Measurement Based Delay and Jitter Constrained Wireless Scheduling With Near-Optimal Spectral Efficiency

Geetha Chandrasekaran, *Member, IEEE*, Gustavo de Veciana, *Fellow, IEEE*, Vishnu V. Ratnam, *Senior Member, IEEE*, Hao Chen, and Charlie Zhang, *Fellow, IEEE*

***Abstract*— We introduce two classes of measurement-based wireless schedulers. The Opportunistic Guaranteed Rate Scheduler (OGRS) meets a user's delay constraints by opportunistically allocating the user the equivalent of a fixed service rate, which for a leaky-bucket constrained traffic ensures the delay requirements are met. By contrast, the Opportunistic Guaranteed Delay Schedulers (OGDS) schedules data transmissions when the current channel is better than what is expected in the time window before packet deadlines expire. Meeting such delay requirements requires a complementary admission control policy. We exhibit a simple measurement based policy, that indirectly accounts for heterogeneity in traffic, channel, and delay constraints by monitoring the statistics of user's aggregate resource usage. We show that the spectral efficiency of our proposed approach is stochastically better than a wireless guaranteed rate scheduler. We bound spectral efficiency by considering an *optimal offline policy* with access to future channel rates and show via extensive simulations that OGRS can be within 10%-40% of the bound whereas OGDS is within 10% of the bound for a range of delay constraints. Additionally, we demonstrate that OGDS can exhibit better spectral efficiency at higher delay deadlines than schedulers leveraging neural network based predictions for future channel rates.**

***Index Terms*— Delay deadline, opportunistic scheduling, low latency, QoS constraints, leaky bucket, URLLC scheduling.**

I. Introduction

THE support of Ultra Reliable Low Latency Communication (URLLC) is expected to be critical towards enabling next generation [1] wireless applications such as industrial automation, augmented and virtual reality, autonomous driving, remote diagnosis, and health care. The key challenge in supporting such applications is their stringent constraints on Quality of Service (QoS). The latency constraints on the wireless downlink for these applications range between 5 and 30 ms, with reliability (percentage of error-free transmissions in packets) requirements of 99.9 to 99.9999%, see e.g., [2]. Moreover, given the limited spectrum available and associated costs, it is also critical to deliver such URLLC based services in a spectrally efficient manner. In general, this is challenging, e.g., one must add substantial upfront redundancy to meet reliability requirements without delays associated with retransmissions, or given low latency requirements one may not be able to exploit opportunism or wait for data to achieve more efficient modes of transmission.

In addition to dealing with the requirements of URLLC traffic, it is also critical to devise resource allocation and scheduling strategies that enable the support, of a mix of traffic, e.g., Enhanced Mobile Broadband (eMBB) and Machine-Type Communications (MTC) traffic, and possibly network slices provisioned to support different classes of applications. Our focus in this paper will be on spectrally efficient scheduling of wireless user traffic with possibly heterogeneous delay deadlines, perhaps the most challenging traffic class, yet we aim to provide an approach that can be combined with other scheduling policies, e.g., proportionally fair or utility maximizing schedulers used to support eMBB traffic, to manage an assortment of services with diverse QoS requirements. There is a substantial literature in wireless (and wireline) scheduling that provides different tools to address the above challenge, yet, as discussed below, it still falls short in many respects. Below, we briefly highlight some of that literature and the associated shortcomings. We then introduce the key contributions of this paper.[1]

A. Related Work

Many works have focused on a setting where users' data queues are *fully backlogged.* When this is the case, one can consider devising schedulers that maximize the sum of the users' utility of their allocated long term rate [5]. For example, Proportionally Fair (PF) wireless scheduling emerges when users have log utility functions, see e.g., [6], and results in a scheduler that realizes a good tradeoff between

Received 14 August 2024; accepted 1 October 2024; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor T. Lan. This work was supported by Samsung Research America, an affiliate of the 6G@University of Texas (UT) Center within the Wireless Networking and Communications Group, The University of Texas at Austin. *(Corresponding author: Geetha Chandrasekaran.)*

Geetha Chandrasekaran and Gustavo de Veciana are with the Department of ECE, The University of Texas at Austin, Austin, TX 78712 USA (e-mail: geethac@utexas.edu).

Vishnu V. Ratnam, Hao Chen, and Charlie Zhang are with Samsung Research America, Mountain View, CA 94043 USA.

Digital Object Identifier 10.1109/TNET.2024.3481066

[1]This manuscript is an extended journal version of our previous work in [3] G. Chandrasekaran, G. de Veciana, V. Ratnam, H. Chen, C. Zhang, *"Spectrally Efficient Guaranteed Rate Scheduling for Heterogeneous QoS Constrained Wireless Networks"*, IEEE WiOpt, Aug 2023. and [4], G. Chandrasekaran, G. de Veciana, V. Ratnam, H. Chen, C. Zhang, *"Delay and Jitter Constrained Wireless Scheduling With Near-Optimal Spectral Efficiency"*, IEEE PIMRC, Sep 2023.

opportunistically scheduling users which have good channels versus achieving a *fair* long term allocation amongst the users. Variations on these ideas have been proposed where the users' utility is a function of the short term throughput, see e.g., [7]. This leads to a more responsive allocation avoiding short term neglect of any user. In practice, PF, and other utility maximizing schedulers, provide a simple and effective strategy for best effort or enhanced Mobile Broadband (eMBB) traffic with no strict delay requirements. In general, utility-maximizing schedulers work best for elastic traffic with no hard deadlines. When hard delay deadlines are considered, either all users have a homogeneous or time-synchronized traffic model [8] or the problem can only be solved if the optimization problem is feasible, i.e., if all users are able to meet the delay deadlines [9]. Still, questions remain as to what happens when user queues are not fully backlogged or how to choose the fairness criterion, i.e., utility functions when there are delay constraints that require high reliability.

In settings where users' queues are not fully backlogged, researchers have focused on devising queue and channel dependent wireless schedulers which are *throughput optimal*, i.e., ensure user queues' stability whenever feasible. These schedulers also address performance objectives, such as Max-Weight [10], which is delay optimal in the idealized symmetric case, Exponential rule [11] which attempts to minimize the max user queue, Log rule [12] which attempts to minimize the mean delay and a variant of Exponential rule that supports real time and non real time QoS [13]. Such schedulers have been adapted to more practical settings, such as the Modified Largest Weighted Delay First (MLWDF) [14] which schedules users based on head-of-line packet delays, current channels, and other hyperparameters (like queue lengths [15]) reflecting user QoS and resource allocation objectives. In practice, such schedulers do meet delay constraints (with high probability) if sufficient resources have been provisioned, yet it is difficult to verify when this is true, and as such provide a graceful degradation across users when this is not the case.

