Computing
https://doi.org/10.1007/s00607-024-01278-x

REGULAR PAPER

Enhancing information freshness in multi-class mobile edge computing systems using a hybrid discipline

Tamer E. Fahim[1,2] · Sherif I. Rabia[2] · Ahmed H. Abd El-Malek[3] · Waheed K. Zahra[1,4]

Received: 26 September 2023 / Accepted: 21 February 2024

Abstract

Timely status updating in mobile edge computing (MEC) systems has recently gained the utmost interest in internet of things (IoT) networks, where status updates may need higher computations to be interpreted. Moreover, in real-life situations, the status update streams may also be of different priority classes according to their importance and timeliness constraints. The classical disciplines used for priority service differentiation, preemptive and non-preemptive disciplines, pose a dilemma of information freshness dissatisfaction for the whole priority network. This work proposes a hybrid preemptive/non-preemptive discipline under an M/M/1/2 priority queueing model to regulate the priority-based contention of the status update streams in MEC systems. For this hybrid discipline, a probabilistic discretionary rule for preemption is deployed to govern the server and buffer access independently, introducing distinct probability parameters to control the system performance. The stochastic hybrid system approach is utilized to analyze the average age of information (AoI) along with its higher moments for any number of classes. Then, a numerical study on a three-class network is conducted by evaluating the average AoI performance and the corresponding dispersion. The numerical observations underpin the significance of the hybrid-discipline parameters in ensuring the reliability of the whole priority network. Hence, four different approaches are introduced to demonstrate the setting of these parameters. Under these approaches, some outstanding features are manifested: exploiting the buffering resources efficiently, conserving the aggregate sensing power, and optimizing the whole network satisfaction. For this last feature, a near-optimal low-complex heuristic method is proposed.

Keywords Mobile edge computing · Status update system · Age of information · Priority queueing model · Stochastic hybrid system

Mathematics Subject Classification 60K25 · 60K30

Extended author information available on the last page of the article

Published online: 19 March 2024

1 Introduction

Currently, the pervasive deployment of the internet of things (IoT) applications sheds light on the significance of the real-time status update system. In such system, the status updates of a physical phenomenon evolving at the transmitter end should be promptly conveyed to the beneficiaries for monitoring and control purposes [1]. In this regard, there are a host of real-life applications, to name a few, reporting the vehicle's position in autonomous vehicular technology [2] and reporting soil properties in smart agricultural systems [3]. The main objective of all such applications is to enhance the information freshness (timeliness) due to the insignificance and the unreliability of the outdated updates [4]. However, in different IoT applications, there are computationally intensive status updates. For instance, in autonomous driving and augmented reality technologies, higher computations are consumed in image processing and voice recognition for each status update packet before being interpreted as an update for monitoring and control purposes. For such applications, depending on the local processor of the end user (having limited computation capacity) or offloading the computations to the central cloud (being far from the end user) contributes to information staleness. As a result, mobile edge computing (MEC) system has emerged [5] as a promising paradigm for information freshness enhancement and low-latency transmission. In such a system, the edge server, with higher computation power than the local processor, handles the intensive computations in proximity to the end user.

The information freshness is to be quantified using the age of information (AoI) metric, which was introduced for the first time in [1]. The AoI, denoted as $\Delta(t)$, is defined as the time span since the generation time instant $u(t)$ related to the freshest received packet at the monitor [6]. Hence, it is formulated as $\Delta(t) = t - u(t)$, and it should be as minimum as possible to ensure timely status updates. In this regard, it has been proved that minimizing the AoI is a distinct problem, which differs from minimizing/maximizing information delay/throughput [7]. There are a host of research works addressing the AoI performance of the real-time status update system, in general, and under the MEC systems, in particular.

For the single-source systems, the authors in [1] initiated the mathematical manipulation of the AoI metric using the queueing abstractions of M/M/1, M/D/1 and D/M/1 under the the first come first served (FCFS) queueing policy. This work was extended, in [8], by the deployment of M/M/1/1*, where the ($*$) is a notation for the preemption in service (PR$^{(s)}$) feature. The newly arrived packet can preempt the ongoing service promptly without the need of a waiting buffer; hence, this model is denoted in our context as PR$^{(s)}$-Bufferless. In [6], the M/M/1/2* queueing model was proposed, where the ($*$) notation here is used to indicate that the preemption is allowed only within the waiting buffer (PR$^{(w)}$) under a non-preemptive service discipline (NP$^{(s)}$). Hence, this model is denoted in our context as NP$^{(s)}$-PR$^{(w)}$. In [9], the authors proved the optimality of the PR$^{(s)}$ -Bufferless approach in the case of a single-source update system with a memoryless server. Since the computational aspects of the computationally intensive messages have not been considered in the aforementioned research work, the study in [10] is the first to analyze the AoI under a single-source MEC system, where an M/M/1 queueing model with FCFS is deployed to model the remote computing scheme. This work was then extended by reusing the M/M/1/2* queueing model for the computing stage [11].

Regarding the multi-source systems, the authors in [12] extended the aforementioned schemes, PR$^{(s)}$-Bufferless and NP$^{(s)}$-PR$^{(w)}$, to be employed under the multiple-source case. In this work, the stochastic hybrid system approach (SHS) is deployed for the first time in the AoI context to formulate the average AoI for the case of memoryless servers. The SHS approach was further generalized, in [13], by analyzing the higher order AoI moments along with the moment generating function (MGF). The AoI performance of the multi-source MEC systems has also been studied in the literature. In [14], the AoI performance of three different computing scenarios was addressed using a FCFS M/M/1 queueing abstraction, which models both the transmission and computing stages.

The majority of the research studies pertaining to multi-source systems assume the same AoI requirements for all status update streams, employing the same service treatment. However, the more realistic scenario is to consider a distinct priority for each stream according to its importance and timeliness sensitivity. To illustrate, in Vehicle-to-Everything (V2X) technology, the status-update streams may be classified into three priority classes [15]: the highest priority is assigned for road safety data (e.g., vehicle's speed); the intermediate priority is assigned for traffic management data (e.g., vehicle's destination); and the lowest priority is dedicated for convenience and entertainment data (e.g., air-conditioning control). Considering the MEC integrated systems, there is also a dire need to consider different priority classes to reflect the urgency in task execution [16].

There is limited AoI research work considering the priority setting for the multi-source status update systems. It can be classified according to the employed service discipline: PR$^{(s)}$ [15, 17–19] and NP$^{(s)}$ [17, 20, 21]. The notion of PR$^{(s)}$ under the priority setting means that the lower priority (LP) class can be preempted either from a higher priority (HP) one or from the same priority class. For the PR$^{(s)}$ priority schemes, the authors in [17] reanalyzed the M/M/1/1* queueing abstraction (proposed in [8, 12]) under the case of multiple prioritized sources. In [18], the previous work was extended by proposing an M/M/1 priority queueing model with separate one-sized buffer for each class under the PR$^{(s)}$ scheme; hence, it can be referred to as PR$^{(s)}$-Multi-buffer in our context. However, in [19], the content-based buffering scheme was proposed in a two-class M/G/1 queueing system. In such a scheme, each class has its own buffering mechanism: the bufferless scheme for the HP class and the infinite buffer scheme under the FCFS policy for the LP class. The buffering mechanism was further generalized in [15] for any number of classes, where each class has its own buffer (finite or infinite), and the Lexicographic optimality approach is used to manage the scheduling of multiple-access requests for a single memoryless server. As regards to the NP$^{(s)}$ priority schemes, the authors in [17] re-studied the M/M/1/2* queueing model (proposed in [6, 12]) under the priority setting. This work was extended in [21], where the buffered packet experiences a deterministic deadline, beyond which it will be dropped. The buffering mechanism was generalized in the work of [20] by considering separate one-sized buffers and separate infinite buffers for each class.

For the aforementioned priority-based schemes, the common feature is that either the $\mathrm{PR}^{(\mathrm{s})}$ or $\mathrm{NP}^{(\mathrm{s})}$ is employed. However, in the context of the general priority queueing system [22], it is stipulated that neither the preemptive nor the non-preemptive priority disciplines can fulfill the satisfaction for all priority classes. To clarify, the $\mathrm{PR}^{(\mathrm{s})}$ is very harsh for the LP classes due to the frequent interruptions. On the other hand, in the $\mathrm{NP}^{(\mathrm{s})}$, the HP classes are dissatisfied with being hindered by the ongoing service of the LP classes. To counter this paradox, the hybrid preemptive/non-preemptive service discipline has been introduced [22]. In this discipline, the server has the power to control the preemption decision using a discretionary rule. In [23], four different variations of this rule are mentioned.

As far as we know, no research work has considered the hybrid preemptive/non-preemptive discipline for the prioritized status-update system under the analytical framework of the AoI, except our first attempt in [24]. Moreover, there is no AoI-based research work pertaining to the status update systems working under multi-source MEC with prioritized update streams. Based on these motivations, in the current work, the hybrid preemptive/non-preemptive discipline is further proposed to manage the contention between multi-priority sources requesting the intensive computations of their status updates at the edge server in an IoT-enabled MEC environment. The edge computing scheme is considered where all computational tasks are offloaded to the edge server for processing. The computing stage at the edge server is modelled as an M/M/1/2 priority queueing model with a shared buffer of size one. Here, two hybrid disciplines are deployed to govern the interaction between priority classes within the shared server and the shared buffer. For these hybrid disciplines, the probabilistic preemption approach (as a discretionary approach for preemption) is utilized. That is, upon the occurrence of a service (resp. buffer) request, the existing served (resp. buffered) packet is given a probabilistic decision either to be preempted or not. The probabilistic preemption in service ($\mathrm{Prob}^{(\mathrm{s})}$) and the probabilistic preemption in waiting ($\mathrm{Prob}^{(\mathrm{w})}$) are governed by distinct probability parameters to ensure the generality of the system. It is worth mentioning that the probabilistic preemption approach has not been extensively addressed in the AoI literature. In [25] and [26], the $\mathrm{NP}^{(\mathrm{s})}$-$\mathrm{Prob}^{(\mathrm{w})}$ was proposed for a single-source system, whereas the $\mathrm{Prob}^{(\mathrm{s})}$-Bufferless was introduced in [27] for multiple unprioritized streams. However, in our previous work [24], the $\mathrm{Prob}^{(\mathrm{s})}$-Bufferless is reused for multiple prioritized streams. To the best of our knowledge, the proposed combined policy (denoted as $\mathrm{Prob}^{(\mathrm{s})}$-$\mathrm{Prob}^{(\mathrm{w})}$) has never been addressed in the AoI context. The SHS approach is employed to analyze the average AoI, along with other higher order moments for any number of priority classes. Subsequently, a numerical study on a three-class network is presented to highlight the effect of the proposed $\mathrm{Prob}^{(\mathrm{s})}$-$\mathrm{Prob}^{(\mathrm{w})}$ policy on the AoI performance, compared with the classical priority-based models [17, 18] besides our previous work [24]. Moreover, four different approaches are introduced to clarify the setting procedure of the proposed preemption probability parameters (controlling parameters). It is revealed that our proposed model, unlike the classical ones, can achieve the satisfaction of the whole priority classes by a thorough adjustment of the controlling parameters. Moreover, compared with our previous work $\mathrm{Prob}^{(\mathrm{s})}$-Bufferless [24], the buffer functionalities under the proposed $\mathrm{Prob}^{(\mathrm{w})}$ policy achieve more promising results.

Based on the foregoing, the main contributions of this work can be summarized as follows:

- The AoI performance of the multi-priority class status update system is addressed under an IoT-enabled MEC environment. To the best of our knowledge, it is the first time to investigate the information freshness performance of the MEC systems with *different priority* tasks.
- The idea of the hybrid preemptive/non-preemptive discipline is proposed under an M/M/1/2 priority queueing model, with the AoI metric being analyzed and investigated. As far as we know, this *hybrid discipline* has never been addressed for the case of multiple prioritized sources under the AoI analytical framework [15, 17–21], except our initial attempt [24].
- The probabilistic preemption approach, as a discretionary priority discipline, is employed at both the server and the buffer independently through the *combined policy* Prob$^{(s)}$-Prob$^{(w)}$. To the best of our knowledge, it is the first time to consider such a combined policy in the AoI analysis.
- The higher order AoI moments, besides the average performance, are analyzed and investigated. Throughout the AoI literature under the priority setting [15, 17–21], no research work has considered these further distributional AoI characteristics.
- Four different approaches are presented to address the controlling parameters' setting by pursuing distinct system objectives. Moreover, a low-complex heuristic generation approach is suggested, yielding a near-optimal performance of the whole network satisfaction.

In the sequel, our work is presented as follows. Section 2 demonstrates the proposed system description and the traffic parameters setting. The AoI analysis is presented in Sect. 3, which is initiated by a brief preliminary on the SHS approach. The numerical study is then presented in Sect. 4 for performance evaluation. Eventually, a summary of the current work and our perception of the envisaged work are presented in Sect. 5.

2 System model

2.1 System model description and assumptions

As depicted in Fig. 1, a typical multi-class IoT-enabled MEC system is considered. Each of M possible sources holds a status update stream with a distinct priority class, where S_1 and S_M are the highest and the lowest priority classes, respectively[1]. These sources contend to transmit their status updates for processing, monitoring and control purposes at the edge base station (BS), which incorporates the MEC server as a processing unit with higher computational power. The detailed description of the

[1] Table 1 summarizes all symbol notations used throughout the article with the order of their mention.

Table 1 Symbol notations used throughout the article

Symbol	Definition
M	Number of priority classes
S_m	A source of a status update stream with class m $(1 \leq m \leq M)$
$p^s_{m,n}$	The probability that a served packet of class n is preempted by class-m packet
$p^w_{m,n}$	The probability that a buffered packet of class n is preempted by class-m packet
λ_m	Class-m average arrival rate
$\hat{\lambda}_m$	Aggregate arrival rate of the HP classes of class m
$\check{\lambda}_m$	Aggregate arrival rate of the LP classes of class m
λ_{total}	Total arrival rate
μ_m	Class-m average service rate
ρ_m	Class-m offered load
ρ_{total}	Total offered load from all classes
$\mathrm{E}[\Delta^k_h]$	kth AoI moment for class h
$\mathbf{P}^s$	Preemption probability matrix containing $p^s_{m,n}$ for $1 \leq m, n \leq M$
$\mathbf{P}^w$	Preemption probability matrix containing $p^w_{m,n}$ for $1 \leq m, n \leq M$
$q(t)$	The system-state discrete component modelling the system occupancy
Q	The discrete state space
$\mathbf{x}(t)$	The system-state continuous component including all AoI processes of interest
$x_0(t)$	The AoI process at the monitor (after successful departure)
$x_j(t)$	The AoI process within the system: $x_1(t)$, AoI at server; and $x_2(t)$, AoI at the buffer
L	All transitions involved in the CTMC $\{q(t)\}$
q_l	The source state of the discrete state transition $l \in L$
q'_l	The destination state of the discrete state transition $l \in L$
$\lambda^{(l)}$	The transition rate of the discrete state transition $l \in L$
$\delta_{q_l,q(t)}$	The Kronecker delta function: $\delta_{q_l,q(t)} = 1$ when $q(t) = q_l$; and $\delta_{q_l,q(t)} = 0$, otherwise
L'_q	All CTMC transitions ending up with state q
L_q	All CTMC originating from state q
$\mathbf{A}_l$	The AoI reset map matrix upon the discrete state transition l
$\mathbf{x}'(t)$	The AoI processes after being reset upon each discrete state transition
$\dot{\mathbf{x}}$	The AoI rate of change
$\pi_q(t)$	The discrete-state probability that $q(t) = q$
$v^{(k)}_{qj}(t)$	The k-th moment correlation value of the AoI process $x_j(t)$ at state $q(t) = q$
$\mathbf{v}^{(k)}_q(t)$	The k-th moment correlation vector at state $q(t) = q$
$\bar{\pi}_q$	The stationary state probability that $q(t) = q$ as $t \to \infty$
$\bar{\mathbf{v}}^{(k)}_q$	The k-th moment correlation vector at state $q(t) = q$ as $t \to \infty$
$\mathbf{P}$	The priority preemption probabilities for $M = 3$, $\mathbf{P} = [p^s_{1,2}, p^s_{1,3}, p^s_{2,3}, p^w_{1,2}, p^w_{1,3}, p^w_{2,3}]$
$\mathbf{P}_{\text{self}}$	The self-preemption probabilities for $M = 3$, $\mathbf{P}_{\text{self}} = [p^s_{1,1}, p^s_{2,2}, p^s_{3,3}, p^w_{1,1}, p^w_{2,2}, p^w_{3,3}]$
$\Delta^{\min}_1$	The optimal average AoI for class 1 (the highest priority class)
CV_m	The coefficient of variation of class-m AoI performance
β_2, β_3	The relative importance for classes 2 and 3, respectively, with respect to class 1
λ^*_{total}	The optimal aggregate traffic intensity reflecting the total sensing power needed
C_{α_2,α_3}	The cost function representing the whole network satisfaction for $M = 3$

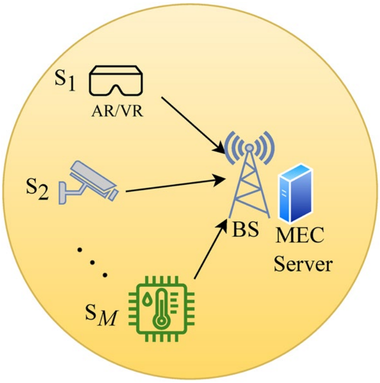


Fig. 1 The proposed multi-class IoT-enabled MEC system

proposed system is demonstrated in Fig. 2. The system is equipped with a waiting buffer of size one to permit the waiting for a busy server. The received status update packet, only after being processed at the server, is converted to readable information to the monitor. Both the server and buffer are managed by the *service policy* and *queueing policy*, respectively, which will be explained shortly. A control unit is provided to deploy the service and queueing policies in governing the multiple access requests at the base station.

