A Grouping-based Scheduler for Efficient Channel Utilization under Age of Information Constraints

Lehan Wang, Jingzhou Sun, Yuxuan Sun, Sheng Zhou, Zhisheng Niu

Beijing National Research Center for Information Science and Technology

Department of Electronic Engineering, Tsinghua University, Beijing 100084, P.R. China

{wang-lh19, sunjz18}@mails.tsinghua.edu.cn, {sunyuxuan, sheng.zhou, niuzhs}@tsinghua.edu.cn

Abstract—**We consider a status information updating system where a fusion center collects the status information from a large number of sources and each of them has its own age of information (AoI) constraints. A novel grouping-based scheduler is proposed to solve this complex large-scale problem by dividing the sources into different scheduling groups. The problem is then transformed into deriving the optimal grouping scheme. A two-step grouping algorithm (TGA) is proposed: 1) Given AoI constraints, we first identify the sources with harmonic AoI constraints, then design a fast grouping method and an optimal scheduler for these sources. Under harmonic AoI constraints, each constraint is divisible by the smallest one and the sum of reciprocals of the constraints with the same value is divisible by the reciprocal of the smallest one. 2) For the other sources without such a special property, we pack the sources which can be scheduled together with minimum update rates into the same group. Simulations show the channel usage of the proposed TGA is significantly reduced as compared to a recent work and is 0.42% larger than a derived lower bound when the number of sources is large.**

I. Introduction

Future 5G and 6G networks are expected to energize industrial Internet of Things (IoT), in which smart machines such as moving vehicles operate autonomously by collecting status information from remote sensors. To guarantee prompt and reliable decision-making, the timeliness of the status information is crucial. For this purpose, a new metric called age of information (AoI) has been proposed in [1]. It is the time elapsed since the generation of the latest status information received by the destination.

AoI has been widely studied by researchers [2]–[15]. Most existing works focus on the average AoI over time and different information sources. Since extreme values of AoI are also important, a policy with optimal average age may not be enough for some time-critical applications, such as vehicle-to-vehicle communications [2]. Therefore, in this paper, the schedulers with AoI constraints are studied. We consider a scenario where a fusion center collects status information from multiple sources. The age of the status information from each source should not exceed their AoI constraints. Assuming at most one source can transmit information in a time slot on a channel, we are interested in the minimum number of channels that can guarantee the AoI constraints of all sources and how to construct a scheduler for them.

The most relevant works are [3], [4]. In [3], the authors assume there is only one channel and can always find a

scheduler when $\sum_{n=1}^{N} \frac{1}{d_n}$, the load of AoI constraints of N sources $[d_1, d_2, \ldots, d_N]$, is less than or equal to $\ln 2$. In [4], the authors propose an algorithm named Aion, which can construct cyclic schedulers for any AoI constraints. The ratio of the number of channels achieved by Aion to the optimal solution is upper bounded by a function of the update frequency of the sources and the load of AoI constraints. Aion is optimal only when the AoI constraints are consecutively divisible, where d_{n+1} is divisible by d_n for $n \in \{1, \ldots, N-1\}$. For the generalized AoI constraints, Aion narrows the search space to the cyclic schedulers with consecutively divisible average transmission intervals and may not be optimal. In this paper, our grouping method efficiently decomposes this complex channel minimization problem into several sub-problems, and does not require the schedulers to provide special average transmission intervals for all the sources.

The contributions of this paper are summarized as follows:

- We propose a grouping method to split the problem into multiple small-scale sub-problems by dividing the sources into different groups, and constructing schedulers for each group separately. Based on the grouping method, we then derive an upper bound of the number of channels and transform the channel minimization problem into finding the optimal grouping method.
- A novel two-step grouping algorithm (TGA) is proposed to solve the grouping problem. Given AoI constraints, we first identify the sources with harmonic AoI constraints, because we can design a fast grouping method and an optimal scheduler for such sources. Under harmonic AoI constraints, all the constraints are divisible by the smallest one, and the sum of reciprocals of the constraints with the same value is divisible by the reciprocal of the smallest AoI constraints. Secondly, for other sources, a heuristic grouping algorithm is proposed. The sources are assigned to the group with the minimum distance. The distance between source n and group g is the difference between $\frac{1}{d_n}$ and the minimum achievable update frequency when n is packed into g, where d_n is AoI constraint of n. Simulations show the number of channels achieved by the proposed TGA is reduced by 20% than that of Aion when there are 300 sources.

The rest of the paper is organized as follows. In Section II, the system model is described. The problem formulation and

motivations of our grouping method are presented in Section III. The schedulers for harmonic AoI constraints are discussed in Section IV while TGA is proposed in Section V. Finally, the simulation results are shown in Section VI, and the conclusions are drawn in Section VII.

II. System Model

We consider a time-slotted system with one fusion center and N information sources sharing K reliable wireless channels, as shown in Fig. 1. At the beginning of each time slot, the scheduled sources transmit the current status to the fusion center. The status information will be received by the center at the end of the slot. Denote the scheduling decision of source n by $U_n(t) \in \{0,1\}$. If $U_n(t) = 1$, then source n is scheduled at time slot t to transmit its status information to the center.
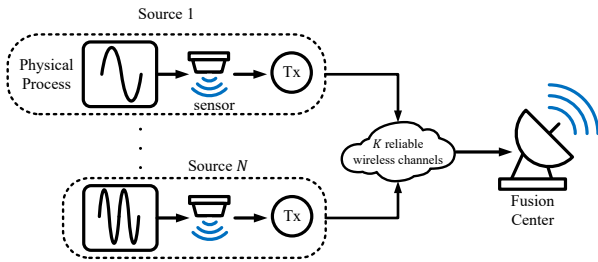

Fig. 1. System model.

AoI is used to measure the freshness of status information. Denote the generation time of the latest received update from source n and the AoI of source n at time slot t by $G_n(t)$ and $A_n(t)$, respectively. Then $A_n(t)$ is the time elapsed since $G_n(t)$, namely $A_n(t) = t - G_n(t)$. If source n is scheduled at time t, then $A_n(t+1)$ will decrease to 1. If $U_n(t) = 0$, $A_n(t+1)$ will increase by 1, i.e.,

$$A_n(t+1) = \begin{cases} 1, & U_n(t) = 1; \\ A_n(t) + 1, & U_n(t) = 0. \end{cases} \quad (1)$$

The AoI constraint of source n is denoted by $d_n \in \mathbb{Z}^+$, where $\mathbb{Z}^+$ denotes positive integers. Without loss of generality, we assume $d_1 \le d_2 \le d_3 \cdots \le d_N$. In addition, each channel can schedule at most one source in a time slot, therefore $K \ge \max_t \sum_{n=1}^N U_n(t)$. Our goal is to find a scheduler π which can satisfy the AoI constraints and minimize the number of channels, namely K. The t-th element of scheduler π contains scheduling decisions of N sources in time slot t, namely $\pi(t) = [U_1(t), \cdots, U_n(t), \cdots, U_N(t)]$.

III. Problem Formulation and Analysis

The problem can be formulated as follow:

$$\min_{\pi} K,$$
$$s.t. \ A_n(t) \le d_n, \forall n = 1, \cdots, N, \forall t = 1, 2, \cdots. \quad (2)$$

Our objective is to find a scheduler π which can satisfy the AoI constraints and minimize the number of channels required. The following lemma, also shown in [4], presents the lower bound of this scheduling problem.

Lemma 1. *Given AoI constraints* $\boldsymbol{d} = [d_1, d_2, \cdots, d_N]$, *we have*

$$K^\star \ge \left\lceil \sum_{n=1}^N \frac{1}{d_n} \right\rceil, \quad (3)$$

where $K^\star$ is the optimal number of channels.