Another class of wireless schedulers was born from modifying/ adapting ideas from wireline scheduling (e.g., traffic shaping and network calculus [16], [17]) to meet QoS requirements under wireless channel variations. For instance, weighted round robin [18] or weighted fair queueing [19] employ user weights drawn from heuristics or tokens [20] based on service deficit [21] to either minimize the average delay or provide a graceful degradation of service. Much of the above mentioned work focuses on scheduling one class of users, e.g., best effort users sensitive to throughput, or traffic that is sensitive to packet delays. In practice, wireless systems need to be shared by heterogeneous user classes.

While many schedulers in the existing literature address delay constraints for real-time traffic, spectral efficiency is often neglected, leading to lesser resource availability for non-real-time traffic and higher network congestion. For instance, there are a variety of online learning algorithms that promise near-optimal packet scheduling for deadline-constrained algorithms. However, they are often limited by either i.i.d assumptions on user traffic [22], or assume symmetric user channel conditions [23], ignoring heterogeneous/ dynamic user link capacities [24] or neglecting the challenges of multi-user scheduling [25]. More recent literature in wireless networks that consider hard delay deadlines [26], [27], [28] either assumes i.i.d traffic arrivals or considers a simplistic binary link capacity instead of the more practical rate adaptive modulation and coding considered in this paper. In this paper, we focus on not only developing a scheduler that meets delay constraints but one that does so in a spectrally efficient manner.

No practical wireless scheduling policy is complete without a complementary strategy for admission control and/ or traffic shaping. Given the uncertainty and heterogeneity associated with traffic, channels, and user requirements in a wireless system, it is virtually impossible to devise good models that would allow one to predict if the users' QoS requirements will be met under a given scheduling policy. While there have been many works in literature that propose Measurement Based Admission Control (MBAC) [29], [30], [31], we note that meeting packet delay and loss targets in buffered systems is challenging [32]. In contrast to traditional MBAC approaches, our approach directly measures the aggregate resource that our delay constrained schedulers are using thus indirectly capturing the impact of the users' traffic, channel variability and delay constraints. Building on [3] and [4], we provide an in-depth analysis of proposed algorithms' spectral efficiency and a variety of practical considerations.

B. Our Contributions

We propose several classes of wireless schedulers which under appropriate assumptions can meet heterogeneous delay deadlines and do so in a spectrally efficient manner such that the more relaxed the constraint the more efficient. A key underlying idea is that if traffic is subject to tight deadlines, the system does not have as much flexibility on when to schedule a user's packets. This in turn forces transmissions to be scheduled when the channel rate may be poor and thus spectrally inefficient. Therefore, it is desirable to have more relaxed delay constraints that the scheduler can exploit to achieve improved spectral efficiency. This permits one to devise a scheduler, the Wireless Guaranteed Rate Service (WGRS), which will ensure a user will see a fixed service rate even with channels that have stochastic variations. If a user's traffic is leaky bucket constrained, one can determine a minimum fixed service rate which will ensure a desired maximum delay.

Any scheduler which allocates at least as much cumulative service as the GRS scheduler over busy periods is GRS *compliant*, and will thus also meet the user's delay deadlines. This observation suggests the possibility of opportunistically serving a user's data ahead of time when channel rates are good, relative to the GRS scheduler, and/ or delaying such service when channel rates are poor, as long as the scheduling is GRS compliant. We devise a class of Opportunistic GRS (OGRS) schedulers that take advantage of this relaxation along with knowledge of the statistics of the users' channel variations, to achieve better spectral efficiency while meeting users' strict delay constraints. While OGRS is opportunistic

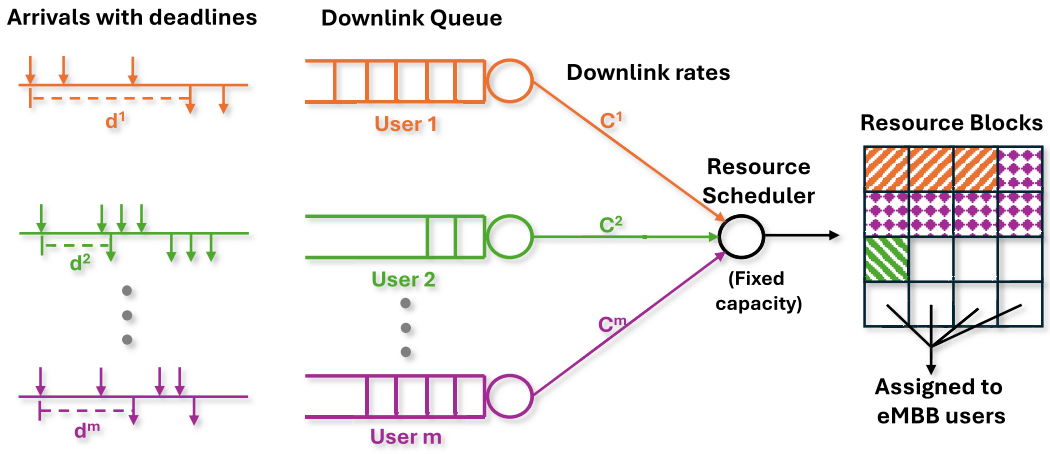


Fig. 1. Illustration of deadline-constrained user traffic with heterogeneous link rates across URLLC users.

and meets the delay constraints, the underlying requirement to provide a minimum rate for each time slot is limiting. We propose an alternative approach, denoted Opportunistic Guaranteed Delay Scheduling (OGDS) that schedules data opportunistically based on the statistics of its channel's temporal variations and the remaining time window until its deadline expires.

First, we establish a stochastic ordering between the resource requirements of OGRS and WGRS scheduling algorithms. Next, by considering *offline policies* with access to future channel capacity realizations, we derive a bound on the spectral efficiency that any delay constrained schedulers could achieve. We show via extensive simulations that OGRS can be within 10% to 40% of the bound as the delay constraint is relaxed. Meanwhile, OGDS is within 10% of the bound for a range of delays that we have simulated so far, up to 10 ms. These gains translate to doubling the eMBB user's throughput even for the users with the weakest wireless channels when URLLC and eMBB traffic share resources. We also observe an increase of up to 57% in the number of users that can be supported. This paper further explores the impact of various additional issues critical to wireless scheduling including transmission errors, Hybrid Automatic Repeat Request (HARQ), user mobility, and the time scales on which to estimate the empirical distribution of channel variations, to show how our proposed approach would fare in practice. Finally, we also compare the spectral efficiency of OGRS and OGDS with delay constrained schedulers leveraging neural network based forecasts of future channel rates. We demonstrate regimes where the spectral efficiency of schedulers using empirical statistics (OGRS, OGDS) is higher than those that employ neural network predictions and vice versa.