The service policy. It controls the access of the newly arrived entity to a busy server. In this regard, the proposed $\mathrm{Prob}^{(\mathrm{s})}$ policy is deployed to resolve the contention between the arrived entity $\mathrm{E_a}$ (class $\mathrm{c_a}$) and the served entity $\mathrm{E_s}$ (class $\mathrm{c_s}$). This policy stipulates that upon the request from $\mathrm{E_a}$, a probabilistic decision is taken, either to preempt $\mathrm{E_s}$ or not, according to the preemption probability parameter $p^{s}_{\mathrm{c_a},\mathrm{c_s}} \in [0, 1]$. The cases of $p^{s}_{\mathrm{c_a},\mathrm{c_s}} = 0$ and $p^{s}_{\mathrm{c_a},\mathrm{c_s}} = 1$ demonstrate the strict non-preemption and the strict preemption cases, respectively. However, the priority setting between status update streams entails that $p^{s}_{\mathrm{c_a},\mathrm{c_s}} = 0$ for $\mathrm{c_a} > \mathrm{c_s}$, where the LP class cannot preempt the ongoing service of the HP one. Furthermore, a *protection feature* is assumed to be deployed alongside the $\mathrm{Prob}^{(\mathrm{s})}$ policy for the sake of the LP classes due to the expected frequent interruptions from the HP ones, especially under the higher traffic loading conditions. Under this assumption, whenever a LP class being served is given the non-preemption decision upon the request from the HP class, it becomes protected from further interruptions from all HP classes until its service completion. The significance of this protection feature will be explained subsequently in the numerical study.

The *queueing policy*. It regulates the access to the waiting buffer. Let $\mathrm{E_{ab}}$ (class $\mathrm{c_{ab}}$) denote the entity requesting the buffer. This request may originate from a newly arrived entity that cannot preempt the served entity or from a preempted served entity. The $\mathrm{Prob}^{(\mathrm{w})}$ policy, independent from the $\mathrm{Prob}^{(\mathrm{s})}$ policy, is proposed to control the preemption possibility of the buffered entity $\mathrm{E_b}$ (class $\mathrm{c_b}$), according to a distinct preemption probability parameter $p^{w}_{\mathrm{c_{ab}},\mathrm{c_b}}$. It is also assumed that $p^{w}_{\mathrm{c_{ab}},\mathrm{c_b}} = 0$, for $\mathrm{c_{ab}} > \mathrm{c_b}$, to reflect the priority setting; however, the protection feature is assumed not to be deployed with the queueing policy. Moreover, it is assumed that at most

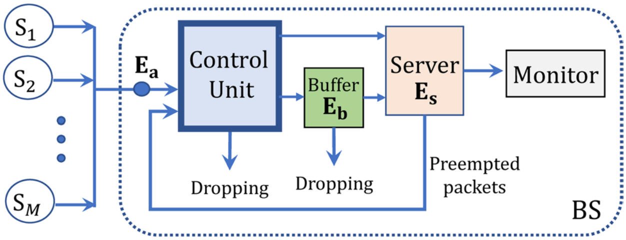


Fig. 2 The detailed system description

one packet related to each class is allowed to exist on the whole system (either in the server or the buffer); hence, $c_b \neq c_s$.

Based on the foregoing, the control unit functionality can be described in the flowchart shown in Fig. 3. The server status (busy/empty) is firstly checked by the control unit to manage the access possibility of E_a. For a busy server, the service preemption possibility is then examined between the entity E_a and the served one E_s. This phase is governed by the $\text{Prob}^{(s)}$ policy and the consideration of the protection feature. If the preemption is permitted, the served packet being preempted will request the buffer ($E_{ab} \leftarrow E_s$), leaving the entity E_a to join the server ($E_s \leftarrow E_a$). On the other hand, the non-preemption decision forces the entity E_a to request the buffer ($E_{ab} \leftarrow E_a$). Regarding the buffer request E_{ab} (if exists), class index duplication check is firstly examined between E_{ab} and E_s ($c_{ab} \overset{?}{=} c_s$). If $c_{ab} = c_s$, the entity E_{ab} will be dropped according to the aforementioned system assumption that $c_b \neq c_s$. Otherwise, the entity E_{ab} joins the empty buffer immediately ($E_b \leftarrow E_{ab}$). If the buffer is busy, the buffer preemption possibility is then checked by employing the $\text{Prob}^{(w)}$ policy. If the preemption is admitted, the buffered entity will be dropped, leaving its place to the entity E_{ab} (i.e., $E_b \leftarrow E_{ab}$). However, upon the non-preemption decision, the entity E_{ab} will be dropped.

For the benefit of our reader, a further demonstration of the proposed system dynamics is presented in Appendix 1 using a working example on three priority classes.

2.2 Traffic parameters

For mathematical manipulation, it is assumed that the arrival process of each prioritized stream m ($1 \leq m \leq M$) follows the Poisson process with rate λ_m. Therefore, for each class m, the total arrival rate of its HP classes and its LP classes can be denoted as $\hat{\lambda}_m = \sum_{i=1}^{m-1} \lambda_i$ and $\check{\lambda}_m = \sum_{i=m+1}^{M} \lambda_i$, respectively. Hence, the total arrival rate can be represented as $\lambda_{\text{total}} = \hat{\lambda}_m + \lambda_m + \check{\lambda}_m$. The exponential service time distribution is also assumed for all priority classes with a service rate μ_m. Accordingly, the total offered load from the whole network is $\rho_{\text{total}} = \sum_{i=1}^{M} \lambda_i/\mu_i$.

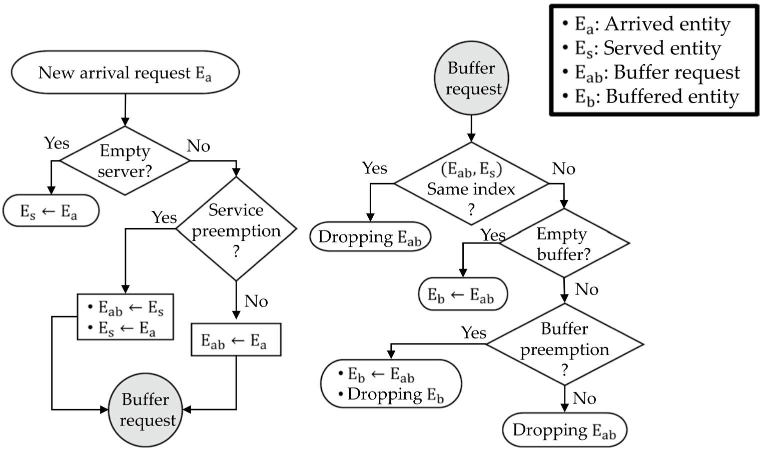


Fig. 3 Flowchart illustrating the control unit functionality for each arrival reques

3 Performance analysis

The aim is to evaluate the kth AoI moment for each class h ($E[\Delta_h^k]$, $k \geq 1$, $1 \leq h \leq M$). Based on the system model description in Sect. 2, the proposed system can be mathematically considered as a modified M/M/1/2 priority queueing system due to the proposed Prob$^{(s)}$-Prob$^{(w)}$ policy; hence, it can be refereed to as M/M/1/2*/($\mathbf{P}^s, \mathbf{P}^w$). The matrix $\mathbf{P}^s$ (resp. $\mathbf{P}^w$) contains all preemption probability parameters $[p_{i,j}^s]$ (resp. $[p_{i,j}^w]$) deployed by the Prob$^{(s)}$ (resp. Prob$^{(w)}$) policy. It should be noted that both $\mathbf{P}^s, \mathbf{P}^w$ are upper triangular matrices to obey the priority setting between classes as mentioned in Sect. 2.

For the proposed queueing system abstraction M/M/1/2*/($\mathbf{P}^s$,$\mathbf{P}^w$), due to its finite-state nature (capacity of two packets) and the assumption of memoryless server, the stochastic hybrid system (SHS) approach is more tractable to be deployed in such case to analyze the AoI performance [12].

Accordingly, this section is organized as follows. A brief preliminary on the AoI-oriented SHS approach is firstly introduced in Sect. 3.1. Then, the detailed SHS analysis of our proposed model is presented in Sect. 3.2, ending up with an algorithm that can be used to extract the AoI moments for the proposed model under a general number of priority classes. After that, Sect. 3.2.1 considers the case of a two-class network, through which closed-form results of the average AoI are provided.

3.1 Preliminary on the AoI-oriented SHS approach

The SHS is a randomly-evolved system, whose state is a hybrid of both continuous and discrete components. In the AoI context, the system behaviour can be described by the hybrid state $(q(t), \mathbf{x}(t))$. The discrete component $q(t) \in Q = \{0, 1, ..., m\}$ signifies all possible states reflecting the system occupancy upon some stochastic events, such as packet arrival or service completion. On the other hand, the continuous component $\mathbf{x}(t) = [x_0(t), x_1(t), ..., x_n(t)] \in \mathbb{R}^{n+1}$ represents the continuous-time growing of the AoI signals measured for a certain source stream. The signal $x_0(t)$ measures the AoI as perceived at the monitor (after successful service completion), while the signals $x_1(t), ..., x_n(t)$ are the corresponding measures as seen by some virtual monitors distributed within the system to track the packet AoI process from its arrival and before its successful departure. For instance, these virtual monitors can be located at the server or at a fixed queue position. Next, the mathematical model related to the evolution of the hybrid state is presented.

First, in the case of a memoryless server, the continuous-time Markov chain (CTMC) $\{q(t)\}$ can be used to model the evolution of the discrete states $q(t)$ with a set of transitions L. Upon each transition $l \in L$, the discrete state will be changed from q_l to q'_l with a transition rate $\lambda^{(l)}\delta_{q_l,q(t)}$. The Kronecker delta function is used here to make the transition rate $\lambda^{(l)}$ pertaining to the source state $q(t) = q_l$ upon the l-th transition. In this regard, $L'_q = \{l \in L q'_l = q\}$ and $L_q = \{l \in L q_l = q\}$ are referred to as the incoming and the outgoing transitions of the state q.

As regards to the continuous states $\mathbf{x}(t)$, their progression is to be modeled in two occasions. Firstly, upon each discrete transition l, $\mathbf{x}(t)$ encounters a resetting: $\mathbf{x}$ resets to $\mathbf{x}' = \mathbf{x}\mathbf{A}_l$. The matrix $\mathbf{A}_l \in \{0, 1\}^{(n+1)\times(n+1)}$ is known as the reset maps matrix. Secondly, during the holding time at each state $q(t)$, $\mathbf{x}(t)$ increases linearly ($\dot{\mathbf{x}} = \mathbf{1}_{n+1}$), abiding by the notion of piecewise linear SHS [12].

Having modelled the SHS behaviour, let declare the following notions for each state $q \in Q$: the state probability, $\pi_q(t) = \mathrm{P}[q(t) = q] = \mathrm{E}[\delta_{q,q(t)}]$; the kth-moment correlation, $v^{(k)}_{q,j}(t) = \mathrm{E}[x^k_j(t)\delta_{q,q(t)}]$, $0 \le j \le n$, $k \ge 0$. For all $0 \le j \le n$, all corresponding correlations can be combined in the k-th moment correlation vector, $\mathbf{v}^{(k)}_q(t) = [v^{(k)}_{q,0}(t), ..., v^{(k)}_{q,n}(t)] = \mathrm{E}[\mathbf{x}^k(t)\delta_{q,q(t)}]$. It should be noted that $\mathbf{v}^{(0)}_q(t) = \mathbf{1}_{n+1}\pi_q(t)$. Hence, using the law of total average, the kth moment of the process $\mathbf{x}^k(t)$ can be deduced as

$$\mathrm{E}[\mathbf{x}^k(t)] = \sum_{q\in Q} \mathrm{E}[\mathbf{x}^k(t)|q(t) = q]\mathrm{P}[q(t) = q] = \sum_{q\in Q} \mathrm{E}[\mathbf{x}^k(t)\delta_{q,q(t)}] = \sum_{q\in Q} \mathbf{v}^{(k)}_q(t). \quad (1)$$

After declaring the foregoing quantities, the SHS analysis starts by evaluating the state probabilities $\pi_q(t)$ ($\forall q \in Q$). Since the ergodicity of the CTMC $\{q(t)\}$ is a substantial assumption in the AoI analysis [12, 13], the state probability $\pi_q(t)$ will converge to $\bar{\pi}_q$. Based on this assumption and following [13, Lemma 1], the stationary state probability $\bar{\pi}_q$ ($\forall q \in Q$) can be deduced by deploying the well-known global balancing equation for each $q \in Q$:

$$\bar{\pi}_q\left(\sum_{l\in L_q} \lambda^{(l)}\right) = \sum_{l\in L'_q} \lambda^{(l)}\bar{\pi}_{q_l}, \quad q \in Q. \quad (2)$$

Then, the constructed system of linear equations is to be solved simultaneously with the normalization equation,

$$\sum_{q\in Q}\bar{\pi}_q = 1. \tag{3}$$

It should be noted that the stationarity of $\pi_q(t) \to \bar{\pi}_q$ does not imply the convergence of the correlation $\mathbf{v}_q^{(k)}(t)$. This is because the correlation $\mathbf{v}_q^{(k)}(t)$ is an AoI measurement process which is independent from the evolution of the system occupancy $q(t)$. In this regard, a detailed explanation for the convergence of $\mathbf{v}_q^{(k)}(t)$ is demonstrated in the work of [12] and [13]. Accordingly, the correlation $\mathbf{v}_q^{(k)}(t)$ will converge to $\bar{\mathbf{v}}_q^{(k)}$ ($k \geq 1$), which satisfies the following system of linear equations, $\forall q \in Q$:

$$\bar{\mathbf{v}}_q^{(k)}\left(\sum_{l\in L_q}\lambda^{(l)}\right) = k\bar{\mathbf{v}}_q^{(k-1)} + \sum_{l\in L_q'}\lambda^{(l)}\bar{\mathbf{v}}_{q_l}^{(k)}\mathbf{A}_l,\ k \geq 1. \tag{4}$$

By (4), solving for the limiting correlation vector $\bar{\mathbf{v}}_q^{(k)}$ ($k \geq 1$) is executed recursively. After that, the kth AoI moment at the monitor can be deduced using equation (1) as follows:

$$\mathrm{E}[\Delta^k] = \mathrm{E}[x_0^k] = \lim_{t\to\infty}\mathrm{E}[x_0^k(t)] = \sum_{q\in Q}\bar{v}_{q,0}^{(k)}. \tag{5}$$

3.2 The SHS analysis of the proposed model

In this section, the mathematical evaluation of the $\mathrm{E}[\Delta_h^k]$($k \geq 1$, $1 \leq h \leq M$) is presented, using the aforementioned SHS analysis. The subsequent analysis is based on the perspective of class h, which is the class of interest.

