Received April 26, 2019, accepted May 16, 2019, date of publication May 27, 2019, date of current version August 2, 2019.
Digital Object Identifier 10.1109/ACCESS.2019.2919320

Scheduling Status Update for Optimizing Age of Information in the Context of Industrial Cyber-Physical System

DEVARPITA SINHA, (Graduate Member, IEEE), AND RAJARSHI ROY, (Senior Member, IEEE)
Department of Electronics and Electrical Communication Engineering, Indian Institute of Technology Kharagpur, Kharagpur 721302, India
Corresponding author: Devarpita Sinha (devarpita.sinha@gmail.com)

ABSTRACT Age of Information has been emerged as an interesting metric in real-time wireless networks that captures the freshness of information in the underlying applications. This topic is motivated by the problem in which the users of a network care about timely information defined as the age of the most recent status update a user has received. In the proposed work, we have studied this concept in the context of an industrial wireless sensor-actuator network for cyber-physical production systems. Such a network with ever-changing dynamics requires continuous updates of the system states and ongoing tasks by exchanging the time-critical, event-driven, and/or time-driven information to maintain the stability of the system, failing which may lead to shut down of the plant and other fatal consequences. Different real-time transmission scheduling algorithms manage how the channel resources are allocated each time depending on the packet arrivals to minimize the age of the information. However, unlike other real-time networks like broadcasting and sensor networks, etc. in cyber-physical systems, cyber and physical devices have different requirements to improve their quality of performance. Balancing between the performance criteria of both cyber and physical units seems to be a great challenge. In this work, the minimization of staleness of the real-time updates by minimizing the age of information and their effect on network performance have been studied extensively for this purpose. Two greedy scheduling policies have been proposed: one for the total age minimization and another one for stale age and jitter minimization. Their performances and complexities are compared with other existing scheduling algorithms. Moreover, the optimality of each of our proposed algorithms are proved analytically and claims are validated via simulation results too. Eventually, these results come to a conclusion that minimum age does not always guarantee the maximum freshness of information and satisfactory system performance at the same time.

INDEX TERMS Age of information, industrial wireless sensor actuator network, industrial cyber-physical system, greedy scheduling, latency, stale age, jitter.

I. INTRODUCTION

In this fast-growing information age [1] numerous time-varying applications require the transmission of information about the state of a process of interest between certain source-destination pairs or nodes through wired or wireless communication networks. Age of Information (AoI) [1] has been receiving vivid attention as a quality metric in real-time status update, particularly for applications which require the time-critical information (such as, breaking news and weather forecast, sensor data for status update in cyber-physical systems or internet of things, health monitoring, energy utilization information of smart grid, smart home or factory, position-velocity-acceleration of a car in smart transportation system), command, control instructions and feedback messages to stay updated about the current status of a real-time system and initiate control action in a timely manner.

AoI is an interesting performance metric to measure the freshness of the information a system has about a process observed remotely in terms of the time interval elapsed since the generation of the packet that was most recently delivered to the destination (ex. monitor, controller, actuator, etc.) [2]. The source generates time-stamped messages that are transmitted in packet forms containing updated information about one or more variables of interest of the source and the time of generation of the sample [3]. If the most recent

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

packet received by the destination carries the time stamp $u(t)$ which characterizes the instant of generation of the packet, then the age at any instant t is defined as the random process $\Delta(t) \triangleq t - u(t)$ [1], [3].

AoI optimization to ensure high freshness of information is not the same as delay minimization or throughput maximization. Simple packet delay of an individual packet measures the time difference between the generation of a packet and its delivery to its receiver(s). Throughput maximization can be achieved by making the sources send packets as fast as possible even at an interval of long period with no delivery. However, this may lead to the destination receiving delayed statuses with higher AoI because these status messages overload the communication network which may eventually cause communication delays and congestions due to limited network resources. Alternatively, delay suffered by the stream of the status packets can be reduced by reducing the rate of update generation. However, this may again lead to the destination having extremely outdated, less valuable or invalid status information and larger age because of infrequent status packets arrivals which in turn degrades system performance. Therefore, the two parameters packet delay and inter-delivery time that influence AoI should be balanced appropriately to receive status updates as soon as it is generated by the source at an optimized updating rate by efficient usage of the available system resources. Controlling only one is insufficient for achieving good AoI performance. A small average age value of information is achieved as long as packets with minimum delay are delivered regularly [2].

Cyber-Physical System (CPS) is large scale, interconnected, and integrated by the cyber world through computation, communication, and components interacting with the physical environment [4]. Here physical processes affect computations and vice versa by transmitting and receiving data via feedback loops as depicted in Fig. 1. CPSs have applications in a wide range of areas and disciplines including medical and health, energy consumption, and power supply, transportation and automobiles, agriculture, industrial automation and manufacturing and many more. The application of CPS to industrial production systems is termed as Cyber-Physical Production Systems (CPPS) [5] which leads to the smart factory.

In CPPS, the typical manufacturing process is merged with intelligent machines, information storage space and production facilities that are able to exchange information autonomously, act accordingly, and control each other independently. AoI plays a major role in industrial CPSs (ICPSs) as the basis of information freshness and timeliness of information exchange not only to perform seamless computation, control, and communication (3Cs) but to provide autonomous, highly flexible, self-organized and reconfigurable manufacturing system too which appears to be humanoid or smart [6].

Wireless sensor-actuator networks (WSANs), as shown in Fig. 2, is a networked smart system consists of a large number of wirelessly interlinked, distributed, battery-powered

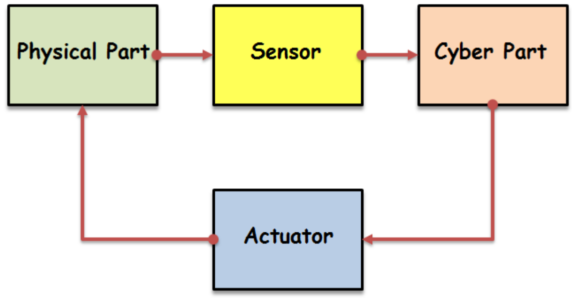


FIGURE 1. CPS as a closed loop system.

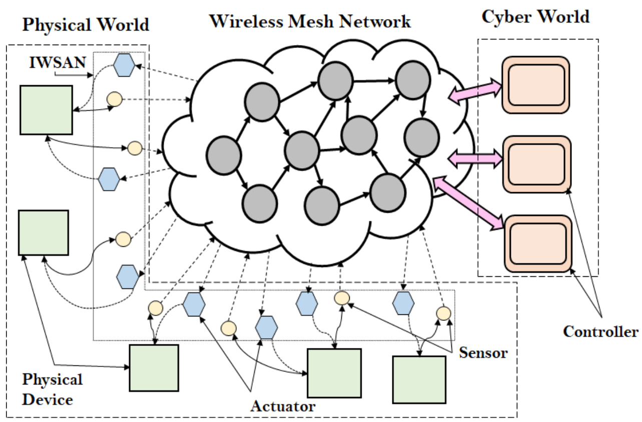


FIGURE 2. WSAN for distributed control of CPPS.

sensors and actuator nodes conjoined with intelligent processors that are principally responsible for monitoring and managing the overall network through communication with sensors and actuators. WSANs are gaining rapid adoption in monitoring and process control applications in ICPSs due to their advantages of enhanced flexibility, scalability, mobility, improved reliability, interoperability, lower deployment effort and maintenance in harsh industrial environments.

Sensors gather information about the state evolution of a process in the plant in a time-triggered or event-triggered manner and the updated status is sent to the controller via single hop or multi-hop communication over the wireless mesh network. Upon receipt of the required data, the controller analyzes and processes the same and makes the necessary decisions on how to react to this information. After that, it sends the appropriate control commands to the actuators in order to control the changed behavior of the physical processes based on the current state of the process. Hence, a closed loop is formed by enabling the application systems to sense, interact, and control the physical behavior of the system by integrating the cyber and physical worlds [7]. Usually, different sections of a process are distributed over several, sometimes redundant, controllers to avoid that a failure in one part of the process affecting other sections of the process. Due to limited availability of bandwidth, energy and other resources, multi-hop communication delays over mesh networks, dynamically changing network topology and channel conditions caused by noise, multipath fading, external interference, moving obstacles and weather conditions it is very challenging to meet the stringent AoI constraint to attain reliability and real-time communication guarantees while supporting uninterrupted execution of physical processes performed by the actuators to avoid production loss, plant shutdown and/or accidents. Some well-established standards for IWSAN are WirelessHART, ISA 100.11a, etc. based on IEEE 802.15.4 [8]. Their MAC layer adopts time division multiple access (TDMA) to avoid potential collision between simultaneous transmissions in the same channel and channel hopping to control access to network resources on the top of IEEE 802.15.4 physical layer [9]. Therefore, the demand for more research on real-time scheduling for resource-constrained WSANs in ICPSs to optimize AoI and network performance is of high priority.

A. CONTRIBUTION AND OUTLINE

In this paper, we have considered a set of sensors, with limited battery life, for monitoring a set of stochastic processes in a smart factory. These sensors are being scheduled to share a single unreliable channel on TDMA basis to transmit their sensed samples to the controller for further processing. Next, the controller sends the necessary control command to the corresponding actuator to control the underlying process. When this control action overs, the changed status of the process is again sensed and the above cycle is followed repeatedly until the maximum iteration is reached. IWSAN model proposed in this paper can be mapped with practical CPPS. Under WITTENSTEIN Bastian's "Future Urban Production" facility in Fellbach, Germany, "Milk Run 4.0" is a prototype of a smart factory that supports demand-driven milk run facility instead of routine service to reduce unnecessary workload [10]. There may be different tasks like filling milk in empty bottles, sealing, labeling, packaging, etc. needed to be done before the bottles are out for delivery. Let sensor i monitors the arrival process of empty bottles through a conveyor belt. It then sends information (say, no. of bottles, size of bottle, etc.) at the scheduled time slot to the controller that sends useful control information to the related actuator. The actuator now controls the milk flow from the milk storage machine and the conveyor moves the filled bottle forward in the production line for other operations to be done sequentially. This action takes a certain time and after it is over the changed state of the process (if any new bottle has come) is again sensed by the sensor i for a preassigned maximum time or the maximum number of iterations. During the scheduling execution and decision distribution period, the network manager assumes that the environment remains static and does not change with time. Therefore, an off-line scheduling scheme is more suitable for this case. This scheduling decision is based on the dynamic priority of a particular sensor sample to fulfill the goal of minimizing the long-term average age of information and improved stability of the network. Here AoI is quantified in terms of the time elapsed since the generation of the last successfully processed sensor sample. Minimizing AoI is equivalent to minimizing the staleness or maximizing the value of information. In this paper terms like sensor sample, sample packet and status update have been used interchangeably. Other than AoI, jitter is another important factor that affects the control quality of a network even results in instability of the whole system. Jitter is related to end to end communication delay which again varies with latency originated from the waiting period before the sensor sample is processed. Therefore, next we provide a deep insight into finding an energy efficient sensor scheduling scheme that will help in minimizing the jitter in addition to AoI or staleness of information. As per our knowledge, this is the first work that tries to implement age metric for control of ISWAN in CPPS.

The rest of the paper is organized as follows. Sec II discusses the related literature review. Sec III describes the system model and derives mathematical formulations related to it. In sec IV, two energy efficient greedy sensor scheduling algorithms are proposed and their optimality is proved analytically in terms of age, latency, and jitter. Results and discussions are given in sec V followed by the conclusion of this paper in sec VI.

II. LITERATURE REVIEW

Many research works have been carried out on minimizing the age metric by modelling the status update of a system as different types of queues, with various arrival/departure processes, number of servers, and queue capacities. Work in [1] tries to find suitable update arrival rate and maximum server utilization for single source single server first come first served (FCFS) M/M/1, M/D/1 and D/M/1 system to achieve minimum age. It is found that deterministic arrival or departure processes achieve a lower average age than memory less Markov processes. Paper [11] computes the age for last come first served (LCFS) M/M/1 queue with and without preemption and finds that LCFS provides equivalently good or even better age performance than that of FCFS. Paper [12] analyzes age for a multiple source-single server FCFS M/M/1 queuing system. Paper [13] deals with random status update generation and their routing through multi-path network resulting in out of order packet delivery with varying packet delivery times that degrades resource utilization and hampers age metric with outdated packets. In this work, it can be observed that the age decreases with the increasing number of servers and service rate, as expected. But, increasing the utilization makes the computation complicated and comes at a cost of increased resource wastage spent on packets that are rendered obsolete. Paper [14] proposes a multi-server queue which balances the out-of-order reception effect for transmission diversity over multiple paths with the in-order queuing. Paper [3], [15] model the source-destination link as a single-server queue with random status updates with different packet management policies at the source node and investigates the average age and peak age for each of these policies. Paper [16] combines the multiple sources for status update model with packet management. Paper [17] tries to optimize the peak age of information (PAoI) for multiclass heterogeneous M/G/1 and M/G/1/1 queues. Paper [18], [19] propose "generate at will" model for status updates. Here the source node (e.g. a sensor) has access to the channel's idle/busy state through acknowledgments (ACKs) and it generates update at its own will only when the channel is free. This completely eliminates the waiting time in the queue and also saves energy consumption. As discussed in the previous section there should be a trade-off between status update frequency and queuing delay to optimize AoI. The results obtained from these paper validates this concept. It shows the zero-wait policy achieves the maximum throughput and the minimum delay but surprisingly, this is not always age optimal. Paper [20] shows the influence of error on peak age for M/M/1 queue. Paper [21] uses scheduled access and slotted ALOHA-like random access approaches for status updates to optimize AoI at the sink. The result reveals that ALOHA-like access, though simple, worsen AoI significantly in comparison to the scheduled access. Paper [22] deals with AoI minimization for information updates through multi-hop networks. In current time more research [2], [23]–[29] is going on investigating scheduling policies to minimize AoI. Paper [23] considers to minimize AoI using age optimal threshold-type status update scheduling policy where the transmission threshold is a function of the energy state of energy harvesting sensor. Paper [24]–[26] resolve minimum age scheduling problem (MASP) for a number of mutually interfering links that share a common channel for transmission. Paper [2], [27]–[30] work on minimizing AoI in the wireless broadcast network. Reference [2] considers four low complexity scheduling policies for periodic status updates transmitted through unreliable channels and compares their routine performance. The result shows that for the symmetric network Greedy, Max-Weight, and Whittle's Index are age optimal whereas for asymmetric network Max-Weight and Whittle's Index policies perform well. References [27] and [28] applies offline and online dynamic scheduling algorithms based on Markov decision processes (MDPs) and reinforcement learning to compare age for both buffered and buffer-less network with random updates. Paper [29] minimizes the long-run average age of the network by applying a transmission scheduling for stochastic arrivals based on Whittle's indexing framework for restless bandits. On the other hand [30] deals with data sampling and link scheduling jointly for minimizing age in multi-flow, multi-hop CPSs. [31]–[33] focus on investigating packet scheduling policies to maximize the freshness of information in CPSs.

This work partially refers to the work used in [2] but unlike [2] it is a one to one communication network with random status update generation. Our system also involves actuators in addition to wireless sensors while making the scheduling policy and therefore considers the total closed loop network control in optimizing AoI.

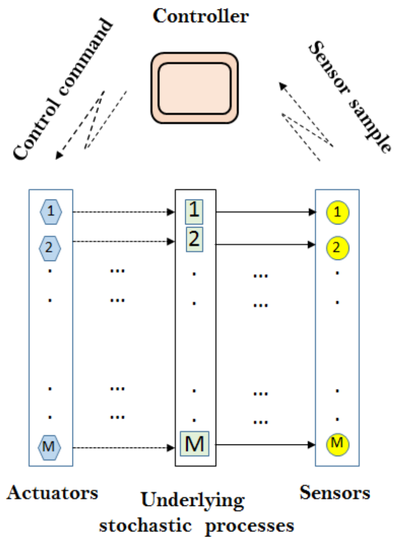


FIGURE 3. System model with M sensors, M actuators, and one controller.