This paper is organized as follows. Section II describes the system model for our work. Section III describes our proposed algorithms for delay constrained scheduling. Section IV presents the main theoretical results of our work. Section V provides extensive simulations for some practical wireless network settings, and also evaluates the spectral efficiency of a natural class of delay constrained schedulers that use neural network based channel rate predictions. Finally, Section VI includes some concluding remarks.

II. System Model

We consider discrete time downlink scheduling for a base station serving a set $\mathcal{U}$ of URLLC users with stochastic arrivals and possibly heterogeneous QoS requirements and a set $\mathcal{E}$ of backlogged eMBB users. We denote by $(A_n^u)_{n\in\mathbb{N}}$ the arrival process for user $u \in \mathcal{U}$, where A_n^u is a random variable denoting the number of bits that arrive and are available for service in time slot n with a transmission deadline of $n + d^u$, where d^u is the delay constraint for user u. In general, it is not possible to ensure delay guarantees to a user without prior knowledge of its traffic statistics or constraints on its traffic. A common approach for the latter is to establish and enforce (through traffic policing/ shaping) apriori constraints on the user's traffic that can be used to design resource allocation mechanisms guaranteed to meet a user's QoS requirements. Fig. 1 illustrates our high level system model with stochastic arrivals for URLLC users, where the delay deadlines are heterogeneous across users and the instantaneous wireless channel rates $(C^u)_{u\in\mathcal{U}}$ are non-identically distributed across users. Note that the leaky bucket parameters for each user are different and selected based on the specific traffic characteristics of the user. Also, resources are first allocated to delay constrained URLLC users with priority, and leftover resources, if any, are assigned to eMBB users.

We will assume each user's traffic satisfies dual leaky bucket constraints [16] with parameters $(\rho^u, \sigma^u, \mu^u)$, where σ^u denotes the token bucket size in bits and ρ^u, μ^u denote the peak and mean bit arrival rate per time slot, respectively. The leaky bucket algorithm is used in data networks to regulate the traffic that a user can send/ receive on a network based on pre-agreed parameters that can be viewed as a service level agreement between the user and service provider reflecting the user's requirements. Enforcement of such constraints is achieved through the leaky bucket algorithm where the users'

arriving data draws on tokens that arrive at a fixed rate to a finite capacity bucket. When traffic satisfies such constraints, one can determine the fixed service rate needed to meet a fixed delay deadline. Note that the leaky bucket parameters should be chosen to represent the users' traffic characteristics.

Remark: Note that while OGRS (introduced in the next section) exploits information on the traffic shaping parameters to design an opportunistic scheduling rule, such strict traffic shaping is not required for the OGDS policy, which only requires the peak bit arrival rate of the user be bounded.

The user's cumulative arrival process $A^u(\cdot,\cdot]$ is thus constrained as follows for all $\tau, n \in \mathbb{N}$,

$$A^u(\tau,\tau+n] = \sum_{k=\tau+1}^{\tau+n} A_k^u \leq \min\left[\rho^u n, \sigma^u + \mu^u n\right]. \quad (1)$$

The base station's transmit resources are modeled as a sequence of frames/ slots comprising multiple Resource Blocks (RBs), which the scheduler can allocate arbitrarily to users on a per slot basis. The BS has access to $\overline{m}$ RBs for user resource allocation. We consider a system where enough resources are available at the base station for both URLLC and eMBB users, such that the total requirement of all URLLC users under our scheduling policy is less than the fixed number of resources $\overline{m}$ available to the BS. This will be ensured by appropriate admission control on URLLC users and prioritizing URLLC traffic over eMBB traffic as needed. This is a fairly reasonable assumption considering the large data rate requirements of eMBB users and the short packet sizes of URLLC users. For each RB, i.e., slice of time and frequency, we let the random variable $C_n^u \in \mathbb{R}^+$ denote the channel rate (bits per RB) that can be transmitted to user u if it is allocated a *single* RB on time slot n. While C_n can be small we assume that each RB has a non-zero effective transmission rate:

Assumption 1 (Connectivity Assumption): The BS can transmit data over an RB at a non-zero channel rate $C_n^u > 0$ with probability 1.

Remark: This can viewed as a coverage/ connectivity requirement for URLLC users which is met using sufficiently strong coding and/ or multiple antennas which is either met with probability 1 or with a probability sufficiently high to far exceed the desired reliability associated with users' QoS guarantees.

A user may be allocated multiple RBs, but we assume a *flat fading* setting where the rate delivered to user u is the same across RBs in a given time slot. Note that one can also address the frequency selective case as follows. To that end, one can adopt a model for frequency selective fading (for URLLC users) where each sub-band is assumed to experience flat fading, see e.g., [33]. The algorithm we proposed in this paper can easily be extended to address frequency-selective channels by empirically tracking the rate distributions *across* subbands, and resources can in turn be allocated based on the thresholds calculated using the overall empirical distribution. Additionally, a single RB may be allocated to only one user in a given time slot. Further, we assume $(C_n^u)_{n\in\mathbb{N}}$ are independent and identically distributed (i.i.d.) across time slots. Additionally, we also assume that a sufficiently large number of RBs are available every time slot to meet each user's QoS requirements. In the sequel, we propose admission control techniques that will limit the total number of users in the system and thus ensure resource availability.

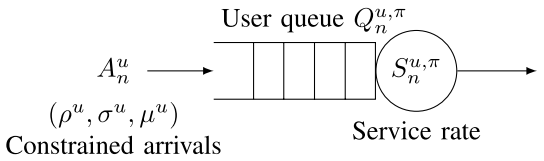


Fig. 2. Leaky bucket constrained arrivals to a discrete time queue with a service rate controlled by scheduling policy π.