Firstly, a two-dimensional CTMC $\{q(t)\}$ is deployed to keep track of the whole system occupancy. More specifically, the first and the second dimensions of $q(t)$ refer to the index of the served and the queued packets, respectively. In this regard, according to the system description in Sect. 2, the discrete state space can be represented as $Q = \mathcal{T} \cup \mathcal{U} \cup \mathcal{V} \cup \mathcal{W}$, where the disjoints sets $\mathcal{T}, \mathcal{U}, \mathcal{V}$, and $\mathcal{W}$ are as follows:

$$\begin{aligned}
&\mathcal{T} = \{(0,0)\}, \quad \mathcal{U} = \{(m,0) | 1 \leq m \leq M\},\\
&\mathcal{V} = \{(m,n) | 1 \leq m \leq M-1, m+1 \leq n \leq M\},\\
&\mathcal{W} = \{(m^*,n) | 2 \leq m \leq M, 1 \leq n \leq M\backslash m\}.
\end{aligned} \tag{6}$$

For more clarification, the set $\mathcal{T}$ represents the case of an empty system. The set $\mathcal{U}$ symbolizes the case of an empty buffer while the server is engaged with any class m ($1 \leq m \leq M$). The two sets $\mathcal{V}$ and $\mathcal{W}$ represent the case of a busy-buffer, busy-server system. The main difference between these two sets is whether the class being served has gained the protection feature or not. The ($*$) notation is used to distinguish those classes gained the protection feature. It should be noted that whenever the served class m possesses the protection feature, the buffered entity may belong to any class except m, as noticed in set $\mathcal{W}$. Otherwise, the LP class is always awaiting the HP served class (as in set $\mathcal{V}$).

According to the system structure described in Sect. 2, three AoI signals is to be tracked $\mathbf{x}(t) = [x_0(t), x_1(t), x_2(t)]$. These signals $x_0(t)$, $x_1(t)$ and $x_2(t)$ can be considered as the dedicated gauges of the AoI processes at the monitor, server and buffer, respectively. Note that the monitor's gauge $x_0(t)$ is only affected by the successful packet departure of the class of interest h; whereas, both $x_1(t)$ and $x_2(t)$ work for any class $1 \leq m \leq M$. Moreover, as mentioned earlier, all these measures increase at a unit rate ($\dot{\mathbf{x}}(t) = 1$) during the holding time at each discrete state $q(t)$.

Table 2 summarizes the SHS CTMC with all discrete state transitions. In this table, the transition sets—$L_{\mathcal{T}}$, $L_{\mathcal{U}}$, $L_{\mathcal{V}}$, and $L_{\mathcal{W}}$— describe the corresponding inner transitions for each discrete set $\mathcal{T}$, $\mathcal{U}$, $\mathcal{V}$ and $\mathcal{W}$, respectively. For each transition set, there are some transitions cases l described by the tuple $a_l = (q_l, q'_l, \lambda^{(l)}, \mathbf{x}'(t), \mathbf{A}_l)$. Moreover, the self transitions (i.e., when $q_l = q'_l$) stand for the self-preemption occurrence within the same class. However, only the self-preemptions related to the class of interest h are considered because the self-preemptions related to others have no effect in the AoI evolution of class h.

In the following, the main stochastic events incorporated in Table 2 are summarized. The detailed explanation of this table is left to Appendix 1 with an illustrative example on the case of three-class network (Fig. 19 and Table 6).

- Departure-to-monitor events. The departed packet may leave the system empty ($L_{\mathcal{T}}$) or permit the buffered packet to take its place ($\mathcal{U}2$, $\mathcal{U}2'$, $\mathcal{U}3$, $\mathcal{U}3'$).
- Fresh arrival entry into an empty server ($\mathcal{U}1$).
- Fresh arrival preemption of the ongoing service. Here, the preempted packet may join an empty buffer ($\mathcal{V}2$), preempt the buffered packet ($\mathcal{V}4$), or fail to preempt the buffered packet and it consequently gets dropped ($\mathcal{V}5$).
- Fresh arrival blocking by the ongoing service. The blocked arrival may instead join the empty buffer ($\mathcal{V}1$ and $\mathcal{W}2$), preempt the buffered packet ($\mathcal{V}3$, $\mathcal{W}1$ and $\mathcal{W}3$), or fail to preempt the buffered packet, so it gets dropped ($\mathcal{W}4$).
- The self-preemptions related to the class of interest h. These preemptions occur in either the server ($\mathcal{U}4$, $\mathcal{V}6$ and $\mathcal{W}5$) or the buffer ($\mathcal{V}7$ and $\mathcal{W}6$).

Going forward, concurrently with the discrete state transitions explained above, the AoI signals will be reset from $\mathbf{x}(t) = [x_0(t), x_1(t), x_2(t)]$ to $\mathbf{x}'(t) = [x'_0(t), x'_1(t), x'_2(t)]$ due to the following actions; otherwise, the AoI measures continue increasing without resetting:

1. If a fresh arrival joins the server (e.g., $\mathcal{U}1$, $\mathcal{U}4$, $\mathcal{V}2$ and $\mathcal{V}4$), the corresponding gauge measure starts from scratch ($x'_1(t) = 0$). Similarly, $x'_2(t) = 0$ upon a fresh arrival entry into the buffer (e.g., $\mathcal{V}1$, $\mathcal{V}3$, $\mathcal{W}1$, $\mathcal{W}2$ and $\mathcal{W}3$).
2. If the buffered packet enters an empty server, its elapsed age process measured by the buffer's gauge $x_2(t)$ will be resumed by the server's gauge, i.e., $x'_1(t) = x_2(t)$ ($\mathcal{U}2$, $\mathcal{U}2'$, $\mathcal{U}3$ and $\mathcal{U}3'$). The same situation exists if the served packet is to be preempted and joins the buffer; therefore, $x'_2(t) = x_1(t)$ ($\mathcal{V}2$ and $\mathcal{V}4$).
3. If the class of interest h departs successfully from the server to the monitor, the monitor's gauge will be reset to the last measure of the server's gauge, i.e., $x'_0(t) = x_1(t)$. This is clear in cases $\mathcal{T}2$, $\mathcal{U}2$ and $\mathcal{U}3$.

Table 2 Transition table of the SHS CTMC

Sets	Cases	$q_l \rightarrow q'_l$	$\lambda^{(l)}$	$\mathbf{x}' = \mathbf{x}\mathbf{A}$	$\mathbf{A}_l$
$L_{\mathcal{T}}$	$\mathcal{T}1$	$(i, 0) \rightarrow (0, 0)$ $1 \leq i \leq M \backslash h$	μ_i	$[x_0, x_1, x_2]$	diag$[1, 1, 1]$
	$\mathcal{T}2$	$(h, 0) \rightarrow (0, 0)$	μ_h	$[x_1, x_1, x_2]$	$\begin{matrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{matrix}$
$L_{\mathcal{U}}$	$\mathcal{U}1$	$(0, 0) \rightarrow (m, 0)$	λ_m	$[x_0, 0, x_2]$	diag$[1, 0, 1]$
	$\mathcal{U}2$	$(i, m) \rightarrow (m, 0)$ $1 \leq i \leq m - 1$	μ_i	$[x_1, x_2, x_2]$ If $i = h$	$\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix}$
	$\mathcal{U}2'$			$[x_0, x_2, x_2]$ If $i \neq h$	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix}$
	$\mathcal{U}3$	$(i^*, m) \rightarrow (m, 0)$ $2 \leq i \leq M \backslash m$	μ_i	$[x_1, x_2, x_2]$ If $i = h$	$\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix}$
	$\mathcal{U}3'$			$[x_0, x_2, x_2]$ If $i \neq h$	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix}$
	$\mathcal{U}4$	$(h, 0) \rightarrow (h, 0)$	$\lambda_h p^s_{h,h}$	$[x_0, 0, x_2]$	diag$[1, 0, 1]$
$L_{\mathcal{V}}$	$\mathcal{V}1$	$(m, 0) \rightarrow (m, n)$	λ_n	$[x_0, x_1, 0]$	diag$[1, 1, 0]$
	$\mathcal{V}2$	$(n, 0) \rightarrow (m, n)$	$\lambda_m p^s_{m,n}$	$[x_0, 0, x_1]$	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$
	$\mathcal{V}3$	$(m, i) \rightarrow (m, n)$ $n + 1 \leq i \leq M$	$\lambda_n p^w_{n,i}$	$[x_0, x_1, 0]$	diag$[1, 1, 0]$
	$\mathcal{V}4$	$(n, i) \rightarrow (m, n)$ $n + 1 \leq i \leq M$	$\lambda_m p^s_{m,n} p^w_{n,i}$	$[x_0, 0, x_1]$	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$
	$\mathcal{V}5$	$(i, n) \rightarrow (m, n)$ $m + 1 \leq i \leq n - 1$	$\lambda_m p^s_{m,i}(1 - p^w_{i,n})$	$[x_0, 0, x_2]$	diag$[1, 0, 1]$
	$\mathcal{V}6$	$(h, n) \rightarrow (h, n)$	$\lambda_h p^s_{h,h}$	$[x_0, 0, x_2]$	diag$[1, 0, 1]$
	$\mathcal{V}7$	$(m, h) \rightarrow (m, h)$	$\lambda_h p^w_{h,h}$	$[x_0, x_1, 0]$	diag$[1, 1, 0]$
$L_{\mathcal{W}}$	$\mathcal{W}1$	$(m^*, i) \rightarrow (m^*, n)$ $n + 1 \leq i \leq M \backslash m$	$\lambda_n p^w_{n,i}$	$[x_0, x_1, 0]$	diag$[1, 1, 0]$
	$\mathcal{W}2$	$(m, 0) \rightarrow (m^*, n)$ $n < m$	$\lambda_n(1 - p^s_{n,m})$	$[x_0, x_1, 0]$	diag$[1, 1, 0]$
	$\mathcal{W}3$	$(m, i) \rightarrow (m^*, n)$ $m + 1 \leq i \leq M$, $n < m$	$\lambda_n(1 - p^s_{n,m})p^w_{n,i}$	$[x_0, x_1, 0]$	diag$[1, 1, 0]$
	$\mathcal{W}4$	$(m, n) \rightarrow (m^*, n)$ $n > m$	$\lambda_i(1 - p^s_{i,m})(1 - p^w_{i,n})$ $1 \leq i \leq m - 1$	$[x_0, x_1, x_2]$	diag$[1, 1, 1]$
	$\mathcal{W}5$	$(h^*, n) \rightarrow (h^*, n)$	$\lambda_h p^s_{h,h}$	$[x_0, 0, x_2]$	diag$[1, 0, 1]$
	$\mathcal{W}6$	$(m^*, h) \rightarrow (m^*, h)$	$\lambda_h p^w_{h,h}$	$[x_0, x_1, 0]$	diag$[1, 1, 0]$

The SHS analysis begins with evaluating $\bar{\pi}_q$ ($\forall q \in Q$) by solving the following global balancing equations with the normalization Eq. (3):

- for $q \in \mathcal{T} = \{(0,0)\}$,

$$\bar{\pi}_q\big(\lambda_{\text{total}}\big) = \sum_{i=1}^{M} \bar{\pi}_{(i,0)}\mu_i, \tag{7}$$

- for $q \in \mathcal{U} = \{(m,0)|1 \le m \le M\}$,

$$\bar{\pi}_q\big(\mu_m + \hat{\lambda}_m + \check{\lambda}_m\big) = \bar{\pi}_{(0,0)}\lambda_m + \sum_{i=1}^{m-1} \bar{\pi}_{(i,m)}\mu_i + \sum_{2\le i\le M\backslash m} \bar{\pi}_{(i^*,m)}\mu_i, \tag{8}$$

- for $q \in \mathcal{V} = \{(m,n)|1 \le m \le M-1, m+1 \le n \le M\}$,

$$\begin{aligned}\bar{\pi}_q\big(\mu_m + \hat{\lambda}_m\big) = {} & \bar{\pi}_{(m,0)}\lambda_n + \bar{\pi}_{(n,0)}\lambda_m p^s_{m,n} + \lambda_n \sum_{i=n+1}^{M} \bar{\pi}_{(m,i)}p^w_{n,i} \\ & + \lambda_m p^s_{m,n} \sum_{i=n+1}^{M} \bar{\pi}_{(n,i)}p^w_{n,i} + \lambda_m \sum_{i=m+1}^{n-1} \bar{\pi}_{(i,n)}p^s_{m,i}(1-p^w_{i,n}),\end{aligned} \tag{9}$$

- for $q \in \mathcal{W} = \{(m^*,n)|2 \le m \le M, 1 \le n \le M\backslash m\}$,

$$\begin{aligned}\bar{\pi}_q\Big(\mu_m + \sum_{1\le i\le n-1\backslash m} \lambda_i p^w_{i,n}\Big) = {} & \lambda_n \sum_{n+1\le i\le M\backslash m} \bar{\pi}_{(m^*,i)}p^w_{n,i} + 1_{n<m}(n)\ \lambda_n(1-p^s_{n,m}) \\ & \Big(\bar{\pi}_{(m,0)} + \sum_{i=m+1}^{M} \bar{\pi}_{(m,i)}p^w_{n,i}\Big) + 1_{n>m}(n)\Big(\bar{\pi}_{(m,n)} \sum_{i=1}^{m-1} \lambda_i(1-p^s_{i,m})(1-p^w_{i,n})\Big).\end{aligned} \tag{10}$$

Then, $\bar{\mathbf{v}}^{(1)}_q$ is to be calculated by applying the system of linear equations described by (4) at $k=1$ as follows:

- for $q \in \mathcal{T} = \{(0,0)\}$,

$$\bar{\mathbf{v}}^{(1)}_q\big(\lambda_{\text{total}}\big) = \bar{\pi}_q\mathbf{1} + \mu_h\bar{\mathbf{v}}^{(1)}_{(h,0)}\mathbf{A}_{\mathcal{T}2} + \sum_{1\le i\le M\backslash h} \mu_i\bar{\mathbf{v}}^{(1)}_{(i,0)}\mathbf{A}_{\mathcal{T}1}, \tag{11}$$

- for $q \in \mathcal{U} = \{(m,0)|1 \le m \le M\}$,

$$\begin{aligned}\bar{\mathbf{v}}^{(1)}_q\big(\mu_m + \hat{\lambda}_m + \check{\lambda}_m + \delta_{h,m}\lambda_h p^s_{h,h}\big) = {} & \bar{\pi}_q\mathbf{1} + \lambda_m\bar{\mathbf{v}}^{(1)}_{(0,0)}\mathbf{A}_{\mathcal{U}1} \\ & + \sum_{i=1}^{m-1} \mu_i\Big(\delta_{h,i}\bar{\mathbf{v}}^{(1)}_{(i,m)}\mathbf{A}_{\mathcal{U}2} + (1-\delta_{h,i})\bar{\mathbf{v}}^{(1)}_{i,m}\mathbf{A}_{\mathcal{U}2'}\Big) + \sum_{2\le i\le M\backslash m} \mu_i\Big(\delta_{h,i}\bar{\mathbf{v}}^{(1)}_{(i^*,m)}\mathbf{A}_{\mathcal{U}3} \\ & + (1-\delta_{h,i})\bar{\mathbf{v}}^{(1)}_{(i^*,m)}\mathbf{A}_{\mathcal{U}3'}\Big) + \delta_{h,m}\lambda_h p^s_{h,h}\bar{\mathbf{v}}^{(1)}_q\mathbf{A}_{\mathcal{U}4},\end{aligned} \tag{12}$$