Proof. Each source requires at least $\frac{1}{d_n}$ of the time slots of a channel, therefore $K^\star \ge \lceil \sum_{n=1}^N \frac{1}{d_n} \rceil$. □

Since scheduler π possesses infinite elements, solving (2) is challenging. If we can narrow the candidate scheduler spaces to cyclic schedulers, which are formally defined in Definition 1, the problem can be significantly simplified.

Definition 1. *A scheduler is cyclic if and only if there exists $C \in \mathbb{Z}^+$ such that $U_n(t) = U_n(t+C), \forall n \in \{1, 2, \cdots, N\}, \forall t$. The cycle length of the scheduler is the smallest integer C which satisfies $U_n(t) = U_n(t+C)$ for N sources.*

The following lemma provides us with the basis to consider only cyclic schedulers and not non-cyclic schedulers.

Lemma 2. *For any AoI constraints* $\boldsymbol{d} = [d_1, d_2, \cdots, d_N]$, *there exists an optimal cyclic scheduler which only requires $K^\star$ channels.*

Proof. To prove this lemma, we define the state of N sources at time t as $\boldsymbol{S}(t) = [A_1(t), \cdots, A_N(t)]$. With scheduling decision $\pi(t)$, the state will transit into $\boldsymbol{S}(t+1) = (\boldsymbol{J}_{1 \times N} - \pi(t)) \odot \boldsymbol{S}(t) + \boldsymbol{J}_{1 \times N}$, where $\boldsymbol{J}_{1 \times N}$ is a N-dimensional vector whose elements are all 1, $\odot$ represents element-wise product between two matrices. For any scheduler which can satisfy the AoI constraints, $A_n(t) \le d_n$ holds for all the sources. Therefore, there are $\prod_{n=1}^N d_n$ different states. We assume there exists an optimal scheduler $\pi^\star$ which requires $K^\star$ channels. With $\pi^\star$, there must be a state which occurs more than once in the time interval from the first slot to the $\left(\prod_{n=1}^N d_n + 1\right)$-th slot. Let t_1 and t_2 denote the time slot when the state appears for the first time, and the second time, respectively. Then we can construct an optimal cyclic scheduler π' with cycle length $C = t_2 - t_1$. For time slot $t' = t + mC$, $t \in [1, C], m \in \mathbb{N}$, let $\pi'(t') = \pi^\star(t)$, where $\mathbb{N}$ denotes natural numbers. Then π' also satisfies the AoI constraints and requires $K^\star$ channels. □

Therefore, to solve problem (2), we first determine all the possible transitions among the states and then identify the loop (a transition path starts and ends at the same state) requiring the minimum number of channels. Due to the large state space, searching for the optimal scheduler is computationally prohibitive. A simple idea is to divide N sources into different groups and design schedulers for each group independently. Such an operation can split the large-scale problem into multiple sub-problems and efficiently narrows down the state space.

Based on this idea, we focus on the design of the grouping scheme and the corresponding cyclic scheduler in this paper. To discuss what kind of sources can be divided into

a same group, we start from the simplest scenario where $d_1 = \cdots = d_N$. In this case, a scheduler can be constructed by simply assigning every d_1 time slots to each source. Under this cyclic scheduler, the average transmission interval of source n is $l_n = \frac{C}{\sum_{t=1}^{C} U_n(t)}$ and $l_n = d_1$ holds for each source. Since each channel can schedule at most d_1 sources, such a scheduler requires $\lceil N/d_1 \rceil$ channels, which meets the lower bound of the problem. Based on this inspiring observation, we can construct a scheduler named grouping for distinct values (GD) given any AoI constraints d. We define the number of distinct values in d as v. $u = \{u_1, \cdots, u_v\}$ includes all the distinct values. Let o_j denote the number of occurrences of u_j in d. In GD, the sources with the same AoI constraints are packed into a same group. Then we design schedulers for each group independently. Since each channel can schedule at most u_j sources with AoI constraints u_j, then o_j sources can be scheduled by $\lceil \frac{o_j}{u_j} \rceil$ channels. By assigning different channels to each group, GD requires $\sum_{j=1}^{v} \lceil \frac{o_j}{u_j} \rceil$ channels. Note that when $v = 1$ or $\frac{o_j}{u_j} \in \mathbb{Z}^+$, $\forall j$, GD is optimal since $\sum_{j=1}^{v} \lceil \frac{o_j}{u_j} \rceil = \lceil \sum_{n=1}^{N} \frac{1}{d_n} \rceil$ in these cases. Based on GD, we can present an upper bound of the problem.

Lemma 3. *Given AoI constraints $d = [d_1, d_2, \cdots, d_N]$, we have*

$$K^\star \leq \sum_{j=1}^{v} \left\lceil \frac{o_j}{u_j} \right\rceil. \tag{4}$$

Proof. The proof is based on the construction of GD. $\square$

Although GD provides the simplest grouping principle and the corresponding cyclic schedulers, GD fails to use the channels efficiently. A channel is *fully utilized* only when all the time slots of the channel are occupied. In $\lceil \frac{o_j}{u_j} \rceil$ channels assigned to sources with AoI constraint u_j, there is at most one channel not fully utilized. Therefore, given any AoI constraints d, there are v channels that are not effectively utilized in the worst case. It is conceivable that when v is large and o_j is relatively small for all the distinct values, the performance of GD is poor. To solve this problem, we should find more effective schedulers to extend the conditions of the sources which can be scheduled in a same group.

IV. SCHEDULERS FOR HARMONIC AoI CONSTRAINTS

Before introducing our two-step grouping method, we first consider some special AoI constraints and design the optimal schedulers for them. The theoretical results provided in this section will be the foundation for the first step of our grouping method. We first present a definition:

Definition 2. *Given $x = [x_1, \cdots, x_N]$, $u = [u_1, \cdots, u_v]$ includes the distinct values of x. x is harmonic if $x_n \in \mathbb{Z}^+, \forall n \in \{1, \cdots, N\}$, and $v \geq 2$, $\frac{u_i}{u_1} \in \mathbb{Z}^+, \forall i \in \{2, \cdots, v\}$. Besides, the number of occurrences of u_i in x is an integral multiple of $\frac{u_i}{u_1}$.*

For example, $x = [2, 4, 4, 4, 4, 6, 6, 6]$ is harmonic, since both 4 and 6 are divisible by the first element in x and the occurrences of 4 and 6 satisfy the conditions in Definition 2.

