
Faculty of Engineering

Faculty Publications

A Whittle Index Approach to Minimizing Age of Multi-Packet Information in IoT Network

Mianlong Chen, Kui Wu, & Linqi Song

2021

© *Chen et al. 2021. This is an open access article licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License*
(<https://creativecommons.org/licenses/by-nc-nd/4.0/>)

This article was originally published at:

<https://doi.org/10.1109/ACCESS.2021.3059966>

Citation for this paper:

Chen, M., Wu, K., & Song, L. (2021). A Whittle index approach to minimizing age of multi-packet information in IoT network. *IEEE Access*, 9, 31467-31480. DOI: 10.1109/ACCESS.2021.3059966

Received January 15, 2021, accepted February 6, 2021, date of publication February 16, 2021, date of current version March 1, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3059966

A Whittle Index Approach to Minimizing Age of Multi-Packet Information in IoT Network

MIANLONG CHEN¹, KUI WU¹, (Senior Member, IEEE), AND LINQI SONG², (Member, IEEE)

¹Department of Computer Science, University of Victoria, Victoria, BC V8W 3P6, Canada

²Department of Computer Science, City University of Hong Kong, Hong Kong

Corresponding author: Kui Wu (wkui@uvic.ca)

This work was supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC) under Grant RGPIN-2018-03896.

ABSTRACT Age of information (AoI) captures the freshness of information and has been used broadly as an important performance metric in big data analytics in the Internet of Things (IoT). We consider a general scenario where a meaningful piece of information consists of multiple packets and the information is not complete until all related packets have been correctly received. Minimizing AoI in this general scenario is challenging in both scheduling algorithm design and theoretical analysis, because we need to track the history of received packets before a complete piece of information can be updated. We first analyse the necessary condition for optimal scheduling, based on which we present an optimal scheduling method. The optimal solution, however, has high time complexity. To address the problem, we investigate the problem with a special type of learning, i.e., learning in restless multi-armed bandits (RMAB), and propose a Whittle index-based scheduling method. We also propose a new transmission strategy based on erasure codes to improve the performance of scheduling policies in lossy networks. Performance evaluation results demonstrate that our solution outperforms other baseline policies such as greedy policy and naïve Whittle index policy in both lossless and lossy networks.

INDEX TERMS Age of information (AoI), multi-packet information, restless multi-armed bandit.

I. INTRODUCTION

A common objective shared by numerous real-time Internet of Things (IoT) systems, such as autonomous driving, health monitoring, and object tracking, is obtaining the most up-to-date information, since outdated information may lead to terrible consequences. For example, a sensor of self-driving vehicle that measures the proximity to obstacles or other vehicles in the vicinity needs to send new location update at a high frequency. Information carried in a lagged update may be obsolete and result in collisions [1], [2]. As another example, home-care services require that any urgent events be reported within a given deadline. All these applications aim at obtaining the up-to-date status information. Hence, a new metric named *age of information* (AoI), or simply age, was initially proposed in [3] to characterize the freshness of received data at *the destination*. AoI is now playing an important role in big data analytics in IoT systems, because AoI-based data transmission/processing can ensure the usefulness and quality of the collected sensing data.

The associate editor coordinating the review of this manuscript and approving it for publication was Yuan Tian¹.

Unlike traditional metrics such as delay and throughput, AoI depicts the time elapsed since the most-recent status update at the destination. The initial study of AoI can be found in [3], [4]. Since then, AoI has attracted a great deal of research [5]–[10]. An excellent survey could be found at [11].

The majority of existing AoI research adopts the following model: time is slotted; in each timeslot, only one source can be served and only one packet can be transmitted. More importantly, they assume that each packet contains a status update and the AoI is updated upon successful reception of a new packet. The single-packet information assumption, however, should not be the norm, since the single-packet-based AoI model cannot be applied in many real-world applications where a meaningful piece of information needs to be encoded in multiple packets.

Autonomous driving and smart manufacturing are two motivating examples that a status update should be performed on the basis of multiple packets. In autonomous driving, short videos captured by front cameras of a vehicle are critical for decision making. Useful information embedded in these short videos can only be explored by performing intensive computation and analysis on the raw data. Therefore, useful information can only be derived and processed after the in-vehicle

processor or road-side unit (RSU) receives a short video encoded in multiple packets [12]. In smart manufacturing, monitoring units that assemble multiple sensors and communication components are used to monitor the status of running machines, including information such as temperature, speed, depth, and vibration [13]. An effective control decision can be made only after the collection of all needed information. In this context, the absence of any aspect will fail the update of control action. Thus, multiple packets are needed for a valid update in terms of AoI. We are thus motivated to extend the single-packet information model to a more general multi-packet information model.

Such an extension, while conceptually simple, poses technical challenges in both data transmission scheduling and theoretical analysis. From the viewpoint of transmission scheduling, a simple policy that achieves the minimum average AoI in the presence of random information size¹ is still unknown. In terms of theoretical performance analysis, it is extremely difficult, if not impossible, to employ the traditional queuing theory-based analysis [4], [8]–[10], because we need to track a random number of packets to determine the time for AoI update. To be more specific, the number of system status may be non-deterministic if we build a Markov chain to track the history of successfully received packets before an AoI update.

We address all the above challenges in this paper and make the following contributions:

- We study the problem of minimizing the average AoI in a multi-source system where complete information consists of multiple packets and the information size is random. Tracking the AoI in such a system is difficult, since we need to record the time when new information starts transmission and the time when all the packets have been received. To address this, we first derive a necessary condition for an optimal solution, based on which we design an optimal solution.
- To reduce the complexity of the optimal solution, we develop an index-based scheduling method. We avoid traditional queuing theory and cast our problem as a special type of learning problem in the framework of restless multi-armed bandits. For this, we propose a new way to track the AoI indirectly, by *approximating* the time elapsed since last AoI update. Such an approximation not only simplifies the analysis, but also allows us to design a Whittle index-based scheduling method that achieves a near-optimal solution.
- For lossy networks, we further propose a packet transmission strategy that takes advantage of erasure codes for AoI update.
- Using simulation, we evaluate the proposed AoI update strategy and compare it with two baseline strategies,

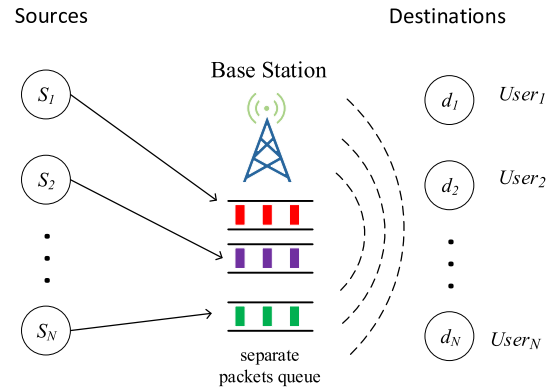


FIGURE 1. The base station (BS) makes transmission schedules to update N users d_1, d_2, \dots, d_N on information of sources s_1, s_2, \dots, s_N , respectively.

greedy policy and naïve Whittle index policy, in both lossless and lossy networks.

The rest of this paper is organized as follows. In Sec. II, we present the system model and the problem formulation. In Sec. III, we analyze the necessary condition for an optimal solution and design an optimal algorithm. In Sec. IV, we develop an index scheduling policy based on the restless multi-armed bandit framework. In Sec. V, we introduce a new transmission strategy for lossy networks based on erasure codes. The simulation results and analysis are presented in Sec. VI. In Sec. VII, we summarize the related work. Finally, this paper is concluded in Sec. VIII.

II. SYSTEM MODEL AND PROBLEM FORMULATION

A. SYSTEM MODEL AND ASSUMPTIONS

We consider a typical IoT network, where a set of N sensing devices (e.g., surveillance video cameras) send information to a base station (BS). The BS forwards the information to the corresponding N destinations (e.g., network users), which process the information and update the AoI accordingly. The system architecture is shown in Fig. 1. Time is divided into intervals (\mathcal{I}) and each interval consists of several timeslots. The system operates as follows.

- **Information generation.** A complete piece of information (e.g., a video frame) I_i consists of r_i ($r_i \geq 1$) packets, where r_i is a random variable following some distribution. We use normal distribution, i.e., $r_i = \max\{1, r'_i\}$, $r'_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$, as an example in our performance evaluation, but the algorithms developed in this paper can be applied for any distribution. The BS maintains a separate queue for each source, as shown in Fig. 1. At the beginning of each \mathcal{I} , the *bursty* packets of the information from s_i arrive at the BS with probability p_i . This semi-periodic information generation model is a good approximation of many real-world applications (e.g., surveillance video cameras or edge-aided industrial systems), where traffic may be neither strictly periodic nor purely random (Poisson).
- **Information transmission.** One packet can be successfully transmitted within one timeslot with probability λ .

¹Information size refers to the number of packets that are needed to transmit the information.

To ease explanation, we first assume that $\lambda = 1$ and then relax this constraint for lossy networks in Sec. V. In addition, we assume a shared wireless channel between the BS and the destinations such that only one destination can communicate with the BS in one timeslot.

- We assume an *information-buffer-free* network. “Information-buffer-free” means that we use the buffer only to store the latest information. In other words, if a source generates new information but the BS has not finished transmitting the source’s old information, the BS stops the transmission of the old information and uses the buffer to store packets of the new information. This assumption is needed to make our later analysis valid. Considering information-buffer is more challenging and is left for future research.