- for $q \in \mathcal{V} = \{(m,n)|1 \le m \le M-1, m+1 \le n \le M\}$,

$$\begin{aligned}&\bar{\mathbf{v}}_q^{(1)}\big(\mu_m + \hat{\lambda}_m + \delta_{h,m}\lambda_h p^s_{h,h} + \delta_{h,n}\lambda_h p^w_{h,h}\big) = \bar{\pi}_q \mathbf{1} + \lambda_n \bar{\mathbf{v}}^{(1)}_{(m,0)}\mathbf{A}_{\mathcal{V}1}\\ &+ \lambda_m p^s_{m,n}\bar{\mathbf{v}}^{(1)}_{(n,0)}\mathbf{A}_{\mathcal{V}2} + \lambda_n \sum_{i=n+1}^{M} p^w_{n,i}\bar{\mathbf{v}}^{(1)}_{(m,i)}\mathbf{A}_{\mathcal{V}3} + \lambda_m p^s_{m,n}\sum_{i=n+1}^{M} p^w_{n,i}\bar{\mathbf{v}}^{(1)}_{(n,i)}\mathbf{A}_{\mathcal{V}4}\\ &+ \lambda_m \sum_{i=m+1}^{n-1} p^s_{m,i}(1-p^w_{i,n})\bar{\mathbf{v}}^{(1)}_{(i,n)}\mathbf{A}_{\mathcal{V}5} + \delta_{h,m}\lambda_h p^s_{h,h}\bar{\mathbf{v}}^{(1)}_q\mathbf{A}_{\mathcal{V}6} + \delta_{h,n}\lambda_h p^w_{h,h}\bar{\mathbf{v}}^{(1)}_q\mathbf{A}_{\mathcal{V}7},\end{aligned} \tag{13}$$

- for $q \in \mathcal{W} = \{(m^*, n) | 2 \le m \le M, 1 \le n \le M \backslash m\}$,

$$\begin{aligned}&\bar{\mathbf{v}}_q^{(1)}\Big(\mu_m + \sum_{1\le i\le n-1\backslash m} \lambda_i p^w_{i,n} + \lambda_h(\delta_{h,m}p^s_{h,h} + \delta_{h,n}p^w_{h,h}\big) = \bar{\pi}_q \mathbf{1}\\ &\quad + \lambda_n \sum_{n+1\le i\le M\backslash m} p^w_{n,i}\bar{\mathbf{v}}^{(1)}_{(m^*,i)}\mathbf{A}_{\mathcal{W}1} + 1_{n<m}(n)\ \lambda_n \times (1-p^s_{n,m}) \times \Big(\bar{\mathbf{v}}^{(1)}_{(m,0)}\mathbf{A}_{\mathcal{W}2}\\ &\quad + \sum_{m+1}^{M} p^w_{n,i}\bar{\mathbf{v}}^{(1)}_{(m,i)}\mathbf{A}_{\mathcal{W}3}\Big) + 1_{n>m}(n)\Big(\bar{\mathbf{v}}^{(1)}_{(m,n)}\mathbf{A}_{\mathcal{W}4}\sum_{i=1}^{m-1}\lambda_i(1-p^s_{i,m})(1-p^w_{i,n})\Big)\\ &+ \delta_{h,m}\lambda_h p^s_{h,h}\bar{\mathbf{v}}^{(1)}_q\mathbf{A}_{\mathcal{W}5} + \delta_{h,n}\lambda_h p^w_{h,h}\bar{\mathbf{v}}^{(1)}_q\mathbf{A}_{\mathcal{W}6}.\end{aligned} \tag{14}$$

After evaluating $\bar{\mathbf{v}}_q^{(1)}$, the higher moment correlations $\bar{\mathbf{v}}_q^{(k)}$ ($k > 1$) can be evaluated recursively by applying the system of linear equation described by (4). Finally, $\mathrm{E}[\Delta_h^k]$ can be evaluated by using Eq. (5). The foregoing steps can be applied for any class of interest h ($1 \le h \le M$).

The algorithmic approach of the foregoing SHS analysis for any number of classes M can be summarized in Algorithm 1 as follows:

Algorithm 1 The SHS algorithm of the proposed scheme

```
Algorithm 1 The SHS algorithm of the proposed scheme
    Input: Number of classes $M$, the class of interest $h$, and the required $k$th AoI moment (where $k \ge 1$).
    Output: $k$th AoI moment of class $h$ $\mathrm{E}[\Delta_h^k]$.
Constructing the discrete state space $Q$, by applying definition (6).
Constructing the transition table, by employing Table 2 with respect to class $h$.
Finding $\bar{\pi}_q$ ($\forall q \in Q$), by solving the system of equations (7), (8), (9), (10) along with the normalization equation (3).
Finding $\bar{\mathbf{v}}_q^{(1)}$ ($\forall q \in Q$), by solving the system of equations (11), (12), (13) and (14).
for $j = 2 : k$ do
    Finding $\bar{\mathbf{v}}_q^{(j)}$ from $\bar{\mathbf{v}}_q^{(j-1)}$ ($\forall q \in Q$), by employing the recursive relation (4).
end for
Evaluating $\mathrm{E}[\Delta_h^k]$, by applying equation (5).
```

By following this algorithm, exact evaluation of $\mathrm{E}[\Delta_h^k]$ (for all $1 \le h \le M$) can be obtained because it is more involved to get closed-form expressions under a general number of priority classes. However, in the sequel of this section, closed-form results for the average AoI are derived under a case study on a two-class network.

3.2.1 Analytical case study on a two-class network

In such a case, the corresponding $\mathbf{P}^s$ and $\mathbf{P}^w$ will be as follow:

$$\mathbf{P}^s = \begin{bmatrix} p^s_{1,1} & p^s_{1,2} \\ 0 & p^s_{2,2} \end{bmatrix}, \quad \mathbf{P}^w = \begin{bmatrix} p^w_{1,1} & \sim \\ 0 & p^w_{2,2} \end{bmatrix}.$$

It should be noted that we ignore the entry $p^w_{1,2}$. This is because whenever class 2 is awaiting the service of class 1, any fresh arrival of class 1 either replaces the existing class-1 packet being served or gets dropped from the system (according to $p^s_{1,1}$). Subsequently, the demonstrated steps in Algorithm 1 is deployed:

1. Constructing the discrete state space Q:

$$\mathcal{T} = \{(0,0)\},\ \mathcal{U} = \{(1,0),(2,0)\},\ \mathcal{V} = \{(1,2)\},\ \mathcal{W} = \{(2^*,1)\}. \tag{15}$$

2. Constructing the transition table while considering that class 1 is the class of interest ($h = 1$), as shown in Table 3.
3. Finding the stationary state probabilities $\bar{\pi}_q$ ($\forall q \in Q$) by solving the following system of equations:

$$\bar{\pi}_{(0,0))}\big(\lambda_1 + \lambda_2\big) = \bar{\pi}_{(1,0)}\mu_1 + \bar{\pi}_{(2,0)}\mu_2, \tag{16}$$

$$\bar{\pi}_{(1,0)}\big(\mu_1 + \lambda_2\big) = \bar{\pi}_{(0,0)}\lambda_1 + \bar{\pi}_{(2^*,1)}\mu_2, \tag{17}$$

$$\bar{\pi}_{(2,0)}\big(\mu_2 + \lambda_1\big) = \bar{\pi}_{(0,0)}\lambda_2 + \bar{\pi}_{(1,2)}\mu_1, \tag{18}$$

$$\bar{\pi}_{(1,2)}\big(\mu_1\big) = \bar{\pi}_{(1,0)}\lambda_2 + \bar{\pi}_{(2,0)}\lambda_1 p^s_{1,2}, \tag{19}$$

$$\bar{\pi}_{(2^*,1)}\big(\mu_2\big) = \bar{\pi}_{(2,0)}\lambda_1(1 - p^s_{1,2}), \tag{20}$$

$$\sum_{q \in Q} \bar{\pi}_q = 1. \tag{21}$$

4. For $h = 1$ (the class of interest), evaluating the 1st-moment correlation vector $\bar{\mathbf{v}}^{(1)}_q$ ($\forall q \in Q$) by solving the following system of equations:

$$\bar{\mathbf{v}}^{(1)}_{(0,0)}\big(\lambda_1 + \lambda_2\big) = \bar{\pi}_{(0,0)}\mathbf{1} + \mu_1\bar{\mathbf{v}}^{(1)}_{(1,0)}\mathbf{A}_{\mathcal{T}2} + \mu_2\bar{\mathbf{v}}^{(1)}_{(2,0)}\mathbf{A}_{\mathcal{T}1}, \tag{22}$$

$$\begin{aligned}\bar{\mathbf{v}}^{(1)}_{(1,0)}\big(\mu_1 + \lambda_2 + \lambda_1 p^s_{1,1}\big) = \pi_{(1,0)}\mathbf{1} + \lambda_1\bar{\mathbf{v}}^{(1)}_{(0,0)}\mathbf{A}_{\mathcal{U}1} + \mu_2\bar{\mathbf{v}}^{(1)}_{(2^*,1)}\mathbf{A}_{\mathcal{U}3} \\ + \lambda_1 p^s_{1,1}\bar{\mathbf{v}}^{(1)}_{(1,0)}\mathbf{A}_{\mathcal{U}4},\end{aligned} \tag{23}$$

Table 3 Transition table of the SHS CTMC, considering class 1 is the one of interest (h=1)

Sets	Cases	$q_l \rightarrow q'_l$	$\lambda^{(l)}$	$\mathbf{x}' = \mathbf{x}\mathbf{A}$	$\mathbf{A}_l$
$L_{\mathcal{T}}$	$\mathcal{T}1$	$(2,0) \rightarrow (0,0)$	μ_2	$[x_0, x_1, x_2]$	diag[1,1,1]
	$\mathcal{T}2$	$(1,0) \rightarrow (0,0)$	μ_1	$[x_1, x_1, x_2]$	$\begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
$L_{\mathcal{U}}$	$\mathcal{U}1$	$(0,0) \rightarrow (1,0)$	λ_1	$[x_0, 0, x_2]$	diag[1,0,1]
		$(0,0) \rightarrow (2,0)$	λ_2		
	$\mathcal{U}2$	$(1,2) \rightarrow (2,0)$	μ_1	$[x_1, x_2, x_2]$	$\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix}$
	$\mathcal{U}3'$	$(2^*,1) \rightarrow (1,0)$	$\mu 2$	$[x_0, x_2, x_2]$	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix}$
	$\mathcal{U}4$	$(1,0) \rightarrow (1,0)$	λ_1	$[x_0, 0, x_2]$	diag[1,0,1]
$L_{\mathcal{V}}$	$\mathcal{V}1$	$(1,0) \rightarrow (1,2)$	λ_2	$[x_0, x_1, 0]$	diag[1,1,0]
	$\mathcal{V}2$	$(2,0) \rightarrow (1,2)$	$\lambda_1 p$	$[x_0, 0, x_1]$	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$
	$\mathcal{V}6$	$(1,2) \rightarrow (1,2)$	λ_1	$[x_0, 0, x_2]$	diag[1,0,1]
$L_{\mathcal{W}}$	$\mathcal{W}2$	$(2,0) \rightarrow (2^*,1)$	$\lambda_1(1-p)$	$[x_0, x_1, 0]$	diag[1,1,0]
	$\mathcal{W}6$	$(2^*,1) \rightarrow (2^*,1)$	λ_1	$[x_0, x_1, 0]$	diag[1,1,0]

$$\bar{\mathbf{v}}^{(1)}_{(2,0)}\left(\mu_2 + \lambda_1\right) = \bar{\pi}_{(2,0)}\mathbf{1} + \lambda_2 \bar{\mathbf{v}}^{(1)}_{(0,0)}\mathbf{A}_{\mathcal{U}1} + \mu_1 \bar{\mathbf{v}}^{(1)}_{(1,2)}\mathbf{A}_{\mathcal{U}2}, \tag{24}$$

$$\begin{aligned} \bar{\mathbf{v}}^{(1)}_{(1,2)}\left(\mu_1 + \lambda_1 p^s_{1,1}\right) = {} & \bar{\pi}_{(1,2)}\mathbf{1} + \lambda_2 \bar{\mathbf{v}}^{(1)}_{(1,0)}\mathbf{A}_{\mathcal{V}1} + \lambda_1 p^s_{1,2} \bar{\mathbf{v}}^{(1)}_{(2,0)}\mathbf{A}_{\mathcal{V}2} \\ & + \lambda_1 p^s_{1,1} \bar{\mathbf{v}}^{(1)}_{(1,2)}\mathbf{A}_{\mathcal{V}6}, \end{aligned} \tag{25}$$

$$\bar{\mathbf{v}}^{(1)}_{(2^*,1)}\left(\mu_2 + \lambda_1 p^w_{1,1}\right) = \bar{\pi}_{(2^*,1)}\mathbf{1} + \lambda_1(1 - p^s_{1,2})\bar{\mathbf{v}}^{(1)}_{(2,0)}\mathbf{A}_{\mathcal{W}2} + \lambda_1 p^w_{1,1} \bar{\mathbf{v}}^{(1)}_{(2^*,1)}\mathbf{A}_{\mathcal{W}6}. \tag{26}$$

5. Evaluating E[Δ_1] using Eq. (5):

$$\mathrm{E}[\Delta_1] = \bar{v}^{(1)}_{(0,0),0} + \bar{v}^{(1)}_{(1,0),0} + \bar{v}^{(1)}_{(2,0),0} + \bar{v}^{(1)}_{(1,2),0} + \bar{v}^{(1)}_{(2^*,1),0}. \tag{27}$$

6. Similarly, finding E[Δ_2] by repeating steps ii, iv and v taking into consideration $h = 2$ (the class of interest).

In order to find closed-form results of the average AoI for the two classes, let's consider homogeneous arrival and service processes, that is, $\lambda_1 = \lambda_2 = \lambda$ and $\mu_1 = \mu_2 = \mu$ (hence $\rho = \frac{\lambda}{\mu}$). This relaxation is considered to make the final results in a quite concise form. Moreover, since we are interested in the average AoI results, the self-preemption is admitted for both classes based on a conclusion reached subsequently in Sect. 4.1.1, that is, $p^s_{1,1} = p^w_{1,1} = 1$ and $p^s_{2,2} = p^w_{2,2} = 1$. Considering $p^s_{1,2} = p_s$, the stationary state probabilities resulting from the aforementioned step (iii) will be as follows:

$$\bar{\pi}_{(0,0)} = \frac{1}{2\rho^2 + 2\rho + 1}, \tag{28}$$

$$\bar{\pi}_{(1,0)} = \frac{\rho(2\rho(p_s - 1) - 1)}{\left(2\rho^2 + 2\rho + 1^2\right)(\rho(p_s - 2) - 1)}, \tag{29}$$

$$\bar{\pi}_{(2,0)} = \frac{\rho(2\rho + 1)}{\left(2\rho^2 + 2\rho + 1^2\right)(\rho(p_s - 2) - 1)}, \tag{30}$$

$$\bar{\pi}_{(1,2)} = \frac{\rho^2(2\rho + p_s + 1)}{\left(2\rho^2 + 2\rho + 1^2\right)(\rho(p_s - 2) - 1)}, \tag{31}$$

$$\bar{\pi}_{(2^*,1)} = \frac{\rho^2(p_s - 1)(2\rho + 1)}{\left(2\rho^2 + 2\rho + 1^2\right)(\rho(p_s - 2) - 1)}. \tag{32}$$