Theorem 1. *For any harmonic AoI constraints $d = [d_1, \cdots, d_N]$, there exists an optimal cyclic scheduler.*

Proof. We prove Theorem 1 by constructing a cyclic scheduler named harmonic scheduler (HS), which requires $\lceil \sum_{n=1}^{N} \frac{1}{d_n} \rceil$ channels under harmonic AoI constraints d. We first construct a scheduler by GD for AoI constraints d' which contains $\sum_{n=1}^{v} \frac{o_n u_1}{u_n}$ elements with value u_1. o_n is the number of occurrences of u_n in d. Since elements in d' are the same, GD is optimal for d'. For ease of notation, resource block (t, k) denotes the t-th time slot of channel k. Define the sequence of resource blocks assigned to the i-th element in d' as $\mathcal{R}_i, i \in \{1, \cdots, \sum_{n=1}^{v} \frac{o_n u_1}{u_n}\}$. The time gap between two consecutive resource blocks in $\mathcal{R}_i$ is u_1. Therefore we assign $\{\mathcal{R}_1, \cdots, \mathcal{R}_{o_1}\}$ to o_1 sources with AoI constraints u_1. For sources with AoI constraints u_n, we find the first sequence of resource blocks that has not been fully occupied, denoted by $\mathcal{R}_q$. Then we identify (t_p, k_p), the first available resource block in $\mathcal{R}_q$, and assign resource blocks $(t_p + m u_n, k_p), m \in \mathbb{N}$ to the source. Therefore each sequence of resource blocks can schedule at most $\frac{u_n}{u_1}$ sources with AoI constraint u_n, and o_n sources with AoI constraints u_n require $\frac{o_n u_1}{u_n}$ sequences of resource blocks. Such requirements can be satisfied because GD provides $\sum_{n=1}^{v} \frac{o_n u_1}{u_n}$ sequences of resource blocks. Since the lower bound of channels are the same for d and d' and HS requires the same number of channels as GD, HS is optimal given any harmonic AoI constraints. $\square$

Theorem 1 expands the range of sources which can be packed into a same group to the sources with harmonic AoI constraints. Also, HS is more efficient compared with GD. For example, given harmonic $d = [2, 4, 4, 4, 4, 6, 6, 6]$, GD requires 3 channels while HS requires 2 channels.

Furthermore, we can also handle the cases when d is a combination of two harmonic AoI constraints with different bases. The base is defined as the smallest element of the harmonic AoI constraints.

Lemma 4. *For AoI constraints with two distinct values: u_1 and u_2, there exists an optimal cyclic scheduler if $\frac{o_1}{u_1} + \frac{o_2}{u_2}$ is an integer. o_1, o_2 is the number of occurrences of u_1 and u_2 in AoI constraints, respectively.*

Proof. We prove Lemma 4 by constructing a scheduler for two distinct values (STV). According to Lemma 1, the lower bound of the minimum number of channels is $K_{\text{LB}} = \frac{o_1}{u_1} + \frac{o_2}{u_2}$. If STV requires K_{LB} channels for any u_1, u_2, o_1, o_2 satisfying the conditions in Lemma 4, then the proof is completed.

For generality, we assume $u_1 < u_2$ and $K_{\text{LB}} = b$. Define $\gcd(u_1, u_2)$ as the greatest common divisor of u_1 and u_2. Then according to linear Diophantine equation, o_1 and o_2 must satisfy the following conditions: $o_1 = x_0 + m u_1/\gcd(u_1, u_2)$, $o_2 = y_0 - m u_2/\gcd(u_1, u_2)$, where m, x_0, y_0 are integers. (x_0, y_0) is any set of solutions to equation $\frac{x}{u_1} + \frac{y}{u_2} = b$. If we choose $x_0 = 0, y_0 = b u_2$, then we can prove that o_1 is divisible by $u_1/\gcd(u_1, u_2)$, and o_2 is divisible by $u_2/\gcd(u_1, u_2)$.

If $\gcd(u_1, u_2) = 1$, then o_1, o_2 are multiples of u_1, u_2, respectively. In this case, STV can be reduced to GD. When

Algorithm 1 Scheduler for two distinct values (STV)

Input: AoI constraint $\boldsymbol{d}$ with two distinct values: $\{u_1, u_2\}$. o_1, o_2 are the number of occurrences of u_1, u_2.

Output: A cyclic scheduler $\boldsymbol{\pi}$;

1: Set the cycle length of $\boldsymbol{\pi}$ as $C = \mathrm{lcm}\{u_1, u_2\}$ and the number of channels required by $\boldsymbol{\pi}$ is $K_{\mathrm{LB}} = \frac{o_1}{u_1} + \frac{o_2}{u_2}$.

2: Divide C slots of $\boldsymbol{\pi}$ into $C/\gcd(u_1, u_2)$ groups, each with $\gcd(u_1, u_2)K_{\mathrm{LB}}$ resource blocks.

3: Assign $\frac{o_1 \gcd(u_1, u_2)}{u_1}$ resource blocks in the first group to sources with AoI constraint u_1. In the other groups, assign resource blocks in the same positions as in the first group to sources with AoI constraint u_1.

4: Schedule sources with AoI constraint u_2 similarly.

$\gcd(u_1, u_2) > 1$, we set the cycle length of STV as the least common multiple of u_1 and u_2, denoted by $\mathrm{lcm}(u_1, u_2)$. Then we divide $\mathrm{lcm}(u_1, u_2)$ time slots in a cycle into $a = \mathrm{lcm}(u_1, u_2)/\gcd(u_1, u_2)$ groups. Each group includes $\gcd(u_1, u_2)$ consecutive time slots and $b\gcd(u_1, u_2)$ resource blocks.

We then separate o_1 sources into the first $u_1/\gcd(u_1, u_2)$ groups, each group contains $\frac{o_1 \gcd(u_1, u_2)}{u_1}$ sources and each source is assigned one resource block. Besides, positions of the resource blocks assigned to the sources with the same AoI constraint should be the same in each group. For the j-th group ($j = eu_1/\gcd(u_1, u_2) + a'$), the scheduler just 'copies' the scheduling decisions in group a', where $a' \in \{1, \cdots, u_1/\gcd(u_1, u_2)\}, e \in \{1, \cdots, \mathrm{lcm}\{u_1, u_2\}/u_1 - 1\}$. The sources with AoI constraints u_2 can be scheduled similarly.

Since $\frac{o_1 \gcd(u_1, u_2)}{u_1} + \frac{o_2 \gcd(u_1, u_2)}{u_2} = b\gcd(u_1, u_2)$, STV requires K_{LB} channels. The lemma is proved. $\square$

Theorem 2. *For two harmonic AoI constraints $\boldsymbol{d}_1 = [d_{1,1}, \cdots, d_{1,i}]$ with length i and $\boldsymbol{d}_2 = [d_{2,1}, \cdots, d_{2,j}]$ with length j, there exists an optimal cyclic scheduler for the $i+j$ sources if $\sum_{n=1}^{i} \frac{1}{d_{1,n}} + \sum_{n=1}^{j} \frac{1}{d_{2,n}} \in \mathbb{Z}^+$.*

Proof. The proof is based on construction. We first use STV to construct a cyclic scheduler for n_1 sources with AoI constraint $d_{1,1}$ and n_2 sources with AoI constraint $d_{2,1}$, where $n_1 = d_{1,1}\sum_{n=1}^{i} \frac{1}{d_{1,n}}, n_2 = d_{2,1}\sum_{n=1}^{j} \frac{1}{d_{2,n}}$. Since $\frac{n_1}{d_{1,1}} + \frac{n_2}{d_{2,1}}$ is an integer, STV is optimal according to Lemma 4. Let $\mathcal{R}_1$ and $\mathcal{R}_2$ denote the sequences of resources blocks allocated to the n_1 and n_2 sources. We use HS to distribute $\mathcal{R}_1$ and $\mathcal{R}_2$ to the sources with AoI constraint $\boldsymbol{d}_1$ and $\boldsymbol{d}_2$. Therefore, such a scheduler requires $\frac{n_1}{d_{1,1}} + \frac{n_2}{d_{2,1}}$ channels, which is also the lower bound of the number of channels. The proof is completed. $\square$

V. TWO-STEP GROUPING ALGORITHM

Now we can design optimal schedulers for harmonic AoI constraints. However, how to construct a scheduler with efficient channel utilization for generalized AoI constraints is still a challenging problem.

To utilize the theoretical results obtained in section IV, we propose a two-step grouping algorithm (TGA):

- Step 1: Given a set of sources $\boldsymbol{N} = \{1, \cdots, N\}$ and the corresponding AoI constraints $\boldsymbol{d} = [d_1, \cdots, d_N]$, we first search the sources with harmonic AoI constraints and design a scheduler for them.
- Step 2: For the other sources without harmonic AoI constraints, we design a new scheduler with optimized channel utilization for them.