III. SYSTEM MODEL

Consider a symmetric industrial WSAN (IWSAN) with M sensors and M corresponding actuators connected through a centralized controller as shown in Fig. 3. Sensors are monitoring M underlying stochastic processes and collecting samples to provide information to the controller about the current status of the system state. For this, these sensor samples are transmitted through a single hop unreliable channel with *ON* probability $p_i = p \in [0, 1]$. This channel is shared on TDMA basis among M sensors. Each of the dedicated slots is represented by notation $t = 1, 2 \ldots . T$ where T is the maximum slot number. Slot t basically indicates the time span from t-1 to t. At any time slot channel state can be *ON* or *OFF* represented by,

$$\begin{aligned} ch &= 1\,(Channel\ ON)\,, w.p. = p \\ &= 0\,(Channel\ OFF)\,, w.p. = (1-p) \end{aligned} \tag{1}$$

At the beginning of each slot t, sensors that are active at that time try to transmit their sensed samples via that shared channel. The controller undergoes a predefined policy π that decides to make controller idle or schedule the processing of one unprocessed sample of an active sensor at each slot t. The controller sends a feedback signal to the sensor reliably and instantaneously after processing sample coming from it. AoI increases linearly in terms of the number of slots since the generation of the last successfully processed sample. If one sample is processed at any slot t, at the end of t AoI is updated to 1. If the sample from an active sensor is not scheduled or failed to be processed at a slot $t-1$ then at the end of the slot sensor discards old sample and senses the current status of the system again to participate in scheduling in the next slot t. Let, $d^{(t)}$ be the sensor whose sample packet has successfully processed in slot t. Then at the end of slot t or in other words at the beginning of slot $t+1$,

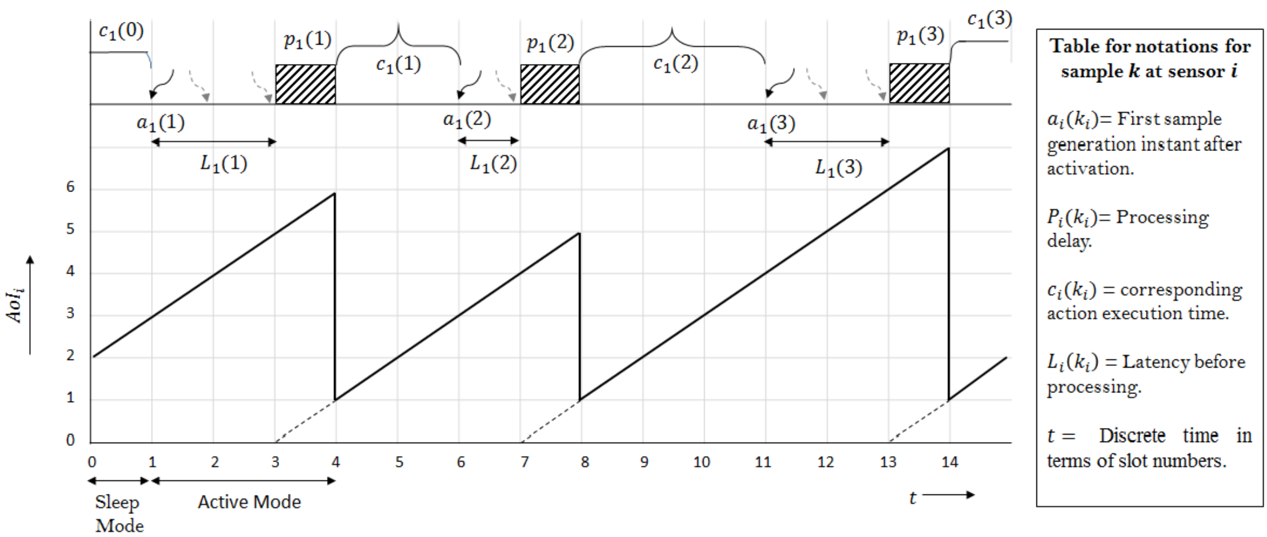


FIGURE 4. Age evolution for sensor i.

the age evolution will be as follows,

$$h_{t+1,i}(k_i) = h_{t,i}(k_i) + 1 \text{ } if \text{ } i \neq d^{(t)}$$
$$h_{t+1,i}(k_i+1) = 1 \text{ } if \text{ } i = d^{(t)} \tag{2}$$

Here $h_{t,i}(k_i) \geq 0$ is the AoI of the sample from sensor i at the beginning of slot $t.k_i$ is the index number of the sample packet from sensor i that needs to be processed at slot t i.e. $k_i - 1$ samples have already been processed from sensor i before slot t. Once k_i^{th} packet of sensor i is processed, packet index changes from k_i to $k_i + 1$. Once sample from sensor i is processed at slot t, the controller generates appropriate control command depending on the information content present in that sample and reliably sends the command packet to the corresponding actuator i in the same slot t to perform necessary actions to control the physical process or device. So, we can say that the scheduling and processing delay in the controller plus communication delay from the controller to actuator takes one slot time in total. After the control command based on the information extracted from sample k_i of sensor i reaches the actuator i, it takes $c_i(k_i) \geq 0$ slots time to make the physical process/device to complete the necessary task. It can be reasonably predicted that only after the completion of previously assigned task significant change in the system state occurs and sensor i again needs to sense this change. So after k_i^{th} sample from sensor i is processed, the sensor waits for $c_i(k_i) + 1$ slots time or age before sensing a new sample and participating in scheduling. Till AoI $c_i(k_i) + 1$ sensor i is said to be inactive or in sleep mode and does not collect any sample or participate in scheduling. Only for $AoI_i \geq c_i(k_i) + 1$, the sensor is active that generates sample in each slot with the replacement of the older one and participate in scheduling till its newly sensed sample is being processed. Being refrained from collecting unnecessary or redundant updates, sensors protect the network from congestion and conflict in using limited network resources (ex. transmission power, frequency, server, etc.). The sleep mode also helps in energy (battery power) conservation for sensors that enhance their lifetime. Excluding inactive sensors from scheduling simplifies the policy to take scheduling decision as comparatively fewer number of sensors compete for transmitting their sample at a particular slot. So it can be concluded from the discussion and definition of age that the age of one newly generated sample from a sensor depends on the action execution time corresponding to the previously processed sample from the same sensor, latency or waiting time of the current sample before its successful processing, and its processing delay.

A. MATHEMATICAL FORMULATIONS

Evolution of the age of sensor i in this proposed system model is illustrated in Fig. 4. The total AoI_i is the total area under the age curve of sensor i. This is obtained by summation of the area under the age curve for each slot t where $t = 1$ to T.

$$AoI_i = \sum_{t=1}^{T} AoI_{t,i}$$
$$= AoI_{1,i} + AoI_{2,i} \ldots\ldots\ldots + AoI_{T,i} \tag{3}$$

At each slot t, $AoI_{t,i}$ as shown in Fig 5, is a combination of a rectangle and an isosceles right angle triangle on the top of it.

$$AoI_{t,i} = \frac{1}{2} \times 1 \times 1 + h_{t,i}(k_i) \times 1 = \frac{1}{2} + h_{t,i}(k_i) \tag{4}$$

The average age of the system with M sensors is,

$$AoI = \frac{1}{TM} \sum_{t=1}^{T} \sum_{i=1}^{M} AoI_{t,i} \tag{5}$$

To find an appropriate scheduling policy to minimize the age of the proposed system the expected weighted sum age

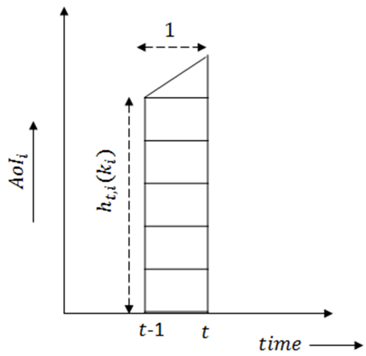


FIGURE 5. Age calculation during each slot *t*.

of information (EXWSAoI) will be considered as the finite horizon objective function as presented in (6).

$$EXWSAoI = \frac{1}{TM} E\left[\sum_{t=1}^{T} \sum_{i=1}^{M} \left\{\alpha_i AoI_{t,i} \,|\, \vec{I}\right\}\right] \quad (6)$$

Here $\vec{I}$ is the vector of initial conditions. $\vec{I} = [\overrightarrow{h_1(1)}, \overrightarrow{c(0)}] = [(h_{1,1}(1), c_1(0)), \ldots, (h_{1,M}(1), c_M(0))]$. $\overrightarrow{h_1(1)}$ and $\overrightarrow{c(0)}$ represent the initial values of $h_{t,i}(k_i)$ and $c_i(k_i - 1)$ respectively for $k_i = 1$. α_i is the positive real valued weight assigned to sensor i. As the network is symmetric, $\alpha_i = \alpha \forall i \in M$. So, replacing the value of $AoI_{t,i}$ in (6) and omitting the initial condition and packet index for notation simplicity we get,

$$\begin{aligned} EXWSAoI &= \frac{1}{TM} E\left[\sum_{t=1}^{T} \sum_{i=1}^{M} \alpha \left\{\frac{1}{2} + h_{t,i}\right\}\right] \\ &= \frac{\alpha}{2} + \frac{\alpha}{TM} E\left[\sum_{t=1}^{T} \sum_{i=1}^{M} \left\{h_{t,i}\right\}\right] \end{aligned} \quad (7)$$

As the constant $\frac{\alpha}{2}$ term in the previous equation does not affect the optimization problem, therefore neglecting the constant part slightly modified expression for (7) will be,

$$EXWSAoI = \frac{\alpha}{TM} E\left[\sum_{t=1}^{T} \sum_{i=1}^{M} \left\{h_{t,i}\right\}\right] \quad (8)$$

Now for any sensor $i \in M$, if sensor i be in sleep mode, it does not take part in scheduling. Only active sensors take part in scheduling. Therefore, it is sufficient to apply the scheduling policy on active sensors only to minimize EXWSAoI. Let, S_t = {Set of active sensors at the beginning of slot t} and $\hat{S}_t$ ={Set of sensors at sleep mode at the beginning of slot t}. For applying scheduling policy only on $i \in S_t$, modified objective function will become,

$$EXWSAoI = \frac{\alpha}{TM} E\left[\sum_{t=1}^{T} \sum_{i \in S_t} h_{t,i}\right] \quad (9)$$

Again, $AoI_{t,i}$ of a sample k from any active sensor $i \in S_t$ at any slot t can be decomposed into three parts viz. execution time $c_i(k_i - 1)$ corresponding to the sample $k_i - 1$ of the same sensor, latency or waiting time of the sample k_i before it is getting service at slot t be $L_{t,i}(k_i) \geq 0$ and processing delay of sample k_i, $P(k_i) \geq 0$. Latency and processing delay together results in the end to end communication delay.

$$h_{t,i}(k_i) = \underbrace{c_i(k_i - 1)}_{\text{Useful age}} + \underbrace{L_{t,i}(k_i) + P_i(k_i)}_{\text{Stale age}} \quad (10)$$

$c_i(k_i - 1)$ is unavoidable age and during this time a useful actuation task goes on. During this time the state change of the process is not taken into consideration and the sensor lies in sleep mode to save battery consumption. So $c_i(k_i - 1)$ amount of age is considered to be 'useful' content in $AoI_{t,i}(k_i)$ and it cannot be controlled externally. Remaining $\Delta_i = L_{t,i}(k_i) + P_i(k_i)$ amount of age is called 'stale age' as it is unnecessarily degrading the performance of the system by prolonging end to end communication delay. As discussed earlier in this system model, processing delay is one slot time that is always constant. Now our next objective is to minimize AoI by minimizing the stale age. Replacing $h_{t,i}(k_i)$ in (9),

$$EXWSAoI = \frac{\alpha}{TM} E\left[c_i(k_i - 1) + L_{t,i}(k_i) + P_i(k_i)\right] \quad (11)$$

Neglecting the constant part and the packet index for notation simplicity, we get expected weighted sum latency (EXWSL) that will be our next objective function.

$$EXWSL = \frac{\alpha}{TM} E\left[\sum_{t=1}^{T} \sum_{i \in S_t} L_{t,i}\right] \quad (12)$$

Finally, our aim is to find two suitable non-anticipative, work-conserving scheduling policies π_1 and π_2 among all the admissible policies Π that minimize the first and second objective functions respectively. Then their performances are compared and the effectiveness of each of them in order to optimize the performance of the overall network are investigated.

$$\text{Objective I: } O_1^{\pi^*} = \min_{\pi_1 \in \Pi} E\left[O_{1T}^{\pi_1}\right],$$

$$\text{Where, } O_{1T}^{\pi_1} = \frac{\alpha}{TM}\left[\sum_{t=1}^{T} \sum_{i \in S_t} h_{t,i}\right]$$

$$\text{Objective II: } O_2^{\pi^*} = \min_{\pi_2 \in \Pi} E\left[O_{2T}^{\pi_2}\right],$$

$$\text{Where, } O_{2T}^{\pi_2} = \frac{\alpha}{TM}\left[\sum_{t=1}^{T} \sum_{i \in S_t} L_{t,i}\right]$$

IV. OPTIMALITY OF THE SCHEDULING POLICIES

In this section first a greedy policy Highest Age First (HAF) is introduced that minimizes the AoI by satisfying objective-I of the finite-horizon scheduling problem as described in sec IV. Next in this section, we are going to propose another greedy scheduling policy Highest Latency First (HLF) that will minimize the stale age by minimizing the objective-II which in turn will sufficiently reduce the total age of the system.

Algorithm 1 Age Evolution Using HAF Algorithm

/* Initialization */
input $\overrightarrow{h_1(1)}$, $\overrightarrow{c(0)}$, T, M
$t \leftarrow 1$
$k_i \leftarrow 1$ for all sensors $i \in M$
while ($t \leq T$)
for all $i \in M$, do
/* Finding active-inactive sensors at slot t */
if $h_{t,i}(k_i) < c_i(k_i - 1) + 1$ then
/* Age evolution of inactive sensors at slot t */
$h_{t+1,i}(k_i) = h_{t,i}(k_i) + 1$
else
Store active sensors in a set S_t^{HAF}
$S_t^{HAF} \leftarrow i$
end
end
Only active sensors take part in the scheduling.
for all $i \in S_t^{HAF}$ do,
/* Scheduling decision in slot t */
HAF schedules the unprocessed active sensor sample with maximum AoI.
$d^{(t)} = \arg\max_{i \in S_t^{HAF}} \{h_{t,i}\}$
/* Post decision age evolution of active sensor at slot t */
Scheduled sensor sample is processed in slot t according to the channel state.
if $ch^{(t)} = 1 \& \, i = d^{(t)}$ then
$k_i \leftarrow k_i + 1$
$h_{t+1,i} = 1$
$c_i(k_i - 1) \leftarrow c_i(k_i)$
else
$h_{t+1,i}(k_i) = h_{t,i}(k_i) + 1$
end
end
$t \leftarrow t + 1$
end

A. GREEDY SCHEDULING POLICY HAF

Definition-I: Greedy policy HAF schedules in each slot t a transmission to the processor of unprocessed active sensor sample with the highest value of AoI ($h_{t,i}$) with ties being broken arbitrarily.

Age evolution using HAF policy is presented in Algorithm-1.