We explain the concept of *status update* in the context of multi-packet information. Each information consists of a (random) number of packets, and each packet has a timestamp indicating when it was generated. The BS sends the information to the intended destination; when the destination receives all the packets in the information, it updates the status regarding the corresponding source node.

The freshness of the knowledge that the destination has about the status of the source node is captured by the concept of the AoI. Like most previous work [7], [11], *this paper only focuses on the transmission scheduling between the BS and the destinations (i.e., the last hop of information delivery)*, since AoI of a source is viewed from the point of the corresponding destination. In this context, when people say “source s_i generates information” or “information I_i is generated”, it means that the BS has the information I_i from s_i in the buffer.

Definition 1: (AoI:) The AoI of a source i at time t is defined as:

$$A_i(t) = t - \mu_i \quad (1)$$

where μ_i denotes the timestamp in the first packet of the information for the most recent status update regarding source i . By default, $\mu_i = 0$ if the destination does not have any status update yet. Since the basic scheduling unit is in terms of timeslot, we use the post-action age when calculating AoI value, i.e., A_i is checked only at the end of a timeslot.

For reference, the main notation in the paper is listed in Table 1. Fig. 2 shows an example on how AoI is updated.

Remark 1: For schedulability purpose, we need the following constraint:

$$T > \sum_{i=1}^N p_i E[r_i], \quad (2)$$

where N is the total number of sources. The above constraint implies that the average information generation rate (calculated as average packets/per \mathcal{I}) in the system must be lower than the system throughput. Otherwise the system would not be stable in the sense that the AoI of some source will go to infinity in the long run.

TABLE 1. List of main notation.

Symbol	Meaning
s_i	the index of source nodes, $i \in \{1, \dots, N\}$;
I_i	a piece of information from source s_i ;
r_i	the number of packets included in information I_i ;
\mathcal{I}	information interval;
$A_i(t)$	age of information of s_i at the end of timeslot t ;
k	index of \mathcal{I} , $k \in \{1, \dots, K\}$;
K	the total number of \mathcal{I} 's;
T	the number of slots in each \mathcal{I} ;
$h_{i,k}$	the number of \mathcal{I} 's since last information I_i has been received;
$t_{k,j}$	the j -th timeslot within k -th \mathcal{I} , $j \in \{1, \dots, T\}$;
p_i	the probability that s_i generates information in each \mathcal{I} ;
$\Lambda_i(k)$	indicator variable whether I_i is generated at the beginning of the k -th \mathcal{I} ;
$u_i(t_{k,j})$	indicator variable whether s_i is selected for update at timeslot $t_{k,j}$;

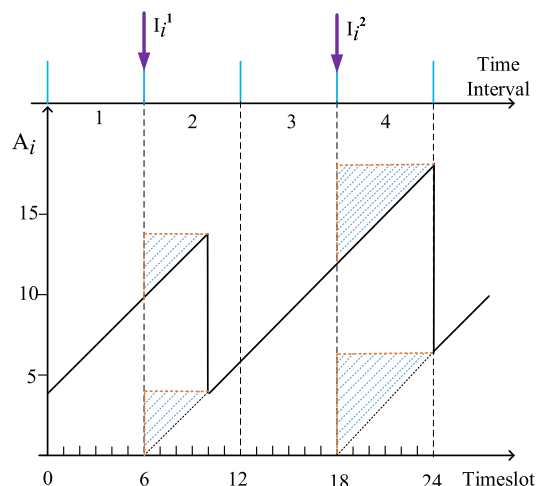


FIGURE 2. An example of AoI update: Assume each time interval contains $T = 6$ timeslots. Information I_i^1 (of length 4 packets) and information I_i^2 (of length 6 packets) arrive in burst at the BS at the beginning of interval 2 and the beginning of interval 4, respectively. The BS finishes the transmission of information I_i^1 at timeslot 10 and hence the destination updates A_i to 4 = (14 – 10); the BS finishes the transmission of information I_i^2 at timeslot 24 and hence the destination updates A_i to 6 = (24 – 18).

B. PROBLEM FORMULATION

A scheduling policy π is defined as a scheduling vector $\mathbf{U} \triangleq \{u_1(t_{k,j}), \dots, u_N(t_{k,j})\}$, where $u_i(t_{k,j}) \in \{0, 1\}$ indicates whether or not the BS transmits s_i 's information at the beginning of timeslot $t_{k,j}$.

As shown in Fig. 3, the area under the AoI line is the *cumulative* AoI calculated within an arbitrary timeslot. As mentioned above, we use the *post-action* age to represent A_i at the end of each timeslot and A_i is calculated in timeslots. Thus we can calculate the long-term *expected average AoI* for source

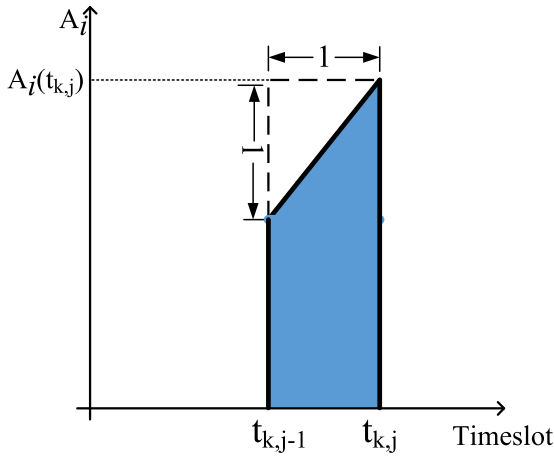


FIGURE 3. The area under A_i for arbitrary timeslot $t_{k,j}$ within interval k .

s_i under scheduling policy π by

$$\begin{aligned} E[A_i^\pi] &= \frac{1}{KT} E \left[\sum_{k=1}^K \sum_{j=1}^T (A_i(t_{k,j}) - \frac{1}{2}) | \bar{A}_i(0) \right] \\ &= \frac{1}{KT} E \left[\sum_{k=1}^K \sum_{j=1}^T A_i(t_{k,j}) | \bar{A}_i(0) \right] - \frac{1}{2} \end{aligned} \quad (3)$$

where the expectation is with respect to the randomness in information generation and the scheduling policy, and $\bar{A}_i(0)$ denotes the initial AoI value of source i . Note that the value of $\frac{1}{2}$ is the size of the triangle shown in Fig. 3, which should be deducted since we use post-action AoI. Because the given initial value will not impact the long-term scheduling decision, we set $\bar{A}_i(0) = 1$ for $\forall i$ and omit $\bar{A}_i(0)$ henceforth. Therefore, for the system we can define the long-term *expected average AoI (EAvgAoI)* as

$$\begin{aligned} E[A^\pi] &= \frac{1}{N} \sum_{i=1}^N E[A_i^\pi] \\ &= \lim_{K \rightarrow \infty} \frac{1}{KNT} E \left[\sum_{k=1}^K \sum_{i=1}^N \sum_{j=1}^T A_i(t_{k,j}) \right] - \frac{1}{2}. \end{aligned} \quad (4)$$

Our goal is to design an optimal scheduling policy π^* that minimizes *EAvgAoI* defined by (4).

III. A NECESSARY CONDITION FOR OPTIMAL SCHEDULING AND AN OPTIMAL SOLUTION

We first consider a simple case where all the information generated at the beginning of an interval is transmitted in the interval.

Lemma 1: Assume that all the information generated at the beginning of an interval is transmitted in the interval. An optimal solution for minimizing the *EAvgAoI* defined in (4) has the following property: once the BS transmits the information of a source, if any, the BS should continuously transmit all the packets in the information without switching to the transmission of another source.

Proof: Take an arbitrary interval and assume that the BS has the information from two sources s_1 and s_2 at the beginning of this interval, with information size of l_1 and l_2 , respectively. Assume that the initial age of s_1 and s_2 at the beginning of the interval is $A_1(0)$ and $A_2(0)$, respectively.² Without loss of generality, assume that the BS finishes s_1 's information before s_2 's information. We can calculate the total AoI of the system in this interval. In particular, the total AoI of s_1 :

$$\begin{aligned} \sum_{j=1}^T A_1(j) &\approx \frac{(A_1(0) + (A_1(0) + l_1 + x))(l_1 + x)}{2} \\ &\quad + \frac{(l_1 + x + T)(T - l_1 - x)}{2} \\ &= A_1(0)(l_1 + x) + \frac{T^2}{2}, \end{aligned} \quad (5)$$

where $x(0 \leq x < l_2)$ denotes the number of packets of source s_2 delivered before finishing s_1 's information. The above equation calculates the size of the shaded area in Fig. 4(a). The total AoI of s_2 , irrespective of the scheduling policy (since its AoI is updated only after both have been delivered), is:

$$\begin{aligned} \sum_{j=1}^T A_2(j) &\approx \frac{(A_2(0) + (A_2(0) + l_1 + l_2))(l_1 + l_2)}{2} \\ &\quad + \frac{(l_1 + l_2 + T)(T - l_1 - l_2)}{2}, \end{aligned} \quad (6)$$

which is the size of the shaded area in Fig. 4(b).

Hence, the total AoI of the system in this interval is: $\mathcal{S}(x) = \sum_{j=1}^T A_1(j) + A_2(j)$. Minimizing $\mathcal{S}(x)$, we get $x = 0$, i.e., continuous transmission of s_1 's information leads to a total AoI no larger than that from non-continuous transmission.