In the following of this section, we first introduce our algorithm of identifying the sources with harmonic AoI constraints and constructing schedulers for them. Secondly, we display the grouping algorithm for the other sources without such a special property.

A. Step 1: Harmonic Sources Identifying & Scheduling

According to Theorems 1 and 2, if $\boldsymbol{d}$ is harmonic or $\boldsymbol{d}$ includes any two harmonic AoI constraints satisfying the conditions in Theorem 2, the corresponding sources can also be packed in a group for efficient channel utilization.

Therefore, if $\boldsymbol{d}$ is a combination of h harmonic AoI constraints, denoted by $\boldsymbol{d}_1, \cdots, \boldsymbol{d}_h$, we can divide the sources into at most h groups. For each group, we use STV and HS to design a scheduler and assign channels independently. The number of channels required by such a procedure is no more than $\sum_{i=1}^{h} Q_i$, where $Q_i = \lceil \sum_{j \in \boldsymbol{d}_i} \frac{1}{d_{i,j}} \rceil$ and $\boldsymbol{d}_i = [d_{i,1}, \cdots, d_{i,j}, \cdots]$.

Based on the analysis above, our harmonic sources identifying (HSI) algorithm is proposed in Algorithm 2. Given AoI constraints $\boldsymbol{d}$, we first derive $\boldsymbol{u} = [u_1, \cdots, u_v]$ which includes all the distinct values of $\boldsymbol{d}$, and arrange $\boldsymbol{u}$ in ascending order. HSI is divided into two parts. The first part is completed in v iterations. In the i-th iteration, we search the sources which possess harmonic AoI constraints with base u_i. $\boldsymbol{d}_i = [d_{i,1}, \cdots, d_{i,m_i}]$ with length m_i denotes the harmonic AoI constraints with base u_i. Besides, the elements in $\boldsymbol{d}_i$ is arranged in ascending order. If we directly construct a scheduler for $\boldsymbol{d}_i$ by HS, there will be at most one channel which is not fully utilized. This is due to the introduction of GD in HS. Since the channels are assigned to each group independently, such an inefficient use of channel seriously degrade the performance of the grouping. Therefore, we only construct a scheduler based on HS for the first p elements in $\boldsymbol{d}_i$, where p satisfies $\sum_{n=1}^{p} \frac{1}{d_{i,n}} = \lfloor \sum_{n=1}^{m_i} \frac{1}{d_{i,n}} \rfloor$. The other $m_i - p$ sources will be scheduled in the second part of HSI or by the heuristic grouping algorithm, which is discussed in the following subsection. After the i-th iteration, p sources with AoI constraints $[d_{i,1}, \cdots, d_{i,p}]$ are excluded from $\boldsymbol{N}$.

In the second part, for distinct value u_i, we find sources with AoI constraint u_j such that $\gcd(u_i, u_j) > 1$. We first construct the harmonic AoI constraints $\boldsymbol{d}_i, \boldsymbol{d}_j$ with bases of u_i and u_j. Then we find the sources with $\boldsymbol{d}_i, \boldsymbol{d}_j$ satisfying the conditions in Theorem 2 and design an optimal scheduler for them based on STV and HS. After the two parts, an optimal scheduler is obtained for all the sources searched by HSI. In addition, we do not take the cases when $\gcd(u_i, u_j) = 1$ into account in the second part. This is because the sum of reciprocals

Algorithm 2 Harmonic Sources Identifying (HSI)

Input: AoI constraint vector d and the set of sources N;

Output: Sources with harmonic AoI constraints N', other sources with generalized constraints $\bar{N}' = N \setminus N'$, a cyclic scheduler π_H for N'.

1: Initialize $N' = \emptyset$. $u = \{u_1, \cdots, u_v\}$ includes distinct values in d. Arrange u in ascending order. o_j, the j-th element in o, is the number of occurrences of u_j in d.

2: **for** $i \in [1, v]$ **do**

3:    $f_i = \{j | \frac{u_j}{u_i} \in \mathbb{Z}^+\}$, $d_i = \emptyset$

4:    **for** $n \in f_i$ **do**

5:       Put $\lfloor \frac{o_n u_i}{u_n} \rfloor \frac{u_n}{u_i}$ elements with value of u_n into d_i.

6:    **end for**

7:    Arrange $d_i = [d_{i,1}, \cdots, d_{i,m_i}]$ in ascending order.

8:    Find p such that $\sum_{n=1}^{p} \frac{1}{d_{i,n}} = \lfloor \sum_{n=1}^{m_i} \frac{1}{d_{i,n}} \rfloor$. Add p sources with AoI constraints $[d_{i,1}, \cdots, d_{i,p}]$ into N'.

9:    Design a scheduler π_i for the p sources based on HS.

10:   $\bar{N}' = N \setminus N'$. Update o for sources in $\bar{N}'$.

11: **end for**

12: **for** $i \in [1, v]$ **do**

13:    **for** $j \in \{j | \gcd(u_j, u_i) > 1, u_j > u_i, \frac{u_j}{u_i} \notin \mathbb{Z}^+\}$ **do**

14:       Repeat Line (3)-(7) and obtain d_i with length m_i.

15:       $f_j = \{e | \frac{u_e}{u_j} \in \mathbb{Z}^+, \frac{u_e}{u_i} \notin \mathbb{Z}^+\}$

16:       Repeat Line (4)-(7) and obtain d_j with length m_j.

17:       Let $s_i = d_{i,1} \sum_{n=1}^{m_i} \frac{1}{d_{i,n}}$, $s_j = d_{j,1} \sum_{n=1}^{m_j} \frac{1}{d_{j,n}}$

18:       Let $b = \lfloor \frac{s_i}{u_i} + \frac{s_j}{u_j} \rfloor$, find the largest positive integer $s_i' \leq s_i$ such that $s_j' = \frac{1}{u_j}(bu_iu_j - u_js_i') \in \mathbb{Z}^+$

19:       **if** s_i' exists and $s_j' \leq s_j$ **then**

20:          Find p_i, p_j such that $\sum_{n=1}^{p_z} \frac{1}{d_{z,n}} = \frac{s_z'}{u_z}$, $z \in \{i, j\}$. Add $p_i + p_j$ sources with AoI constraints $[d_{i,1}, \cdots, d_{i,p_i}, d_{j,1}, \cdots, d_{j,p_j}]$ into N'.

21:          Design a scheduler π_i' for the $p_i + p_j$ sources based on STV and HS.

22:          $\bar{N}' = N \setminus N'$ and update o for sources in $\bar{N}'$.

23:       **end if**

24:    **end for**

25: **end for**

26: π_H is a combination of π_i and π_i', $\forall i \in \{1, \cdots, v\}$. The number of channels required by π_H is $\sum_{n \in N'} \frac{1}{d_n}$.

---

of constraints in the two harmonic vectors will be integers if $\gcd(u_i, u_j) = 1$, and such sources are found in the first part.

B. Step 2: heuristic grouping algorithm

After HSI, we find harmonic groups and design an optimal cyclic scheduler for them. For the other sources without harmonic AoI constraints, HS is not optimal, hence we have to design a new scheduler for them. Before introducing the heuristic grouping algorithm (HGA), a definition is presented.