Next, a property of HAF and a lemma are given that will lead to the optimality proof of HAF in Theorem-I.

Property–I: If the winner of a particular slot fails to get the service, it may lose the game in the next slot.

It follows that if a sensor sample having the maximum age is scheduled at a particular slot but fails to be transmitted due to bad channel condition then the same may or may not be scheduled at the next slot. Transmission failure for bad channel condition and not getting chance to transmit again in the next slot seems to be like double punishment for that particular sensor and may lead to starvation.

Proof: Please see Appendix A.

Lemma-I: Greedy scheduling policy HAF attains the minimum age of an active sensor sample arranged in decreasing order of age at any time slot.

That means for any slot t sample from any active sensor i' in the active set $S_t^{'\pi_1}$ obtained by π_1 policy and arranged in decreasing manner of age has higher AoI than that of the sample from the same sensor index i' in the similarly arranged active set $S_t^{'HAF}$ obtained by policy HAF. It follows, $h_{t,i'}^{\pi_1} \geq h_{t,i'}^{HAF} \forall i' \in S_t' =$ Set of active sensors at slot t arranged in decreasing order of age.

Proof: Please see Appendix B.

Theorem-I: For symmetric IWSAN network with an unreliable time-shared channel from the sensor to processor, among the class of admissible policies, greedy scheduling policy HAF attains the minimum expected weighted sum age of information ($O_1{}_T^{\pi^*}$).

Proof: Please see Appendix C.

B. GREEDY SCHEDULING POLICY HLF

Definition-II: Greedy policy HLF schedules in each slot t a transmission to the processor of unprocessed active sensor sample with the highest value of latency ($L_{t,i}$) with ties being broken by scheduling sensors with the highest AoI ($h_{t,i}$) first.

Age evolution using HLF policy is presented in Algorithm-2.

Next, we will discuss two properties of HLF, one corollary and a lemma that will lead to the optimality proof of HLF in Theorem-II.

Property–II: Once the winner always wins the game until it gets the service.

It follows that if a sensor sample having the maximum latency is scheduled at a particular slot but fails to transmit due to bad channel condition then it will surely get a chance to transmit at the next slot as well. It means, scheduling decision switches only after successful sample packet transmission of the scheduled sensor.

Proof: Please see Appendix D.

Property-III: HLF schedules active sensor samples to be served in the first come first serve (FCFS) basis.

Proof: Please see Appendix E.

Corollary-I: HLF based queue attains the minimum average jitter.

Proof: Please see Appendix F.

Lemma-II: Greedy scheduling policy HLF attains the minimum latency of an active sensor sample arranged in decreasing order of latency at any time slot.

That means for any time slot t sample from any active sensor i' in the active set $S_t^{''\pi_2}$, obtained by π_2 policy and arranged in decreasing manner of latency, has higher latency than that of the sample from the same sensor index i' in the similarly arranged active set $S_t^{''HLF}$ obtained by policy HLF. It follows, $L_{t,i'}^{\pi_1} \geq L_{t,i'}^{HLF} \forall i' \in S_t' =$ Set of active sensors at slot t arranged in decreasing order of latency.

Proof: Please see Appendix G.

Algorithm 2 Age Evolution Using HLF Algorithm

/* Initialization */
input $\overrightarrow{h_1(1)}$, $\overrightarrow{c(0)}$, T, M
$t \leftarrow 1$
$k_i \leftarrow 1$ for all sensors $i \in M$
while ($t \leq T$)
  for all $i \in M$, do
  /* Finding active-inactive sensors at slot t */
    if $h_{t,i}(k_i) < c_i(k_i - 1) + 1$ then
    /* Age evolution of inactive sensors at slot t */
    $h_{t+1,i}(k_i) = h_{t,i}(k_i) + 1$
    else
    Store active sensors in a set S_t^{HLF}.
    $S_t^{HLF} \leftarrow i$
    end
  end
  Only active sensors take part in the scheduling
  for all $i \in S_t^{HLF}$ do
  /∗ Calculate latency for each active sensor at slot t */
  $L_{t,i}(k_i) = h_{t,i}(k_i) - c_i(k_i - 1) - 1$
  /* Scheduling decision in slot t */
  HLF schedules the unprocessed active sensor sample with maximum latency.
  $d^{(t)} = \arg\max_{i \in S_t^{HLF}} \{L_{t,i}(k_i)\}$
/∗ Post decision age evolution of active sensor at slot t */
Scheduled sensor sample is processed in slot t according to the channel state.
  if $ch^{(t)} = 1 \& i = d^{(t)}$ then
  $k_i \leftarrow k_i + 1$
  $h_{t+1,i} = 1$
  $c_i(k_i - 1) \leftarrow c_i(k_i)$
  else
  $h_{t+1,i}(k_i) = h_{t,i}(k_i) + 1$
  end
 end
$t \leftarrow t + 1$
end

Theorem-II: For symmetric IWSAN network with an unreliable time-shared channel from the sensor to processor among the class of admissible policies, greedy scheduling policy HLF attains the minimum expected weighted sum latency ($O_{2T}^{\pi^*}$).

Proof: Please see Appendix H.

V. RESULT AND DISCUSSION

In this section extensive simulations of the proposed scheduling algorithms HAF and HLF for symmetric IWSAN network with M sensors, M actuators and a controller/processor as described in section IV are performed and their results are compared with different existing algorithms viz. Work Conserving Round Robbin (WCRR) [34], Least Served First (LSF) [35], Power of Two Random Choice (PoTRC) [36] and simple Random Choice (RC) [36] to evaluate the performance of the system in terms of expected weighted sum age of information (EXWSAoI), expected weighted sum latency (EXWSL) and RMS jitter ($\overline{JT}$) (all in terms of the number of time slots t). WCRR schedules the active sensor that is immediately next to the sensor last successfully served. LSF schedules the active sensor from which least number of sample packets (k_i) have been served till the current slot considering higher age first as the tie-breaker condition. RC picks up any random active sensor and serves it whereas PoTRC chooses any two random sensors that are currently active and schedules the sensor with higher age between those two. LSF and our proposed algorithms HAF, HLF have time complexity of $O(n)$ whereas WCRR, PoTRC, and RC have the complexity of order $O(1)$. In terms of computation complexity too RC is the simplest algorithm. However, parameters used for all simulations are listed in the table-1:

TABLE 1. Simulation parameters.

Parameters	Symbol	Value Assigned
The weight assigned to each sensor	$\alpha_i = \alpha$ $\forall\, i \in M$	1
Transmission channel reliability	$p_i = p$ $\forall\, i \in M$	0.8
Time slot (as per WirelessHart standard [8])	t	10 ms
Initial Age [37]	$\overrightarrow{h_1(1)}$	[1,25]
Initial useful age	$\overrightarrow{c(0)}$	[1,24]

In Fig. 6. comparison of EXWSAoI obtained by proposed HAF, HLF and other popular algorithms (as mentioned earlier) with respect to the maximum number of time slots T and for different values of M has been shown. From this, we can see that EXWSAoI obtained by HAF is always minimum than that obtained by other algorithms for any value of M.

Fig. 7. plots the mean value of EXWSAoI averaged over $T = 10$ *to* 1000 with respect to different values of M. It shows that HAF provides the lower limit and RC provides the upper limit of mean EXWSAoI for this system model. Age calculated by other algorithms lie between these two. For a very small value of M ($M < 10$), mean EXWSAoI calculated by HLF, WCRR and LSF are very close to the lower limit. As the number of sensors are increasing this difference is reducing effectively but as the number of sensors are increasing EXWSAoI is also increasing and that may lead to outdated data transmission. On the other hand, EXWSAoI for PoTRC and RC are always substantially higher than the lower limit. This difference is minimum for the lower value of M (for $M < 20$) as shown in Fig. 7. Again, one important feature for IWSAN is scalability and it is expected to support as many numbers of nodes as possible. However, to meet the required refresh rate of few milliseconds [9] number of nodes should be chosen very carefully.

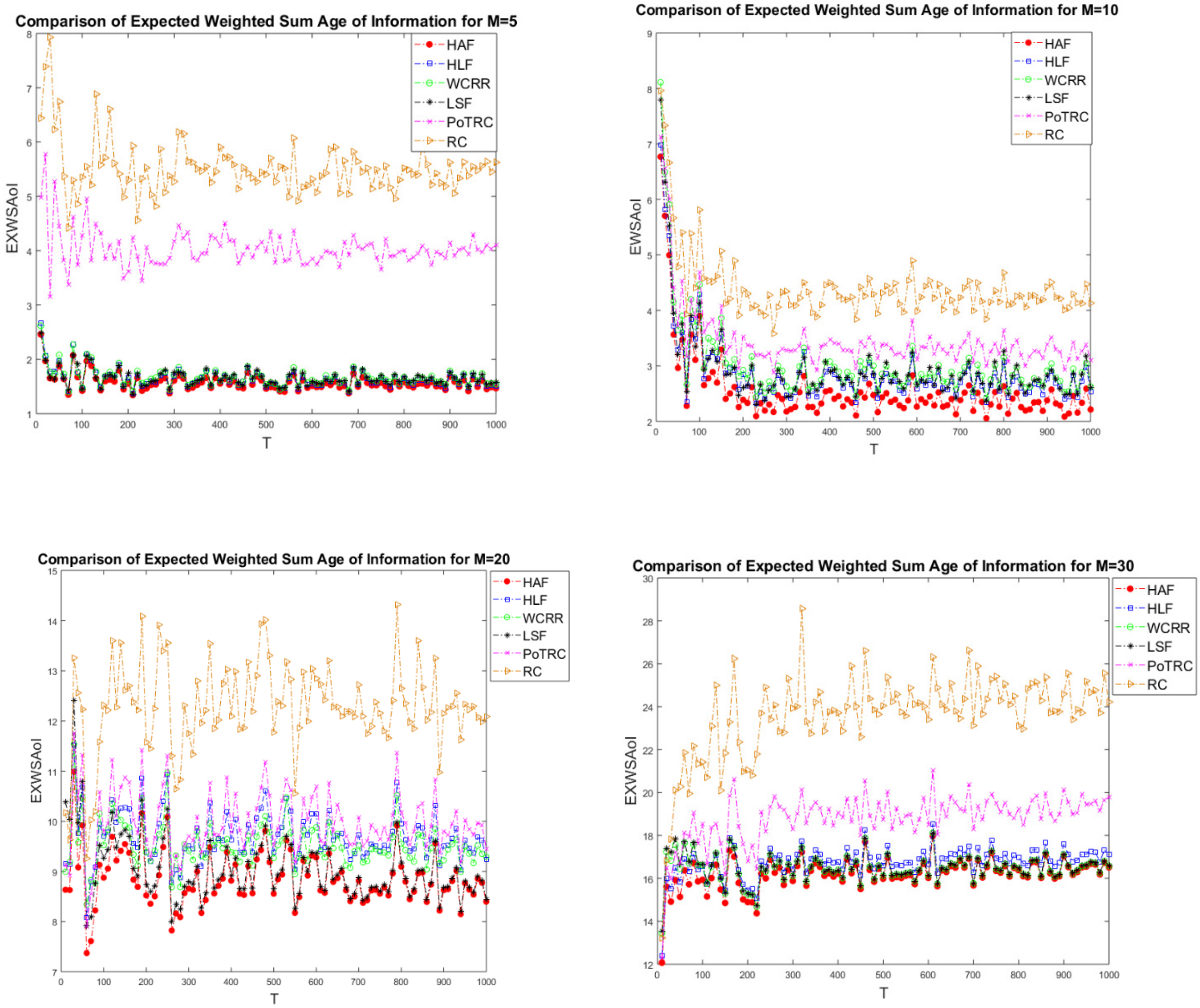


FIGURE 6. Comparison of EXWSAoI using various algorithms for different M.

In Fig. 8. comparison of EXWSAoI and EXWSL with respect to T is plotted for two proposed and other above-mentioned traditional scheduling algorithms considering $M = 16$ [8]. From this plot, it can be observed that though EXWSAoI obtained by HAF is minimum, EXWSL is always minimum for HLF. This minimum value of EXWSL obtained by HLF is always considerably lower than the same of HAF though EXWSAoI obtained by HLF is slightly higher than that of HAF. That means minimizing latency to minimize the stale age part from the total age provides a comparable result with allover minimization of total age as a whole.

Fig. 9. compares the performance of different scheduling algorithms in terms of mean EXWSAoI and mean EXWSL for this proposed system. As discussed earlier, HAF gives the best result and RC gives the worst result in terms of mean EXWSAoI. Among other scheduling algorithms, LSF provides marginally higher result than that of HAF and mean EXWSAoI for PoRTC, HLF, and WCRR are very close to each other but substantially lower than the worst case value.

On the contrary, minimum mean EXWSL is obtained by HLF while the maximum of that is provided by RC only similar to mean EXWSAoI case. Mean EXWSL for WCRR is close to HLF but HAF, PoTRC and LSF provides EXWSL comparably higher than the lower limit obtained by HLF.

Fig 10 compares the performance of different scheduling algorithms in terms of mean RMS jitter with respect to T for $M = 16$. It is found that the lowest jitter value is produced by HLF and the highest jitter is obtained by RC. Next to HLF, WCRR provides sufficiently good RMS jitter. However, other scheduling policies HAF, PoTRC, and LSF produce jitters that are close to each other but sufficiently higher than that of HLF.

From the earlier discussion of this paper, it is clearly understood that AoI should be minimized to maintain the information freshness for accurate decision making in the

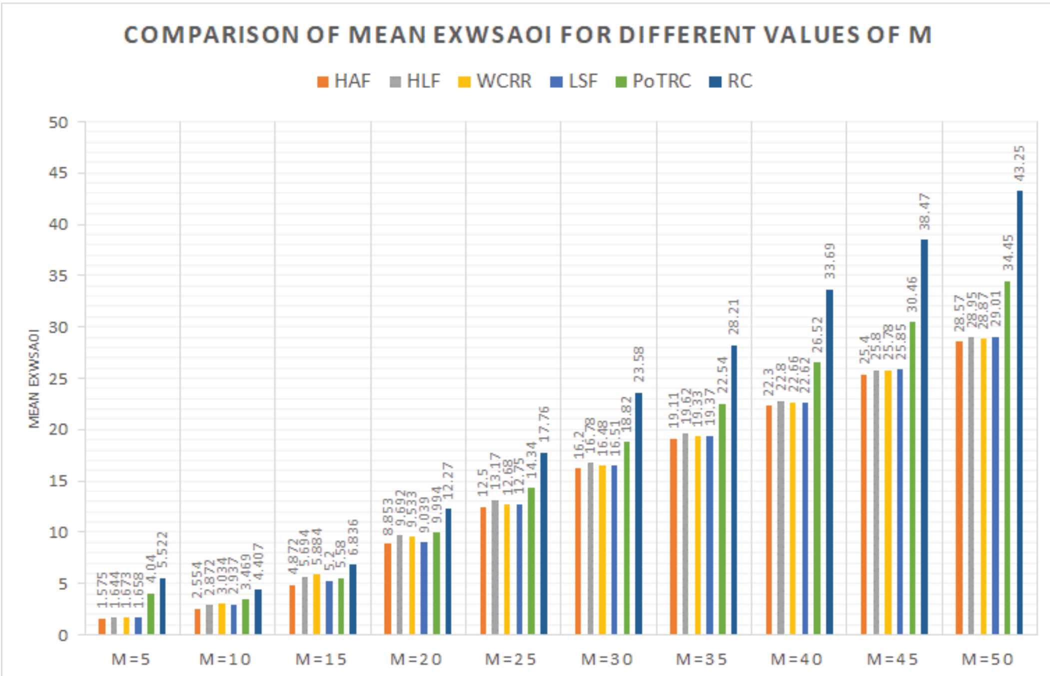


FIGURE 7. Comparison of mean EXWSAoI for different M.