We consider a system model where a scheduling policy, say π, decides the number of RBs to be allocated to each user in each time slot. The decision of policy π at time n is assumed to be causal concerning knowledge of the current and past channel rates $(C_\tau^u)_{\tau=0}^n$, arrivals and queue lengths, allowing for *opportunistic scheduling*, i.e., taking advantage of capacity variations across time. In particular, we let $M_n^{u,\pi} \in \mathbb{R}^+$ denote the number of RBs allocated to user u on slot n by a policy π given the observed history. Such an allocation provides an overall service rate $S_n^{u,\pi}$ (total bits transmitted with potentially multiple RBs allocated) to the user u on time slot n given by,

$$S_n^{u,\pi} = M_n^{u,\pi} C_n^u,$$

and we define the cumulative service over an interval $(\tau,\tau+n]$ as follows,

$$S^{u,\pi}(\tau,\tau+n] = \sum_{k=\tau+1}^{\tau+n} S_k^{u,\pi}. \quad (2)$$

A user's data queue (in bits) is modeled as a First Come First Serve (FCFS) discrete time queue with arrivals A_n^u and service rate $S_n^{u,\pi}$ as shown in Fig. 2. We let $Q_{n+1}^{u,\pi}$ denote the number of bits in the user's queue at the start of slot $n+1$, then

$$Q_{n+1}^{u,\pi} = [Q_n^{u,\pi} - S_n^{u,\pi}]^+ + A_{n+1}^u. \quad (3)$$

III. Opportunistic Delay Constrained Schedulers

In this section, we will introduce several delay constrained wireless schedulers. For the sake of brevity, we will drop the user index (marked by superscript u) as we consider per-user schedulers. The user index will be reintroduced in the sequel when we consider admission control. We propose a scheduler that evaluates the rate required for each user's QoS requirements and provision resources such that delay deadlines can be met. To do so, the scheduler needs information on each user's queue length and the distribution of the wireless channel realizations. Catering to strict delay deadlines requires a disciplined approach to resource allocation with a certain level of regularity in service. We use leaky bucket traffic constraints to perform a worst-case delay analysis that provides a minimum rate of service required to satisfy latency constraints. The assumed traffic shaping serves to constrain the peak *burstiness* of the arrival process. We show that it is feasible

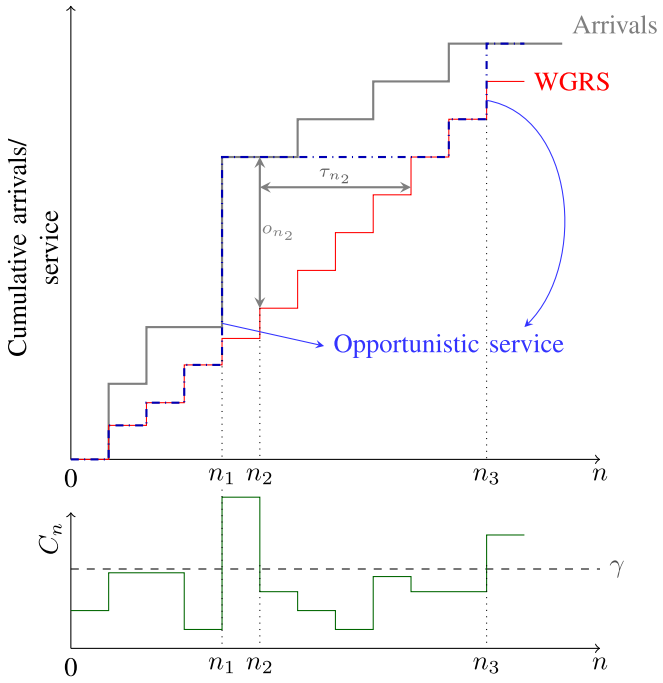


Fig. 3. Temporal channel variations and opportunistic service based on bits in the queue.

to schedule opportunistically over temporal variations in the wireless channel without compromising on QoS guarantees.

A. Opportunistic Guaranteed Rate Schedulers

Definition 1 (Threshold-Based Opportunistic GRS(s)): The basic principle underlying a threshold-based OGRS(s) scheduling policy π is as follows: if on time slot n the channel rate C_n exceeds a threshold γ_n^π, then a sufficient number of RBs are allocated by the scheduler to clear the queue backlog, i.e, $M_n^\pi = Q_n^\pi/C_n$. Otherwise, a minimal number of RBs are allocated to ensure the cumulative service allocated keeps up with that of the GRS(s) scheduler over its busy cycles.

OGRS(s) schedulers ensure that the user will see a fixed service rate of s or better. If s is chosen appropriately based on the user's leaky bucket parameters and desired delay constraint, this will ensure the QoS requirements are met. Such schedulers exploit the freedom to decide when to allocate RBs to the user and in particular, to do so when the channels are particularly good. Note, that the threshold γ_n^π can be time/ state dependent and controls how the algorithm exploits channel rate fluctuations – this will be explained in the sequel.

Fig. 3 exhibits the perspective underlying Opportunistic GRS scheduling. *The key idea is to exploit temporal channel rate variability to improve spectral efficiency without impacting delay guarantees.* We observe that at times n_1 and n_3 the user's channels are particularly good, and the user has queued data significantly higher than s. Our proposed scheduler chooses to exploit these good user channels, by serving much more data at those times than the minimal service rate required by GRS(s) scheduling, see the blue dash-dotted curve in Fig. 3. In principle, since the user's channel is good at those times, the number of RBs allocated by the wireless scheduler would be much lesser as compared to the WGRS(s) scheduler (introduced in the previous section). Next, we formally introduce a class of Opportunistic GRS(s) scheduling policies.

Algorithm 1 exhibits the details of the threshold based OGRS(s) scheduler which operates with respect to the cumulative service a virtual GRS scheduler would provide for the same arrival process. To start with, consider a GRS(s) busy cycle that without loss of generality begins at 0. Then one can express the user queue length and service for the GRS(s) at time n as follows,

$$Q_n^{\text{GRS}} = [Q_{n-1}^{\text{GRS}} - s]^+ + A_n,$$
$$S_n^{\text{GRS}} = \min[Q_n^{\text{GRS}}, s].$$

Therefore, the cumulative service provided by GRS(s) over an interval $(0, n]$ can be expressed as,

$$S^{\text{GRS}}(0, n] = S^{\text{GRS}}(0, n-1] + \min[Q_n^{\text{GRS}}, s]\,.$$

Next, we have the OGRS policy which needs a metric that can measure the amount of service provided in excess of the guaranteed minimum rate s per time slot. Let O_n^π denote the amount of data that has been (opportunistically) sent ahead of time n relative to the GRS(s) scheduler, i.e.,

$$O_n^\pi = S^\pi(0, n-1] - S^{\text{GRS}}(0, n-1].$$

We initialize $O_0^\pi = 0$ at the start of a busy cycle. Note that the duration of a busy cycle of a GRS(s) compliant scheduling policy with leaky bucket constrained arrivals is upper bounded [16] by $b_{\max}(s) = \frac{\sigma}{s-\mu}$, which bounds O_n^π and guarantees it will eventually return to 0. Note that any amount of opportunism O_n^π gained translates to the scheduler being $\tau_n = \left\lceil \frac{O_n^\pi}{s} \right\rceil$ time slots ahead of the service deadline, see $o_{n_2}^\pi$ and τ_{n_2} as marked in Fig. 3.