Subsequently, going through steps (iv) and (v), the average AoI for both classes can be formulated as follows:

$$\begin{aligned}\mathrm{E}[\Delta_1] = \frac{1}{\mathcal{A}(\rho + 1)(2\rho + 1)}\Big(&8\rho^8(2p_s - 3) + 4\rho^7(19p_s - 31) - 2\rho^6\left(5{p_s}^2 - 85p_s + 143\right)\\ &+ \rho^5\left(-12{p_s}^2 + 187p_s - 365\right) + \rho^4\left(-5{p_s}^2 + 117p_s - 293\right) - \rho^3\left({p_s}^2 - 45p_s + 156\right)\\ &+ 2\rho^2(5p_s - 27) + \rho(p_s - 11) - 1\Big),\end{aligned} \tag{33}$$

$$\begin{aligned}\mathrm{E}[\Delta_2] = \frac{1}{\mathcal{A}\left(2\rho^2 - \rho(p_s - 3) + 1^2\right)(\rho(p_s - 1) - 1)}\Big(&8\rho^9\left({p_s}^2 - 3p_s + 3\right)\\ &- 4\rho^8\left({p_s}^3 - 13{p_s}^2 + 36p_s - 37\right) - 2\rho^7\left(7{p_s}^3 - 68{p_s}^2 + 194p_s - 205\right)\\ &+ \rho^6\left(-15{p_s}^3 + 174{p_s}^2 - 574p_s + 651\right) + \rho^5\left(-11{p_s}^3 + 137{p_s}^2 - 526p_s + 658\right)\\ &+ \rho^4\left(-5{p_s}^3 + 71{p_s}^2 - 318p_s + 449\right) - \rho^3\left({p_s}^3 - 22{p_s}^2 + 125p_s - 210\right)\\ &+ \rho^2\left(3{p_s}^2 - 29p_s + 65\right) - 3\rho(p_s - 4) + 1\Big),\end{aligned} \tag{34}$$

where

$$\mathcal{A} = \rho\mu(\rho + 1)(2\rho + 1)\left(2\rho^2 + 2\rho + 1\right)(\rho(p_s - 2) - 1).$$

4 Numerical results

In this section, the analytical work presented in Sect. 3 will be numerically elaborated for performance evaluation. In the subsequent studies, a network of three prioritized classes is considered. Such setting is motivated by an example of three-class status update system in V2X technology [15] as illustrated in Sect. 1. Unless otherwise indicated, the homogeneous arrival and service processes are considered, where $\lambda_1 = \lambda_2 = \lambda_3 = \frac{1}{3}\lambda_{\text{total}}$ and $\mu_1 = \mu_2 = \mu_3 = \mu = 1$. Moreover, for performance evaluation, a comparative study will be conducted between our proposed model and some priority-based classical schemes (incorporated in Sect. 1). By classical, we mean the models that employ either the strict preemption or non-preemption schemes. In this regard, the two priority-based classical schemes proposed in [17] will be considered: PR$^{(s)}$-Bufferless scheme and NP$^{(s)}$-PR$^{(w)}$ scheme. Moreover, the classical scheme introduced in [18], PR$^{(s)}$-Multi-buffer scheme, will be considered in one of the subsequent numerical studies. In the sequel, our proposed model will be denoted relative to its combined policy Prob$^{(s)}$-Prob$^{(w)}$.

Firstly, the effect of the proposed model on the AoI performance will be investigated in Sect. 4.1, incorporating the study of the average performance and the corresponding dispersion. Subsequently, four different methods are presented in Sect. 4.2 to manifest how to set the system controlling parameters $\mathbf{P}^s, \mathbf{P}^w$. However, for presentation convenience, the content of these matrices will be written as follows: priority preemption probabilities between classes, $\mathbf{P} = [p^s_{1,2}, p^s_{1,3}, p^s_{2,3}, p^w_{1,2}, p^w_{1,3}, p^w_{2,3}]$; and self preemption probabilities, $\mathbf{P}_{\text{self}} = [p^s_{1,1}, p^s_{2,2}, p^s_{3,3}, p^w_{1,1}, p^w_{2,2}, p^w_{3,3}]$. Lastly, the validation of our analysis is presented in Sect. 4.3.

4.1 AoI performance evaluation

In this study, the AoI performance under the proposed Prob$^{(s)}$-Prob$^{(w)}$ model will be compared with the two classical schemes proposed in [17]: PR$^{(s)}$-Bufferless scheme and the NP$^{(s)}$-PR$^{(w)}$ scheme. Throughout this comparative study, the AoI performance for each priority class will be investigated in terms of its average and the corresponding dispersion (variability) beyond the average in Sects. 4.1.1 and 4.1.2, respectively. In both studies, the proposed controlling parameters are set as $\mathbf{P} = [0.2, 0.2, 0.2, 0.8, 0.8, 0.8]$. However, two cases of the self-preemption (SP) modes will be compared: strict SP ($\mathbf{P}_{\text{self}} = \mathbf{1}_{1\times 6}$) and probabilistic SP ($\mathbf{P}_{\text{self}} = 0.5 \times \mathbf{1}_{1\times 6}$).

4.1.1 Average AoI performance

In this study, the average AoI performance for each prioritized class will be addressed as shown in Figs. 4, 5 and 6. Based on theses illustrations, the following investigations can be laid out:

- Regarding the PR$^{(s)}$-Bufferless model, as shown in Fig. 4, it is considered as the optimal approach for the highest priority class (class 1): the result that has been proved in [9]. Accordingly, let this optimal performance be denoted as $\Delta_1^{\min}$. However, this comes at the cost of a harsh degradation on the LP classes due to the frequent interruptions (as shown in Figs. 5 and 6). Conversely, the NP$^{(s)}$ -PR$^{(w)}$ model significantly enhances the performance of the LP classes (Figs. 5 and 6) due to the relief from service interruptions, to the detriment of class 1 (Fig. 4), which is imposed to wait for the ongoing service of the LP classes.
- The foregoing paradox between classical models is resolved by employing the proposed Prob$^{(s)}$-Prob$^{(w)}$ model, by which a substantial enhancement for class 2 (Fig. 5) and class 3 (Fig. 6) is noticed under a compromise performance for class 1 (Fig. 4). This is because the hybrid preemptive/non-preemptive disciplines, employed at the server and the buffer, relax the adverse consequences of the strict preemption and non-preemption schemes.
- From another perspective, by a proper adjustment of the controlling parameters ($\mathbf{P}$ and $\mathbf{P}_{\text{self}}$), the stability of the AoI performance for the LP network can be guaranteed. By stable performance, we mean that the average AoI is always bounded with respect to the increasing offered load of the network. As shown in Fig. 5, $E[\Delta_2]|_{\text{Prob}^{(s)}\text{-Prob}^{(w)}}$ becomes stable despite being unstable in the classical schemes. In addition, as shown in Fig. 6, $E[\Delta_3]|_{\text{Prob}^{(s)}\text{-Prob}^{(w)}}$ experiences a marginal increase at the higher loading condition (almost stable) in contrast with the dramatic increase in case of classical schemes. After doing further numerical elaborations in this regard (not presented here due to space constraints), it turned out that the protection feature assumption for the LP classes has a significant contribution in such stability privilege.
- Regarding the SP setting of the proposed Prob$^{(s)}$-Prob$^{(w)}$ model, the strict SP is the preferable setting, rather than the probabilistic one, for the sake of the average AoI for each class. This is because the strict SP setting ($\mathbf{P}_{\text{self}} = \mathbf{1}_{1\times 6}$) permits the replacement of the stale packet (in the server or buffer) with a fresh one of the same class, the case that enhances the average AoI. Moreover, it was found that employing the strict SP mode for a certain class has no adverse consequences on the others. This is due to the memoryless property of the assumed exponential service time distribution.

4.1.2 AoI dispersion

This study implies the evaluation of the standard deviation σ ($\sigma^2 = E[\Delta^2] - (E[\Delta])^2$) for each priority class. However, due to the difference in the average AoI performance of the models incorporated in the comparative study, the coefficient of variation metric ($\text{CV} = \sigma / E[\Delta]$) is to be used instead of the standard deviation.

The CV performance of all priority classes is demonstrated in Figs. 7, 8 and 9. The main remarks are summarized as follows:

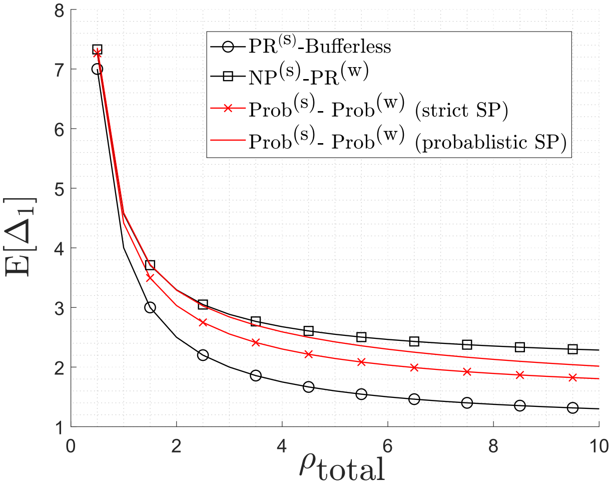


Fig. 4 Class 1's average AoI

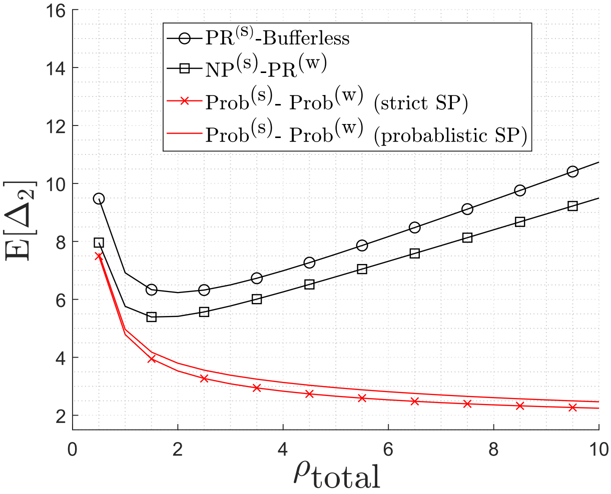


Fig. 5 Class 2's average AoI

- For class 1 performance depicted in Fig. 7, the PR$^{(s)}$-Bufferless scheme suffers from higher AoI variability compared with the NP$^{(s)}$-PR$^{(w)}$ scheme, contrasting with the situation of its average performance (Fig. 4). Hence, employing any of these classical schemes for class 1 results in a trade-off between the enhancement of the average AoI performance and its dispersion. However, this trade-off is compromised by adopting the proposed Prob$^{(s)}$-Prob$^{(w)}$ model, as noticed in Figs. 4 and 7.

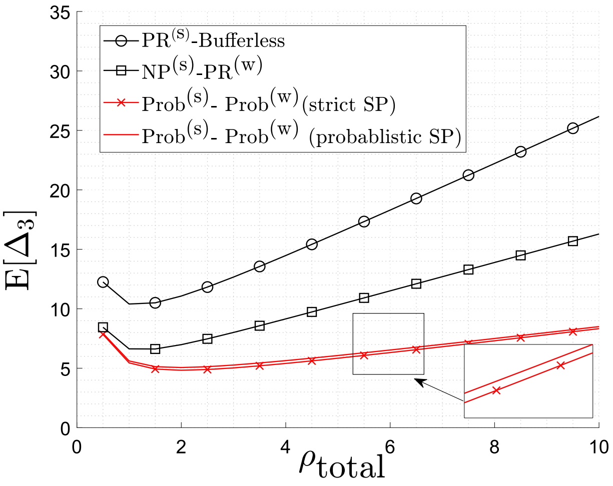


Fig. 6 Class 3's average AoI

- For the LP classes, as shown in Figs. 8 and 9, the proposed model besides the NP$^{(s)}$-NP$^{(w)}$ scheme yield significant enhancements of the AoI variability, compared with the PR$^{(s)}$-Bufferless.
- Regarding the SP setting, it should be noted that, unlike the average performance, the AoI variability is significantly improved by the probabilistic SP setting rather than the strict mode. This result manifests the importance of the probabilistic SP setting (characterized by $\mathbf{P}_{\text{self}}$) in adjusting the average AoI and the corresponding dispersion to some acceptable levels.

According to the foregoing study on the average AoI performance and its dispersion, the usefulness of our proposed model is corroborated. By thoroughly adjusting the controlling parameters ($\mathbf{P}$ and $\mathbf{P}_{\text{self}}$), the performance of the LP classes can be enhanced significantly without violating the strict AoI sensitivity limit of class 1 (determined by the application). This in turn promotes the reliability of the proposed model towards the whole priority classes, compared with the classical schemes. Accordingly, in the next section, we will exhibit different methods by which the controlling parameters can be adjusted so that the whole network satisfaction is fulfilled.

4.2 Controlling parameters setting

In this section, different approaches are introduced to manifest how to adjust the proposed controlling parameters $\mathbf{P}$ and $\mathbf{P}_{\text{self}}$. However, our main interest will be placed on the average AoI performance; hence, the strict SP setting will be assumed henceforth ($\mathbf{P}_{\text{self}} = \mathbf{1}_{1\times 6}$). In this regard, four different approaches are proposed for the setting of $\mathbf{P}$. For each approach, there is a distinct system objective to be accomplished as follows: HP class protection, optimal aggregate traffic intensity, optimal network satisfaction, and low-complex near-optimal network satisfaction.

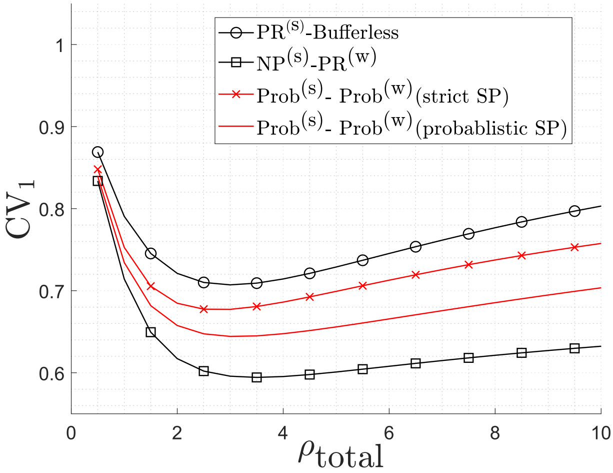


Fig. 7 Class 1's coefficient of variation performance

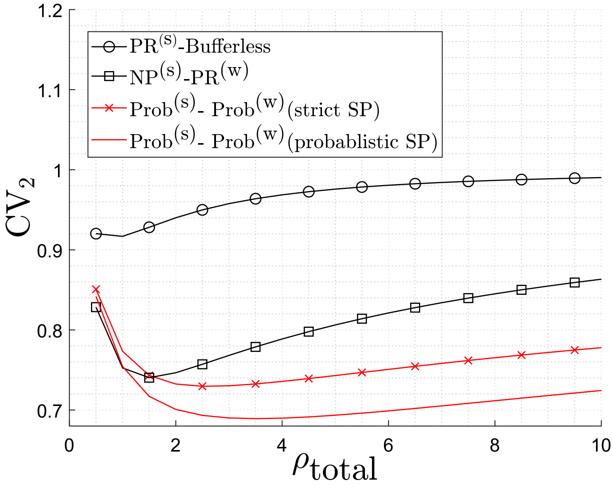


Fig. 8 Class 2's coefficient of variation performance

4.2.1 Approach 1: HP-class protection

In this approach, the main objective is to adjust the controlling parameters so that the AoI performance of the LP classes can be enhanced, while class 1 is kept gaining its optimal performance $\Delta_1^{\min}$. To this end, the server should be governed by the full PR$^{(s)}$ policy so that any new request from class 1 can be handled by the server promptly. Hence, an extracted version of the proposed model is considered, that is, PR$^{(s)}$-Prob$^{(w)}$ where $p^s_{1,2} = p^s_{1,3} = p^s_{2,3} = 1$. In such case, it is clear that $p^w_{1,2}$ and $p^w_{1,3}$ will be useless and the only parameter that has the control on the performance is $p^w_{2,3}$.