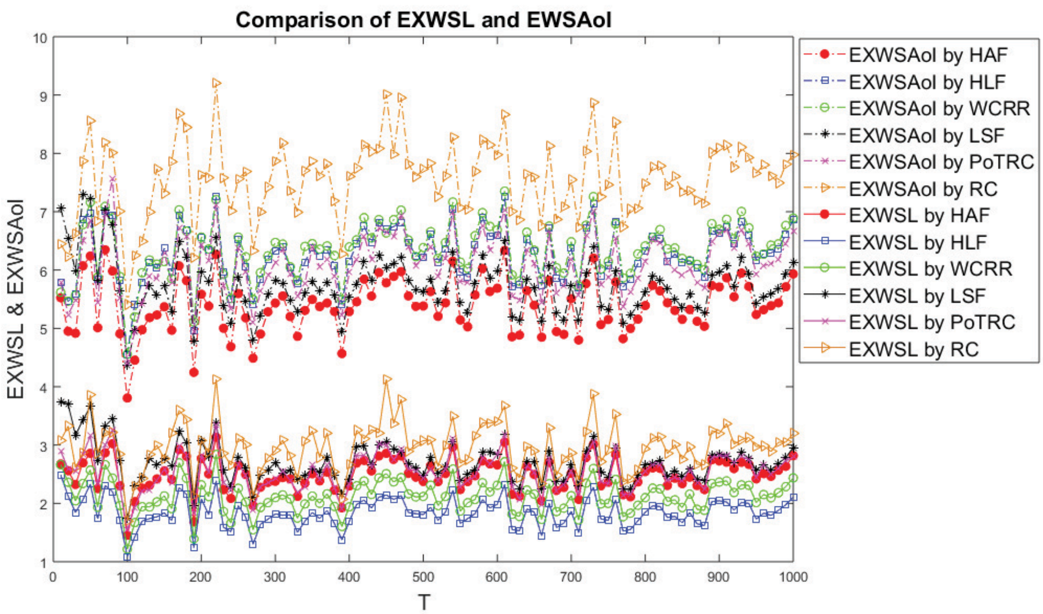


FIGURE 8. Comparison of EXWSAoI and EXWSL for M = 16.

cyber systems whereas latency and jitter are important parameters to maintain the quality of performance of the physical devices. The more jitter arises more the system performance degrades. As a result, in the CPSs, a thoughtful trade-off between the information freshness and jitter minimization is required while maintaining computational complexity as low as possible.

In CPSs, the physical part is not as fast as the cyber part. Any degradation in the performance of the physical plant has a wider effect that cannot be mitigated easily and quickly. Little latency causes higher jitter that has a huge effect on physical devices but the amount of change in information content caused by slightly higher age has very minor or negligible effect on decision making as the state of the physical

TABLE 2. Performance comparison of various algorithms.

Performance (Best to Worst)	Computation Complexity		EXWSAoI		EXWSL		RMS Jitter	
	Algorithm	Time Complexity	Algorithm	% Improvement compared to RC	Algorithm	% Improvement compared to RC	Algorithm	% Improvement compared to RC
↓	RC	$O(1)$	HAF	28.7	HLF	36.94	HLF	45.73
	PoTRC	$O(1)$	LSF	24.09	WCRR	28.45	WCRR	33.26
	WCRR	$O(1)$	PoTRC	19.22	HAF	16.39	HAF	22.06
	HAF	$O(n)$**	HLF	17.10	PoTRC	14.41	PoTRC	17.63
	HLF	$O(n)$**	WCRR	16.13	LSF	9.02	LSF	16.42
	LSF	$O(n)$	RC	0	RC	0	RC	0

**Detailed proof is given in appendix I

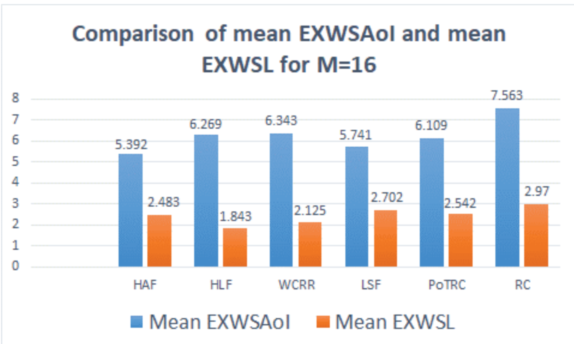


FIGURE 9. Comparison of mean EXWSAoI and EXWSL for M = 16.

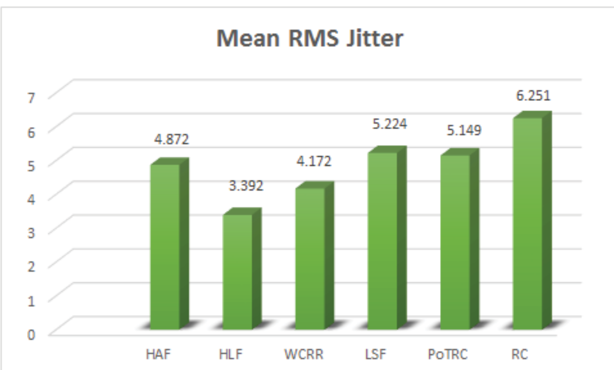


FIGURE 10. Comparison of mean RMS jitter for M = 16.

world does not change so fast. So minimizing latency and jitter seems to be more important at a cost of marginally higher AoI. Again latency minimization eradicates the staleness of information which is the main purpose for dealing with age as information freshness is monotonically decreasing function of stale age. That means finding the optimal value of AoI that satisfies minimum jitter constraint in addition to maximum information freshness is more effective for CPS than dealing with minimum AoI only. The following Table-2 compares the performance of all the above-discussed algorithms and from there it is clearly visible that our proposed greedy scheduling algorithm HLF provides considerably lower latency as well as jitter besides a sufficiently lower value of the age of information that is necessary for the better performance of IWSAN whereas another proposed scheduling algorithm HAF deals with minimization of the total age of information only. Among other traditional algorithms, RC is though simplest but produce worst results in terms of all performance metrics i.e. age, latency, and jitter. WCRR and PoTRC have better performance than RC at a cost of same time complexity but still they are not up to the mark in terms of EXWSAoI and EXWSL respectively compared to HAF and HLF. LSF has complexity same as HAF and HLF and its age performance is quite good but its latency and jitter performance is too poor to be acceptable. Hence, after looking into merits and demerits of all the above-mentioned algorithms it can be aptly concluded that HLF is more compatible scheduling policy in the context of industrial CPS.

VI. CONCLUSION

This paper considers a symmetric IWSAN with a time-shared unreliable channel for sending time-sensitive information about the state of interest of the system to a processor/controller to generate necessary actions accordingly. Two energy efficient greedy sensor scheduling algorithms HAF and HLF are proposed for that purpose. We studied their optimality analytically with respect to age, latency, and jitter and compared their effect on system performance by extensive simulations. Finally, it is concluded that the attempt to minimize overall age of information seems to be a less effective practice than minimizing the stale part imbedded in total the age term in case of both cyber and physical world simultaneously as a closed loop networked control system. Mitigating stateless in turn assures the overall freshness of the information along with the smooth execution of physical actions by minimizing jitter. Investigation of age optimal scheduling algorithm for asymmetric IWSAN, incorporating deadline and emergency tasks and scheduling policies for multi-hop, multi-channel communication network may be considered as some of the interesting future research directions of this work.

APPENDIX A
PROOF OF PROPERTY I

At slot t, HAF schedules unprocessed active sensor sample from sensor $j^{HAF(t)} = arg \max_{i \in S_t^{HAF}} \left\{ h_{t,i}^{HAF} \right\}$. If sensor sample $j^{HAF(t)}$ is processed successfully at slot t, the age evolution at the end of slot t from (2) will be as follows,

$$h_{t+1,i}^{HAF} = h_{t,i}^{HAF} + 1 \forall i \in S_t^{HAF} \setminus \left\{ j^{HAF(t)} \right\}$$
$$h_{t+1,i}^{HAF} = 1 \, for \; i = j^{HAF(t)}. \quad (13)$$

At slot $t+1$, the set of active sensors $S_{t+1}^{HAF} = \left[S_t^{HAF} \setminus \left\{ j^{HAF(t)} \right\} \right] \cup N_{t+1}$, N_{t+1} is the set of sensors that just become active at the beginning of slot $t+1$ and AoI of any sensor sample $n \in N_{t+1}$ is $h_{t,i}^{HAF} = h_{t+1,n}$. So at slot $t+1$, $j^{HAF(t+1)} = arg \max_{i \in S_{t+1}^{HAF}} \left\{ h_{t+1,i}^{HAF} \right\} \neq j^{HAF(t)}$. But if the sensor sample $j^{HAF(t)}$ fails to be processed at slot t due to bad channel condition, the age evolution at the end of slot t will be, $h_{t+1,i}^{HAF} = h_{t,i}^{HAF} + 1 \forall i \in S_t^{HAF}$. At slot $t+1$, active sensors $i \in S_{t+1}^{HAF} = S_t^{HAF} \cup N_{t+1}$ and for all $n \in N_{t+1}$, $h_{t+1,i}^{HAF} = h_{t+1,n}$. If any $h_{t+1,n} \leq h_{t+1,j^{HAF(t)}}^{HAF}$, then $j^{HAF(t+1)} = arg \max_{i \in S_{t+1}^{HAF}} \left\{ h_{t+1,i}^{HAF} \right\} = j^{HAF(t)}$ but if any one of $h_{t+1,n} \geq h_{t+1,j^{HAF(t)}}^{HAF}$, then $j^{HAF(t+1)} = arg \max_{i \in S_{t+1}^{HAF}} \left\{ h_{t+1,i}^{HAF} \right\} \neq j^{HAF(t)}$.

Thus property-I holds.

APPENDIX B
PROOF OF LEMMA I

Consider a symmetric IWSAN network with M sensors, M actuators and a controller/processor as described in sec. IV. The weight assigned to each sensor is $\alpha_i = \alpha > 0 \, \forall i \in M$ and time shared unreliable channel from sensor to processor has reliability $p_i = p \in [0, 1]$. The channel condition *ON* or *OFF* as eqn. (1) is time varying and at any slot t it is independent of sensors and the underlying scheduling policy. We use induction method to prove the age optimality of HAF as compared to any other admissible scheduling policy $\pi_1 \in \prod$. At any slot t, let $R^{(t)}$ be the total number of active sensors and the set of active sensors be $S_t = \{x_i\}_t$, $i = 1 \ldots R^{(t)}$. Without the loss of generality if the sensors in S_t are arranged in decreasing order of AoI of the sensor samples then the modified set will be say, S_t' and elements in S_t' are represented by $e_{i'}$ for $i' = 1 \ldots R^{(t)}$. Let us assume, the sample from active sensor $d^{(t)}$ is processed successfully at slot t. So at the end of slot t age will be, $h_{t+1,i} = h_{t,i'} + 1 \forall i' \in S_t' \setminus \left\{ d^{(t)} \right\}$, $i \in S_{t+1}$. Let, $N_{t+1} = \{y_n\}_{t+1}$, $n = 1 \ldots r^{(t+1)}$ be the set of $r^{(t+1)}$ new sensors that just become active at the beginning of slot $t+1$ with age $h_{t+1,n} \forall n \in N_{t+1}$. Then at $t+1$ the set of active sensors $S_{t+1} = \left[S_t' \setminus \left\{ d^{(t)} \right\} \right] \cup N_{t+1}$ and S_{t+1}' is sorted S_{t+1} in decreasing order of AoI of the sensor samples. For comparing the performance of proposed HAF with any other scheduling policy $\pi_1 \in \prod$ we assume that the set of newly active sensors N_{t+1} added at the beginning of each slot is independent of underlying scheduling policy used. According to HAF, always $arg \max_{i' \in S_t'^{HAF}} \left\{ h_{t,i'}^{HAF} \right\} = \{e_1\}_t^{HAF}$ is processed in slot t whereas in π_1 any element $\{e_k\}_t^{\pi_1} = d^{\pi_1(t)} \neq \{e_1\}_t^{\pi_1} = j^{\pi_1(t)} = arg \max_{i \in S_t^{\pi_1}} \left\{ h_{t,i}^{\pi_1} \right\} = arg \max_{i' \in S_t'^{\pi_1}} \left\{ h_{t,i'}^{\pi_1} \right\}$ is processed in t.

Remark-1: From the above discussion we can say, $R^{(t+1)} = R^{(t)} - 1 + r^{(t+1)}$ if $d^{(t)}$ is processed successfully at slot t. Otherwise $R^{(t+1)} = R^{(t)} + r^{(t+1)}$.

Remark-2: The total number of elements $R^{(t)}$ is same for both HAF and π_1 at any slot t. Starting from the same initial condition only one of the active sensor samples is being processed per slot. No sample is being processed if either channel is OFF or scheduling policy remains idle as no active sensor sample is present in that slot. Remaining active sensors are carried forward to the next slot in addition to the same number of new sensors that are being added in each slot irrespective of the underlying scheduling policy.

Base case: As initial conditions $\overrightarrow{h_1(1)}$ and $\overrightarrow{c(0)}$ are same for both the policies, the set of active sensors S_t is same for both HAF and π_1 at slot $t = 1$. So, $S_1'^{HAF} = S_1'^{\pi_1} = S_1'$, $\{e_{i'}\}_1^{HAF} = \{e_{i'}\}_1^{\pi_1} = \{e_{i'}\}_1$ and $h_{1,i'}^{HAF} = h_{1,i'}^{\pi_1} = h_{1,i'}$.