Definition 3. *A vector $x = [x_1, x_2, \cdots, x_N]$ is consecutively divisible if $x_i \in \mathbb{R}^+$, $x_i \geq 1$, $\forall i \in \{1, \cdots, N\}$ and $x_i/x_{i-1} \in \mathbb{Z}^+$, $\forall i \in \{2, \cdots, N\}$.*

Algorithm 3 Consecutively divisible Scheduler (CS)

Input: $l = [l_1, \cdots, l_N]$ with $l_1 \leq \cdots \leq l_N$;

Output: A cyclic scheduler π_S;

1: Find the minimum integer a satisfying $al_1 \in \mathbb{Z}^+$. Construct π' requiring $\lceil \frac{1}{a} \lceil \sum_{n=1}^{N} \frac{1}{l_n} \rceil \rceil$ channels. The cycle length $C_{\pi'} = al_N$ if $l_N \in \mathbb{Z}^+$, otherwise $C_{\pi'} = a^2 l_N$.

2: Let $M = [M_1, \cdots, M_{C_{\pi'}/a}]$ with each $M_j = \lceil \sum_{n=1}^{N} \frac{1}{l_n} \rceil$

3: **for** $i \in [1, N]$ **do**

4:    Find p such that M_p is the first largest element in $[M_1, \cdots, M_{\lceil l_i \rceil}]$

5:    Find available resource block (t_p, k_p) such that $\sum_{n=1}^{N} U_n(t_p)$ is the first smallest element in $[\sum_{n=1}^{N} U_n((p-1)a+1), \cdots, \sum_{n=1}^{N} U_n(\min\{pa, al_i\})]$.

6:    Assign resource blocks $(t_p + mal_i, k_p)$, $m \in \{0, \cdots, C_{\pi'}/(al_i) - 1\}$ to the source i. Let $M_r = M_r - 1$, for $r = \lceil \frac{t_p + mal_i}{a} \rceil$, $m \in \{0, \cdots, C_{\pi'}/(al_i) - 1\}$

7: **end for**

8: Set the cycle length of π_S as $C_S = C_{\pi'}/a$.

9: The number of channels required by π_S is $\lceil \sum_{n=1}^{N} \frac{1}{l_n} \rceil$.

10: **for** $i \in [1, C_S]$ **do**

11:    N_i denotes the sources scheduled by π' in slots $[(i-1)a+1, ia]$.

12:    In π_S, set $U_n(i) = 1$ for each source n in N_i.

13: **end for**

In Definition 3, $\mathbb{R}^+$ denotes positive real numbers. If $[d_1, \cdots, d_N]$ is consecutively divisible, then a consecutively divisible AoI constraints scheduler (CAS) requiring $\lceil \sum_{n=1}^{N} \frac{1}{d_n} \rceil$ channels can be constructed [4]. CAS can be constructed within N iterations. In the n-th iteration, we find the smallest $t_n \leq d_n$ and the smallest $k_n \leq K$ such that resource block (t_n, k_n) is available. Then resource blocks $\mathcal{R}_n = \{(t, k_n) | t = t_n + md_n, m \in \mathbb{N}\}$ are assigned to source n, namely $U_n(t_n + md_n) = 1$. According to Lemma 1, CAS is optimal for consecutively divisible AoI constraints.

For generalized AoI constraints, CAS can not be used directly. According to [4], if there exist consecutively divisible average transmission intervals $l = [l_1, l_2, \cdots, l_N]$ such that $l_n \leq d_n, \forall n \in \{1, \cdots, N\}$, a consecutively divisible scheduler (CS) requiring $\lceil \sum_{n=1}^{N} \frac{1}{l_n} \rceil$ channels can be constructed as a generalized version of CAS. The details of CS is shown in Algorithm 3.

With CS, we can design a scheduler under any AoI constraints. Given generalized AoI constraints, a direct idea is to find consecutively divisible average transmission intervals requiring the minimum number of channels based on CS, i.e.,

$$\min_{l} \left\lceil \sum_{n=1}^{N} \frac{1}{l_n} \right\rceil,$$
$$s.t. 1 \leq l_n \leq d_n, \forall n = 1, 2, \cdots, N, \quad (5)$$
$$l \text{ is consecutively divisible.}$$

Let K_1 denote the number of channels required by the solution of (5). However, such a method may lead to inefficient channel utilization, which is shown in the following example.

Example 1. *Consider sources $\{A, B, C, D, E, F, G, H, I, J\}$ and generalized AoI constraints $\boldsymbol{d} = [3, 5, 5, 5, 6, 6, 6, 7, 7, 7]$, then $\boldsymbol{l}^\star = [2.5, 5, 5, 5, 5, 5, 5, 5, 5, 5]$ is the solution of (5). The cyclic scheduler constructed by CS is:*

time slot	1	2	3	4	5	6	7	8	9	10	
channel 1	A	B	A	C	D	A	B	A	C	D	$\cdots$
channel 2	E	F	G	H	I	E	F	G	H	I	$\cdots$
channel 3	J	□	□	□	□	J	□	□	□	□	$\cdots$

CS requires 3 channels while the lower bound of the number of channels is 2. Besides, channel 3 is not fully utilized. We can construct a cyclic scheduler which meets the lower bound.

time slot	1	2	3	4	5	6	7	8	9	10	11	12	
channel 1	A	B	A	C	D	A	B	A	C	D	A	B	$\cdots$
channel 2	E	F	G	H	I	J	E	F	G	H	I	J	$\cdots$

The scheduler above possesses average transmission intervals $\boldsymbol{l} = [2.5, 5, 5, 5, 6, 6, 6, 6, 6, 6]$, which is not consecutively divisible. The cycle length C is the least common multiple of cycle length of channel 1 and 2, namely $C = \text{lcm}\{5, 6\} = 30$.

$\boldsymbol{l}$ in Example 1 is actually a combination of two different consecutively divisible vectors, namely $\boldsymbol{l}_1 = [2.5, 5, 5, 5]$ and $\boldsymbol{l}_2 = [6, 6, 6, 6, 6, 6]$. The cyclic scheduler is also a combination of two schedulers derived by CS, with average transmission interval vectors $\boldsymbol{l}_1$ and $\boldsymbol{l}_2$, respectively. This phenomenon leads to the following theorem.

Theorem 3. *For any average transmission interval vector $\boldsymbol{l}$ which is a combination of h consecutively divisible vectors $\boldsymbol{l}_1, \boldsymbol{l}_2, \cdots, \boldsymbol{l}_h$, there exists a cyclic scheduler requiring $\sum_{j=1}^{h} Q_j$ channels, where $Q_j = \lceil \sum_{i \in \boldsymbol{l}_j} \frac{1}{l_{j,i}} \rceil$ and $\boldsymbol{l}_j = [l_{j,1}, \cdots, l_{j,i}, \cdots]$.*

Based on Theorem 3, we can also divide the sources into different groups and design a scheduler for each group independently given generalized AoI constraints. We still need to find a grouping method for sources such that the corresponding average transmission intervals lead to minimum number of channels. The problem can be formulated as follow:

$$\min_{\mathcal{G}, \boldsymbol{l}} \sum_{j \in \mathcal{G}} \left\lceil \sum_{i \in \boldsymbol{g}_j} \frac{1}{l_i} \right\rceil,$$

$$s.t. \begin{cases} 1 \leq l_n \leq d_n, \forall n = 1, 2, \ldots, N \\ \boldsymbol{l}_j \text{ is consecutively divisible}, \forall j = 1, \ldots, |\mathcal{G}|, \end{cases} \quad (6)$$

where $\mathcal{G}$ represents a grouping method and $|\mathcal{G}|$ is the number of groups. $\boldsymbol{g}_j$ includes the sources in the j-th group. The average transmission interval of source n is l_n and $\boldsymbol{l}_j$ denotes the consecutively divisible average transmission intervals of the j-th group.