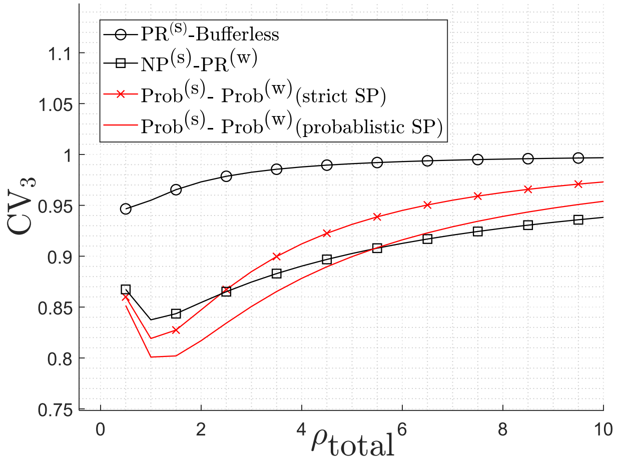


Fig. 9 Class 3's coefficient of variation performance

Based on the foregoing, in the following numerical example, the setting of the controlling parameters will be $\mathbf{P} = [1, 1, 1, 1, 1, 0.2]$. The extracted version of our model PR$^{(s)}$-Prob$^{(w)}$ is to be compared with other related classical models that adopt full PR$^{(s)}$ policy: PR$^{(s)}$-Bufferless model [17] and PR$^{(s)}$-Multi-buffer model [18]. This comparative study is depicted in terms of the average AoI for each class, as shown in Figs. 10, 11 and 12. As noticed in Fig. 10, the common performance $\Delta_1^{\min}$ is yielded for all models under study. However, the effect on the LP network can be explained as follows:

- Compared with the PR$^{(s)}$-Bufferless scheme, the PR$^{(s)}$-multi-buffer model significantly improves class 2 performance (Fig. 11). Whereas, class 3 performance (Fig. 12) gets worse after ($\rho^* \approx 6.5$). This trend is due to the deployed multi-buffer mechanism, by which the HP classes constitute a heavy workload over the lowest priority class, especially at the higher traffic loading conditions.
- On the other hand, comparing with PR$^{(s)}$-multi-buffer model, the extracted version of our model PR$^{(s)}$-Prob$^{(w)}$, by its controlling parameter $p_{2,3}^{w} = 0.2$, yields a significant enhancement for class 3 at the cost of marginal degradation for class 2. As noticed in Fig. 12, the crossover point is shifted to $\rho^{**} \approx 16.5$.

To conclude, the deployment of a single buffer controlled by the proposed Prob$^{(w)}$ policy can outperform the multi-buffer case. Hence, comparatively speaking, our proposed system is less costly but more efficient: it makes the best use of its buffer resources owing to the proposed Prob$^{(w)}$ policy.

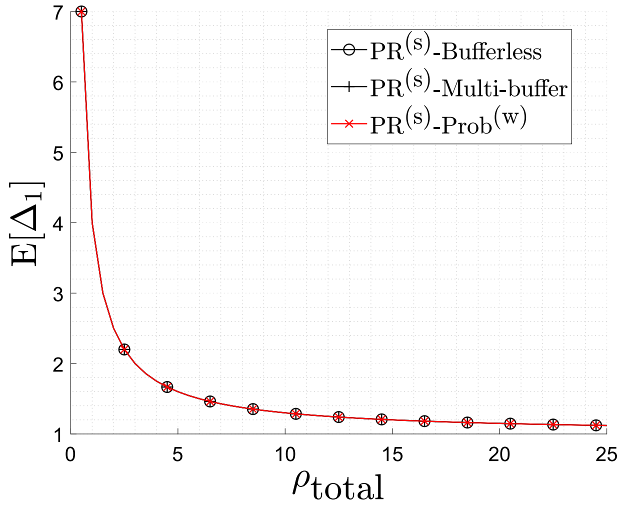


Fig. 10 Class 1's average AoI under the approach of HP-class protection

4.2.2 Approach 2: optimal aggregate traffic intensity

In this approach, under a fixed assignment of the controlling parameters **P**, the main objective is to adapt the traffic intensities $(\lambda_1, \lambda_2, \lambda_3)$ so that the aggregate traffic intensity λ_{total} (the total packet generation rate) is minimized. However, this minimization problem is limited by the fulfillment of some AoI constraints related to each class. This problem is motivated by the fact that minimizing the total generation rate λ_{total} is a sign of minimizing the total sensing power needed for the whole network. Hence, in our context, the sensing power and the generation rate are used interchangeably.

Based on the aforementioned description, the AoI constraints are represented as follows:

$$\mathrm{E}[\Delta_1] < \Delta_{\mathrm{max}}, \quad \mathrm{E}[\Delta_2] < \frac{1}{\beta_2}\Delta_{\mathrm{max}}, \quad \mathrm{E}[\Delta_3] < \frac{1}{\beta_3}\Delta_{\mathrm{max}},$$

where β_2 and β_3 are the importance factors for classes 2 and 3, respectively, relative to the importance of class 1; hence, $0 < \beta_2, \beta_3 \leq 1$. In the prioritized case, these factors can be set such that $\beta_2 > \beta_3$. According to these constraints, whenever class 2 and 3 become nowhere near as important as class 1 $(\beta_2, \beta_3 \ll 1)$, their corresponding constraints will be much less strict (increasing the AoI bound). On the other hand, $\beta_2 = \beta_3 = 1$ is the case of the strictest QoS requirements, where a unique AoI bound (Δ_{max}) should be obeyed for all classes.

In this regard, Table 4 demonstrates the resulting optimal aggregate intensity $\lambda^*_{\mathrm{total}}$ in a comparative study with the classical models in addition to our previous work [24], which employs the Prob$^{(\mathrm{s})}$-Bufferless scheme. In this study, the controlling parameters **P** are set fixed as $\mathbf{P} = 0.5 \times \mathbf{1}_{1\times 6}$. Moreover, through this comparative

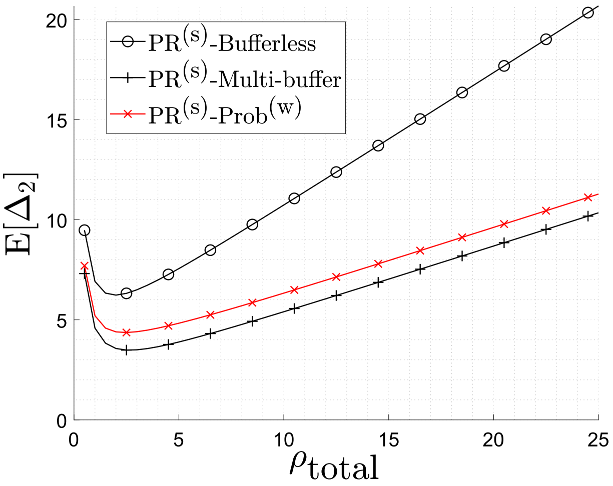


Fig. 11 Class 2's average AoI under the approach of HP-class protection

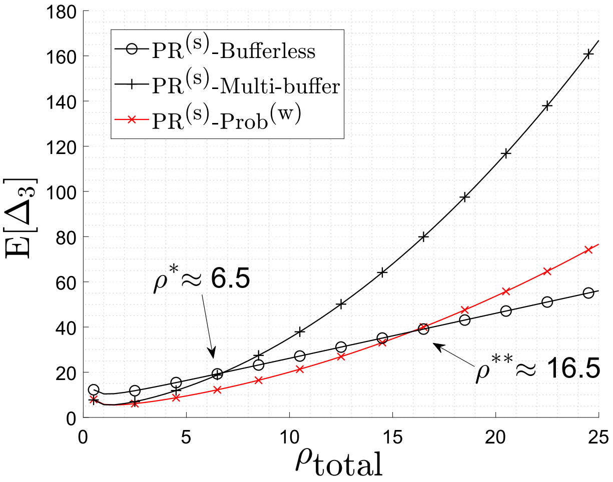


Fig. 12 Class 3's average AoI under the approach of HP-class protection

study, we consider different values of β_2 and β_3, by which different AoI constraints of the LP classes are applied. However, the AoI bound for class 1 is set to be $\Delta_{\max} = 4$.

According to Table 4, it should be noted that the total generation rate has an upward trend as β_2 and β_3 approach 1 (i.e., as the AoI constraints become stricter). This is because in such cases higher generation rate is needed to cope with the confined AoI constraints. Moreover, our proposed model outperforms other models by satisfying the AoI bounds for all classes with minimum sensing power needed. Conversely, the classical models need a higher energy as the constraints become stricter. More specifically, in this study, it was found that the classical models cannot at all satisfy the strictest AoI bounds (at $\beta_2 = \beta_3 = 1$). Furthermore, compared with our

Table 4 The optimal aggregate traffic intensity λ^*_{total} under the four different models

(β_2, β_3)	Prob$^{(s)}$-Prob$^{(w)}$	Prob$^{(s)}$-bufferless	PR$^{(s)}$-bufferless	NP$^{(s)}$-PR$^{(w)}$
$(0.8^5, 0.8^6)$	0.55	0.80	0.75	0.70
$(0.8^4, 0.8^5)$	0.70	0.90	0.90	0.75
$(0.8^3, 0.8^4)$	0.75	1.15	1.05	0.85
$(0.8^2, 0.8^3)$	0.90	1.40	1.30	1.15
$(0.8^1, 0.8^2)$	1.05	2.00	1.85	1.65
(1, 1)	1.80	4.35	N/A	N/A

previous work Prob$^{(s)}$-Bufferless, the current work is superior due to the buffering functionalities with the corresponding Prob$^{(w)}$ policy.

Based on the foregoing investigations, our proposed scheme can be considered as an energy-efficient module, which is capable of fulfilling the AoI restrictions of all prioritized classes with a minimum power needed.

4.2.3 Approach 3: optimal network satisfaction

In this approach, the fulfillment of the whole network satisfaction is represented as an optimization problem, whose decision variables are the system controlling parameters **P** to be explored. In this regard, the network cost function to be minimized is denoted as C_{α_2,α_3}, which is defined as the weighted sum of the average AoI for all priority classes. Hence, the minimization problem can be constructed as follows:

$$\min_{\mathbf{P}} C_{\alpha_2,\alpha_3} = \mathrm{E}[\Delta_1] + \alpha_2 \times \mathrm{E}[\Delta_2] + \alpha_3 \times \mathrm{E}[\Delta_3], 0 \leq \alpha_2, \alpha_3 \leq 1. \tag{35}$$

The weighting parameters α_2, α_3 are deployed to determine to what extent the whole priority network satisfaction is sensitive to the performance of the LP classes. Under the priority setting, it is intuitive that $\alpha_2 > \alpha_3$.

In this regard, the squirrel search algorithm (SSA) [28] is used to solve the optimization problem (35) (finding the optimal controlling parameters $\mathbf{P}^*$). In this nature-inspired algorithm, the efficient foraging behaviour of southern flying squirrels (FSs) is exploited, where a special locomotion technique (called gliding) is used instead of flying so that spacious areas of the forest can be covered swiftly with minimum energy consumption. The hickory nut tree is always the optimal destination for all FSs. In winter, it is the main food source to fulfill their energy requirements. In autumn, FSs seek the hickory nut trees for storage purposes to be abundant in winter. Accordingly, the location of the hickory nut tree is the optimal solution to be explored $\mathbf{P}^*$. Hence, we assume a 6th-dimensional search space for the FSs to suit the dimension of $\mathbf{P}^*$. Moreover, assuming the number of 50 FSs through the search space, the location of each one is a candidate solution of the problem. Furthermore, using the same stopping criterion introduced in [28], we set the maximum number of iterations at 100. Other deployed SSA parameters are set as recommended in [28]. Owing to the SSA location-update strategy, its accuracy and convergence rate are proved to outperform other nature-inspired schemes [28]. Further details in this regard can be found through the work of [28].

By solving the optimization problem (35), the optimal capabilities of our proposed model can be investigated. This problem is solved under three different schemes: our previous work [24] (Prob$^{(s)}$-Bufferless); and two cases of our current proposed model (Prob$^{(s)}$-PR$^{(w)}$ and Prob$^{(s)}$-Prob$^{(w)}$). It should be noted that Prob$^{(s)}$-PR$^{(w)}$ scheme is obtained from the proposed model by imposing $p^w_{1,2} = p^w_{1,3} = p^w_{2,3} = 1$. This scheme is incorporated in the comparative study to clarify whether the system privilege comes from using the buffer only or from using the buffer with its PR$^{(w)}$ policy.

Having obtained the optimization results for the aforementioned schemes, a comparative study is conducted with the classical schemes, PR$^{(s)}$-Bufferless and NP$^{(s)}$-PR$^{(w)}$ [17], in terms of $C_{\frac{1}{2},\frac{1}{4}}$, which is a representative of the whole satisfaction of the priority network. Assuming $\mu_1 = \mu_2 = \mu_3 = 1$, three different traffic cases (TC1, TC2 and TC3) of the arrival processes are addressed: TC1, $\lambda_1 = 2 \times \lambda_2 = 4 \times \lambda_3$ (Fig. 13); TC2, $\lambda_1 = \frac{1}{2} \times \lambda_2 = \frac{1}{4} \times \lambda_3$ (Fig. 14); and TC3, $\lambda_1 = \lambda_2 = \lambda_3$ (Fig. 15). All traffic cases form the same offered load $\rho_{\text{total}} = [0.5, 30]$.

Overall, Figs. 13, 14 and 15 demonstrate that the optimal network satisfaction in all traffic cases is achieved by employing the proposed Prob$^{(s)}$-Prob$^{(w)}$ policy owing to its controlling parameters. Moreover, the superiority of the proposed Prob$^{(s)}$-Prob$^{(w)}$ policy compared with the Prob$^{(s)}$-PR$^{(w)}$ policy is an indication for the importance of the Prob$^{(w)}$ policy to regulate the contention on the buffer especially at the higher traffic loading conditions: the result that underpins the conclusion reached in Sect. 4.2.1. Hence, the deployment of Prob$^{(s)}$ policy combined with the Prob$^{(w)}$ policy is essential to ensure the robustness of the system at all traffic conditions. Furthermore, compared with the classical approaches, the performance gain of our model becomes more significant in the case of TC1 ($\lambda_1 > \lambda_2 > \lambda_3$) as shown in Fig. 13), where the LP classes are adversely affected by the excess load of the HP ones.

4.2.4 Approach 4: low-complex near-optimal network satisfaction

In this section, heuristic generation formulas for the controlling parameters **P** are proposed to introduce a near-optimal solution for the optimization problem (35), without any computational complexity. Accordingly, these formulas will undergo different studies to substantiate the near-optimality feature.

First of all, the general proposed heuristic formula of all controlling parameters involved in **P** is expressed as follows, for $1 \le m \le M - 1$ and $m + 1 \le n \le M$:

$$p^s_{m,n} = p^w_{m,n} = e^{-\left[\frac{\alpha_n}{\alpha_m} \times \frac{\rho_m}{\rho_n}\right]\rho_m}. \tag{36}$$

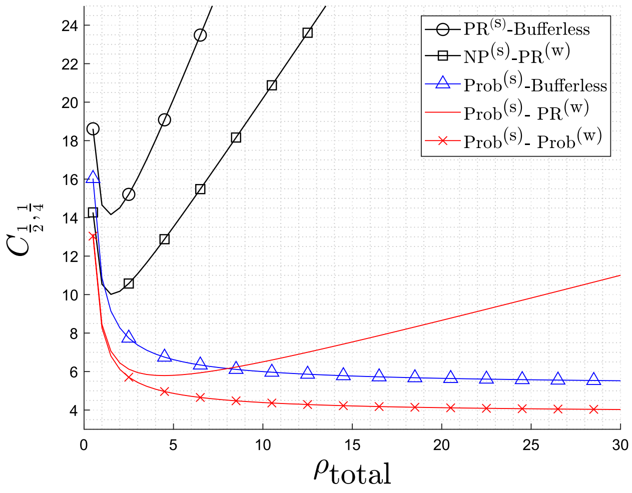


Fig. 13 The cost function $C_{\frac{1}{2},\frac{1}{4}}$ under the traffic case TC1, $\lambda_1 > \lambda_2 > \lambda_3$

By this generation rule, we take into account the following considerations: First, $p^s_{m,n}$ ($p^w_{m,n}$) decreases exponentially with the increase of the offered load of class m (the source of interruption). Secondly, the decaying rate of $p^s_{m,n}$ ($p^w_{m,n}$) depends on the exponent coefficients $0 \leq \frac{\alpha_n}{\alpha_m} \leq 1$ and $\frac{\rho_m}{\rho_n} > 0$. Therefore, when $\rho_m < \rho_n$, the decreasing regression of $p^s_{m,n}$ ($p^w_{m,n}$) will be marginal for the sake of the HP class m, which is adversely affected by the excess load of the LP class n. However, in the reversed situation ($\rho_m > \rho_n$), $p^s_{m,n}$ ($p^w_{m,n}$) decreases swiftly to relieve the LP class n from the expected increase in HP class m preemptions. Based on this design, the controlling parameters enable the system to respond in favour of the class experiencing the harsh conditions. From another perspective, by the importance ratio ($\frac{\alpha_n}{\alpha_m}$), the design of the controlling parameters can reflect the differentiated importance between the prioritized classes. Here, class-1 importance can be regarded as $\alpha_1 = 1$.