As mentioned in the policy Definition 1 above, if $C_n > \gamma_n^\pi$ then π serves all the data in the user queue, i.e., $S_n^\pi = Q_n^\pi$. Clearly, the metric O_n^π must be positive if $Q_{n-1}^\pi > s$, because π has served all the traffic that has entered the queue since the start of the busy cycle, while GRS(s) only the bare minimum service it guarantees.

When the channel rate is not so good, i.e., if $C_n \le \gamma_n^\pi$ then OGRS can choose to **not** schedule any RBs if at least s bits had been transmitted in advance. Specifically, if the amount of excess service at time $n-1$ falls short of s then π only serves the minimum number of bits to ensure it keeps up with the GRS(s) scheduler, i.e.,

$$S_n^\pi = [\min[s, Q_n^\pi] - O_n^\pi]^+\,.$$

B. OGRS Threshold Selection

In this section, we propose various ways to design the thresholds $(\gamma_n^\pi)_n$ driving the behavior of the threshold-based OGRS scheduler. We shall assume that the scheduler has access to $F_C(\cdot)$, the CDF for the users' channel rate variations. We also assume that $F_C^{-1}(\cdot)$ is an appropriately defined inverse CDF. As explained in the sequel, in practice the CDF can be inferred, as in [34] to possibly adapt to changes over time.

Algorithm 1 Guaranteed Rate Scheduling With Opportunism Over Temporal Variations

initialize $O_0^\pi = 0, S_0^\pi = 0, S_0^{\text{GRS}} = 0$;
while $n > 0$ **do**
  if $C_n > \gamma_n^\pi$ **then**
    $S_n^\pi = Q_n^\pi$;
  else
    $S_n^\pi = [\min[s, Q_n^\pi] - O_n^\pi]^+$;
  end
  $M_n^\pi = S_n^\pi / C_n$;
  $S_n^{\text{GRS}} = \min[s, Q_n^{\text{GRS}}]$;
  $Q_{n+1}^{\text{GRS}} = Q_n^{\text{GRS}} - S_n^{\text{GRS}} + A_{n+1}$;
  $Q_{n+1}^\pi = Q_n^\pi - S_n^\pi + A_{n+1}$;
  $O_{n+1}^\pi = S^\pi(0,n] - S^{\text{GRS}}(0,n]$;
end

1) ST(α): Static Threshold: Our first threshold design is a static percentile, i.e., $\gamma_n^\pi = \gamma^\pi$ corresponding to the α-percentile of the channel rate CDF, where $\alpha \in (0,1)$, so,

$$F_C(\gamma^\pi) = \alpha \implies \gamma^\pi = F_C^{-1}(\alpha). \tag{4}$$

For example, with a choice of $\alpha = 0.8$ the OGRS(s) triggers an opportunistic scheduling of the user's queued data only if the current channel rate has exceeded the 80^{th} percentile, i.e., $c_n > \gamma^\pi$. Note that the choice percentile α is a design parameter that can in principle be optimized to minimize the mean resources (RBs) allocated by the associated OGRS(s) scheduler.

2) DTP(δ): Dynamic Threshold Based on Probability: Next, we consider thresholds based on a dynamic percentile of the channel rate CDF $F_C(\cdot)$. Recall that $O_n^\pi = o_n^\pi$ denotes the amount of data that our OGRS(s) policy has delivered *ahead* of time as compared to GRS(s) at time n. Given that the GRS(s) must serve at least s bits per slot, an OGRS(s) policy could in principle wait for $\tau_n = \left\lfloor \frac{o_n^\pi}{s} \right\rfloor$ time slots before the GRS(s) scheduler catches up and is forced to schedule at time $\tau_n + 1$. Ideally, the data should be scheduled on slot n if the current rate realization c_n is better than that to be observed in the next $\tau_n + 1$ time slots with high probability, i.e.,

$$\mathbb{P}\left(c_n > \max_{i=1,\ldots,\tau_n+1} C_{n+i}\right) \geq \delta. \tag{5}$$

The following lemma translates the above requirement to a threshold on c_n. Note that δ is a design parameter that needs to be carefully chosen to minimize the number of RBs required.

Lemma 1: Let $(C_n)_n$ be i.i.d random variables with the same marginal distribution $F_C(\cdot)$ and appropriately defined inverse $F_C^{-1}(\cdot)$. If c_n exceeds the threshold $F_C^{-1}\left(\delta^{\frac{1}{\tau_n+1}}\right)$ then (5) is satisfied.

Proof: The current channel rate realization c_n is considered good for opportunistic scheduling if,

$$\max_{i=1,\ldots,\tau_n+1} C_{n+i} < c_n,$$

$$\begin{aligned}
&\iff F_C\left(\max_{i=1,\ldots,\tau_n+1} C_{n+i}\right) \overset{(a)}{<} F_C(c_n),\\
&\iff \max_{i=1,\ldots,\tau_n+1} F_C\left(C_{n+i}\right) \overset{(b)}{<} F_C(c_n),\\
&\iff \max_{i=1,\ldots,\tau_n+1} U_i \overset{(c)}{<} F_C(c_n).
\end{aligned} \tag{6}$$

where step (a) follows from the monotonicity of the cumulative distribution function (CDF) $F_C(\cdot)$ of the wireless channel strength and step (b) follows from the commutative property of the max function with CDF $F_C(\cdot)$. Step (c) follows from the fact that $F_C(C_{n+i}) \sim U_i$ are i.i.d. Uniform$[0,1]$.