However, searching for the optimal grouping method is computationally prohibitive. At least $2^{N-1} - 1$ grouping methods already exist for the case of just dividing N sources into two groups. We have three naive tricks to narrow down the search space: 1) Problem (5) is a special case of problem (6) when we only have one group, then if K_1 equals the lower bound in Lemma 1, there is no need to find other grouping

schemes. 2) When K_1 is larger than the lower bound, to achieve better performance, the optional number of groups should be larger than 1. Additionally, by Theorem 3, each group is assigned at least one channel. Therefore the optional number of groups is upper bounded by $K_1 - 1$. 3) The grouping scheme $\mathcal{G}$ with the lower bound $\sum_{j \in \mathcal{G}} \lceil \sum_{i \in \boldsymbol{g}_j} \frac{1}{d_i} \rceil > K_1$ can be excluded.

Solving problem (6) is challenging with the tricks above because there are still many optional grouping methods left. Therefore, we propose a heuristic grouping algorithm (HGA).

Note that the grouping problem in (6) is similar to a clustering problem in the field of machine learning. Some definitions inspired by the clustering are presented.

Definition 4. *The center of a group is the source whose average transmission interval obtained by solving problem (5) for the sources in this group, equals its AoI constraint.*

Based on property 2 in [4], if $\boldsymbol{l}^\star = \{l_1^\star, \cdots, l_N^\star\}$ is the solution to (5) with minimum $\sum_{n=1}^{N} \frac{1}{l_n^\star}$, then there exists at least one element i in $\boldsymbol{l}^\star$ such that $l_i^\star = d_i$. Since the average transmission interval obtained by solving problem (5) is consecutively divisible, this property can be used to narrow down the search space of the grouping method. To achieve better performance, we should make the gap between the AoI constraints and the average transmission intervals obtained by solving problem (5) in each group as small as possible. Given source μ_g as the center of group g, we define the distance between group g and any source n as follows:

Definition 5. *The distance between group g and source n is:*

$$D_{g,n} = \begin{cases} \dfrac{1}{\lfloor \frac{d_n}{d_{\mu_g}} \rfloor d_{\mu_g}} - \dfrac{1}{d_n}, & d_n \geq d_{\mu_g}; \\ \dfrac{\lceil \frac{d_{\mu_g}}{d_n} \rceil}{d_{\mu_g}} - \dfrac{1}{d_n}, & d_n < d_{\mu_g}. \end{cases} \quad (7)$$

If n is put into group g, then there are two different cases: 1) when $d_n \geq d_{\mu_g}$, d_{μ_g} should be a divisor of l_n, then the largest l_n is $\lfloor \frac{d_n}{d_{\mu_g}} \rfloor d_{\mu_g}$; 2) when $d_n < d_{\mu_g}$, d_{μ_g} should be divisible by l_n, hence $l_n \leq d_{\mu_g} \frac{1}{\lceil \frac{d_{\mu_g}}{d_n} \rceil}$. Therefore, the distance between group g and source n is the difference between the reciprocal of d_n and the largest average transmission interval of source n when n is put into group g. In addition, there may be more than one source with the same average transmission interval as AoI constraint in a group. For simplicity, the distance is calculated based on only one center. The simulation results in the next section validate this simplification.

To approach the minimum number of channels, we should put the sources into the group with the minimum distance. However, such a procedure may lead to inefficient use of channels. In Example 1, if the centers of group 1 and 2 are A and B. Then $\{E, F, G, H, I, J\}$ are put into group 1 while C, D are packed into group 2. It is frustrating that such an operation requires 3 channels. If we rearrange source I, J and put them in group 2, then we only need 2 channels.

To illustrate this phenomenon, we define the utilization of an average transmission interval vector $\boldsymbol{l} = [l_1, \cdots, l_N]$

Algorithm 4 Heuristic Grouping Algorithm (HGA)

Input: The set of sources N and the AoI constraints d;
Output: A cyclic scheduler π_G;

1: Obtain $\boldsymbol{u} = \{u_1, \cdots, u_v\}$ including all the distinct values in $\boldsymbol{d}$. For sources in $\boldsymbol{N}$, solve problem (5) for average transmission intervals $\{l_1, \cdots, l_N\}$ and $K_1 = \lceil \sum_{i \in \boldsymbol{N}} \frac{1}{l_i} \rceil$.
2: **for** $i \in [2, K_1 - 1]$ **do**
3:     The optional set of centers: $\boldsymbol{u}_{ce} = \{\boldsymbol{z} | \boldsymbol{z} \subset \boldsymbol{u}, |\boldsymbol{z}| = i\}$
4:     **for** each optional set of centers **do**
5:         Put sources in the group with the minimum distance. $\boldsymbol{G}$ denotes the groups with unused part larger than γ.
6:         **for** each group j in $\boldsymbol{G}$ **do**
7:             $\boldsymbol{r} = \{D_{j,n} + \frac{1}{d_n} | n \in \boldsymbol{g}_j\}$, where $\boldsymbol{g}_j$ includes all the sources in j-th group and has $|\boldsymbol{g}_j|$ elements.
8:             Arrange $\boldsymbol{r}$ in descending order. Find the maximum m such that $\sum_{n=1}^{m} r_n \leq \lfloor \sum_{n=1}^{|\boldsymbol{g}_j|} r_n \rfloor$.
9:             Put the sources corresponding to $\{r_{m+1}, \cdots, r_{|\boldsymbol{g}_j|}\}$ to other group with the minimum distance and enough unused part. If we can not find such groups, put the sources into the first group.
10:         **end for**
11:         Derive the minimum number of channels for each group by solving problem (5). K_G denotes the number of channels required by all the groups.
12:         **if** $K_G = \lceil \sum_{n=1}^{N} \frac{1}{d_n} \rceil$ **then**
13:             Construct π_G for the current grouping scheme by CS. Terminate the algorithm.
14:         **end if**
15:     **end for**
16: **end for**
17: Obtain the optimal grouping method over all the optional number of groups and choices of centers. Construct π_G by CS.

Algorithm 5 Two-step Grouping Algorithm (TGA)

Input: The set of sources N and the AoI constraints d;
Output: A cyclic scheduler π_T;

1: Search harmonic sources by HSI, construct π_H
2: For the other sources in N, design π_G by HGA
3: π_T is the combination of π_H and π_G. The number of channels required by π_T is the sum of numbers of channels required by π_H and π_G.

---

group have distinct AoI constraints. If there are two centers μ_g and $\mu_{g'}$ with the same AoI constraint, then for any source n, the two groups are the same because distance $D_{g,n} = D_{g',n}$. Therefore, if we divide the sources into i groups, there are $\binom{v}{i}$ ways to choose centers, where v is the number of distinct values in $\boldsymbol{d}$. For each optional combination of centers, we put the sources into the group with the minimum distance. After that, for each group with large unused part of channels (unused part larger than γ), we find the sources leading to the inefficient utilization and put them into the other groups with the minimum distance and enough unused parts of channels. Here, 'enough' means that if source n is rearranged and put into group g', then $D_{g',n} + \frac{1}{d_n}$ is no larger than the unused part of g'. When the grouping method for a certain combination of centers is derived, we obtain the number of channels by solving (5) for each group. If the number of channels required by the current grouping scheme equals the lower bound of the problem, we stop searching for a better solution and terminate the algorithm. Otherwise, we will construct a grouping scheme for each optional number of groups and combination of centers to derive the optimal grouping method. Finally, we construct scheduler π_G by using CS for each group.