Inductive step: First it is assumed that for any slot t, $h_{t,i'}^{\pi_1} \geq h_{t,i'}^{HAF} \forall i' \in S_t'$. To prove $h_{t+1,i'}^{\pi_1} \geq h_{t+1,i'}^{HAF} \forall i' \in S_t'$ we can write from the initial discussion that,

$$\begin{aligned} S_{t+1}^{HAF} &= \left\{ x_1, x_2, \ldots, x_{R^{(t+1)}} \right\}_{t+1}^{HAF} \\ &= S_t'^{HAF} \setminus \{e_1\}_t^{HAF} \cup N_{t+1} \\ &= \left\{ \cancel{e_1}, e_2, e_3, \ldots, e_{R^{(t)}} \right\}_t'^{HAF} \cup \left\{ y_1, y_2, \ldots, y_{r^{(t+1)}} \right\}_{t+1} \\ &= \left\{ e_2, e_3, \ldots, e_{R^{(t)}} \right\}_t'^{HAF} \cup \left\{ y_1, y_2, \ldots, y_{r^{(t+1)}} \right\}_{t+1} \\ &= \left\{ \underbrace{x_1, x_2 \ldots x_{R^{(t)}-1}}_{\substack{\text{Inherited from} \\ S_t'^{HAF} \setminus \{e_1\}_t^{HAF}}}, \underbrace{x_{R^{(t)}}, x_{R^{(t)}+1} \cdots x_{R^{(t+1)}}}_{\substack{\text{Newly added } r^{(t+1)} \text{sensors} \\ \text{from } N_{t+1}}} \right\}_{t+1}^{HAF} \end{aligned} \quad (14)$$

From above equation the relation between S_{t+1}^{HAF} and $S_t'^{HAF}$ can be found as, $\{x_i\}_{t+1}^{HAF} = \left\{ e_{i'+1} \right\}_t^{HAF}$ for $1 \leq i = i' < R^{(t)}$. $\{x_i\}_{t+1}^{HAF} = \{y_n\}_{t+1}$ for $n = 1 \; to \; r^{(t+1)}$ and $R^{(t)} \leq i \leq R^{(t+1)}$. Age for any element $i \in S_{t+1}^{HAF}$ is as follows,

$$\begin{aligned} h_{t+1,i}^{HAF} &= h_{t,i'+1}^{HAF} + 1 \, for \; 1 \leq i = i' < R^{(t)} \\ &= h_{t+1,n} \, for \; n = 1 \; to \; r^{(t+1)}, R^{(t)} \leq i \leq R^{(t+1)} \end{aligned} \quad (15)$$

Similarly for π_1,

$$\begin{aligned} S_{t+1}^{\pi_1} &= \left\{ x_1, x_2, \ldots, x_{R^{(t+1)}} \right\}_{t+1}^{\pi_1} \end{aligned}$$

$$= S_t^{\prime\pi_1} \setminus \{e_k\}_t^{\pi_1} \cup N_{t+1}$$
$$= \{e_1, e_2, .., e_{k-1}, \cancel{e_k}, e_{k+1}, .., e_{R(t)}\}_t^{\prime\pi_1} \cup \{y_1, y_2, .., y_{r(t+1)}\}_{t+1}$$
$$= \{e_1, e_2, \ldots, e_{k-1}, e_{k+1}, \ldots, e_{R^{(t)}}\}_t^{\prime\pi_1} \cup \{y_1, y_2, \ldots, y_{r^{(t+1)}}\}_{t+1}$$
$$= \left\{ \underbrace{x_1, x_2, \ldots, x_k, x_{k+1} \ldots x_{R^{(t)}-1}}_{\substack{Inherited\ from \\ S_t^{\prime\prime\pi}2\setminus\{g_k\}_t^{\pi_1}}}, \underbrace{x_{R^{(t)}}, x_{R^{(t)}+1}, \cdots, x_{R^{(t+1)}}}_{\substack{Newly\ added\ r^{t+1} sensors \\ from\ N_{t+1}}} \right\}_{t+1}^{\pi_1} \quad (16)$$

The relation between $S_{t+1}^{\pi_1}$ and $S_t^{\prime\pi_1}$ can be found as, $\{x_i\}_{t+1}^{\pi_1} = \{e_{i'}\}_t^{\pi_1} for 1 \leq i = i' < k$. $\{x_i\}_{t+1}^{\pi_1} = \{e_{i'+1}\}_t^{\pi_1}$ $for\ k \leq i = i' < R^{(t)}$. $\{x_i\}_{t+1}^{\pi_1} = \{y_n\}_{t+1}^{\pi_1}$ $for\ n = 1\ to\ r^{(t+1)}$ and $R^{(t)} \leq i \leq R^{(t+1)}$. Age for any element $i \in S_{t+1}^{\pi_1}$ is as follows,

$$h_{t+1,i}^{\pi_1} = h_{t,i'}^{\pi_1} + 1\,for \text{ and } 1 \leq i = i' < k$$
$$= h_{t,i'+1}^{\pi_1} + 1\,for\ k \leq i = i' < R^{(t)}$$
$$= h_{t+1,n}\,for\ n = 1\ to\ r^{(t+1)}, R^{(t)} \leq i \leq R^{(t+1)} \quad (17)$$

As $h_{t,i'}^{\pi_1} \geq h_{t,i'}^{HAF}$ and $h_{t,i'}^{HAF} \geq h_{t,i'+1}^{HAF}$ thus $h_{t,i'}^{\pi_1} \geq h_{t,i'+1}^{HAF}$ for sure. Now comparing $h_{t+1,i}^{HAF}$ and $h_{t+1,i}^{\pi_1}$ for the above three segments of the value of i it can be proved that, $h_{t+1,i}^{\pi_1} \geq h_{t+1,i}^{HAF} \forall i \in S_{t+1}$. Next, $S_{t+1}^{\pi_1}$ and S_{t+1}^{HAF} are arranged in decreasing order of age to $S_{t+1}^{\prime\pi_1}$ and $S_{t+1}^{\prime HAF}$ respectively. For any $n \in N_{t+1}$ and $i \in S_{t+1}$ if,

Case i: $h_{t+1,n} \geq h_{t+1,i}^{HAF} \geq h_{t+1,i}^{\pi_1}$, then $S_{t+1}^{\prime G-I} = \{\ldots y_n, x_i, x_{i+1} \ldots\}_{t+1}^{HAF}$ and $S_{t+1}^{\prime\pi_1} = \{\ldots y_n, x_i, x_{i+1} \ldots\}_{t+1}^{\pi_1}$.

Case ii: $h_{t+1,i}^{\pi_1} \geq h_{t+1,n} \geq h_{t+1,i}^{HAF}$, then $S_{t+1}^{\prime HAF} = \{\ldots y_n, x_i, x_{i+1} \ldots\}_{t+1}^{HAF}$ and $S_{t+1}^{\prime\pi_1} = \{\ldots x_i, y_n, x_{i+1} \ldots\}_{t+1}^{\pi_1}$.

Case iii: $h_{t+1,i}^{\pi_1} \geq h_{t+1,i}^{HAF} \geq h_{t+1,n}$, then $S_{t+1}^{\prime HAF} = \{\ldots x_i, y_n, x_{i+1} \ldots\}_{t+1}^{HAF}$ and $S_{t+1}^{\prime\pi_1} = \{\ldots x_i, y_n, x_{i+1} \ldots\}_{t+1}^{\pi_1}$.

From the above discussion and previous result $h_{t+1,i}^{\pi_1} \geq h_{t+1,i}^{HAF} \forall i \in S_{t+1}$, we compare $h_{t+1,i'}^{HAF}$ and $h_{t+1,i'}^{\pi_1}$ for three cases and can remark that $h_{t+1,i'}^{\pi_1} \geq h_{t+1,i'}^{HAF} \forall i' \in S'_{t+1}$ **(Induction complete)**. It readily follows Lemma-I.

APPENDIX C
PROOF OF THEOREM-I

For finding the optimality of the proposed HAF algorithm, the value of between objective-I calculated by HAF and any other admissible scheduling policy $\pi_1 \in \prod$ are compared.

Let $H_t = \sum_{i \in S_t} h_{t,i}$ be the sum of the ages of all the active sensors in the active sensor set S_t at slot t. If we can prove $H_t^{HAF} \leq H_t^{\pi_1}\ \forall t \in T$, it is sufficient to state that $O1_T^{\pi^*} = O1_T^{HAF} \leq O1_T^{\pi_1} \forall \pi_1 \in \prod$. Let's say, the sample from the active sensor $d^{(t)}$ is processed successfully at slot t. So at the end of slot t age will be, $h_{t+1,i} = h_{t,i} + 1\ if\ i \neq d^{(t)}$ and $h_{t+1,i} = 1\ if\ i = d^{(t)}$ from eqn (2). At slot $t+1$ the set of active sensors $S_{t+1} = \left[S_t \setminus \{d^{(t)}\}\right] \cup N_{t+1}$ where N_{t+1} is the set of sensors that just become active at the beginning of slot $t+1$ and AoI of any sensor sample $n \in N_{t+1}$ be $h_{t+1,i} = h_{t+1,n}$ and $H_{N_{t+1}} = \sum_{n \in N_{t+1}} h_{t+1,n}$. At slot $t+1$, $H_{t+1} = \sum_{i \in S_{t+1}} h_{t+1,i}$ can be expressed as,

$$H_{t+1} = H_t - h_{t,d^{(t)}} + \sum_{i \in S_t} \mathbb{I}_t(i) + H_{N_{t+1}} \quad (18)$$

Here $\mathbb{I}_t(.)$ is an indicator function which takes the value $\mathbb{I}_t(i) = 1\ if\ i \neq d^{(t)}$ and $\mathbb{I}_t(i) = 0\ if\ i = d^{(t)}$. Consider the same assumption as Lemma-I that the numbers of newly active sensors added at the beginning of each slot is independent of the policy used. According to HAF, always $d^{HAF(t)} = j^{HAF(t)} = arg \max_{i \in S_t^{HAF}} \left\{h_{t,i}^{HAF}\right\}$ is processed in slot t whereas in policy π_1 any element $d^{\pi_1(t)} \neq j^{\pi_1(t)} = arg \max_{i \in S_t^{\pi_1}} \left\{h_{t,i}^{\pi_1}\right\}$ is processed.

Base case: As initial conditions $\overrightarrow{h_1(1)}$ and $\overrightarrow{c(0)}$ are same for both the policies, the sets of active sensors $S_1^{HAF} = S_1^{\pi_1} = S_1$ are same for both HAF and π_1 at slot $t = 1$. So, $H_1^{HAF} = H_1^{\pi_1} = H_1$.

Inductive step: First it is assumed that for any slot $t \in T$, $H_t^{HAF} \leq H_t^{\pi_1}$. The expression H_{t+1}^{HAF} and $H_{t+1}^{\pi_1}$ can be obtained from eqn. (18) as follows,

$$H_{t+1}^{HAF} = H_t^{HAF} - h_{t,d^{HAF(t)}} + \sum_{i \in S_t^{HAF}} \mathbb{I}_t(i) + H_{N_{t+1}} \quad (19)$$
$$H_{t+1}^{\pi_1} = H_t^{\pi_1} - h_{t,d^{\pi_1(t)}} + \sum_{i \in S_t^{\pi_1}} \mathbb{I}_t(i) + H_{N_{t+1}} \quad (20)$$

From the remark-II drawn in Lemma-I, the total number of elements $R^{(t)}$ is same for both HAF and π_1 at any slot t. So $\sum_{i \in S_t^{HAF}} \mathbb{I}_t(i) = \sum_{i \in S_t^{\pi_1}} \mathbb{I}_t(i)$. Therefore the comparison of H_{t+1}^{HAF} and $H_{t+1}^{\pi_1}$ depends on first two terms of eqn. (19) and (20).

$$(H_{t+1}^{\pi_1} - H_{t+1}^{HAF}) = (H_t^{\pi_1} - H_t^{HAF}) + (h_{t,d^{HAF(t)}} - h_{t,d^{\pi_1(t)}}) \quad (21)$$

Now, $(H_t^{\pi_1} - H_t^{HAF}) = \delta_t(\text{say}) \geq 0$ from the assumption in the inductive step. To make $(H_{t+1}^{\pi_1} - H_{t+1}^{HAF}) = \delta_{t+1} \geq 0$ either of the two following conditions must be satisfied.

$$Condition\ 1:\ \left(h_{t,d^{HAF(t)}} - h_{t,d^{\pi_1(t)}}\right) \geq 0$$
$$Condition\ 2:\ -\delta_t \leq \left(h_{t,d^{HAF(t)}} - h_{t,d^{\pi_1(t)}}\right) < 0 \Leftrightarrow 0 \leq \left(h_{t,d^{\pi_1(t)}} - h_{t,d^{HAF(t)}}\right) \leq \delta_t$$

Next, we arrange $S_t^{\pi_1}$ and S_t^{HAF} in decreasing order of age to $S_t^{'\pi_1}$ and $S_t^{'HAF}$ respectively. Elements in $S_t^{'\pi_1}$ and $S_t^{'HAF}$ are represented by $\{e_{i'}\}_t^{\pi_1}$ and $\{e_{i'}\}_t^{HAF}$ respectively for $i' = 1 \ldots R^{(t)}$. From the result of Lemma-I for any slot $t \in T, h_{t,i'}^{\pi_1} \geq h_{t,i'}^{HAF}\ \forall i' \in \{1 \ldots R^{(t)}\}$. Hence, $\{e_1\}_t^{\pi_1} \geq \{e_1\}_t^{HAF}$. As $d^{HAF(t)} = arg \max_{i \in S_t^{HAF}} \left\{h_{t,i}^{HAF}\right\} = \{e_1\}_t^{HAF}$ and $d^{\pi_1(t)} = \{e_k\}_t^{\pi_1} \leq \{e_1\}_t^{\pi_1} = arg \max_{i \in S_t^{\pi_1}} \left\{h_{t,i}^{\pi_1}\right\}$, two cases may happen.

Case i: $\{e_1\}_t^{HAF} \geq \{e_k\}_t^{\pi_1}$ *i.e.* $\left(h_{t,d^{HAF(t)}} - h_{t,d^{\pi_1(t)}}\right) \geq 0$. Then *condition 1* is satisfied.

Case ii. $\{e_1\}_t^{HAF} \leq \{e_k\}_t^{\pi_1}$ *i.e.* $\left(h_{t,d^{HAF(t)}} - h_{t,d^{\pi_1(t)}}\right) < 0 \Leftrightarrow \left(h_{t,d^{\pi_1(t)}} - h_{t,d^{HAF(t)}}\right) > 0$. Now, if we can prove that $max\left(h_{t,d^{\pi_1(t)}} - h_{t,d^{HAF(t)}}\right) \leq \delta_t$ then our purpose will be fulfilled.

The elements present in $S_t^{\pi_1}$ and $S_t^{'\pi_1}$ are same but in a different order. The similar thing happens between S_t^{HAF} and $S_t^{'HAF}$. So, $H_t^{\pi_1} = \sum_{i \in S_t^{\pi_1}} h_{t,i} = \sum_{i' \in S_t^{'\pi_1}} h_{t,i'}$ and $H_t^{HAF} = \sum_{i \in S_t^{HAF}} h_{t,i} = \sum_{i' \in S_t^{'HAF}} h_{t,i'}$.

$$
\begin{aligned}
(H_t^{\pi_1} - H_t^{HAF}) &= \sum_{i' \in S_t^{'\pi_1}} h_{t,i'} - \sum_{i' \in S_t^{'HAF}} h_{t,i'} \\
&= \left[\{e_1\}_t^{\pi_1} - \{e_1\}_t^{HAF}\right] + \left[\{e_2\}_t^{\pi_1} - \{e_2\}_t^{HAF}\right] \\
&\quad + \ldots + \left[\left\{e_{R^{(t)}}\right\}_t^{\pi_1} - \left\{e_{R^{(t)}}\right\}_t^{HAF}\right] \qquad (22)
\end{aligned}
$$

Let $\left[\{e_{i'}\}_t^{\pi_1} - \{e_{i'}\}_t^{HAF}\right] = \alpha_{i'} \geq 0$ (from Lemma-I) and $(H_t^{\pi_1} - H_t^{HAF}) = \delta_t \geq 0$. Substituting $\alpha_{i'}$ and δ_t in eqn. (22)

$$\delta_t = \alpha_1 + \alpha_2 + \ldots + \alpha_{R^{(t)}} \qquad (23)$$

Now, the maximum possible value of $\left(h_{t,d^{\pi_1(t)}} - h_{t,d^{HAF(t)}}\right)$ is obtained when $d^{\pi_1(t)} = \{e_k\}_t^{\pi_1} = \{e_1\}_t^{\pi_1}$. $max\left(h_{t,d^{\pi_1(t)}} - h_{t,d^{HAF(t)}}\right) = \{e_1\}_t^{\pi_1} - \{e_1\}_t^{HAF} = \alpha_1$. From (43) it can be seen that, $\alpha_1 \leq \delta_t$ that gives, $max\left(h_{t,d^{\pi_1(t)}} - h_{t,d^{HAF(t)}}\right) \leq \delta_t$. Hence, *condition 2* is satisfied. Combining the results of *case i* and *case ii* we can conclude that $\left(h_{t,d^{HAF(t)}} - h_{t,d^{\pi_1(t)}}\right) \geq -\delta_t$ that means,$(H_{t+1}^{\pi_1} - H_{t+1}^{HAF}) = \delta_{t+1} > 0$ or $H_{t+1}^{HAF} \leq H_{t+1}^{\pi_1}$ for any slot $t \in T$ **(Induction complete)**. Hence Theorem-I is proved.