Now assume that the BS has information of $m(m > 2)$ sources at the beginning of the interval. For any two sources among them, s_1 and s_2 , assume that the BS finishes s_1 's information before s_2 's information without loss of generality. If s_1 's and s_2 's information transmission times do not overlap each other, we do not need to do any adjustment. Otherwise, we can adjust their transmission order such that s_1 's information is transmitted continuously before s_2 's information being transmitted. To be more specific, if the k -th packet s_2^k from s_2 is transmitted during the transmission of s_1 's information, the adjustment simply switches the order by moving s_2^k right before s_2^{k+1} , as shown in Fig. 5. Based on the two-source case analysis, the above adjustment leads to a total AoI no larger than that before the adjustment. The lemma thus holds. ■

Remark 2: Lemma 1 does not imply that our problem is equivalent to AoI minimization of single-packets of various lengths with earliest deadline first (EDF) or largest/smallest job first. For instance, consider a simple network with two sources s_1 and s_2 . The information size of source s_1 is 3 and

²Since AoI is viewed at the destinations, the destinations should piggy-back the age information to the BS in acknowledgements.

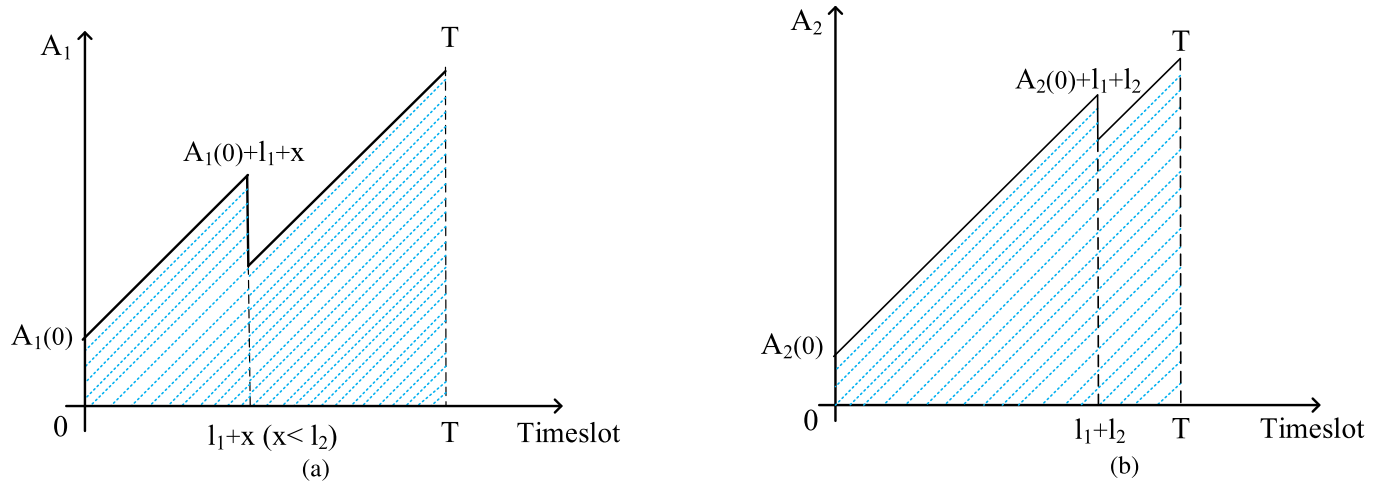


FIGURE 4. Illustration of Aol changes of two sources.

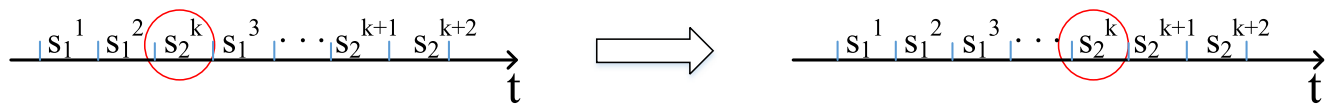


FIGURE 5. Adjustment of transmission schedule (s_1^1, s_1^2, \dots are packets in the information of s_1 , and s_2^1, s_2^2, \dots are packets in the information of s_2).

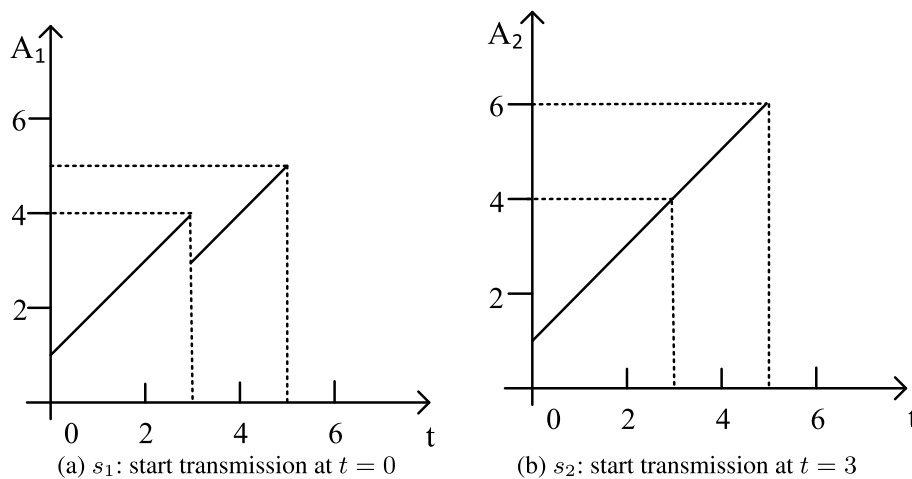


FIGURE 6. Scheduling with earliest deadline first (EDF). Total Aol = 33.

the information will be outdated in 6 timeslots; the information size of s_2 is 2 and the information will be outdated in 10 timeslots. Using the EDF scheduling policy, source s_1 has a higher priority. Fig. 6 and Fig. 7 plot the scheduling results with EDF and with Lemma 1, respectively. It can be seen that EDF leads to a higher total age.

While Lemma 1 is based on the assumption that all the information generated at the beginning of an interval is transmitted in the interval, we can use this lemma to design a scheduling algorithm by relaxing this assumption with information carryover: (a) we find the transmission orders for the current interval by enumerating the results for all possible transmission orders; (b) if some information cannot be transmitted in the current interval with the optimal order, we carry

the remaining packets and the age of their corresponding sources, i.e., the pairs (A_i, l_i) where A_i denotes the age of source s_i and l_i denotes number of packets remaining at the end of the current interval, into the next interval. The carry-over information is outdated and replaced if new information from the same source arrives in the next interval. Repeat steps (a) and (b). The optimal schedule is the one that leads to the smallest $E\text{AvgAoI}$.

We call this algorithm *Opt*. It is optimal because it is essentially a brute-force search for all possible transmission orders. The worst-case complexity of *Opt* is $O((N!)^K)$. Note that the optimal transmission orders in each interval together do not necessarily lead to the global optimal, because an optimal transmission order in the current interval may lead to initial

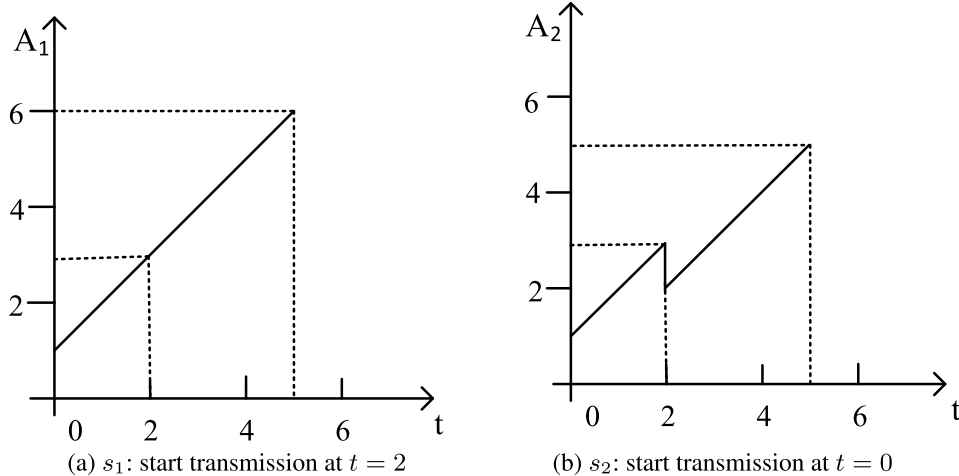


FIGURE 7. Scheduling with Lemma 1 where BS transmits information of s_2 first. Total AoI = 32.

AoI values which are not optimal for the next interval. Due to this reason, brute-force search is necessary for guaranteeing the global optimality.

By reduction from the 3-satisfiability (3-SAT) problem, He *et al.* [14] proved that the minimum age scheduling problem (MASP) is NP-hard in single-packet information systems. While there are subtle differences between the system model in [14] and that in this paper (e.g., He *et al.* assumed a detailed signal-to-interference-and-noise ratio (SINR) transmission model), these differences do not change the core argument in the 3-SAT reduction. Since single-packet information is a special case of multi-packet information, our problem is NP-hard. Due to this reason, Algorithm *Opt* is only used as a benchmark for small-scale problems. When the number of sources becomes higher, we need a fast approximate solution.