One could design a dynamic threshold to ensure that the probability of *not* seeing a better channel rate realization in the next $\tau_n + 1$ time slots is greater than a pre-specified $\delta \in (0,1)$. Such a design criterion would lead to the following,

$$\begin{aligned}
\mathbb{P}\left(\max_{i=1,\ldots,\tau_n+1} F_C\left(C_{n+i}\right) < F_C(c_n)\right) &\geq \delta\,,\\
\mathbb{P}\left(\max_{i=1,\ldots,\tau_n+1} U_i < F_C(c_n)\right) &\geq \delta\,,\\
\left(F_C(c_n)\right)^{\tau_n+1} &\overset{(a)}{\geq} \delta\,,\\
(\tau_n+1)\log F_C(c_n) &\geq \log\delta\,,\\
\implies F_C(c_n) &\geq \delta^{\frac{1}{\tau_n+1}}
\end{aligned} \tag{7}$$

where step (a) follows from the CDF of the maximum of $\tau_n + 1$ independent uniformly distributed random variables. Consequently, the threshold is chosen to be,

$$\gamma_n^\pi = F_C^{-1}\left(\delta^{\frac{1}{\tau_n+1}}\right). \tag{8}$$

□

3) DTE: Dynamic Threshold Based on Expectation: A user with a current channel rate c_n might choose not to schedule transmissions on the current slot in the hope of seeing a better channel in the next τ_n slots. The previous threshold design was based on the inequality in (5) being satisfied with high probability. Alternatively, the current channel rate c_n might be considered good if one can ensure the inequality holds on average. Taking the expectation of the inequality on the right hand side and computing the expectation of the max of uniform random variables in (6) gives,

$$F_C(c_n) > \mathbb{E}\left[\max_{i=1,\ldots,\tau_n+1} U_i\right] = 1 - \frac{1}{\tau_n+2}\,.$$

Under this rough approximation an associated threshold on c_n depends on τ_n which can be set to,

$$\gamma_n^\pi = F_C^{-1}\left(1 - \frac{1}{\tau_n+2}\right). \tag{9}$$

This captures the key insight that with a larger number of slots τ_n, an OGRS(s) scheduler can choose to wait until the channel rate exceeds the $1 - 1/(\tau_n + 2)$ percentile. Furthermore, this threshold selection mechanism does not have any design parameter which makes it easier to implement in practice.

C. Opportunistic Guaranteed Deadline Scheduling

Suppose we start with an empty user queue at $t = 0$, then the arrival process $A(0,\tau]$ delayed by d would be such that

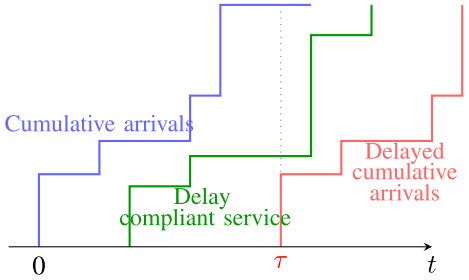


Fig. 4. Delay compliant cumulative service along with worst case delayed service curve.

$A(-d, \tau - d] = A(0, \tau - d]$. Fig. 4 depicts the cumulative arrivals $A(0, \tau]$ in blue and the corresponding delayed version in red $A(0, \tau - d]$. The delayed arrivals curve represents the worst case cumulative service that a server could provide without violating the delay constraint on each packet. Any cumulative service curve that lies within the arrivals and worst case departures curve will be delay compliant.

Definition 2: ***GDS(d)*** *We let the Guaranteed Deadline Scheduler with parameter d, GDS(d), be a scheduling policy that guarantees each bit in the user data queue be serviced within a delay of d since its arrival. Clearly,*

$$A(0, \tau] \geq S^{GDS}(0, \tau] \geq A(-d, \tau - d].$$

Definition 3 (Opportunistic GDS(d)): A threshold based OGDS(d) scheduling policy π is as follows: whenever the channel rate C_n exceeds a threshold γ_n^π, a sufficient number of RBs are allocated by the scheduler to completely clear the queue backlog, i.e, $M_n^\pi = Q_n^\pi / C_n$. Otherwise, a minimal number of RBs are allocated so as to ensure that the cumulative service of π at slot n exceeds or matches that of the d delayed cumulative arrival curve.

At each time slot, the number of slots τ_n over which there is flexibility to pick when to serve the data in the user queue depends on the residual time until the earliest deadline. Note that any data whose deadline is due to expire at a given time slot will be allocated resources in the same time slot. In case the current channel rate is expected to be better than those in the next τ_n time slots (i.e., the current rate exceeds the threshold γ_n^π), the entire queue backlog is cleared.

Algorithm 2 Guaranteed Deadline Scheduling With Opportunism Over Temporal Variations

```
initialize S_0^π = 0;
while n > 0 do
    τ_n = min[k : k ≥ n, A(0, k − d] ≥ S^π(0, k]];
    if C_n > γ_n^π then
        S_n^π = Q_n^π ;
    else
        S_n^π = [A(0, n − d] − S^π(0, n − 1]]^+ ;
    end
    M_n^π = S_n^π / C_n;
    Q_{n+1}^π = Q_n^π − S_n^π + A_{n+1} ;
end
```

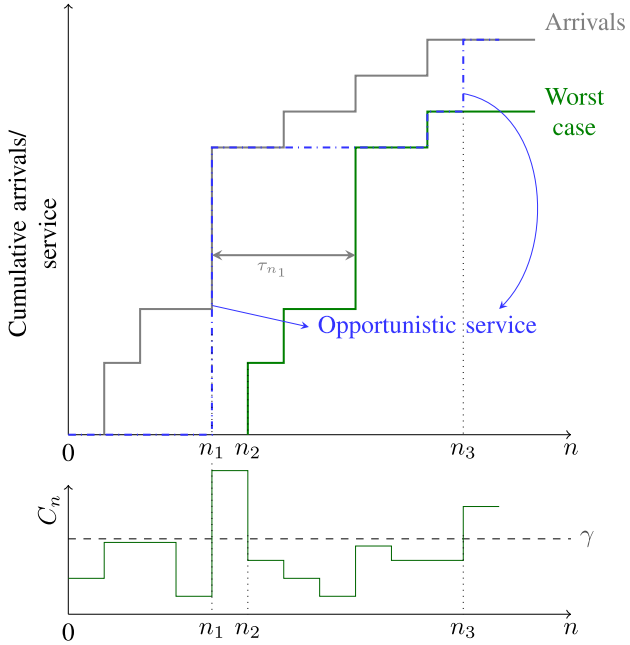


Fig. 5. Illustration of the slack available to schedule cumulative arrivals based on the worst case service curve. The bottom figure shows the time varying nature of the wireless channel rates with a fixed threshold γ to determine the channel quality.