APPENDIX D
PROOF OF PROPERTY-II

At slot t, HLF schedules unprocessed active sensor sample from sensor $j^{HLF(t)} = arg \max_{i \in S_t} \{L_{t,i}\}$. If the sensor sample $j^{HLF(t)}$ is processed successfully at slot t, the age evolution at the end of slot t will be,

$$
\begin{aligned}
&h_{t+1,i}^{HLF} = h_{t,i}^{HLF} + 1 \forall i \in S_t^{HLF} \setminus \left\{j^{HLF(t)}\right\} \\
&h_{t+1,i}^{HLF} = 1\, for\ i = j^{HLF(t)}. \qquad (24)
\end{aligned}
$$

This age hike is causing due to latency increment of unprocessed sample from any sensor i at slot t. From (10) and (24) it can be stated that at the end of slot t latency increment will be as follows,

$$L_{t+1,i}^{HLF} = L_{t,i}^{HLF} + 1 \forall i \in S_t^{HLF} \setminus \left\{j^{HLF(t)}\right\} \qquad (25)$$

At slot $t+1$, the set of active sensors $S_{t+1}^{HLF} = \left[S_t^{HLF} \setminus \left\{j^{HLF(t)}\right\}\right] \cup N_{t+1}$, N_{t+1} is the set of sensors that just become active at the beginning of slot $t+1$. So the latency of any sensor sample $n \in N_{t+1}$ is $L_{t+1,i}^{HLF} = L_{t+1,n} = 0$. Then at slot $t+1$, $j^{HLF(t+1)} = arg \max_{i \in S_{t+1}^{HLF}} \left\{L_{t+1,i}^{HLF}\right\} \neq j^{HLF(t)}$. If the sensor sample $j^{HLF(t)} \in S_t^{HLF}$ fails to be processed at slot t due to bad channel condition, the age evolution and latency increment at the end of slot t will be, $h_{t+1,i}^{HLF} = h_{t,i}^{HLF} + 1 \forall i \in S_t^{HLF}$ and $L_{t+1,i}^{HLF} = L_{t,i}^{HLF} + 1 \forall i \in S_t^{HLF}$ respectively. As $L_{t,i}^{HLF} \geq 0, L_{t+1,i}^{HLF} \geq 1 \forall i \in S_t^{HLF}$. At slot $t+1$ as active sensors $i \in S_{t+1}^{HLF} = S_t^{HLF} \cup N_{t+1}$ where $\forall n \in N_{t+1}$, $L_{t+1,i} = L_{t+1,n} = 0$. So, $L_{t+1,n} \leq L_{t+1,j^{HLF(t)}}^{HLF}$ always. Then $j^{HLF(t+1)} = arg \max_{i \in S_{t+1}^{HLF}} \left\{L_{t+1,i}^{HLF}\right\} = j^{HLF(t)}$ always. Thus property-II holds.

APPENDIX E
PROOF OF PROPERTY-III

Consider the same symmetric IWSAN network containing M sensors, M actuators and a controller/processor as used in Lemma-I. At any slot t, let $R^{(t)}$ be the total number of active sensors and the set of active sensors be $S_t^{HLF} = \{x_i\}_t^{HLF}$, $i = 1 \ldots R^{(t)}$. Without the loss of generality, the sensors in S_t^{HLF} are arranged in decreasing order of latency ($L_{t,i}^{HLF}$) of the sensor samples. The sensor that has become active earlier or in other words, sensors that entered active set S_t^{HLF} earlier has higher latency. According to HLF, if two sensor samples have the same latency then the sample with higher age will get the higher priority. That means it will be placed before the other one in the ordered set. Say, the modified set will be $S_t^{''HLF}$ and elements in $S_t^{''HLF}$ are represented by $g_{i'}$ for $i' = 1 \ldots R^{(t)}$. So in $S_t^{''HLF}, L_{t,i'}^{HLF} \geq L_{t,i'+1}^{HLF}$. If $L_{t,i'}^{HLF} = L_{t,i'+1}^{HLF}$ then by definition of HLF, $h_{t,i'}^{HLF} \geq h_{t,i'+1}^{HLF}$.

Let, $N_{t+1} = \{y_n\}_{t+1}$ for $n = 1 \ldots r^{(t+1)}$ be the set of $r^{(t+1)}$ new sensors that just become active at the beginning of slot $t+1$ with age $h_{t+1,n}$ and latency $L_{t+1,n} = 0 \forall n \in N_{t+1}$. $N_{t+1}'' = \{z_n\}_{t+1}$ for $n = 1 \ldots r^{(t+1)}$ be the modified set from N_{t+1} arranged in decreasing order of age $h_{t+1,n}$. From property-II it is known that according to HLF, always $d^{HLF(t)} = arg \max_{i \in S_t^{HLF}} \left\{L_{t,i}^{HLF}\right\} = arg \max_{i' \in S_t^{''HLF}} \left\{L_{t,i'}^{HLF}\right\} = \{g_1\}_t^{HLF}$ is scheduled to be processed in slot t until it is getting service successfully. So, at $t+1$ set of active sensors $S_{t+1}^{HLF} = \left\{S_t^{''HLF} \setminus \{g_1\}_t^{HLF}\right\} \cup N_{t+1}'' = S_{t+1}^{''HLF}$ as follows,

$$
\begin{aligned}
&S_{t+1}^{HLF}\\
&=\{x_1,x_2,\ldots,x_{R^{(t+1)}}\}_{t+1}^{HLF}\\
&=\left\{S_t''^{HLF}\setminus\{g_1\}_t^{HLF}\right\}\cup N_{t+1}''\\
&=\{\cancel{g_1},g_2,g_3,\ldots,g_{R^{(t)}}\}_t''^{HLF}\cup\{z_1,z_2,\ldots,z_{r^{(t+1)}}\}_{t+1}''\\
&=\left\{\underbrace{x_1,x_2,\ldots,x_{R^{(t)}-1}}_{\substack{\text{Inherited from}\\ S_t''^{HLF}\setminus\{g_1\}_t^{HLF}}},\ \underbrace{x_{R^{(t)}},x_{R^{(t)}+1},\ldots,x_{R^{(t+1)}}}_{\substack{\text{Newly added } r^{t+1}\text{sensors}\\ \text{from } N_{t+1}''}}\right\}_{t+1}^{HLF}\\
&=S_{t+1}''^{HLF} \qquad (26)
\end{aligned}
$$

From the above expression and property-II, we can conclude that sensor samples are being served in the same order as they are arranged in the active set $S_t''^{HLF}$. That means HLF repeatedly schedule $\{g_1\}_t^{HLF}$ to transmit sensor sample until it is getting service successfully. When the active sensor $\{g_1\}_t^{HLF}$ is served, it leaves the active set $S_t''^{HLF}$ and latency of the remaining packets increase by 1 which in turn increase the AoI of those remaining sensors by 1. So at the end of slot t, these remaining sensors in $S_t''^{HLF}\setminus\{g_1\}_t^{HLF}$ inherits to the active set S_{t+1}^{HLF} in the next slot $t+1$ with age $h_{t+1,i}=h_{t,i'}+1$ and latency $L_{t+1,i}=L_{t,i'}+1\forall i'\in S_t''^{HLF}\setminus\{g_1\}_t^{HLF}$, $i\in S_{t+1}$. In addition to that, a decreasing age wise arranged set of newly active sensors N_{t+1}'' appends at the end of $S_t''^{HLF}\setminus\{g_1\}_t^{HLF}$ in S_{t+1}^{HLF} and S_{t+1}^{HLF} is found to be same as $S_{t+1}''^{HLF}$. The same procedure goes on until the maximum number of slots T is reached. So the order of the processing of sensor samples is maintained in the same order they are becoming active and entering the active set. From this, we can conclude that the working policy of this HLF scheduling algorithm can be mapped as that of a FCFS queue.

APPENDIX F
PROOF OF COROLARRY-I

The jitter of any active sensor $i\in S_t$ is the square term of stale age as represented in eqn. (10).

$$JT_{t,i}=\left|\Delta_{t,i}\right|^2=\left|L_{t,i}+1\right|^2 \qquad (27)$$

Average jitter $\overline{JT}$ will be,

$$\overline{JT}=\frac{1}{TM}\left[\sum_{t=1}^{T}\sum_{i\in S_t}JT_{t,i}\right] \qquad (28)$$

Let, $\overline{JT}_t=\sum_{i\in S_t}JT_{t,i}$ be total jitter at the end of any slot t. To prove $\overline{JT}^{HLF}=\min_{\pi\in\Pi}\overline{JT}^{\pi}$ it is sufficient to show $\overline{JT}_t^{FCFS}=\min_{\pi\in\Pi}\overline{JT}_t^{\pi}\forall t\in T$. Now, property-III shows that HLF maps active set S_t as a queue and serves active sensor $i\in S_t$ in FCFS manner. If we can prove that FCFS queue provides minimum jitter than any other service policy then our purpose will be fulfilled.

Fig. 11. shows that the active sensors in S_t at any slot $t\in T$ are queued according to their arrival instant to get service. At slot t queue in Fig. 11(a) serves sample packet 1 in FCFS manner whereas queue in Fig. 11(b) serves any packet $k\neq 1$ following any other service policy say $S\pi$. Next, we calculate jitter at the end of slot t for each of FCFS and $S\pi$ service policy and show that, $\overline{JT}_t^{FCFS}\leq\overline{JT}_t^{s\pi}$ for any $t\in T$.

Let the total jitter at the end of slot t without any departure be,

$$
\begin{aligned}
\overline{JT}_t=&\left|L_{t,1}+1\right|^2+\ldots+\left|L_{t,k-1}+1\right|^2+\left|L_{t,k}+1\right|^2\\
&+\left|L_{t,k+1}+1\right|^2+\ldots+\left|L_{t,\tau}+1\right|^2 \qquad (29)
\end{aligned}
$$

From Fig. 11, the total jitter at the end of slot t for FCFS and $S\pi$ queue are as follows,

$$
\begin{aligned}
\overline{JT}_t^{FCFS}&=\cancel{\left|L_{t,1}+1\right|^2}+\ldots+\left|L_{t,k-1}+1\right|^2+\left|L_{t,k}+1\right|^2\\
&\quad+\left|L_{t,k+1}+1\right|^2+\ldots+\left|L_{t,\tau}+1\right|^2\\
&=\overline{JT}_t-\left|L_{t,1}+1\right|^2 \qquad (30)\\
\overline{JT}_t^{s\pi}&=\left|L_{t,1}+1\right|^2+\ldots\left|L_{t,k-1}+1\right|^2+\cancel{\left|L_{t,k}+1\right|^2}\\
&\quad+\left|L_{t,k+1}+1\right|^2+\ldots+\left|L_{t,\tau}+1\right|^2\\
&=\overline{JT}_t-\left|L_{t,k}+1\right|^2 \qquad (31)
\end{aligned}
$$

Now from Fig. 11, $L_{t,i}\geq L_{t,i+1}$, then, $L_{t,1}\geq L_{t,k}$ and $\overline{JT}_t^{FCFS}=\overline{JT}_t-\left|L_{t,1}+1\right|^2\leq\overline{JT}_t-\left|L_{t,k}+1\right|^2=\overline{JT}_t^{s\pi}$. From this result it is proved that FCFS queue provides minimum jitter than any other service policy which in turn shows that HLF attains the minimum average jitter.

APPENDIX G
PROOF OF LEMMA-II

M sensors in symmetric IWSAN network as used in Lemma-I share unreliable channel from the sensor to the processor on TDMA basis. This channel has reliability $p_i=p\in[0,1]$. The channel condition *ON* or *OFF* as eqn. (1) is time varying and at any slot t is independent of the sensor and the underlying scheduling policy. Induction method is used to prove the latency optimality of HLF as compared to any other admissible scheduling policy $\pi_2\in\prod$. These policies are compared under the same sample path input [38].

Let, the set of total $R^{(t)}$ number of active sensors at slot t is $S_t=\{x_i\}_t$, $i=1\ldots R^{(t)}$. $S_t''=\{g_{i'}\}_t$, $i'=1\ldots R^{(t)}$ is modified set S_t arranged in decreasing order of latency $(L_{t,i})$ with ties being broken by arranging sensors with the highest AoI $(h_{t,i})$ first. So in S_t'', $L_{t,i'}\geq L_{t,i'+1}$. If $L_{t,i'}=L_{t,i'+1}$ then by tie-breaking condition $h_{t,i'}\geq h_{t,i'+1}$. $N_{t+1}=\{y_n\}_{t+1}$, $n=1\ldots r^{(t+1)}$ be the set of $r^{(t+1)}$ new sensors that just become active at the beginning of slot $t+1$ with age $h_{t+1,n}$ and latency $L_{t+1,n}=0\,\forall n\in N_{t+1}$. $N_{t+1}''=\{z_n\}_{t+1}$, $n=1\ldots r^{(t+1)}$ be the modified set from N_{t+1} arranged in decreasing order of age $h_{t+1,n}$. So without loss of generality, at slot $t+1$ active sensor set $S_{t+1}=\left[S_t''\setminus\left\{d^{(t)}\right\}\right]\cup N_{t+1}''$. For comparing the performance of proposed HLF with any other scheduling policy $\pi_2\in\prod$ same assumption as Lemma-I is considered that the set of newly active sensors N_{t+1} added at the beginning

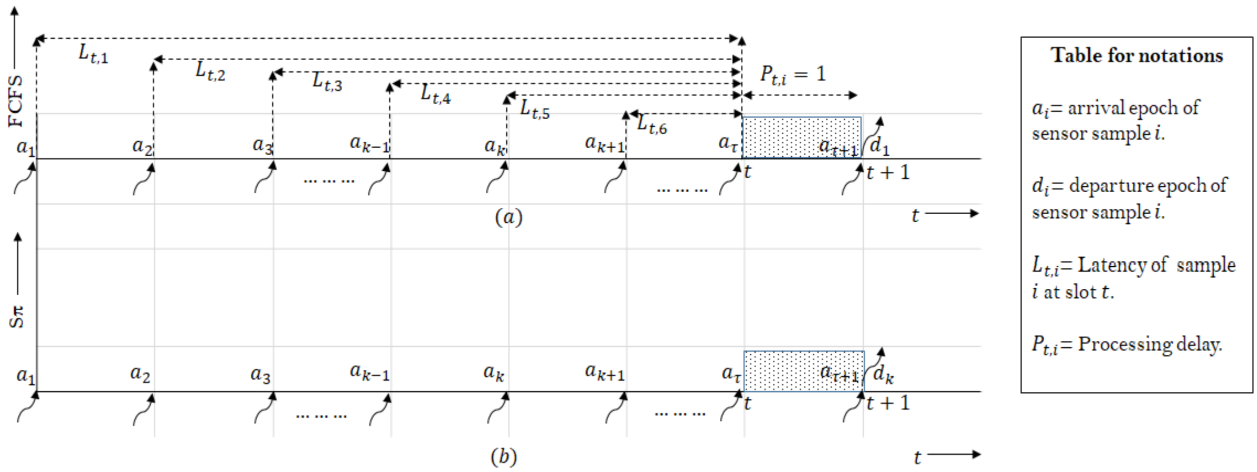


FIGURE 11. **Arrival and departure of sample packets for (a) FCFS and (b) $S\pi$ queue.**

of each slot is independent of underlying scheduling policy used. From Property-III it is known that HLF serves active sensors in S_t as FCFS basis. So, $d^{HLF(t)} = \{g_1\}_t^{HLF} = arg \max_{i\in S_t^{HLF}} \left\{L_{t,i}^{HLF}\right\} = arg \max_{i'\in S_t^{\prime\prime HLF}} \left\{L_{t,i'}^{HLF}\right\}$ whereas, in policy π_2 any element $d^{\pi_2(t)} = \{g_k\}_t^{\pi_2} \neq \{g_1\}_t^{\pi_2} = j^{\pi_2(t)} = arg \max_{i\in S_t^{\pi_2}} \left\{L_{t,i}^{\pi_2}\right\} = arg \max_{i'\in S_t^{\prime\prime\pi_2}} \left\{L_{t,i'}^{\pi_2}\right\}$ is processed in t.