IV. INDEX-BASED SCHEDULING POLICY DESIGN AND ANALYSIS

The *Opt* solution has a high worst-case complexity. To tackle the challenge, we develop an index-based scheduling policy based on the framework of *restless bandits* [15]. We propose an *indirect* way to track the AoI, which simplifies the design of index-based policy. We then transform our problem into a *restless multi-armed bandit* problem (RMBP) and develop a scheduling policy with Whittle index, which runs much faster and achieves nearly optimal results in our later evaluation.

A. APPROXIMATING AoI

As shown in Fig. 2, the information I_i (if any) is generated at the beginning of an interval, and may be delivered at any timeslot within that interval. Unlike previous work such as [6], [16], we only consider information update upon the arrival of all related packets, and the transmission time for information from different sources may be different. It is thus challenging to track the exact value of A_i since we need to not only monitor the update, but also track the history of all transmitted packets.

To address this difficulty, we introduce a variable $h_{i,k}$ to represent the number of intervals that have passed (checked at the end of k -th interval) since the last information update for s_i . In particular,

$$h_{i,k+1} = \begin{cases} 1, & \text{if } \sum_{j=1}^T u_i(t_{k,j}) > 0 \\ h_{i,k} + 1, & \text{otherwise} \end{cases} \quad (7)$$

Note that $\sum_{j=1}^T u_i(t_{k,j}) > 0$ means that in the k -th interval, the BS transmits s_i 's information and thus it must have information to deliver at the beginning of this interval. Due to the schedulability assumption (Remark 1), we assume³ s_i 's information can be updated in the k -th interval and thus $h_{i,k+1}$ is reset to 1. Otherwise (i.e., $\sum_{j=1}^T u_i(t_{k,j}) = 0$), the BS has no information of s_i at the beginning of the k -th interval and thus $h_{i,k+1} = h_{i,k} + 1$.

We then use $h_{i,k}$ to estimate the value of A_i . Note that in an index-based solution, the main goal of such approximation is to keep *the relative order of different age values rather than their exact values*. For this reason, we thus approximate

$$A_i(t_{k,j}) \approx h_{i,k} \cdot T, \quad \forall j \in \{1, \dots, T\} \quad (8)$$

Since the value of $h_{i,k}$ is updated only at the end of intervals, the above approximation allows an easy calculation of A_i and tractable performance analysis in the following section.

Remark 3: The approximation of A_i with $h_{i,k}$ is to ease analysis, based on which we can design an effective index-based policy to solve the scheduling problem raised in Sec. II. In the actual performance evaluation in Sec. VI, however, we calculate the exact A_i value whenever all packets in the information have been received.

³This assumption is to simplify analysis and allows us to develop an effective index-based policy. It is reasonable since $1 > \sum_{i=1}^N p_i \frac{E[r_i]}{T}$ due to Remark 1. Our later simulation-based evaluation, however, does not depend on this assumption.

In the following we develop a low-complexity scheduling algorithm whose $E\text{AvgAoI}$ is close to the minimum by leveraging the Whittle's methodology.

B. SCHEDULING WITH WHITTLE INDEX

We first introduce the *restless multi-armed bandit* (RMAB) framework and show how our problem is mapped to RMAB. We then propose a scheduling policy with the Whittle index [15].

1) RMAB

RMAB is a generalization of the classical *multi-armed bandit problem* (MAB) [17]. In MAB, a player, with full knowledge of the current state of each arm, chooses one out of N arms to activate at each time and receives a reward determined by the state of the activated arm. Only the activated arm may change state and the states of passive arms remain unchanged. Whittle generalized MAB to RMAB by allowing $M(1 \leq M \leq N)$ arms to be played simultaneously and allowing passive arms to change states even if they are not played. RMAB has been shown to be PSPACE-hard by Papadimitriou and Tsitsiklis in [18]. Hence, Whittle proposed an optimal index policy for the RMAB problem under a relaxed constraint: the number of activated arms can vary over time but its average over the infinite horizon equals M . With this relaxation, Whittle applied the Lagrangian approach to decouple the RMAB problem into multiple sub-problems, namely the decoupled model.

It is easy to see that our problem belongs to the relaxed RMAB, since (1) we can regard each source as an arm and all arms are restless, and (2) an action for each arm is made in *each interval*, and the value of $\sum_{i=1}^N p_i$ means the average number of active arms in the long term. Refer to the next subsection for the bandit model and the actions.

Hence N sub-problems can be decoupled by applying Whittle's approach. Each sub-problem corresponds to information transmission for a single source s_i . To obtain the optimal packet transmission policy, our goal becomes to find a way of calculating the Whittle index that matches our context. To be more specific, the resulting Whittle index should help the BS determine whose information should be transmitted at the beginning of each timeslot.

2) DECOUPLED MODEL AND THE PROPERTIES OF OPTIMAL POLICY

Using the AoI approximation in Sec. IV-A, we can analyze the dynamics of variable $h_{i,k}$ to develop an index policy. Since each sub-problem corresponds to only one source, in the following analysis we *omit the source index* i for simplicity. We first transform the sub-problem into a Markov Decision process (MDP) [19], whose basic components (states, actions, transitions, and objective) are defined as follows:

- **States:** To characterize the state $s(k)$ of the MDP, we define $s(k) = (\Lambda(k), h_k)$, where $\Lambda(k) = 1$ indicates

that information is generated at the beginning of the k -th interval and $\Lambda(k) = 0$ if not.

- **Actions:** We use $a(k) \in \{0, 1\}$ to denote the action taken in the k -th interval. To be specific, $a(k) = 1$ if the source's status is updated and $a(k) = 0$ if not.
- **Transition:** Given the action $a(k) = a$, the state transition probability from current state $s = (\Lambda, h)$ to next state s' is as follows
 - 1) if action $a = 1$:

$$P[s' = (1, h + 1) | s = (0, h)] = p;$$

$$P[s' = (0, h + 1) | s = (0, h)] = 1 - p;$$

$$P[s' = (1, 1) | s = (1, h)] = p;$$

$$P[s' = (0, 1) | s = (1, h)] = 1 - p;$$
 - 2) if action $a = 0$:

$$P[s' = (1, h + 1) | s = (\Lambda, h)] = p;$$

$$P[s' = (0, h + 1) | s = (\Lambda, h)] = 1 - p;$$
- **Objective:** There are two actions that the BS can execute, active ($a = 1$) or idle ($a = 0$). Action $a = 1$ incurs a cost of transmitting the information; and action $a = 0$ incurs a cost of information aging for not transmitting the information. Combining these two types of cost, we define the total cost of executing action a given state $s = (\Lambda, h)$ as follows:

$$\Delta(s, a) \triangleq ((1 - \Lambda \cdot a) \cdot h + 1) \cdot \omega + C \cdot r \cdot a \cdot (1 - \omega) \quad (9)$$

where $(1 - \Lambda \cdot a) \cdot h + 1$ represents the age evolution of h if the action $a(k) = a$ is taken, C represents the cost for transmitting one packet, and $\omega \in (0, 1)$ is a weight parameter.

Now we can define the objective function under policy π as:

$$\Upsilon^\pi = \lim_{K \rightarrow \infty} \frac{1}{K} E \left[\sum_{t=1}^K \Delta(s(t), a(t)) \right] \quad (10)$$

A policy is called cost-optimal if it minimizes the average cost defined by (10). Proved in [7], a cost-optimal policy is stationary and deterministic. In particular, a policy is stationary if for $\forall k_1, k_2 \in \{1, \dots, K\}$, we have $a(k_1) = a(k_2)$ when $s(k_1) = s(k_2)$. Recall that we always update the value of h at the end of intervals, hence during an arbitrary interval, the optimal policy either idles in every slot, or transmits until the information is delivered. Hence, the policy is stationary.

For analyzing the system in steady-state, we extend K to an infinite horizon. Then, the Bellman equation can be formulated as

$$\theta(s, a) + \beta = \min_{a \in \{0, 1\}} \{U_a(s)\} \quad (11)$$

where $\theta(s, a)$ is the cost-to-go function and β is the optimal average cost. More specially

$$U_0(s) = p \cdot \theta((1, h + 1), a) + (1 - p) \cdot \theta((0, h + 1), a) + h + 1;$$

$$U_1(s) = p \cdot [\theta((1, h + 1), a) + \theta((1, 1), a)] + (1 - p) \cdot [\theta((0, h + 1), a) + \theta((0, 1), a)] + r \cdot C + 1 \quad (12)$$

Further, we show that the deterministic stationary scheduling policy is threshold-type. Obviously, for state $s(0, h)$, the optimal action is to be idle (since $C \geq 0$). For state $s(1, h)$, if we assume the optimal action is to update, i.e.,

$$U_1(s(1, h), 1) - U_0(s(1, h), 0) \leq 0 \quad (13)$$

Then, for state $s(1, h + 1)$, we have

$$\begin{aligned} & U_1(s(1, h + 1), 1) - U_0(s(1, h + 1), 0) \\ &= (rC + 1 + E[J(s')]) - (h + 2 + E[J(s')]) \\ &\leq (rC + 1 + E[J(s')]) - (h + 1 + E[J(s')]) \\ &= U_1(s(1, h), 1) - U_0(s(1, h), 0) \leq 0 \end{aligned} \quad (14)$$

where $E[J(s')]$ is the expectation of cost taken over all next state s' that are reachable from state s , and the inequality in the second step is due to the non-decreasing property of $E[J(s')]$ [7]. Hence, *the cost-optimal policy must be threshold-type*.