To verify the near-optimality feature of the proposed heuristic method with the optimal solution of (35) presented in Sect. 4.2.3, some studies are carried out as follows: studying different traffic cases; and experimenting with different cost functions, $C_{\frac{1}{2},\frac{1}{4}}$ and $C_{\frac{1}{4},\frac{1}{8}}$. In this study, five traffic cases will be considered: the aforementioned cases (TC1, TC2 and TC3); TC4, $\mu_1 = 2, \mu_2 = 1, \mu_3 = \frac{2}{3}$; and TC5, $\mu_1 = \frac{2}{3}, \mu_2 = 1, \mu_3 = 2$. In the cases TC4 and TC5, the arrival processes are homogeneous $\lambda_1 = \lambda_2 = \lambda_3$. However, all traffic cases constitute the same total offered load ρ_{total}. Under each traffic case, the maximum percentage error between the heuristic solution and the optimal one is calculated over the span $\rho_{\text{total}} = [0.5, 15]$. This accuracy percentage is tabulated in Table 5, where acceptable accuracy levels are demonstrated in all cases under study. Moreover, under each traffic case, it is clear that the accuracy gets improved whenever class 2 and class 3 become nowhere near as important as class 1 ($\alpha_2, \alpha_3 \ll 1$). Consequently,

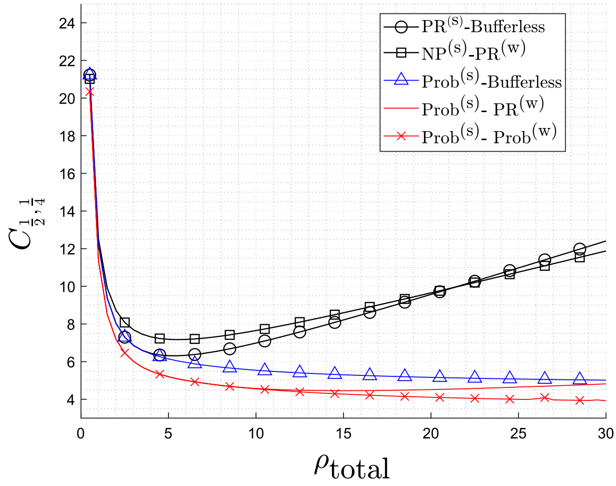


Fig. 14 The cost function $C_{\frac{1}{2},\frac{1}{4}}$ under the traffic case TC2, $\lambda_1 < \lambda_2 < \lambda_3$

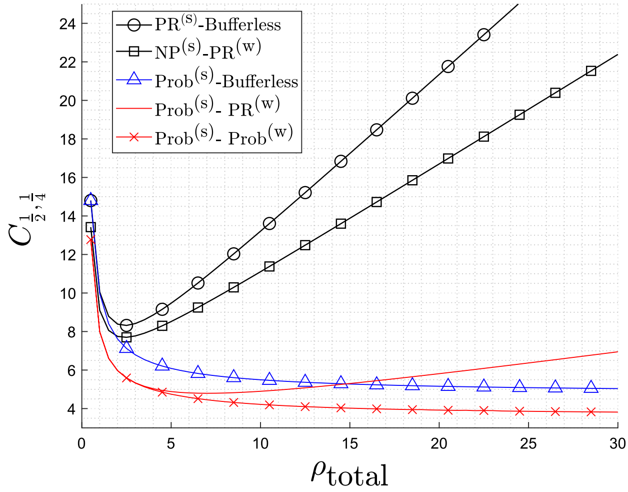


Fig. 15 The cost function $C_{\frac{1}{2},\frac{1}{4}}$ under the traffic case TC3, $\lambda_1 = \lambda_2 = \lambda_3$

the proposed low-complex heuristic generating method of the controlling parameters can obtain near-optimal performance of the whole network satisfaction.

4.3 Analytical model validation

Throughout this section, the presented analytical study in Sect. 3 is verified by building up a simulation environment, similar to the proposed system described in Sect. 2, using MATLAB R2015a. The stationary results can be ensured by setting a

Table 5 Maximum percentage error between the heuristic and the optimal solutions over the span $\rho_{\text{total}} = [0.5, 15]$

	TC1 (%)	TC2 (%)	TC3 (%)	TC4 (%)	TC5 (%)
$C_{\frac{1}{2},\frac{1}{4}}$	6.0	11.7	7.9	6.0	3.0
$C_{\frac{1}{4},\frac{1}{8}}$	5.0	5.1	4.3	4.5	2.6

large simulation time (10^6 time units). To this end, a workstation with the following specifications is deployed: Intel(R) Xeon(R) Gold 6230R CPU, 2.10 GHz (2 processors); 128 GB RAM; and 64 bit Windows 10 pro operating system.

In this study, the controlling parameters, $\mathbf{P}$ and $\mathbf{P}_{\text{self}}$ are set as follows: $\mathbf{P} = [0.5, 0.5, 0.5, 0.3, 0.3, 0.3]$ and $\mathbf{P}_{\text{self}} = \mathbf{1}_{1\times6}$. Accordingly, the average AoI $\mathrm{E}[\Delta]$ and the corresponding standard deviation σ are evaluated under both the analytical and simulation framework. As depicted in Figs. 16 and 17, the simulation results substantiate the validity of the analytical results with maximum percentages errors of 0.5775 % and 1.2621 % for $\mathrm{E}[\Delta]$ (Fig. 16) and σ (Fig. 17), respectively.

5 Conclusion

This work enhanced the AoI performance of multi-class IoT-enabled MEC systems by proposing a hybrid preemptive/non-preemptive discipline under an M/M/1/2 priority queueing model. Through the proposed $\text{Prob}^{(\text{s})}$-$\text{Prob}^{(\text{w})}$ policy, the probabilistic preemption approach (as a discretionary discipline for preemption) is deployed to govern the rivalry between priority classes within the shared server and buffer. Independent probabilistic preemption decisions are taken for the $\text{Prob}^{(\text{s})}$ and the $\text{Prob}^{(\text{w})}$ policies, and with distinct probability parameters (system controlling parameters). The SHS approach is deployed to analyze the average AoI in addition to the corresponding higher order moments. An algorithmic approach is presented to extract the AoI moments for any specific number of classes. However, under a case study on a two-class network, closed-form results of the average AoI are derived. Then, a numerical study on a three-class network is presented for performance evaluation. Throughout this study, the performance of the proposed model is compared with some of the priority-based classical schemes in terms of the average AoI and its coefficient of variation. In this regard, it is demonstrated that the thorough adjustment of the system controlling parameters ensures the reliability of the proposed system not for specific class but for all priority classes. Hence, four different approaches are presented to explain the setting of the proposed controlling parameters. Based on these approaches, it turned out that our proposed model—by its controlling parameters— acquires different privileges. Firstly, the performance of the LP classes can be manipulated without violating the optimal performance of the highest priority (age-sensitive) class. In addition, owing to the proposed $\text{Prob}^{(\text{w})}$ policy, the system exploits the buffer resources efficiently, making the system less costly. Moreover, the proposed model is an energy-efficient, where the AoI bounds of all classes can be fulfilled with minimum sensing power needed. Furthermore, using the proposed

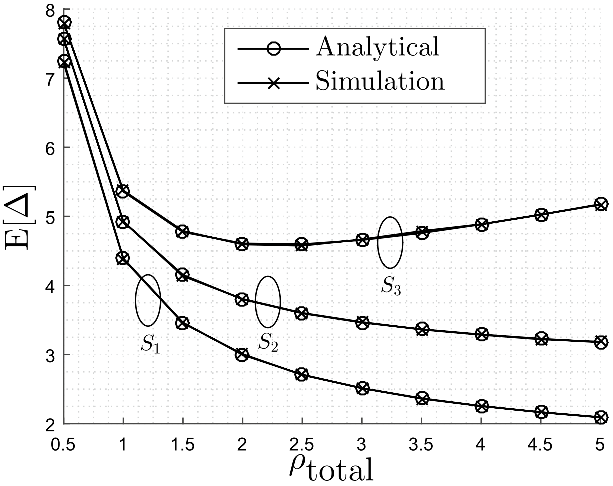


Fig. 16 Analytical model validation in terms of the average AoI

scheme, the optimal satisfaction of the whole priority network can be achieved by exploring the suitable setting of the controlling parameters according to the working traffic conditions. In this study, it is also demonstrated that the $\text{Prob}^{(\text{w})}$ policy has a significant impact on maintaining the optimality of the proposed scheme under the heavy loading conditions. Finally, a heuristic generation method for the controlling parameters is also proposed to provide a near-optimal solution without any computational complexity.

As a future work, it is expected to extend our work to incorporate other discretionary rules for the hybrid discipline instead of the probabilistic preemption approach. Moreover, the extension to a general service time distribution is an inevitable phase in our future work. From another perspective, it is envisaged that the proposed model can be incorporated into a multi-tier computing network considering the following: local servers with limited computation power, heterogeneous edge servers with higher computational power, and central clouds with super computation power.

Appendix 1 Explanation of the proposed system dynamics with an illustrative example

In this section, a further demonstration of the proposed system model is presented using a timeline representation under an example of three priority classes as shown in Fig. 18. This example demonstrates only a sample of all possible stochastic events explained in Sect. 3.2. Here, class A and C constitute the highest and the lowest priority classes, respectively, whereas class B has the middle priority. Let the n^{th} packet of class A, B, and C be denoted as A_n, B_n, and C_n,

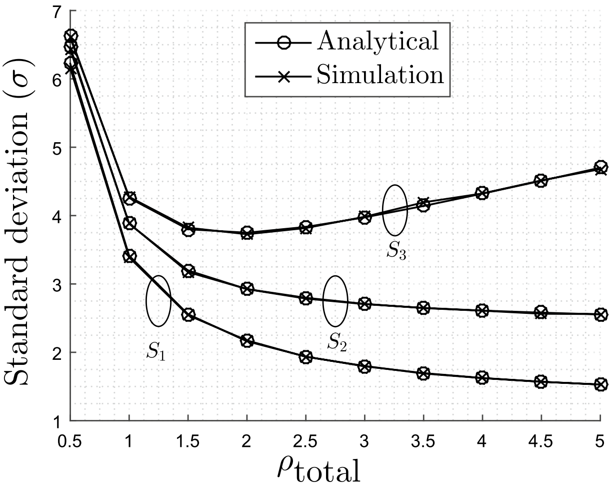


Fig. 17 Analytical model validation in terms of the AoI standard deviation

respectively. As shown in Fig. 18, all packets annotated in red color are those that fail to achieve a successful departure to the monitor due to the forced dropping they encounter before initiating or completing the service. Moreover, it should be noted that the proposed Prob$^{(s)}$ and Prob$^{(w)}$ policies are represented in this example by a Yes/No probabilistic decisions for preemption based on a certain probability parameter. For example, as shown in Fig. 18, the decision "$p^{w}_{\text{B,C}}$: No" means that the preemption-in-waiting request from class B upon class C is declined based on the probability parameter $p^{w}_{\text{B,C}}$. In the following, a summary is introduced about the stochastic events that occur at the time epochs $t = \{t_1, t_2, \cdots, t_{12}\}$:

- Fresh arrival entry to an empty server. This event happens at $t = t_1$ for packet A_1.
- Fresh arrival blocking by the ongoing service. The blocked arrival may instead join the empty buffer (as in the cases of C_1 at $t = t_2$ and B_2 at $t = t_5$) or be dropped from the system due to its failure to preempt the buffered packet (as in the cases of B_1 at $t = t_3$, A_2 at $t = t_6$ and C_2 at $t = t_{10}$). It should be noted that the ongoing service of C_1 gains the protection feature (explained in Sect. 2.1) concurrently with the probabilistic decision "$p^{s}_{\text{B,C}}$: No" at $t = t_3$. That is why the preemption request from packet A_2 is further declined by the ongoing service of C_1 at $t = t_6$.
- Fresh arrival preemption of the ongoing service. The packet preempted from the server may join an empty buffer (as in the case of B_2 at $t = t_8$) or be dropped from the system due to the system condition that no more than one packet of the same class can exist in the system (as in the case of A_3 at $t = t_9$).
- Successful departure to monitor. The departed packet may leave the system empty (as in the case of B_2 at $t = t_{12}$) or permit the buffered packet to take its place (as in the cases of A_1 at $t = t_4$, C_1 at $t = t_7$ and A_4 at $t = t_{11}$).

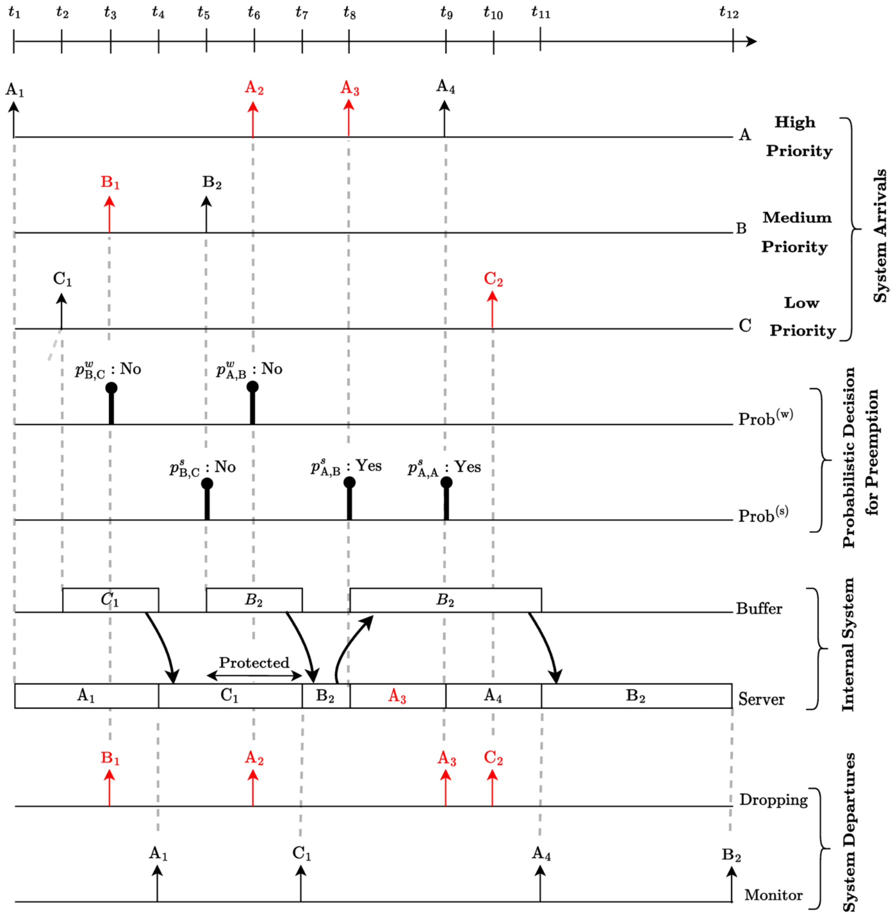


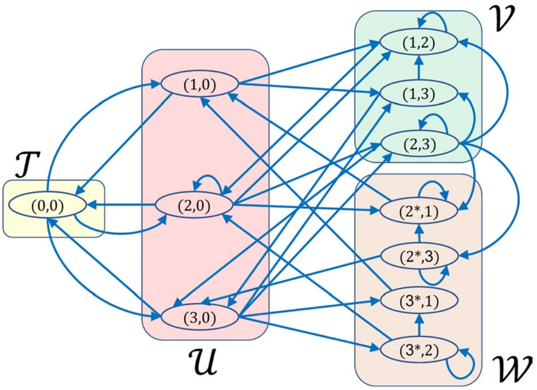
Fig. 18 A timeline representation on a three-class network

Appendix 2 Explanation of Table 2 with an illustrative example

In the following, the discrete state transitions incorporated in Table 2 are exhaustively explained:

- $L_{\mathcal{T}}$: This transition set represents all incoming transitions into state (0, 0). This in turn describes the successful departure-to-monitor events that occur at empty-buffer, busy-server system. The departed packet may belong to the class of interest h (case "$\mathcal{T}2$") or any of the other classes (case "$\mathcal{T}1$").
- $L_{\mathcal{U}}$: This transition set combines all entering transitions into state $q \in \mathcal{U}$. This means that the system becomes/remains an empty-buffer, busy-server system, as explained in the following cases:
  - Case "$\mathcal{U}1$": A fresh class m packet ($1 \le m \le M$) arrives at an empty-buffer, empty-server system.