Base case: As initial conditions $\overrightarrow{h_1(1)}$ and $\overrightarrow{c(0)}$ are same for both the policies, the set of active sensors S_t is same for both HLF and π_2 at slot $t = 1$. So, $S_1^{\prime\prime HLF} = S_1^{\prime\prime\pi_2} = S_1^{\prime\prime}$, $\{g_{i'}\}_1^{HLF} = \{g_{i'}\}_1^{\pi_2} = \{g_{i'}\}_1$, $h_{1,i'}^{HLF} = h_{1,i'}^{\pi_2} = h_{1,i'}$ and $L_{1,i'}^{HLF} = L_{1,i'}^{\pi_2} = L_{1,i'}$.

Inductive step: First it is assumed that for any slot t, $L_{t,i'}^{\pi_2} \geq L_{t,i'}^{HLF} \forall i' \in S_t^{\prime\prime}$. To prove $L_{t+1,i'}^{\pi_2} \geq L_{t+1,i'}^{HLF} \forall i' \in S_t^{\prime\prime}$ we can write from the initial discussion that,

$$\begin{aligned}
&S_{t+1}^{HLF}\\
&= \left\{x_1, x_2, \ldots, x_{R^{(t+1)}}\right\}_{t+1}^{HLF}\\
&= \left\{S_t^{\prime\prime HLF} \setminus \{g_1\}_t^{HLF}\right\} \cup N_{t+1}^{\prime\prime}\\
&= \left\{\cancel{g_1}, g_2, g_3, \ldots, g_{R^{(t)}}\right\}_t^{\prime\prime HLF} \cup \left\{z_1, z_2, \ldots, z_{r^{(t+1)}}\right\}_{t+1}^{\prime\prime}\\
&= \left\{\underbrace{x_1, x_2, \ldots, x_{R^{(t)}-1}}_{\substack{Inherited\ from\\ S_t^{\prime\prime HLF}\setminus\{g_1\}_t^{HLF}}}, \underbrace{x_{R^{(t)}}, x_{R^{(t)}+1}, \ldots, x_{R^{(t+1)}}}_{\substack{Newly\ added\ r^{(t+1)} sensors\\ from\ N_{t+1}^{\prime\prime}}}\right\}_{t+1}^{HLF}\\
&= S_{t+1}^{\prime\prime HLF}\\
&= \left\{g_1, g_2, g_3, \ldots, g_{R^{(t+1)}}\right\}_{t+1}^{\prime\prime HLF}
\end{aligned} \tag{32}$$

From above equation the relation between $S_{t+1}^{HLF} = S_{t+1}^{\prime\prime HLF}$ and $S_t^{\prime\prime HLF}$ can be found as, $\{x_i\}_{t+1}^{HLF} = \{g_{i'}\}_{t+1}^{HLF} = \{g_{i'+1}\}_t^{HLF}$ *for* $1 \leq i = i' < R^{(t)}$. $\{x_i\}_{t+1}^{HLF} = \{g_{i'}\}_{t+1}^{HLF} = \{z_n\}_{t+1}$ *for* $n = 1$ *to* $r^{(t+1)}$, $R^{(t)} \leq i = i' \leq R^{(t+1)}$. Age and latency for any element $i \in S_{t+1}^{HLF}$ is as follows,

$$\begin{aligned}
h_{t+1,i'}^{HLF} &= h_{t+1,i}^{HLF} = h_{t,i'+1}^{HLF} + 1 \; for \; 1 \leq i = i' < R^{(t)}\\
&= h_{t+1,n} \; for \; n = 1 \; to \; r^{(t+1)},\\
&\qquad R^{(t)} \leq i = i' \leq R^{(t+1)}
\end{aligned} \tag{33}$$

$$\begin{aligned}
L_{t+1,i'}^{HLF} &= L_{t+1,i}^{HLF} = L_{t,i'+1} + 1 \; for \; 1 \leq i = i' < R^{(t)}\\
&= 0 \; for \; n = 1 \; to \; r^{(t+1)},\\
&\qquad R^{(t)} \leq i = i' \leq R^{(t+1)}
\end{aligned} \tag{34}$$

Similarly for π_2,

$$\begin{aligned}
&S_{t+1}^{\pi_2}\\
&= \left\{x_1, x_2, \ldots, x_{R^{(t+1)}}\right\}_{t+1}^{\pi_2}\\
&= \left\{S_t^{\prime\prime\pi_2} \setminus \{g_k\}_t^{\pi_2}\right\} \cup N_{t+1}^{\prime\prime}\\
&= \left\{g_1, g_2, \ldots, g_{k-1}, \cancel{g_k}, g_{k+1}, \ldots, g_{R^{(t)}}\right\}_t^{\prime\prime\pi_2} \cup \left\{z_1, z_2, \ldots, z_{r^{(t)}}\right\}_{t+1}^{\prime\prime}\\
&= \left\{\underbrace{x_1, x_2, \ldots, x_k, x_{k+1} \ldots x_{R^{(t)}-1}}_{\substack{Inherited\ from\\ S_t^{\prime\prime\pi_2}\setminus\{g_k\}_t^{\pi_2}}}, \underbrace{x_{R^{(t)}}, x_{R^{(t)}+1}, \cdots, x_{R^{(t+1)}}}_{\substack{Newly\ added\ r^{(t+1)} sensors\\ from\ N_{t+1}}}\right\}_2^{\pi_2}\\
&= S_{t+1}^{\prime\prime\pi_2}\\
&= \left\{g_1, g_2, g_3, \ldots, g_{R^{(t+1)}}\right\}_{t+1}^{\prime\prime\pi_2}
\end{aligned} \tag{35}$$

From above equation the relation between $S_{t+1}^{\pi_2} = S_{t+1}^{\prime\prime\pi_2}$ and $S_t^{\prime\prime\pi_2}$ can be found as, $\{x_i\}_{t+1}^{\pi_2} = \{g_{i'}\}_{t+1}^{\pi_2} = \{g_{i'}\}_t^{\pi_2}$ *for* $1 \leq$

$i = i' < k, \{x_i\}_{t+1}^{\pi_2} = \{g_{i'}\}_{t+1}^{\pi_2} = \left\{g_{i'+1}\right\}_{t+1}^{\pi_2}$ *for* $k \leq i = i' < R^{(t)}$. $\{x_i\}_{t+1}^{\pi_2} = \{g_{i'}\}_{t+1}^{\pi_2} = \{z_n\}_{t+1}$ *for* $n = 1$ *to* $r^{(t+1)}, R^{(t)} \leq i = i' \leq R^{(t+1)}$. Age and latency for any element $i \in S_{t+1}^{\pi_2}$ is as follows,

$$\begin{aligned} h_{t+1,i'}^{\pi_2} &= h_{t+1,i}^{\pi_2} = h_{t,i'}^{\pi_2} + 1 \; for \; 1 \leq i = i' < k \\ &= h_{t,i'+1}^{\pi_2} + 1 \; for \; k \leq i = i' < R^{(t)} \\ &= h_{t+1,n} \; for \; n = 1 \; to \; r^{(t+1)}, \\ &\qquad R^{(t)} \leq i = i' \leq R^{(t+1)} \end{aligned} \tag{36}$$

$$\begin{aligned} L_{t+1,i'}^{\pi_2} &= L_{t+1,i}^{\pi_2} = L_{t,i'}^{\pi_2} + 1 \; for \; 1 \leq i = i' < k \\ &= L_{t,i'}^{\pi_2} + 1 \; for \; k \leq i = i' < R^{(t)} \\ &= 0 \; for \; n = 1 \; to \; r^{(t+1)}, \\ &\qquad R^{(t)} \leq i = i' \leq R^{(t+1)} \end{aligned} \tag{37}$$

Now, $L_{t,i'}^{\pi_2} \geq L_{t,i'}^{HLF}$ and $L_{t,i'}^{HLF} \geq L_{t,i'+1}^{HLF}$. So $L_{t,i'}^{\pi_2} \geq L_{t,i'+1}^{HLF}$. Next, comparing $L_{t+1,i}^{HLF}$ and $L_{t+1,i}^{\pi_2}$ for the above three segments of the value of i it can be proved that $L_{t,i'} \geq L_{t,i'+1}$. By comparing $L_{t+1,i'}^{HLF}$ and $L_{t+1,i'}^{\pi_2}$ for all three segments of i' it is proved that $L_{t+1,i'}^{HLF} \leq L_{t+1,i'}^{\pi_2}$ always $\forall 1 \leq i' \leq R^{(t+1)}$ (**Induction complete**). It readily follows Lemma-II.

APPENDIX H
PROOF OF THEOREM-II

For finding the optimality of the proposed HLF algorithm we compare the value of objective II calculated by HLF and any other admissible scheduling policy $\pi_2 \in \prod$.

Let $V_t = \sum_{i \in S_t} L_{t,i}$ be the sum of the latencies of all the active sensors in the active sensor set S_t at slot t. If we can prove $V_t^{HLF} \leq V_t^{\pi_2} \; \forall t \in T$, it is sufficient to state that $O2_T^{\pi^*} = O2_T^{HLF} \leq O2_T^{\pi_2} \forall \pi_2 \in \prod$. Let say, the sample from active sensor $d^{(t)}$ is processed successfully at slot t. So at the end of slot t latency will be, $L_{t+1,i} = L_{t,i} + 1$ *if* $i \neq d^{(t)}$. At slot $t+1$ the set of active sensors $S_{t+1} = \left[S_t \backslash \left\{d^{(t)}\right\}\right] \cup N_{t+1}$ where N_{t+1} is the set of sensors that just become active at the beginning of slot $t+1$. So, latency of any sensor sample $n \in N_{t+1}$ be $L_{t+1,i} = L_{t+1,n} = 0$ and $L_{N_{t+1}} = \sum\limits_{n \in N_{t+1}} L_{t+1,n} = 0$. At slot $t+1$, $V_{t+1} = \sum\limits_{i \in S_{t+1}} L_{t+1,i}$ can be expressed as,

$$\begin{aligned} V_{t+1} &= V_t - L_{t,d^{(t)}} + \sum\nolimits_{i \in S_t} \mathbb{I}_t(i) + L_{N_{t+1}} \\ &= V_t - L_{t,d^{(t)}} + \sum\nolimits_{i \in S_t} \mathbb{I}_t(i) \end{aligned} \tag{38}$$

Here, $\mathbb{I}_t(.)$ is an indicator function which takes the value $\mathbb{I}_t(i) = 1$ *if* $i \neq d^{(t)}$ and $\mathbb{I}_t(i) = 0$ *if* $i = d^{(t)}$. Consider the same assumption as Lemma-II that the numbers of newly active sensors added at the beginning of each slot is independent of policy used. According to HLF, always $d^{HLF(t)} = j^{HLF(t)} = arg \max\limits_{i \in S_t^{HLF}} \left\{L_{t,i}^{HLF}\right\}$ is processed in slot t whereas in policy π_2 any element $d^{\pi_2(t)} \neq j^{\pi_2(t)} = arg \max\limits_{i \in S_t^{\pi_2}} \left\{L_{t,i}^{\pi_2}\right\}$ is processed in t.

Base case: As initial conditions $\overrightarrow{h_1(1)}$ and $\overrightarrow{c(0)}$ are same for both the policies, the set of active sensors $S_1^{HLF} = S_1^{\pi_2} = S_1$ is same for both HLF and π_2 at slot $t = 1$. So, $V_1^{HLF} = V_1^{\pi_1} = V_1$.

Inductive step: First it is assumed that for any slot $t \in T$, $V_t^{HLF} \leq V_t^{\pi_2}$. The expression V_{t+1}^{HLF} and $V_{t+1}^{\pi_2}$ can be obtained from eqn. (37) as follows,

$$V_{t+1}^{HLF} = V_t^{HLF} - L_{t,d^{HLF(t)}} + \sum\nolimits_{i \in S_t^{HLF}} \mathbb{I}_t(i) \tag{39}$$

$$V_{t+1}^{\pi_2} = V_t^{\pi_2} - L_{t,d^{\pi_2(t)}} + \sum\nolimits_{i \in S_t^{\pi_2}} \mathbb{I}_t(i) \tag{40}$$

From the remark-II drawn in Lemma-I, the total number of elements $R^{(t)}$ is same for both HLF and π_2 at any slot t, it can be said that $\sum_{i \in S_t^{HLF}} \mathbb{I}_t(i) = \sum_{i \in S_t^{\pi_2}} \mathbb{I}_t(i)$. So the comparison of V_{t+1}^{HLF} and $V_{t+1}^{\pi_2}$ depends on the first two terms of eqn. (38) and eqn. (39).

$$(V_{t+1}^{\pi_2} - V_{t+1}^{HLF}) = (V_t^{\pi_2} - V_t^{HLF}) + \left(L_{t,d^{HLF(t)}} - L_{t,d^{\pi_2(t)}}\right) \tag{41}$$

Now, $(V_t^{\pi_2} - V_t^{HLF}) = \gamma_t(\text{Say}) \geq 0$ from the assumption in the inductive step. To make $(V_{t+1}^{\pi_2} - V_{t+1}^{HLF}) = \gamma_{t+1} \geq 0$ either of the two following conditions must be satisfied.

$$\begin{aligned} &\textit{Condition 1:} \; \left(L_{t,d^{HLF(t)}} - L_{t,d^{\pi_2(t)}}\right) \geq 0 \\ &\textit{Condition 2:} \; -\gamma_t \leq \left(L_{t,d^{HLF(t)}} - L_{t,d^{\pi_2(t)}}\right) < 0 \Leftrightarrow \\ &\qquad 0 \leq \left(L_{t,d^{\pi_2(t)}} - L_{t,d^{HLF(t)}}\right) \leq \gamma_t \end{aligned}$$

Next, we arrange $S_t^{\pi_2}$ and S_t^{HLF} in decreasing order of latency to $S_t^{\prime\prime\pi_2}$ and $S_t^{\prime\prime HLF}$ respectively. Elements in $S_t^{\prime\prime\pi_2}$ and $S_t^{\prime\prime HLF}$ are represented by $\{g_{i'}\}_t^{\pi_2}$ and $\{g_{i'}\}_t^{HLF}$ respectively for $i' = 1 \ldots R^{(t)}$. From the result of Lemma-II, for any slot $t \in T$, $L_{t,i'}^{\pi_2} \geq L_{t,i'}^{HLF} \; \forall i' \in \{1 \ldots R^{(t)}\}$. Hence, $\{g_1\}_t^{\pi_2} \geq \{g_1\}_t^{HLF}$. As $d^{HLF(t)} = arg \max\limits_{i \in S_t^{HLF}} \left\{L_{t,i}^{HLF}\right\} = \{g_1\}_t^{HLF}$ and $d^{\pi_2(t)} = \{g_k\}_t^{\pi_2} \leq \{g_1\}_t^{\pi_2} = arg \max\limits_{i \in S_t^{\pi_2}} \left\{L_{t,i}^{\pi_2}\right\}$ two cases may happen.