By applying the threshold-type scheduling policy, the BS is idle when $1 \leq h < \bar{H}$ and transmits if $h \geq \bar{H}$, where $\bar{H} \in \{1, 2, \dots\}$ is the threshold. Next, we explicitly investigate the relation between the average cost and the threshold, based on which we derive the Whittle index.

Remark 4: The Markov Decision Process introduced above does not directly indicate transmission scheduling. Instead, it is to help design an index policy (Sec. IV-C), based on which the transmission schedule is made for each timeslot (Sec. IV-D).

C. DERIVATION OF WHITTLE INDEX

Prior to deriving the Whittle index, we first form a discrete-time Markov Chain (DTMC) that depicts the state transition of the variable h . Within each interval, different actions, if taken, incur different costs. For example, the DTMC incurs the cost $rC + 1$ when h is reduced to 1 at the end of the interval, which means a transmission opportunity is allocated and hence the AoI is updated. Otherwise, the cost is not considered, and h is increased by 1. Hence, the state space of the DTMC is $\{1, 2, \dots, \bar{H}, \dots\}$.

Lemma 2: Denote the steady-state distribution of the above DTMC as $Q = \{q_1, q_2, \dots\}$. We have

$$q_i = \begin{cases} \varphi(\bar{H}), & \text{if } i \leq \bar{H} \\ \varphi(\bar{H})(1-p)^{i-\bar{H}} & \text{otherwise} \end{cases} \quad (15)$$

where $\varphi(x) = \frac{p}{1+p(x-1)}$ and \bar{H} is the threshold.

Proof: In accordance with the *threshold-type* policy, we know $q_1 = q_2 = \dots = q_i$ when $i \leq \bar{H}$. Denote $q = q_1 = \dots = q_i$ for $\forall i \leq \bar{H}$. Once h is larger than \bar{H} , it either increases (with probability $1 - p$) or reduces to 1 (with probability p), as analyzed in Sec. IV-B2. Hence, $q_i = q \times (1 - p)^{i-\bar{H}}$ when $i > \bar{H}$. Since the sum of the corresponding probability in Q must equal 1, we have

$$\lim_{n \rightarrow \infty} \bar{H} \times q + q \times (1 - p) + \dots + q \times (1 - p)^{n-\bar{H}} = 1 \quad (16)$$

Solving (16), we obtain that $q = \frac{p}{1+p(\bar{H}-1)}$. The lemma thus holds. ■

Therefore, the average cost can be calculated as the expectation over all states

$$\begin{aligned} \Phi(\bar{H}) &= \sum_{i=2}^{\bar{H}} i \cdot \varphi(\bar{H}) \cdot \omega + (1 \cdot \omega + C \cdot r \cdot (1 - \omega)) \cdot \varphi(\bar{H}) \\ &\quad + \sum_{i=\bar{H}+1}^{\infty} i \cdot \varphi(\bar{H}) \cdot (1 - p)^{i-\bar{H}} \cdot \omega \\ &= \frac{(\frac{\bar{H}^2}{2} + (\frac{1}{p} - \frac{1}{2})\bar{H} + \frac{1}{p^2} - \frac{1}{p})\omega + rC(1 - \omega)}{\bar{H} + \frac{1-p}{p}} \end{aligned} \quad (17)$$

We can regard the threshold as a variable of h . Note that $\Phi(h)$ is strictly convex in the domain. Let h^* be the value at the minimization point of $\Phi(h)$. Then, the value of an optimal threshold for minimizing the average cost $\Phi(\bar{H})$ is either $\lfloor h^* \rfloor$ or $\lceil h^* \rceil$, i.e.,

$$\bar{H}^* = \begin{cases} \lfloor h^* \rfloor, & \text{if } \Phi(\lfloor h^* \rfloor) \leq \Phi(\lceil h^* \rceil) \\ \lceil h^* \rceil & \text{if } \Phi(\lceil h^* \rceil) \leq \Phi(\lfloor h^* \rfloor) \end{cases} \quad (18)$$

Thus, both actions for state $s = (1, h)$ are equivalent if $\Phi(h) = \Phi(h + 1)$. By solving this equation, we can explicitly derive the Whittle index, denoted by $I(s = (\Lambda, h))$ as

$$I(\Lambda, h) = \left(\frac{h^2}{2r} - \frac{h}{2r} + \frac{h}{pr} \right) \cdot \frac{\omega}{1 - \omega} \quad (19)$$

Next, we show that the original AoI minimization problem is indexable. Let $\mathcal{P}(C)$ be the set of system states where the optimal action is to be idle when the transmission cost is C . We give the definition of indexability.

Definition 2: (Indexability): If $\mathcal{P}(C)$ increases monotonically from \emptyset to the entire state space, as C increases from 0 to $+\infty$, we say that the decoupled problem corresponding to information of source s_i is indexable. The age of information minimization problem is called indexable if the decoupled problem is indexable for every source.

From (17) and (19), we know that the cost C is a function of the threshold of h . In addition, by calculating the derivative of (19), we have

$$\frac{\omega}{(1 - \omega)r} \cdot \left(h - \frac{1}{2} + \frac{1}{p} \right) > 0, \quad (20)$$

since h is non-negative integer, $\omega \in (0, 1)$ and $p \in (0, 1]$. Hence, the index function monotonically increases. Moreover, substituting $h = 0$ yields $C = 0$, implying that $\mathcal{P}(C) = \emptyset$, and $h \rightarrow +\infty$ results in $C \rightarrow +\infty$, consequently, $\mathcal{P}(C) = \mathbb{N}$. Thus, the decoupled problem corresponding to source s_i is indexable. Because source s_i is arbitrary, the original AoI minimization problem is hence indexable.

D. SCHEDULING ALGORITHM WITH THE WHITTLE INDEX

We next design the scheduling policy based on the Whittle index. Taking $I(\Lambda, h_k)$ as the Whittle index for each source s_i , the index policy works as follows: at the beginning of

each timeslot, we select the source with the maximum Whittle index for transmission (randomly select one if there is a tie), and if the source with the maximum Whittle index has finished transmission of all packets for the current information, select the source with the next largest Whittle index to transmit. Repeat the above process until the BS has no information to send.

We call the above scheduling policy as Multi-packet Whittle Index Policy (MWIP). An important advantage of MWIP is the low complexity in scheduling packet transmissions, especially when the information is composed of multiple packets.

E. WHY DOES OUR SOLUTION WORK WELL

Since we do not update the Whittle index until the end of the interval, within each interval, once the BS decides to transmit information for a source, it continuously transmits all the packets in the information. This matches the necessary condition of an optimal solution stated in Lemma 1. In contrast, other existing baseline methods, such as the Greedy Policy (GP) [5] and the Naïve Whittle Index Policy (NWIP) as shown in the later evaluation (Sec. VI), do not meet the necessary condition. This partially explains why our method outperforms other existing methods.

It is worth noting that in all index-based solutions, indexes play a critical role in determining the transmission order. Other research such as [20], [21] may help design a better index-based solution for multi-packet information systems in the future. Nevertheless, no matter what the new solution is, the indexes must not break the necessary condition in Lemma 1. This insight is significant for any future effort in designing better algorithms for AoI minimization in multi-packet information systems.

V. ERASURE CODE-AIDED AoI MINIMIZATION IN LOSSY NETWORKS

In this section, we relax the assumption of reliable packet transmission in each timeslot. Instead, we assume a lossy network where packets are not guaranteed to deliver to the intended recipient. Using *erasure code*, we improve the performance of Whittle index-based scheduling policy in lossy network.

Let q ($0 < q < 1$) be the *bit error rate* (BER) associated with the wireless channel. Assume that information needs to be transmitted to the destination and the total amount of bits of the information is m . Then the probability that the information is successfully delivered to the destination without error is

$$\lambda = (1 - q)^m \quad (21)$$

Note that, $q = 0$ means the channel is noiseless and the information is delivered with no errors, which is the case studied in Sec. IV.

A. REGULAR TRANSMISSION

Recall that in our system model, each piece of information consists of r packets. The recipient cannot update the

AoI until all packets are received. In lossy networks, packet transmission may fail and the sender has to re-transmit it. As a result, the transmission of next packet will not start until the current packet has been successfully received by the destination, which needs to send back an *acknowledgement* to the sender. We name this transmission scheme as *Regular Transmission* (Reg Trans).

Let X and Y be the number of bits contained in the packet and the *acknowledgement*, respectively. According to (21), the *packet delivery probability* (PDP) (i.e., the probability that a packet is confirmed to be successfully received by the recipient) of *Reg Trans* scheme is

$$\lambda_1 = (1 - q)^{X+Y}. \quad (22)$$

B. ERASURE CODE-BASED TRANSMISSION

Erasure code (EC) is a technology of data protection and recovery. It has been widely used in fields such as data storage systems [22], [23]. The key idea behind erasure code is that k blocks of data are expanded into n ($n > k$) blocks of encoded data, such that *any* subset of k (out of n) encoded blocks suffices to reconstruct the original data. Such a code is named as (n, k) code and allows the recipient to recover from up to $n - k$ losses in a group of n encoded blocks of data. One such code is Reed-Solomon code, whose details can be found in [24]. We omit the encoding/decoding details since they are out of the focus of the paper.