Algorithm 2 details the steps involved in OGDS(d) scheduling. When the channel rate is above a certain threshold $C_n > \gamma_n^\pi$, the OGDS policy π serves all data in the user queue,

$$S_n^\pi = Q_n^\pi.$$

Otherwise, the scheduler π allocates only the minimum number of RBs required to meet the worst case delayed service curve, i.e.,

$$S_n^\pi = [A(0, n - d] - S^\pi(0, n - 1]]^+ .$$

Specifically, if the cumulative service provided by π until time $n-1$ is greater than that of the worst case delayed cumulative service at time n, then policy π can completely refrain from allocating any resources at time n if the channel rate is below the threshold.

OGDS Threshold selection: Define τ_n as the slack available to the scheduler before it is forced to schedule data to maintain delay guarantees, i.e.,

$$\tau_n = \min\left[k : k \geq n, A(0, k - d] \geq S^\pi(0, k]\right]. \tag{10}$$

The threshold selection is illustrated in Fig. 5. At the time n_1, the OGDS scheduler has a slack of τ_{n_1} time slots before it is forced to start servicing the user queue. Therefore, for any particular channel rate realization c_n, the channel condition is considered good for opportunistic scheduling if,

$$\mathbb{E}\left[\max_{i=1,\ldots,\tau_n+1} U_i\right] < F_C(c_n). \tag{11}$$

The left hand side is a maximum of τ_n+1 i.i.d uniform random variables which can be shown to be (see proof of Lemma 1),

$$\mathbb{E}\left[\max_{i=1,\ldots,\tau_n+1} U_i\right] = 1 - \frac{1}{\tau_n + 2} < F_C(c_n).$$

With this rough approximation an associated threshold on c_n depends on τ_n which can be set to,

$$\gamma_n^{\pi} = F_C^{-1}\left(1 - \frac{1}{\tau_n + 2}\right). \tag{12}$$

This captures the key insight that with a larger number of slots τ_n where π is not going to be forced to schedule user data, an OGRS(s) scheduler might choose to wait unless indeed it currently has a channel rate in the $1 - 1/(\tau_n + 2)$ percentile. Furthermore, the current threshold selection does not have any design parameter which makes it highly convenient for usage in practice. In the discussion above, we have assumed that the user's channel rate CDF is available. Typically, the serving BS tracks the user's Channel State Information (CSI) for adaptive modulation and coding, therefore, it is reasonable to assume that we can empirically estimate the channel rate CDF using CSI [34]. Also note that when the channel rates are discrete, we could use linear interpolation to invert the empirical CDF and compute the percentiles for the channel rate threshold.

1) Modified OGDS: While most scheduling policies for delay constrained traffic focus on improving key performance metrics such as energy efficiency [35], reliability [36], and delay, jitter is often neglected. It is a particularly important metric when transmitting periodic updates to networked real-time control and/ or interactive AR/ VR gaming applications. Disparate transmission delays across users can be undesirable/ intolerable, especially in scenarios that need synchronization of updates across all users. There are multiple ways to measure the variability of transmission delay. In this work, we define jitter in terms of the standard deviation of delay for data transmissions that are periodic.

We propose an elementary modification to the OGDS algorithm that provides a way to trade off between spectral efficiency, delay, and jitter, by carefully selecting a transmission window over which resources are allocated to the user. One could either wait for a predetermined number of transmit instants, say ζ, or artificially advance the targeted delay deadline to $d - \zeta$ to reduce packet jitter. A shorter window for transmission reduces the number of opportunities available for a user to be efficient, nevertheless, it reduces the variability in delay. Specifically, in Algorithm 2, step 10, the user queue update equation could be modified as follows,

$$Q_{n+1}^{\pi} = Q_n^{\pi} - S_n^{\pi} + A_{n+1+\zeta}\,, \tag{13}$$

where S_n^{π} denotes the service provided at time n, and $A_{n+1+\zeta}$ stands for the arrivals at time $n+1+\zeta$. In the sequel, we will refer to the parameter ζ as the jitter control parameter and demonstrate how the modified OGDS policy performs in terms of spectral efficiency and jitter.

IV. Main Results

A. Lower Bound on Spectral Efficiency

In this subsection, we state the theorem on a lower bound on the *minimum number of resource blocks* required by any wireless scheduler meeting the delay deadlines. The lower bound is based on considering an *offline* policy with complete knowledge of the future channel realizations and thus not achievable in an online setting, yet a good benchmark.

Consider a user with an arrival process $(A_n)_n$ and a time varying channel rate $(C_n)_n$ per resource block, whose traffic is subject to a delay constraint of at most d slots.

Theorem 1: For any scheduling policy π meeting the delay constraint, let N_n^{π} denote the (possibly fractional) number of resource blocks used to serve the arrivals A_n, these RBs may be allocated at the earliest on slot n, but no later than the deadline $n+d$. Similarly, we let M_n^{π} denote the total number of RBs allocated on slot n. It then follows that,

$$N_n^{\pi} \geq A_n \min_{0\leq j\leq d}\left[\frac{1}{C_{n+j}}\right] \ \ a.s.\,. \tag{14}$$

Furthermore, if the arrivals and channel rate processes are stationary and independent of each other and the policy π is such that,

$$\lim_{n\to\infty}\frac{1}{n}\sum_{\tau=1}^{n} N_{\tau}^{\pi} = \bar{N}^{\pi},$$

then the time average of $(M_n^{\pi})_n$ also converges to a limit $\bar{M}^{\pi}$, which satisfies

$$\bar{M}^{\pi} = \bar{N}^{\pi} \geq \mathbb{E}[A_1]\,\mathbb{E}\left[\frac{1}{\max\limits_{0\leq j\leq d} C_{1+j}}\right]. \tag{15}$$

Proof: See proof of [4, Theorem 1]. □

B. Stochastic Dominance

Consider a user with an arrival process $(A_n)_n$ and a time varying channel rate $(C_n)_n$ per resource block, whose traffic is subject to a delay constraint of at most d slots. The arrivals A_n are leaky bucket constrained (ρ, μ, σ) where bits arrive and are available for service at time n. Then the following theorem establishes how the number of RBs required for a given user by the WGRS policy stochastically dominates that of OGRS-DTE.

Theorem 2: For a system in steady state, the mean RBs required by OGRS per time slot is stochastically dominated by that required by WGRS.

$$\mathbb{E}[M^{WGRS}] \geq \mathbb{E}[M^{OGRS}]. \tag{16}$$

Proof: See Appendix for proof and Section V-C for simulation based results of the above theorem. □

The above theorem establishes the superiority of the OGRS-DTE policy over the strict sense service policy WGRS in the ergodic sense, which directly implies that the OGRS-DTE enables the overall network scheduler to support existing eMBB users at a higher data rate than the WGRS policy.