C. Complexity

In HSI, we first search the sources with harmonic AoI constraints with a complexity of $O(N^3)$. Then we construct a scheduler for them with a complexity of $O(C_H N)$, C_H is the cycle length of π_H constructed by HSI. In HGA, we use a fast algorithm proposed in [4] to solve problem (5) and the complexity is $O(N^4 (d_N)^2)$. Since we have at most 2^v possible combinations of centers, the complexity of HGA is $O(2^v N^4 (d_N)^2)$, where v is the number of distinct values in $\boldsymbol{d}$. In conclusion, TGA has a complexity of $O(2^v N^4 (d_N)^2)$.

D. Combination of HSI and HGA

As mentioned earlier, TGA is a combination of HSI and HGA, shown in Algorithm 5. Although HGA can design a scheduler for any AoI constraints, HSI plays a key role in TGA. Firstly, the number of sources needed to be handled by HGA is decreased after HSI. Secondly, since the optional number of groups is upper bounded by $K_1 - 1$, HGA can reduce K_1 if the sources founded by HSI, denoted by $\boldsymbol{N}'$, is not an emptyset. Therefore HSI can efficiently narrow down the search space of HGA. Thirdly, since HSI has strong theoretical guarantee, taking HSI as the first step of TGA can achieve better performance compared with pure HGA. Denote

---

as $\sum_{n=1}^{N} \frac{1}{l_n}$. The number of channels assigned to $\boldsymbol{l}$ is $\lceil \sum_{n=1}^{N} \frac{1}{l_n} \rceil$, when $\boldsymbol{l}$ is consecutively divisible. Then we use $\lceil \sum_{n=1}^{N} \frac{1}{l_n} \rceil - \sum_{n=1}^{N} \frac{1}{l_n}$ to approximate the unused part of the channels. A large unused part indicates the inefficient utilization of channels. When n is packed into group g, l_n can be approximated by reciprocal of $D_{g,n} + \frac{1}{d_n}$. Therefore, if group 1 contains $\{A, E, F, G, H, I, J\}$, then the approximated average transmission intervals are $[3, 6, 6, 6, 6, 6, 6]$ and the unused part of the channel is $2 - \frac{1}{3} - \frac{6}{6} = \frac{2}{3}$. For group 2 with $\{B, C, D\}$, the unused part of the channel is $1 - \frac{3}{5} = \frac{2}{5}$. Both two groups inefficiently use the channels. If group 1 and 2 contain $\{A, E, F, G, H\}$ and $\{B, C, D, I, J\}$, the approximated average transmission interval vectors are $[3, 6, 6, 6, 6]$ and $[5, 5, 5, 5, 5]$ while the unused part of channels are zero for both groups. Therefore, after putting the sources to the group with the minimum distance, we should rearrange the sources which lead to inefficient use of the channels.

Finally, we propose HGA in Algorithm 4. We first derive K_1 by solving problem (5), hence the optional number of groups is $\{2, \cdots, K_1 - 1\}$. Secondly, we assume the centers of each

the other sources with generalized AoI constraints by $\bar{N}' = N \setminus N'$. Define $l^\star = [l_1^\star, \cdots, l_N^\star]$ and $l' = [l_1', \cdots, l_n', \cdots]$ as the optimal average transmission intervals of problem (5) for N sources and sources in $\bar{N}'$, respectively. Let K_{WA} and K_{OA} denote the number of channels achieved by TGA with and without HSI when we only allow the sources in $\bar{N}'$ to be packed into the same group, then

$$
\begin{aligned}
K_{OA} &= \left\lceil \sum_{n \in N} \frac{1}{l_n^\star} \right\rceil = \left\lceil \sum_{n \in N'} \frac{1}{l_n^\star} + \sum_{n \in \bar{N}'} \frac{1}{l_n^\star} \right\rceil \\
&\overset{(a)}{\geq} \left\lceil \sum_{n \in N'} \frac{1}{d_n} + \sum_{n \in \bar{N}'} \frac{1}{l_n^\star} \right\rceil \overset{(b)}{=} \left\lceil \sum_{n \in \bar{N}'} \frac{1}{l_n^\star} \right\rceil + \sum_{n \in N'} \frac{1}{d_n} \\
&\overset{(c)}{\geq} \left\lceil \sum_{n \in \bar{N}'} \frac{1}{l_n'} \right\rceil + \sum_{n \in N'} \frac{1}{d_n} = K_{WA}.
\end{aligned}
$$
(8)

Inequality (a) comes from the characteristics of average transmission interval. Equality (b) is based on the fact that the scheduler constructed by HSI is optimal and achieves full channel utilization for all the occupied channels, thus $\sum_{n \in N'} \frac{1}{d_n}$ is an integer. Since l' is the solution of problem (5) for sources in $\bar{N}'$, then for any $\bar{N}'$, we have $\lceil \sum_{n \in \bar{N}'} \frac{1}{l_n'} \rceil \leq \lceil \sum_{n \in \bar{N}'} \frac{1}{l_n^\star} \rceil$, then inequality (c) holds. If $\lceil \sum_{n \in \bar{N}'} \frac{1}{l_n'} \rceil < \lceil \sum_{n \in \bar{N}'} \frac{1}{l_n^\star} \rceil$, the introduction of HSI leads to better results when all the sources are packed into a single group. Since K_1 is the upper bound for the results of HGA, better performance can also be achieved by HSI when the sources in $\bar{N}'$ are divided into several groups.

Since HGA is a heuristic method for solving problem (6), we can not guarantee that the more sources scheduled by HSI, the better the performance achieved by TGA. Based on the simulations, we find a trick to improve the performance of the proposed method. If the distinct values of AoI constraints of sources in $\bar{N}'$ after the first part of HSI are the same as that after the second part, we then only schedule the sources found in the first part by HSI. For example, if AoI constraints for the sources in $\bar{N}'$ after the first part is $\bar{d}' = [6, 6, 6, 6, 6, 7, 7, 9, 9, 9, 9, 9, 9, 9]$. Then after the second part, AoI constraints are $[6, 7, 7, 9, 9, 9, 9]$. Such a procedure needs 3 channels for $\bar{d}'$. If we use HGA directly for $\bar{d}'$, we only require 2 channels by dividing them into two groups.

E. Remarks

Aion, proposed in [4], is a fast algorithm for solving problem (5) and constructing a corresponding scheduler. Compared with Aion, our TGA has the following main differences:

- Aion is optimal with only consecutively divisible AoI constraints while HSI (the first step of TGA) is optimal for sources with harmonic AoI constraints.
- Given AoI constraints d, Aion directly solves problem (5) for d assuming all the sources are packed into a same group, while HGA aims at solving our grouping problem (6). HGA divides the sources into different groups, each

group can possess different cyclic schedulers and cycle lengths. Since problem (5) is only a special case of (6), the number of channels achieved by Aion will be the upper bound of HGA.

Therefore, if HSI is excluded from TGA and N sources are only allowed to be packed into a same group, TGA will be reduced to Aion.

VI. SIMULATION RESULTS

In this section, we compare the performance of Aion and TGA. Besides, the number of channels required by consecutively divisible $l = [l_1, \cdots, l_N]$ is derived by $\left\lceil \sum_{n=1}^{N} \frac{1}{l_n} \right\rceil$, not by using CS. Fig. 2 shows the number of channels achieved by TGA ($\gamma = 0.5$), GD, and Aion. We also show the lower bound in Lemma 1. In Fig. 2, we vary the number of sources, namely N, from 10 to 300 with a step size of 10. For each N, the AoI constraints are generated by following a uniform distribution which takes value in interval $[2, 10]$. The number of required channels is averaged over 1000 instances.