Letting $k = r$, we can regard each packet as a block of data. After encoding, the r original packets are expanded and encoded to n ($n > r$) new packets. To deliver the information, we transmit the n encoded packets without retransmissions. As such, the source node does not require the destination node to send back *acknowledgement* to confirm the correct reception of every packet. Instead, as soon as any r (out of the n) packets are successfully received by the destination, the destination sends an *acknowledgement* to terminate the transmission session and then reconstructs the source information for AoI update. As a result, at most n timeslots are needed to complete the transmission process. We name this transmission scheme as *Erasure Code Transmission* (EC Trans). Since packets (except the last one in an information piece) are transmitted without the of *acknowledgement*, the PDP of *EC Trans* scheme is

$$\lambda_2 = (1 - q)^X. \quad (23)$$

Since for each information (instead of each packet) the receiver needs to send back an *acknowledgement* to the source node, we denote

$$\lambda_3 = (1 - q)^Y. \quad (24)$$

We next calculate the expected time for successfully delivering information, namely *Expected Transmission Time* (ETT). To avoid unfairly penalizing *Reg Trans*, we assume that the data/acknowledgement two-way handshake can be finished within one timeslot.

- For *Reg Trans*, the expectation of successful single packet transmission time is $1/\lambda_1$. Since the transmission of each packet is independent, the total *expected transmission time* of a complete piece of information is

$$E^{Reg}[T] = \frac{r}{\lambda_1} \quad (25)$$

- For *EC Trans*, as long as any k of the n encoded packets are successfully received, the transmission is terminated with an *acknowledgement* message. Hence, the expected transmission time can be calculated as follows

$$\begin{aligned} E^{EC}[T] &= \lambda_3 \{ k \lambda_2^k \\ &+ (k+1) \binom{k}{k-1} \lambda_2^{k-1} (1-\lambda_2) \lambda_2 \\ &+ (k+2) \binom{k+1}{k-1} \lambda_2^{k-1} (1-\lambda_2)^2 \lambda_2 + \dots \\ &+ n \binom{n-1}{k-1} \lambda_2^{k-1} (1-\lambda_2)^{(n-1)-(k-1)} \lambda_2 \} \\ &+ (1-\lambda_3)n + \Gamma n \\ &= \lambda_3 \sum_{i=0}^{n-k} (k+i) \binom{k+i-1}{k-1} \lambda_2^k (1-\lambda_2)^i \\ &+ (1-\lambda_3 + \Gamma)n, \end{aligned} \quad (26)$$

where i represents the number of failed packets during the transmission process, Γ is the probability that no k (out of n) packets have been correctly received, i.e.,

$$\Gamma = 1 - \sum_{i=0}^{n-k} \binom{k+i-1}{k-1} \lambda_2^k (1-\lambda_2)^i, \quad (27)$$

and λ_3 represents the probability that the BS correctly receives the *acknowledgement*. Note that the BS might continue sending packets (up to n) if the *acknowledgement* is lost.

Remark 5: There is a subtle difference between the two schemes in handling transmission failures. Reg Trans keeps trying until all packets are successfully transmitted, while EC Trans stops trying after transmitting n packets. In this sense, the above comparison can only give us a rough idea on the transmission time of Reg Trans and EC Trans. It may be unfair to compare the expected transmission time, since in theory Reg Trans may have infinitely long transmission time. Nevertheless, if information is stored in the BS for too long, it will be replaced by new information, meaning that Reg Trans would not result in infinite re-transmission in practice.

As an example, we illustrate the ETT values obtained when $X = 4000$ bits, $Y = 400$ bits and BER varies from 10^{-6} to 10^{-4} in Fig. 8. Obviously, *EC Trans* outperforms *Reg Trans* when the BER is high (e.g., higher than 10^{-5}). This makes sense since in this case multiple re-transmissions may be needed to successfully transmit a packet, resulting in a high ETT for *Reg Trans*.

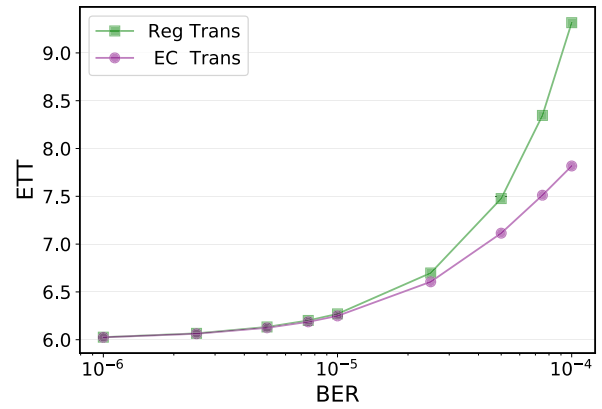


FIGURE 8. Illustration of effect of BER on ETT, (6, 8) erasure code is used for *EC Trans*.

VI. PERFORMANCE EVALUATION

In this section, we evaluate the performance of MWIP with simulation. We compare MWIP with the following two baseline scheduling policies:

- Greedy Policy (GP) [5]: A scheduling policy that always selects the source with the maximum AoI for transmission.
- Naïve Whittle Index Policy (NWIP): In [7], the Whittle index has been used for AoI minimization when each packet includes complete information and thus causes a status update. This Whittle index policy cannot be applied directly to our case, and thus it is impossible to fairly compare the method in [7] with ours. Nevertheless, we can modify the method in [7] to be applicable to our case by disabling the update of status and the Whittle index if the received packet is not the last packet of the information. We call this modified Whittle index policy as naïve Whittle index policy (NWIP).

We are interested in the expected average AoI, which is defined by (4). We evaluate the performance of different network scenarios with different number of sources and different information generation rates. In addition, we evaluate the benefit of coding in the presence of packet losses.

A. COMPARISON WITH THE OPTIMAL SOLUTION

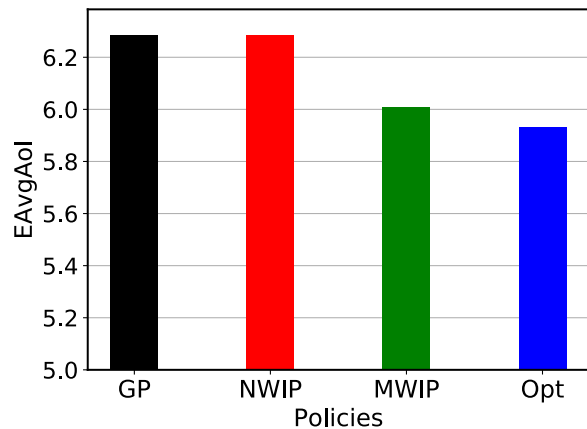
First, we implement the optimal solution *Opt* and use it as the benchmark for the above three methods. Nevertheless, due to the high complexity of *Opt*, such comparison is only possible for a small network over a small time period. Hence, we simulate the following simple networking scenario

- There are 2 sources and 2 corresponding destinations in the network.
- The information generation rate is 0.9 for each source.
- For an arbitrary source, $r_i \in \{2, 3\}$. To diversify the traffic, we set $r_i \neq r_j$ if $i \neq j$.
- The total time length is 20 intervals, i.e., $K = 20$. Moreover, we set the number of timeslots of each interval as $T = 5$, in accordance with (2).

The simulation result is plotted in Fig. 9. This shows that our scheduling policy MWIP achieves a smaller $EAvG\text{AoI}$,

TABLE 2. Impact of number of sources on EAvGAoI.

Policy	Number of Sources									
	5	10	15	20	25	30	35	40	45	50
GP	27.28878	39.71936	67.68997	90.82541	111.09172	133.69518	163.09424	190.16638	203.67403	230.79568
NWIP	27.28878	39.79056	67.68603	90.66856	110.97294	133.62292	163.07634	190.00626	203.79690	230.87717
MWIP	27.28878	38.45890	60.89890	81.46848	101.56829	121.01118	147.64967	172.59384	187.68068	211.94173

FIGURE 9. Comparison with the *Opt* algorithm.

compared with that of GP and NWIP and it is very close to the *EAvGAoI* achieved by *Opt*.

In addition, the *EAvGAoI* of GP and NWIP is nearly identical. Recall that when the information generation rate is the same for all sources, the Whittle index in [7] is only affected by AoI. What's more, in [7] the index calculation formula is monotonic, which means a higher AoI will result in a bigger Whittle index. In other words, the source with a higher AoI will be scheduled for transmission first if NWIP is applied. This matches the key idea of GP. Hence, the GP and NWIP policies make consistent decisions with a high chance.

Regarding the running speed, we run the simulation on a commodity Linux machine with Intel Core i7-4790 CPU @ 3.60 GHz and 8 GB memory. NWIP returns the result in 0.3 ms; GP returns the result in 0.25 ms; MWIP returns the result in 0.3 ms. *Opt*, however, has a much higher running time, which is 314572.8 ms. This is because *in the worst case* the number of scheduling order combinations that meet the necessary condition (Sec. III) is up to $(2!)^{20}$, and *opt* searches for the optimal value in the entire space.

B. IMPACT OF NUMBER OF SOURCES

Based on the system model described in Sec. II, we simulate a large network with various number of sources. We first test a lossless network. We set the parameters $K = 1000$ and $\omega = 0.5$.