Case i: $\{g_1\}_t^{HLF} \geq \{g_k\}_t^{\pi_2}$ *i.e.* $\left(L_{t,d^{HLF(t)}} - L_{t,d^{\pi_2(t)}}\right) \geq 0$. Then condition 1 is satisfied.

Case ii: $\{g_1\}_t^{HLF} \leq \{g_k\}_t^{\pi_2}$ *i.e.* $\left(L_{t,d^{HLF(t)}} - h_{t,d^{\pi_2(t)}}\right) < 0 \Leftrightarrow \left(L_{t,d^{\pi_2(t)}} - L_{t,d^{HLF(t)}}\right) > 0$. Now, if we can prove that $max \left(L_{t,d^{\pi_2(t)}} - L_{t,d^{HLF(t)}}\right) \leq \gamma_t$ then, our purpose will be fulfilled.

The elements present in $S_t^{\pi_2}$ and $S_t^{\prime\prime\pi_2}$ are same but in a different order. Similar thing happens between S_t^{HLF} and $S_t^{\prime\prime HLF}$. So, $V_t^{\pi_1} = \sum_{i \in S_t^{\pi_2}} L_{t,i} = \sum_{i' \in S_t^{\prime\prime\pi_2}} L_{t,i'}$ and $V_t^{HLF} = \sum_{i \in S_t^{HLF}} L_{t,i} = \sum_{i' \in S_t^{\prime\prime HLF}} L_{t,i'}$.

$$\begin{aligned} (V_t^{\pi_2} - V_t^{HLF}) &= \sum\nolimits_{i' \in S_t^{\prime\prime\pi_2}} L_{t,i'} - \sum\nolimits_{i' \in S_t^{\prime\prime HLF}} L_{t,i'} \\ &= \left[\{g_1\}_t^{\pi_2} - \{g_1\}_t^{HLF}\right] + \left[\{g_2\}_t^{\pi_2} - \{g_2\}_t^{HLF}\right] \\ &\quad + \ldots + \left[\left\{g_{R^{(t)}}\right\}_t^{\pi_2} - \left\{g_{R^{(t)}}\right\}_t^{HLF}\right] \end{aligned} \tag{42}$$

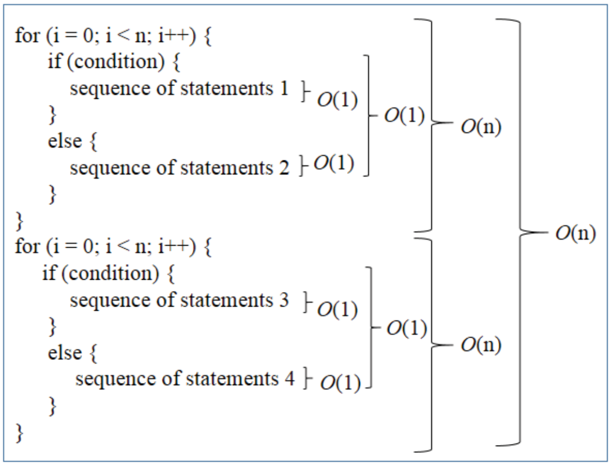


FIGURE 12. Time complexity calculation for HAF and HLF algorithms.

Let, $\left[\{g_{i'}\}_t^{\pi_2} - \{g_{i'}\}_t^{HLF}\right] = \beta_{i'} \geq 0$ (from Lemma-II) and $(V_t^{\pi_2} - V_t^{HLF}) = \gamma_t \geq 0$. Substituting $\beta_{i'}$ and γ_t in (42),

$$\gamma_t = \beta_1 + \beta_2 + \ldots + \beta_{R^{(t)}} \tag{43}$$

Now, the maximum possible value of $\left(L_{t,d^{\pi_2(t)}} - L_{t,d^{HLF(t)}}\right)$ is obtained when $d^{\pi_2(t)} = \{g_k\}_t^{\pi_2} = \{g_1\}_t^{\pi_2}$. $max\left(L_{t,d^{\pi_2(t)}} - L_{t,d^{HLF(t)}}\right) = \{g_1\}_t^{\pi_2} - \{g_1\}_t^{HLF} = \beta_1$. From (43) it can be seen that, $\beta_1 \leq \gamma_t$ that gives, $max\left(L_{t,d^{\pi_2(t)}} - L_{t,d^{HLF(t)}}\right) \leq \gamma_t$. Hence, *condition 2* is satisfied. Combining the results of *case i* and *case ii* we can conclude that $\left(L_{t,d^{HLF(t)}} - L_{t,d^{\pi_2(t)}}\right) \geq -\gamma_t$ that means, $(V_{t+1}^{\pi_2} - V_{t+1}^{HLF}) = \gamma_{t+1} \geq 0$ or $V_{t+1}^{HLF} \leq V_{t+1}^{\pi_2}$ for any slot $t \in T$ (**Induction complete**). This result shows that for an unreliable time-shared channel from the sensor to the processor in a symmetric IWSAN network, our proposed greedy scheduling policy HLF attains the minimum expected weighted sum latency as given in objective-II.

APPENDIX I
PROOF OF TIME COMPLEXITY OF HAF AND HLF

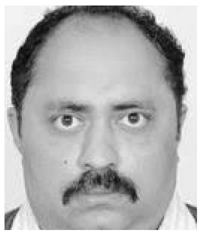
From algorithms 1 & 2 it can be said that the algorithms of both the HAF and HLF are in the form as given in Fig. 12. There are two in loops in a row. Each of the ‘‘for loops’’ consists of an ‘‘if-then-else’’ statement free from any kind of loop. Therefore, the worst-case runtime [39] of the algorithm will be, T(n) = n*O(1) + n*O(1) = O(n) + O(n) that is basically O(n) itself. This can also be written as O(max(n,n)) = O(n). Therefore, it is proved that for both the HAF and HLF algorithms, time complexity is O(n).

REFERENCES

[1] S. Kaul, R. Yates, and M. Gruteser, ‘‘Real-time status: How often should one update?’’ in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Orlando, FL, USA, Mar. 2012, pp. 2731–2735.
[2] I. Kadota, A. Sinha, and E. Modiano, ‘‘Scheduling policies for minimizing age of information in broadcast wireless networks,’’ *IEEE/ACM Trans. Netw.*, submitted for publication. Accessed: May 31, 2018. [Online]. Available: https://arxiv.org/abs/1801.01803
[3] 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.
[4] R. Rajkumar, I. Lee, L. Sha, and J. Stankovic, ‘‘Cyber-physical systems: The next computing revolution,’’ in *Proc. 47th ACM/IEEE Design Automat. Conf. (DAC)*, Anaheim, CA, USA, Jun. 2010, pp. 731–736.
[5] F. Pérez, E. Irisarri, D. Orive, M. Marcos, and E. Estevez, ‘‘A CPPS architecture approach for industry 4.0,’’ in *Proc. IEEE 20th Conf. Emerg. Technol. Factory Automat. (ETFA)*, Luxembourg, Sep. 2015, pp. 1–4.
[6] S. Wang, J. Wan, D. Li, and C. Zhang, ‘‘Implementing smart factory of industrie 4.0: An outlook,’’ *Int. J. Distrib. Sensor Netw.*, vol. 12, no. 1, pp. 1–10, Jan. 2016.
[7] 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.
[8] C. Lu, A. Saifullah, M. Sha, H. Gonzalez, D. Gunatilaka, C. Wu, L. Nie, and Y. Chen, ‘‘Real-time wireless sensor-actuator networks for industrial cyber-physical systems,’’ *Proc. IEEE*, vol. 104, no. 5, pp. 1013–1024, May 2016.
[9] J. Åkerberg, M. Gidlund, and M. Björkman, ‘‘Future research challenges in wireless sensor and actuator networks targeting industrial automation,’’ in *Proc. IEEE 9th Int. Conf. Ind. Inform. (INDIN)*, Lisbon, Portugal, Jul. 2011, pp. 410–415.
[10] M. Hermann, T. Pentek, and B. Otto, ‘‘Design principles for industrie 4.0 scenarios: A literature review,’’ Techn. Univ. Dortmund-Audi Stiftungslehrstuhl Supply Net Order Manage., Working Paper 01, 2015. Accessed: Jun. 19, 2017.
[11] S. K. Kaul, R. D. Yates, and M. Gruteser, ‘‘Status updates through queues,’’ in *Proc. 46th Annu. Conf. Inf. Sci. Syst. (CISS)*, Princeton, NJ, USA, Mar. 2012, pp. 1–6.
[12] R. D. Yates and S. Kaul, ‘‘Real-time status updating: Multiple sources,’’ in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Cambridge, MA, USA, Jul. 2012, pp. 2666–2670.
[13] C. Kam, S. Kompella, and A. Ephremides, ‘‘Age of information under random updates,’’ in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Istanbul, Turkey, Jul. 2013, pp. 66–70.
[14] C. Kam, S. Kompella, and A. Ephremides, ‘‘Effect of message transmission diversity on status age,’’ in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Honolulu, HI, USA, Jun./Jul. 2014, pp. 2411–2415.
[15] M. Costa, M. Codreanu, and A. Ephremides, ‘‘Age of information with packet management,’’ in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Honolulu, HI, USA, Jun./Jul. 2014, pp. 1583—1587.
[16] N. Pappas, J. Gunnarsson, L. Kratz, M. Kountourisy, and V. Angelakis, ‘‘Age of information of multiple sources with queue management,’’ in *Proc. IEEE Int. Conf. Commun. (ICC)*, London, U.K., Jun. 2015, pp. 5935–5940.
[17] L. Huang and E. Modiano, ‘‘Optimizing age-of-information in a multiclass queueing system,’’ in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Hong Kong, Jun. 2015, pp. 1681–1685.
[18] R. D. Yates, ‘‘Lazy is timely: Status updates by an energy harvesting source,’’ in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Hong Kong, Jun. 2015, pp. 3008–3012.
[19] Y. Sun, E. Uysal-Biyikoglu, R. D. Yates, C. E. Koksal, and N. B. Shroff, ‘‘Update or wait: How to keep your data fresh,’’ *IEEE Trans. Inf. Theory*, vol. 63, no. 11, pp. 7492–7508, Nov. 2017.
[20] K. Chen and L. Huang, ‘‘Age-of-information in the presence of error,’’ in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Barcelona, Spain, Jul. 2016, pp. 2579–2583.
[21] S. K. Kaul and R. D. Yates, ‘‘Status updates over unreliable multiaccess channels,’’ in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Aachen, Germany, Jun. 2017, pp. 331–335.
[22] A. M. Bedewy, Y. Sun, and N. B. Shroff, ‘‘Age-optimal information updates in multihop networks,’’ in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Aachen, Germany, Jun. 2017, pp. 576–580.
[23] B. T. Bacinoglu and E. Uysal-Biyikoglu, ‘‘Scheduling status updates to minimize age of information with an energy harvesting sensor,’’ in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Aachen, Germany, Jun. 2017, pp. 1122–1126.
[24] Q. He, D. Yuan, and A. Ephremides, ‘‘On optimal link scheduling with min-max peak age of information in wireless systems,’’ in *Proc. IEEE Int. Conf. Commun. (ICC)*, Kuala Lumpur, Malaysia, May 2016, pp. 1–7.
[25] 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.

[26] Q. He, D. Yuan, and A. Ephremides, "Optimizing freshness of information: On minimum age link scheduling in wireless systems," in *Proc. 14th IEEE Int. Symp. Modeling Optim. Mobile, Ad Hoc, Wireless Netw. (WiOpt)*, Tempe, AZ, USA, May 2016, pp. 1–8.
[27] 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.
[28] Y. P. Hsu, E. Modiano, and L. Duan, "Scheduling algorithms for minimizing age of information in wireless broadcast networks with random arrivals: The no-buffer case," 2017, *arXiv:1712.07419*. Accessed: May 31, 2018. [Online]. Available: https://arxiv.org/abs/1712.07419
[29] Y. P. Hsu, "Age of information: Whittle index for scheduling stochastic arrivals," Accessed: May 31, 2018, *arXiv:1801.03422*. [Online]. Available: https://arxiv.org/abs/1801.03422
[30] Z. Wang, X. Qin, B. Liu, and P. Zhang, "Joint data sampling and link scheduling for age minimization in multihop cyber-physical systems," *IEEE Wireless Commun. Lett.*, to be published. doi: 10.1109/LWC.2019.2891598.
[31] C. Joo and A. Eryilmaz, "Wireless scheduling for information freshness and synchrony: Drift-based design and heavy-traffic analysis," in *Proc. 15th IEEE Int. Symp. Modeling Optim. Mobile, Ad Hoc, Wireless Netw. (WiOpt)*, Paris, France, May 2017, pp. 1–8.
[32] S. Han, K. Y. Lam, D. Chen, M. Xiong, J. Wang, K. Ramamritham, and A. K. Mok, "Online mode switch algorithms for maintaining data freshness in dynamic cyber-physical systems," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 3, pp. 756–769, Mar. 2016.
[33] G. Li, C. Zhao, J. Li, and B. Guo, "Maintaining data freshness in distributed cyber-physical systems," *IEEE Trans. Comput.*, to be published. doi: 10.1109/TC.2018.2889456.
[34] J. Kurose and K. Ross, *Computer Networking: A Top Down Approach Using the Internet*. 6th ed. London, U.K.: Pearson, 2013.
[35] P. Serrano, D. Lerrabeiti, and Á. León, "Tunable least served first: A new scheduling algorithm with tunable fairness," in *Proc. Netw. Appl. Towards Ubiquitously Connected World, IFIP Int. Workshop Netw. Appl. (EUNICE)*, Madrid, Spain, Jul. 2005, pp. 1–11.
[36] M. Mitzenmacher, "The power of two choices in randomized load balancing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 12, no. 10, pp. 1094–1104, Oct. 2001.
[37] J. Åkerberg, M. Gidlund, J. Neander, T. Lennvall, and M. Björkman, "Deterministic downlink transmission in wirelesshart networks enabling wireless control applications," in *Proc. 36th Annu. Conf. IEEE Ind. Electron. Soc. (IECON)*, Glendale, AZ, USA, Nov. 2010, pp. 2120–2125.
[38] Z. Lieu, P. Nain, and D. Towsley, "Sample path methods in the control of queues," *Queuing Syst.*, vol. 21, pp. 293–335, Sep. 1995.
[39] F. Carrano and W. Savitch, *Data Structures and Abstractions With Java*. Upper Saddle River, NJ, USA: Prentice-Hall, 2003.

DEVARPITA SINHA received the B.Tech. degree in electronics and communication engineering from the West Bengal University of Technology, Kolkata, India, in 2014, and the M.E. degree with first class with Distinction in electronics and communication engineering (with the specialization in wireless communication) from the Birla Institute of Technology (BIT), Ranchi, in 2016. She is currently pursuing the Ph.D. degree with the Department of Electronics and Electrical Communication Engineering, IIT Kharagpur, India. Her research interests include resource allocation problems, performance evaluation and control in communication networks, and scheduling and decision making.



RAJARSHI ROY received the B.E. degree with first class (Hons.) in electronics and telecommunication engineering from Jadavpur University, Kolkata, India, in 1992, the M.Sc. (engineering) degree in electrical communication engineering from IISc, Bengaluru, India, in 1995, and the Ph.D. degree in electrical engineering from Polytechnic University, Brooklyn, NY, USA (Currently known as New York University, Tandon School of Engineering), in 2001. He is currently an Associate Professor with the Department of Electronics and Electrical Communication Engineering, IIT Kharagpur, India. His research interests include queuing theory, Markov decision theory, 5G/6G wireless communications, complex communication networks, MIMO, network coding, cooperative communication, cyber-physical systems, and cognitive technology.