The performance achieved by TGA is close to the lower bound. When N increases, the gap between Aion and the lower bound increases from 0.343 to 14.771 while the gap for TGA is no more than 0.124. This phenomenon illustrates that TGA can still ensure good performance when N is large. When $N = 300$, the average lower bound is 64.867 while the average numbers of channels required by Aion and TGA are 79.638 and 64.961, respectively. Therefore, with HSI and HGA, the average number of channels decreases by 18.43%. This is because Aion can only provide optimal schedulers for consecutively divisible AoI constraints. For the generalized AoI constraints, Aion has to find the optimal consecutively divisible average transmission intervals. Such an operation results in a high update frequency for most sources and requires more channels. Our TGA can not only provide optimal schedulers for harmonic AoI constraints by HSI but also decrease the gap between the average transmission intervals and the AoI constraints of the sources by HGA.

In addition, we find that the number of channels achieved by Aion exceeds GD when the number of sources is larger than 70. Recall that GD assigns independent channels to the sources with distinct AoI constraints, hence GD allows generalized average transmission interval vectors and achieves the optimal performance for the sources which occupy exactly an integer number of channels. Therefore, when N increases, GD can find more sources that can occupy an integer number of channels and achieve better performance than Aion.

Fig. 3 shows the comparison among the three methods mentioned above when the number of distinct values of AoI constraints increases. In Fig. 3, 1000 instances of AoI constraint vectors are generated by following a uniform distribution $[2, 20]$. When N increases, the gap between Aion and the lower bound increases from 0.241 to 11.146 while the gap for TGA is no more than 0.225. When $N = 300$, the average lower bound is 41.439 while the average numbers of channels required by Aion and TGA are 52.585 and 41.615. The gap between TGA and the lower bound is no more than 0.42%.
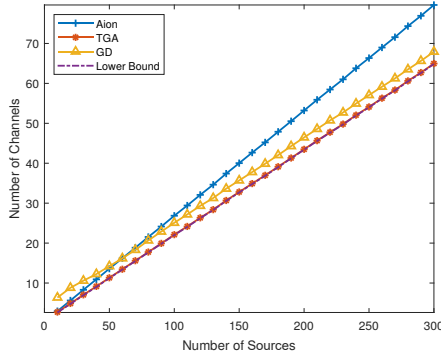
Fig. 2. Performance of Aion, GD and TGA over 1000 instances, where AoI constraints are generated by uniform distribution $[2, 10]$.
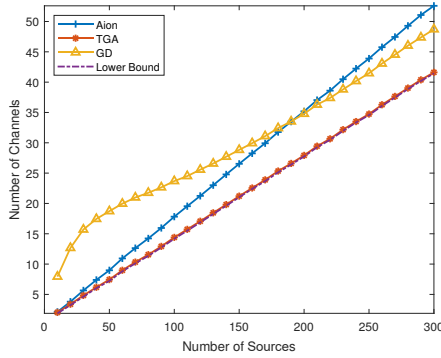

Fig. 3. Performance of Aion, GD and TGA over 1000 instances, where AoI constraints are generated by uniform distribution $[2, 20]$.

Besides, with HSI and HGA, the average number of channels decreases 20.86% compared with Aion. Comparing with the results in Fig. 2, the gap between Aion and TGA increases when the AoI constraints are more distinct. This is because when the number of distinct values increases, Aion may lead to more sources with average transmission intervals smaller than the AoI constraints and thus require more channels. The introduction of grouping can solve this problem by increasing the solution space of the average transmission interval vectors.

Secondly, the number of sources with the performance of GD lower than Aion increases from 70 to 210 when the maximum AoI constraint increases from 10 to 20. Given N, then the number of sources with the same AoI constraint decreases when the number of distinct values of AoI constraints increases. In this case, GD will waste more channel resources and achieve worse performance. Therefore, when the number of distinct values increases, the performance of GD is better than that of Aion only when N is large.

VII. CONCLUSION

A scheduling problem for minimizing the number of channels with AoI guarantee is studied in this paper. To split the large-scale scheduling problem into multiple small-scale problems, we propose a grouping method which divides the sources into different groups and designs schedulers for each group independently. Therefore, the scheduling problem is then transformed into finding the optimal grouping scheme. A novel two-step grouping algorithm (TGA) is proposed. In the first step, we identify the sources with harmonic AoI constraints and design an optimal scheduler for them. Then for the other sources with generalized AoI constraints, we propose a heuristic grouping algorithm to pack the sources which can be scheduled together with minimum update rates into a same group. In the simulations, TGA shows only 0.42% gap compared with the lower bound of the problem when the number of sources is large.

VIII. ACKNOWLEDGMENT

REFERENCES

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

[2] M. K. Abdel-Aziz, S. Samarakoon, C.-F. Liu, M. Bennis, and W. Saad, "Optimized age of information tail for ultra-reliable low-latency communications in vehicular networks," *IEEE Transactions on Communications*, vol. 68, no. 3, pp. 1911–1924, 2019.

[3] C. Li, Q. Liu, S. Li, Y. Chen, Y. T. Hou, W. Lou, and S. Kompella, "Scheduling with age of information guarantee," *IEEE/ACM Transactions on Networking*, 2022.

[4] Q. Liu, C. Li, Y. T. Hou, W. Lou, and S. Kompella, "Aion: A bandwidth optimized scheduler with aoi guarantee," in *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*. IEEE, 2021, pp. 1–10.

[5] H. Tang, J. Wang, L. Song, and J. Song, "Minimizing age of information with power constraints: Multi-user opportunistic scheduling in multi-state time-varying channels," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 5, pp. 854–868, 2020.

[6] J. Sun, Y. Sun, S. Zhou, and Z. Niu, "Status update for accurate remote estimation: Centralized and decentralized schemes," *IEICE Transactions on Communications*, vol. 105, no. 2, pp. 131–139, 2022.

[7] J. Sun, L. Wang, Z. Jiang, S. Zhou, and Z. Niu, "Age-optimal scheduling for heterogeneous traffic with timely throughput constraints," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 5, pp. 1485–1498, 2021.

[8] R. Devassy, G. Durisi, G. C. Ferrante, O. Simeone, and E. Uysal-Biyikoglu, "Delay and peak-age violation probability in short-packet transmissions," in *2018 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2018, pp. 2471–2475.

[9] I. Kadota, A. Sinha, E. Uysal-Biyikoglu, R. Singh, and E. Modiano, "Scheduling policies for minimizing age of information in broadcast wireless networks," *IEEE/ACM Transactions on Networking*, vol. 26, no. 6, pp. 2637–2650, 2018.

[10] C. Li, Q. Liu, S. Li, Y. Chen, Y. T. Hou, and W. Lou, "On scheduling with aoi violation tolerance," in *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*. IEEE, 2021, pp. 1–9.

[11] A. Maatouk, Y. Sun, A. Ephremides, and M. Assaad, "Timely updates with priorities: Lexicographic age optimality," *IEEE Transactions on Communications*, 2022.

[12] F. Peng, Z. Jiang, S. Zhang, and S. Xu, "Age of information optimized mac in v2x sidelink via piggyback-based collaboration," *IEEE Transactions on Wireless Communications*, vol. 20, no. 1, pp. 607–622, 2020.

[13] J. Song, D. Gunduz, and W. Choi, "Optimal scheduling policy for minimizing age of information with a relay," *arXiv preprint arXiv:2009.02716*, 2020.

[14] Y. Sun and B. Cyr, "Information aging through queues: A mutual information perspective," in *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*. IEEE, 2018, pp. 1–5.

[15] L. Wang, J. Sun, Y. Sun, S. Zhou, and Z. Niu, "A uoi-optimal policy for timely status updates with resource constraint," *Entropy*, vol. 23, no. 8, p. 1084, 2021.