For each source i , we assume that each information includes a random number of packets r_i , which is randomly selected from $\{1, 2, 3, 4, 5, 6\}$ (with replacement). The information generation rate of each source is set to 0.8. We use the *minimum T* that satisfies (2) in our simulation, that is

$T = \left\lceil 0.8 \times \sum_{i=1}^N r_i \right\rceil$. Note that this value of T changes with the number of sources. To obtain the average value of *EAvGAoI*, we run the simulation 100 times for all policies.

The average simulation results of policies GP, NWIP, and MWIP are listed in Table 2. We can see that MWIP outperforms the other two when the number of sources is big. NWIP and GP achieve very similar *EAvGAoI*, as explained in Sec. VI-A. By analyzing the result obtained under the same policy, we observe that a larger number of sources causes a higher *EAvGAoI*. This is because when the information generation rate of each source is fixed, a larger number of sources means a higher system workload, resulting in higher AoI.

We store all the numerical data obtained in simulation in a dataset and draw its Box-and-Whisker plot in Fig. 10 to show the data distribution through their quartiles. Obviously, compared across the policies, MWIP has an advantage regarding the minimum, lower quartile, median, upper quartile and the maximum of the dataset. The maximum value of MWIP is even lower than the minimum values of GP and NWIP. This also explains why we get much smaller *EAvGAoI* in Table 2 for MWIP.

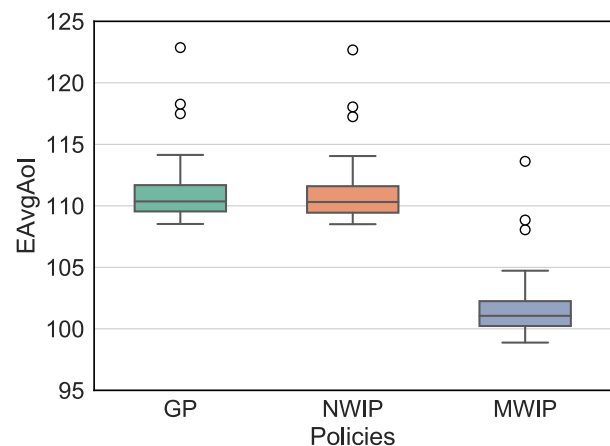


FIGURE 10. Box-and-Whisker plot depicting the distribution of numerical results for simulation with 25 sources. Each box contains five values of the dataset, the minimum, lower quartile, median (second quartile), upper quartile and maximum of *EAvGAoI*, from bottom to top. The white points above the maximum line are outliers.

C. BENEFIT OF CODING IN LOSSY NETWORKS

In this section, we evaluate the benefits of *EC Trans* strategy in lossy networks. We simulate a network with 10 sources.

By setting the bit error rate $BER = 4 \times 10^{-4}$ and $X = 4000$, $Y = 400$, it is easy to calculate that the *packet delivery probability* for the two transmission strategies, *Reg Trans* and *EC Trans*, are $\lambda_1 = 0.64$ and $\lambda_2 = 0.67$, respectively.

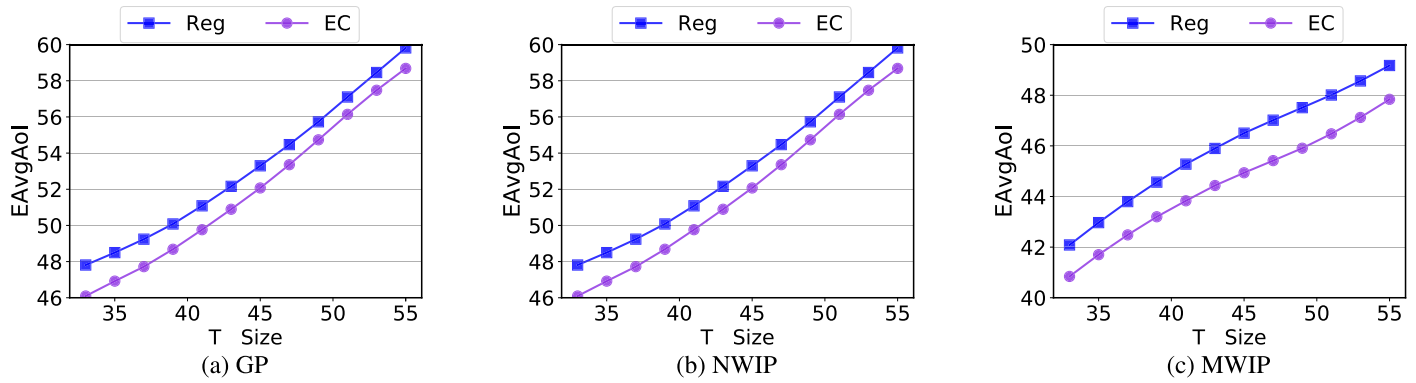


FIGURE 11. Analysis of T size: $K = 3000$, $p_i = 1.0$.

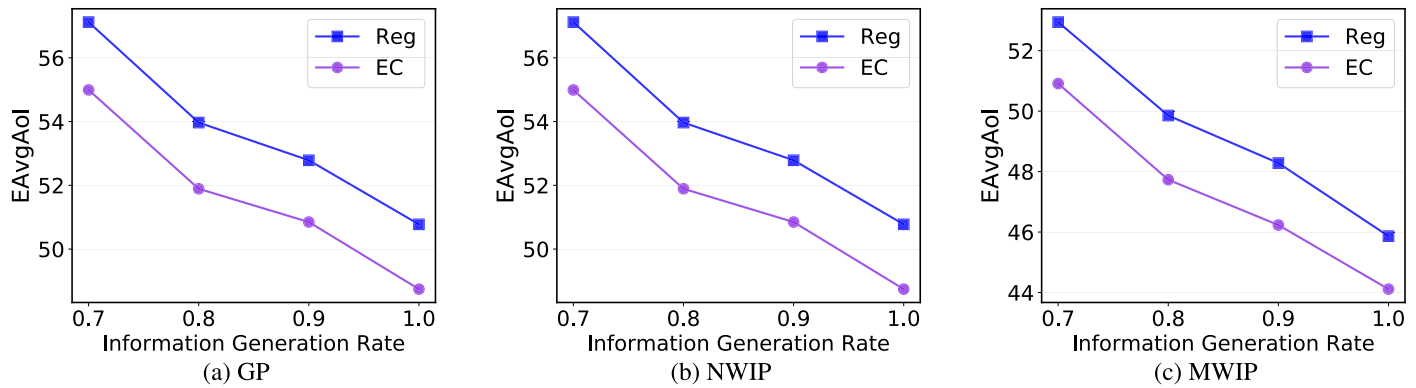


FIGURE 12. Analysis of information generation rate: $K = 3000$, $p_i \in \{0.7, 0.8, 0.9, 1.0\}$.

Considering the uncertainty of channel status, we use (r_i, T) code for *EC Trans* transmission strategy, where T is the interval size obtained according to (2). We first investigate the effect of T on the $E\text{AvgAoI}$, by increasing T from the *minimum T* to double this value. Note that the *minimum T* is obtained using the approach in Sec. VI-B. We also run the simulation for 100 times and calculate the average.

Fig. 11 presents the simulation results with different T values. The *EC Trans* strategy achieves lower $E\text{AvgAoI}$ for all policies compared with that of the *Reg Trans* strategy.

We also investigated the impact of information generation rate by varying it from 0.7 to 1.0. Following the change of information generation rate, the *minimum T* also changes. By using the *minimum T* for each rate, we plot the average result in Fig. 12. We observe that a high information generation rate leads to a low $E\text{AvgAoI}$ for all scheduling policies. This is because with a high information generation rate, information is transmitted more frequently to update the knowledge at the destinations. Similarly, the *EC Trans* strategy achieves a lower $E\text{AvgAoI}$.

Overall, compared to GP and NWIP, MWIP achieves a lower $E\text{AvgAoI}$ in all the tested scenarios. In addition, the *EC Trans* strategy can reduce the $E\text{AvgAoI}$ in lossy networks.

VII. RELATED WORK

The importance of providing timely information has been recognized in different application domains, including for example environment protection, health monitoring and intelligent traffic. The concept of AoI was firstly formalized as a network performance metric of interest in [3]. Aimed at addressing the problem of congestion and packet collision in vehicular networks with a large number of nodes, the authors proposed a rate control algorithm to minimize the system age.

In [4] and [8]–[10], the authors studied AoI minimization problem using queuing theory. The key idea is to understand the fundamental characteristics of the process describing the AoI changes. In [4], the author investigated the metric *average age* and considered Poisson arrivals. The authors modeled the system as an M/M/1 queueing system by assuming that the time for transmitting a packet is exponentially distributed. Similarly, the *average age* for M/D/1 and D/M/1 models was also analyzed, corresponding to the cases with deterministic service time and periodic sampling, respectively. In addition, the authors in [9] proposed a new metric called *peak age*, which denotes the maximum value of age immediately before performing AoI update. They showed that *peak age* is a suitable metric to characterize the AoI in applications imposing an upper threshold on the value of AoI.