Fig. 19 State transition diagram



- Case “$\mathcal{U}2$”, “$\mathcal{U}2'$”,“$\mathcal{U}3$” and “$\mathcal{U}3'$”: In all these cases, the served packet has been successfully departed. This is before the queued packet can takes its place at the server.
- Case “$\mathcal{U}4$”: It describes the potential self-preemption over the ongoing service of class h (class of interest) with probability $p^s_{h,h}$.

- $L_{\mathcal{V}}$: This transitions set is an aggregate of all transitions scenarios ending up with the destination state $q'_l \in \mathcal{V} = \{(m, n)|1 \le m \le M-1, m+1 \le n \le M\}$. Here, the served class m has not acquired the protection feature so far. Accordingly, the corresponding cases can be explained as follows.

  - Case “$\mathcal{V}1$” and “$\mathcal{V}3$”: A fresh packet arrival of class n cannot preempt the ongoing service of the HP class m. Instead, it can either join the empty buffer (case “$\mathcal{V}1$”) or preempt the queued class i, $n+1 \le i \le M$, with probability $p^w_{n,i}$ (case “$\mathcal{V}3$”).
  - Case “$\mathcal{V}2$” and “$\mathcal{V}4$”: A fresh packet arrival of class m preempts the current service of class n with probability $p^s_{m,n}$. After that, the preempted class n can either join an empty buffer (case “$\mathcal{V}2$”) or preempt the buffered class i $(n+1 \le i \le M)$ with probability $p^w_{n,i}$ (case “$\mathcal{V}4$”).
  - Case “$\mathcal{V}5$”: A new arrival of class m preempts the ongoing service of class i $(m+1 \le i \le n-1)$ with probability $p^s_{m,i}$. However, the preempted class i fails to preempt the waiting class n with probability $(1 - p^w_{i,n})$.
  - Case “$\mathcal{V}6$” and “$\mathcal{V}7$”: These cases constitute the self-preemption possibility of the class of interest h, whenever being at the server (case “$\mathcal{V}6$”) or at the buffer (case “$\mathcal{V}7$”).

- $L_{\mathcal{W}}$: This transition set incorporates all transitions l, whose destination state $q'_l \in \mathcal{W} = \{(m^*, n)|2 \le m \le M, 1 \le n \le M \backslash m\}$. It should be noted that, by these transitions, the protection feature of the served class has been gained/kept. Accordingly, we have the following cases:

  - Case “$\mathcal{W}1$”: The fresh packet arrival of class n cannot preempt the protected service of class m. Instead, it can preempt the buffered class i $(n+1 \le i \le M \backslash m)$ with probability $p^w_{n,i}$.

Table 6 Transition table of the SHS CTMC, considering that class 2 is the one of interest (h=2)

sets	cases	$q_l \to q_l'$	$\lambda^{(l)}$
$L_{\mathcal{T}}$	$\mathcal{T}1$	$(1,0) \to (0,0)$	μ_1
		$(3,0) \to (0,0)$	μ_3
	$\mathcal{T}2$	$(2,0) \to (0,0)$	μ_2
$L_{\mathcal{U}}$	$\mathcal{U}1$	$(0,0) \to (1,0)$	λ_1
		$(0,0) \to (2,0)$	λ_2
		$(0,0) \to (3,0)$	λ_3
	$\mathcal{U}2$	$(2,3) \to (3,0)$	μ_2
	$\mathcal{U}2'$	$(1,3) \to (3,0)$	μ_1
		$(1,2) \to (2,0)$	μ_1
	$\mathcal{U}3$	$(2^*,1) \to (1,0)$	μ_2
		$(2^*,3) \to (3,0)$	μ_2
	$\mathcal{U}3'$	$(3^*,1) \to (1,0)$	μ_3
		$(3^*,2) \to (2,0)$	μ_3
	$\mathcal{U}4$	$(2,0) \to (2,0)$	$\lambda_2 p^s_{2,2}$
$L_{\mathcal{V}}$	$\mathcal{V}1$	$(1,0) \to (1,2)$	λ_2
		$(1,0) \to (1,3)$	λ_3
		$(2,0) \to (2,3)$	λ_3
	$\mathcal{V}2$	$(2,0) \to (1,2)$	$\lambda_1 p^s_{1,2}$
		$(3,0) \to (1,3)$	$\lambda_1 p^s_{1,3}$
		$(3,0) \to (2,3)$	$\lambda_2 p^s_{2,3}$
	$\mathcal{V}3$	$(1,3) \to (1,2)$	$\lambda_2 p^w_{2,3}$
	$\mathcal{V}4$	$(2,3) \to (1,2)$	$\lambda_1 p^s_{1,2} p^w_{2,3}$
	$\mathcal{V}5$	$(2,3) \to (1,3)$	$\lambda_1 p^s_{1,2} \times (1 - p^w_{2,3})$
	$\mathcal{V}6$	$(2,3) \to (2,3)$	$\lambda_2 p^s_{2,2}$
	$\mathcal{V}7$	$(1,2) \to (1,2)$	$\lambda_2 p^w_{2,2}$
$L_{\mathcal{W}}$	$\mathcal{W}1$	$(2^*,3) \to (2^*,1)$	$\lambda_1 p^w_{1,3}$
		$(3^*,2) \to (3^*,1)$	$\lambda_1 p^w_{1,2}$
	$\mathcal{W}2$	$(2,0) \to (2^*,1)$	$\lambda_1 (1 - p^s_{1,2})$
		$(3,0) \to (3^*,1)$	$\lambda_1 (1 - p^s_{1,3})$
		$(3,0) \to (3^*,2)$	$\lambda_2 (1 - p^s_{2,3})$
	$\mathcal{W}3$	$(2,3) \to (2^*,1)$	$\lambda_1 (1 - p^s_{1,2}) \times p^w_{1,3}$
	$\mathcal{W}4$	$(2,3) \to (2^*,3)$	$\lambda_1 (1 - p^s_{1,2}) \times (1 - p^w_{1,3})$
	$\mathcal{W}5$	$(2^*,1) \to (2^*,1)$	$\lambda_2 p^s_{2,2}$
		$(2^*,3) \to (2^*,3)$	$\lambda_2 p^s_{2,2}$
	$\mathcal{W}6$	$(3^*,2) \to (3^*,2)$	$\lambda_2 p^w_{2,2}$

- Case “$\mathcal{W}2$” and “$\mathcal{W}3$”: the unprotected served class m has gained the protection feature due to the fail preemption from a HP class n $(n < m)$ with probability $(1 - p^s_{n,m})$. Hence, the fresh arrival class n can instead enter the empty buffer (case “$\mathcal{W}2$”) or preempt the buffered class i, for $m + 1 \leq i \leq M$ (case “$\mathcal{W}3$”).

- Case "$\mathcal{W}4$": the unprotected served class m has acquired the protection feature due to the fail preemption from a HP class i $(1 \leq i \leq m-1)$ with probability $(1-p^{s}_{i,m})$. In addition, the fresh arrival class i fails also to preempt the buffered class n with probability $(1-p^{w}_{i,n})$.
- Case "$\mathcal{W}5$" and "$\mathcal{W}6$": These cases describe the self-preemptions of the class of interest h. The cases that seem to be similar to the aforementioned cases "$\mathcal{V}6$" and "$\mathcal{V}7$".

According to the foregoing explanation of the transition Table 2, an example on a three-class network is demonstrated through the state transition diagram (Fig. 19) and the corresponding transition table (Table 6), considering that class 2 is the one of interest.

Acknowledgements This research work is supported by the Egyptian Ministry of Higher Education (MoHE) grant under the supervision of the Egypt-Japan University of Science and Technology (E-JUST).

Author contributions The individual contributions of the authors are as follows: TEF conceptualization; formal Analysis; investigation; methodology; software; visualization; writing, original draft preparation; writing, review and editing. SIR conceptualization; formal analysis; methodology; writing, review and editing; supervision; project administration. AHA conceptualization; formal analysis; investigation; methodology; writing, review and editing; supervision. WKZ conceptualization; investigation; writing, review and editing; supervision.

Funding Open access funding provided by The Science, Technology & Innovation Funding Authority (STDF) in cooperation with The Egyptian Knowledge Bank (EKB).

Data availability Not applicable

Code availability Not applicable

Declarations

Conflict of interest The authors declare that they have no competing interests as defined by Springer, or other interests that might be perceived to influence the results and/or discussion reported in this paper.

Ethical approval Not applicable.

References

1. Kaul SK, Yates RD, Gruteser M (2012) Real-time status: how often should one update? In: 2012 Proceedings IEEE INFOCOM. IEEE, Orlando, FL, USA, pp 2731–2735

2. Talak R, Karaman S, Modiano E (2016) Speed limits in autonomous vehicular networks due to communication constraints. In: IEEE 55th conference on decision and control (CDC), Las Vegas, NV, USA, pp 4998–5003
3. Abbas Q, Zeb S, Hassan SA, Mumtaz R, Zaidi SAR (2020) Joint optimization of age of information and energy efficiency in IoT networks. In: Proceeding of IEEE 91st vehicular technol. Conference (VTC2020-Spring), Antwerp, Belgium, pp 1–5
4. Sun Y, Uysal-Biyikoglu E, Yates RD, Koksal CE, Shroff NB (2017) Update or wait: how to keep your data fresh. IEEE Trans Inf Theory 63(11):7492–7508
5. Yu W, Liang F, He X, Hatcher WG, Lu C, Lin J, Yang X (2017) A survey on the edge computing for the internet of things. IEEE Access 6:6900–6919
6. Costa M, Codreanu M, Ephremides A (2016) On the age of information in status update systems with packet management. IEEE Trans Inf Theory 62(4):1897–1910
7. Abbas Q, Hassan SA, Pervaiz H, Ni Q (2021) A Markovian model for the analysis of age of information in IoT networks. IEEE Wirel Commun Lett 10(7):1596–1600
8. Kaul SK, Yates RD, Gruteser M (2012) Status updates through queues. In: 2012 46th annual conference on information sciences and systems (CISS), Princeton, NJ, USA, pp 1–6
9. Bedewy AM, Sun Y, Shroff NB (2017) Age-optimal information updates in multihop networks. In: Proceeding of IEEE international symposium on information theory (ISIT), Aachen, Germany, pp 576–580
10. Kuang Q, Gong J, Chen X, Ma X (2019) Age-of-information for computation-intensive messages in mobile edge computing. In: 2019 11th international conference on wireless communications and signal processing (WCSP), pp 1–6. IEEE
11. Gong J, Kuang Q, Chen X, Ma X (2019) Reducing age-of-information for computation-intensive messages via packet replacement. In: 2019 11th international conference on wireless communications and signal processing (WCSP), pp 1–6. IEEE
12. Yates RD, Kaul SK (2018) The age of information: real-time status updating by multiple sources. IEEE Trans Inf Theory 65(3):1807–1827
13. Yates RD (2020) The age of information in networks: moments, distributions, and sampling. IEEE Trans Inf Theory 66(9):5712–5728
14. Tang Z, Sun Z, Yang N, Zhou X (2023) Age of information of multi-user mobile edge computing systems. IEEE Open J Commun Soc
15. Maatouk A, Sun Y, Ephremides A, Assaad M (2020) Status updates with priorities: lexicographic optimality. In: Proceeding of IEEE 18th international symposium on modeling and optimization in mobile, ad hoc, and wireless networks (WiOPT), Volos, Greece, pp 1–8
16. Paymard P, Mokari N (2022) Resource allocation in PD-NOMA-based mobile edge computing system: multiuser and multitask priority. Trans Emerg Telecommun Technol 33(3):3631
17. Kaul SK, Yates RD (2018) Age of information: updates with priority. In: Proceeding of IEEE international symposium on information theory (ISIT), Vail, CO, USA, pp 2644–2648
18. Maatouk A, Assaad M, Ephremides A (2019) Age of information with prioritized streams: when to buffer preempted packets? In: Proc. IEEE international symposium on information theory (ISIT), Paris, France, pp 325–329
19. Najm E, Nasser R, Telatar E (2019) Content based status updates. IEEE Trans Inf Theory 66(6):3846–3863
20. Xu J, Gautam N (2020) Peak age of information in priority queuing systems. IEEE Trans Inf Theory 67(1):373–390
21. Abdollahi MP, Azarhava H, Haghrah A, Niya JM (2022) On the rate and age of information for non-preemptive systems with prioritized arrivals and deterministic packet deadlines in IoT networks. Ad Hoc Netw 124:102717–102729
22. Jaiswal NK (1968) Priority queues. Elsevier, The Netherlands
23. Kim K (2012) T-preemptive priority queue and its application to the analysis of an opportunistic spectrum access in cognitive radio networks. Comput Oper Res 39(7):1394–1401
24. Fahim TE, Rabia SI, Abd El-Malek AH, Zahra WK (2022) Analyzing age of information in prioritized status update systems using probabilistic hybrid discipline. In: Proceeding of 12th international conference on simulation and modeling methodologies, technologies and applications (SIMULTECH), Lisbon, Portugal, pp 151–162
25. Zhang J, Xu Y (2022) Age analysis of status updating system with probabilistic packet preemption. Entropy 24(6):785–820

Authors and Affiliations

Tamer E. Fahim[1,2] · **Sherif I. Rabia**[2] · **Ahmed H. Abd El-Malek**[3] · **Waheed K. Zahra**[1,4]

✉ Tamer E. Fahim
tamer.essam@ejust.edu.eg; tamer.essam@alexu.edu.eg

Sherif I. Rabia
sherif.rabia@alexu.edu.eg

Ahmed H. Abd El-Malek
ahmed.abdelmalek@ejust.edu.eg

Waheed K. Zahra
waheed.zahra@ejust.edu.eg

[1] Institute of Basic and Applied Sciences, Egypt-Japan University of Science and Technology (E-JUST), New Borg El-Arab City, Alexandria 21934, Egypt

[2] Department of Engineering Mathematics and Physics, Faculty of Engineering, Alexandria University, Alexandria 21544, Egypt

[3] Department of Electronics and Communications Engineering, Egypt-Japan University of Science and Technology (E-JUST), New Borg El-Arab City, Alexandria 21934, Egypt

[4] Department of Engineering Physics and Mathematics, Faculty of Engineering, Tanta University, Tanta 31521, Egypt

26. Akar N, Doğan O, Atay EU (2020) Finding the exact distribution of (peak) age of information for queues of PH/PH/1/1 and M/PH/1/2 type. IEEE Trans Commun 68(9):5661–5672
27. Doğan O, Akar N (2021) The multi-source probabilistically preemptive M/PH/1/1 queue with packet errors. IEEE Trans Commun 69(11):7297–7308
28. Jain M, Singh V, Rani A (2019) A novel nature-inspired algorithm for optimization: squirrel search algorithm. Swarm Evolut Comput 44:148–175

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.