEE Infocom*, May 2022.

[13] C. C. Wang. How useful is delayed feedback in aoi minimization — a study on systems with queues in both forward and backward directions. In *IEEE ISIT*, June 2022.

[14] A. Arafa, R. D. Yates, and H. V. Poor. Timely cloud computing: Preemption and waiting. In *Allerton*, September 2019.

[15] B. Wang, S. Feng, and J. Yang. When to preempt? Age of information minimization under link capacity constraint. *Journal of Communications and Networks*, 21(3):220–232, June 2019.

[16] B. Wang, S. Feng, and J. Yang. To skip or to switch? Minimizing age of information under link capacity constraint. In *IEEE SPAWC*, June 2018.

[17] K. N. Varma and K. Jagannathan. An erasure queue-channel with feedback: Optimal transmission control to maximize capacity. In *IEEE ITW*, April 2023.

[18] S. Liyanaarachchi and S. Ulukus. The role of early sampling in age of information minimization in the presence of ACK delays. Available online at arXiv:2308.07905.