The work in [5]–[7] and [25] is more close to ours. In [5], Kadota *et al.* studied the age under unreliable wireless transmission channel. The impact of transmission errors on scheduling decision was analyzed and two scheduling algorithms were evaluated with deterministic packet generation rate (every T timeslots). The authors extended their work in [6] under the constraint of the minimum throughput. To minimize AoI, they developed three low-complexity scheduling policies subject to the minimum throughput requirement and evaluated their performance against the optimal policy. Similarly, the work in [7] also focused on wireless networks, while the communication channel is supposed to be noiseless and the arrivals of packets containing time-sensitive status update information are stochastic. All the above work assumed that each single packet contains a complete information update.

In [26], Yates investigated the problem of status updates transmitted through a server powered by an energy harvesting system. He proved that an optimal policy to minimize the AoI contains server idle moments, even though the remaining energy of the server is enough to transmit an update. Intuitively, the system transmits an update only if it can result in significant reduction of the AoI at the recipient.

VIII. CONCLUSION AND FUTURE WORK

Age of information (AoI), a new metric to capture the freshness of information, has attracted great attention in big data analytics of real-time IoT systems. In many real-world systems such as autonomous driving, timely information update is critical since outdated information not only wastes network bandwidth but also may lead to disastrous consequences. More often than not, a piece of information that is meaningful to the applications may be embedded in multiple packets, and as such we need to extend existing models from the single-packet information case to a more general multi-packet information scenario. This paper addressed this challenge. We first analyzed the necessary condition for optimal AoI scheduling, based on which we proposed an optimal solution. To address the high complexity of the optimal solution, we proposed a new way to approximate the age of information. The approximation greatly facilitates the design of a Whittle index-based solution, called Multi-packet Whittle Index Policy (MWIP), that runs much faster and achieves close to optimal results. For networks with unreliable transmission, we proposed a transmission strategy using erasure codes. With extensive simulation, we evaluated the performance of MWIP and compared it with two baseline methods, Greedy Policy (GP) and Naïve Whittle Index Policy (NWIP). In conclusion, MWIP reduces the average age in the multi-packet information model in both reliable and lossy networks.

It may still have room to design better solutions for the AoI optimization problem in multi-packet information systems. This paper imparts some important insights into the future endeavor in this direction and opens other research opportunities. First, a good solution, either index-based or any other kind, should continuously transmit all the packets in

the information to achieve the global minimum. Second, in multi-packet information systems, the goal of minimizing AoI may conflict with fairness among sources. It is interesting future research to consider the AoI optimization within some fairness constraints. Third, we decoupled the Whittle index and channel conditions in this paper. It would be interesting research to investigate whether considering channel conditions in the Whittle index leads to a better solution. Finally, even partial information may have a value in practice. Considering partial information and its associated partial value is another research for future work.

REFERENCES

- [1] S. Taghvaeeyan and R. Rajamani, "Portable roadside sensors for vehicle counting, classification, and speed measurement," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 1, pp. 73–83, Feb. 2014.
- [2] N. Gageik, P. Benz, and S. Montenegro, "Obstacle detection and collision avoidance for a UAV with complementary low-cost sensors," *IEEE Access*, vol. 3, pp. 599–609, 2015.
- [3] S. Kaul, M. Gruteser, V. Rai, and J. Kenney, "Minimizing age of information in vehicular networks," in *Proc. 8th Annu. IEEE Commun. Soc. Conf. Sensor, Mesh Ad Hoc Commun. Netw.*, Jun. 2011, pp. 350–358.
- [4] S. Kaul, R. Yates, and M. Gruteser, "Real-time status: How often should one update?" in *Proc. IEEE INFOCOM*, Mar. 2012, pp. 2731–2735.
- [5] I. Kadota, E. Uysal-Biyikoglu, R. Singh, and E. Modiano, "Minimizing the age of information in broadcast wireless networks," in *Proc. 54th Annu. Allerton Conf. Commun., Control, Comput. (Allerton)*, Sep. 2016, pp. 844–851.
- [6] I. Kadota, A. Sinha, and E. Modiano, "Optimizing age of information in wireless networks with throughput constraints," in *Proc. IEEE INFOCOM-IEEE Conf. Comput. Commun.*, Apr. 2018, pp. 1844–1852.
- [7] Y.-P. Hsu, "Age of information: Whittle index for scheduling stochastic arrivals," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2018, pp. 2634–2638.
- [8] M. Costa, M. Codreanu, and A. Ephremides, "Age of information with packet management," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2014, pp. 1583–1587.
- [9] M. Costa, M. Codreanu, and A. Ephremides, "On the age of information in status update systems with packet management," *IEEE Trans. Inf. Theory*, vol. 62, no. 4, pp. 1897–1910, Apr. 2016.
- [10] L. Huang and E. Modiano, "Optimizing age-of-information in a multi-class queueing system," 2015, *arXiv:1504.05103*. [Online]. Available: <http://arxiv.org/abs/1504.05103>
- [11] A. Kosta, N. Pappas, and V. Angelakis, "Age of information: A new concept, metric, and tool," *Found. Trends Netw.*, vol. 12, no. 3, pp. 162–259, 2017.
- [12] N. H. Amer, H. Zamzuri, K. Hudha, and Z. A. Kadir, "Modelling and control strategies in path tracking control for autonomous ground vehicles: A review of state of the art and challenges," *J. Intell. Robot. Syst.*, vol. 86, no. 2, pp. 225–254, May 2017.
- [13] S. Mittal, M. A. Khan, D. Romero, and T. Wuest, "Smart manufacturing: Characteristics, technologies and enabling factors," *Proc. Inst. Mech. Eng., B, J. Eng. Manuf.*, vol. 233, no. 5, pp. 1342–1361, Apr. 2019.
- [14] Q. He, D. Yuan, and A. Ephremides, "Optimal link scheduling for age minimization in wireless systems," *IEEE Trans. Inf. Theory*, vol. 64, no. 7, pp. 5381–5394, Jul. 2018.
- [15] P. Whittle, "Restless bandits: Activity allocation in a changing world," *J. Appl. Probab.*, vol. 25, no. A, pp. 287–298, 1988.
- [16] Y.-P. Hsu, E. Modiano, and L. Duan, "Age of information: Design and analysis of optimal scheduling algorithms," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2017, pp. 561–565.
- [17] W. R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, vol. 25, nos. 3–4, pp. 285–294, Dec. 1933.
- [18] C. H. Papadimitriou and J. N. Tsitsiklis, "The complexity of optimal queueing network control," *Math. Oper. Res.*, vol. 24, no. 2, pp. 293–305, May 1999.
- [19] M. L. Puterman, "Markov decision processes: Discrete stochastic dynamic programming," *J. Oper. Res. Soc.*, vol. 46, no. 6, p. 792, 1995.

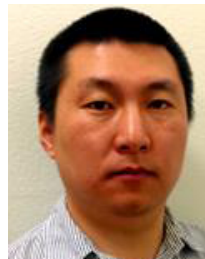
- [20] S. Feng and J. Yang, "Age-optimal transmission of rateless codes in an erasure channel," in *Proc. ICC-IEEE Int. Conf. Commun. (ICC)*, May 2019, pp. 1–6.
- [21] Z. Jiang, B. Krishnamachari, S. Zhou, and Z. Niu, "Can decentralized status update achieve universally near-optimal age-of-information in wireless multiaccess channels?" in *Proc. 30th Int. Teletraffic Congr. (ITC)*, vol. 1, Sep. 2018, pp. 144–152.
- [22] A. G. Dimakis, V. Prabhakaran, and K. Ramchandran, "Decentralized erasure codes for distributed networked storage," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2809–2816, Jun. 2006.
- [23] H.-Y. Lin and W.-G. Tzeng, "A secure erasure code-based cloud storage system with secure data forwarding," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 6, pp. 995–1003, Jun. 2012.
- [24] S. B. Wicker and V. K. Bhargava, *Reed-Solomon Codes and Their Applications*. Hoboken, NJ, USA: Wiley, 1999.
- [25] B. Zhou and W. Saad, "Minimizing age of information in the Internet of Things with non-uniform status packet sizes," in *Proc. ICC-IEEE Int. Conf. Commun. (ICC)*, May 2019, pp. 1–6.
- [26] R. D. Yates, "Lazy is timely: Status updates by an energy harvesting source," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jun. 2015, pp. 3008–3012.



MIANLONG CHEN received the B.Sc. degree in computer science and engineering from the China University of Geosciences, Wuhan, China, in 2017, and the M.Sc. degree in computer science from the University of Victoria, Canada, in 2020. He is currently a Network Engineer with Shanghai Mint Company Ltd. His research interests include network optimization and information theory.



KUI WU (Senior Member, IEEE) received the B.Sc. and M.Sc. degrees from Wuhan University, China, in 1990 and 1993, respectively, both in computer science, and the Ph.D. degree in computing science from the University of Alberta, Canada, in 2002. He joined the Department of Computer Science, University of Victoria, Victoria, BC, Canada, in 2002, where he is currently a Full Professor. His research interests include network tomography, network calculus, edge computing, and network security.



LINQI SONG (Member, IEEE) received the Ph.D. degree in electrical engineering from the University of California at Los Angeles (UCLA), Los Angeles, and the B.S. and M.S. degrees from Tsinghua University. He was a Postdoctoral Scholar with the Department of Electrical and Computer Engineering, UCLA. He is currently an Assistant Professor with the Department of Computer Science, City University of Hong Kong. His research interests include information theory, machine learning, and big data. He has received the Hong Kong RGC Early Career Scheme, in 2019, and the Best Paper Award from IEEE MIPR 2020.

...