V. Simulation Results

We consider a BS serving a set of URLLC and eMBB users with each user's channel rate C_n per RB determined by the corresponding received Signal to Noise Ratio (SNR). The received SNR was modelled using the 3GPP Urban-Micro path loss model [1], with Rayleigh distributed small scale fading. We assume bounded channel realizations, where the SNR lies between -6.934 dB $\leq$ SNR $\leq$ 20 dB. For simplicity, we shall use the 3GPP MCS table (see [37, Table 5.2.2.1-2]) to determine the rate obtained per RB

TABLE I
LEAKY BUCKET PARAMETERS FOR MULTIPLE USERS

User	distance (m)	Delay(ms)	ρ	μ	σ
1	300	5	10	5	50
2	500	3	20	10	50
3	700	7	10	5	50

C_n, where each RB is a time frequency slice of duration 1ms with bandwidth $B = 10$ KHz. The traffic model is stochastic arrivals with a packet size of 1024 bits that arrive each time slot shaped by the leaky bucket with parameters (in packets per time slot) $\sigma = 50, \rho = 10, \mu = 5$, unless otherwise specified. Finally, to determine the channel quality thresholds, we need the CDF $F_C(\cdot)$ for the channel rate per RB on a given slot, for each user. We used the last 100 channel SNR realizations to determine the empirical CDF of the user's SNR at any given instant. Note that all plots in this section were generated over 10^6 slots, resulting in a ± 0.1 error for the estimated mean number of allocated RBs per slot $\bar{M}^\pi$ with 99% confidence interval. Additionally, we will use the WGRS policy as a baseline and also other benchmark policies such as the multicarrier version of the MLWDF [14] policy and schedulers that use neural network based forecasts of future channel rates to evaluate the promise of our proposed algorithms.

A. Performance of Opportunistic Scheduling Policies

1) Improvement in eMBB Throughput: In this subsection, we demonstrate the throughput improvement for eMBB users where the BS supports multiple heterogeneous users, 3 URLLC, and 5 eMBB users. We consider ON-OFF bursty arrivals for URLLC users, where packets arrive at a peak rate ρ during the ON period. The ON, OFF cycles are of duration $\frac{\sigma}{\rho-\mu}, \frac{\sigma}{\rho}$, respectively. Distance from the BS and leaky bucket parameters for the 3 URLLC users are tabulated below:

The eMBB users are located at distances $250, 560, 650, 720,$ and 800 meters from the BS. Note that we assume that eMBB users are infinitely backlogged and do not have any stringent deadlines. A total of 6000 RBs are available to all the users connected to the BS and the URLLC users are allocated resources with priority. Furthermore, all leftover RBs after assignment to URLLC users are allocated through proportionally fair scheduling to one eMBB user during each time slot. The throughput performance of the oracle-aided policy is also included to provide a *bound* on the best feasible spectral efficiency for URLLC users, which translates to higher throughput for eMBB users.

Fig. 6 showcases the throughput gains for eMBB users for the various algorithms. When compared to the baseline and benchmark scheduling policies, OGDS policy is indeed closer to the throughput gain bound set by the oracle-aided scheduling policy with access to future channel rates.

B. Admission Control

We consider a set of 100 users with ON-OFF bursty traffic and leaky bucket constrained arrivals. The ON OFF duration

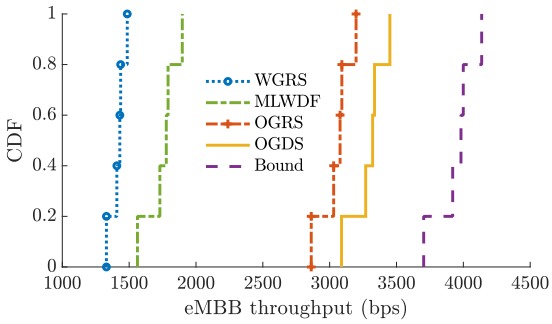


Fig. 6. Long term throughput distribution for eMBB users.

is set to $\frac{\sigma}{\rho-\mu}, \frac{\sigma}{\rho}$, with parameters (in packets per time slot) $\sigma = 50, \rho = 10, \mu = 5$. The delay deadline and user location (distance from the BS) are drawn uniformly random from the sample spaces $\{200, 250, \ldots, 800\}$ and $\{2, 3, \ldots, 10\}$, respectively. The arrivals and channel variations are generated over 10^6 time slots to simulate the number of users that can be admitted for various system capacities $\overline{m}$ (the total number of RBs available in the system). If $Y \sim \mathcal{N}(\hat{\mu}, \hat{\nu})$ denotes the random resource requirement of a new user, then the probability that the total $X_u + Y$ will exceed $\overline{m}$ is approximated using the following inequality, see [38],

$$\mathbb{P}\left(X_u + Y \geq \overline{m}\right) \leq \exp\left(-\frac{\overline{m} - \mu}{2\nu}\right), \qquad (17)$$

where $X_u \sim \mathcal{N}(\mu, \nu)$, $\mu = \mu_u + \hat{\mu}$ and variance $\nu = \nu_u + \hat{\nu}$. Note that the inequality in (17) provides a computationally reasonable expression that can be used to decide if the new user can be admitted without exceeding the reliability requirement δ. Therefore, for the same set of users, we also use the Gaussian approximation for the aggregate resource requirement X_u to determine the number of users that can be admitted, shown as dashed lines in Fig. 7.

It can be seen in Fig. 7 that the Gaussian approximation provides a conservative estimate of the number of users that can be admitted into the system for both the OGRS and WGRS scheduling policies. For OGDS, there is a cross-over point where the CLT estimate is above the simulation based users admitted until a system capacity of 3500 RBs, after which the CLT estimate becomes conservative. This can be attributed to the more bursty nature of OGDS scheduling as compared to the other scheduling policies, making the CLT approximation for OGDS reasonable only at higher system capacities than OGRS/ WGRS. Also interesting to note is that both OGRS and OGDS scheduling policies admit more users as compared to the WGRS policy, which is interesting given that WGRS is more deterministic in resource provisioning, whereas the others are more bursty.

C. Stochastic Dominance of WGRS Policy

Through extensive simulations, we were able to observe first-order stochastic dominance of the number of resource blocks allocated during a WGRS busy cycle over both OGRS-DTE policy and OGDS. We observed that on average OGRS-DTE required lesser resources than WGRS during a

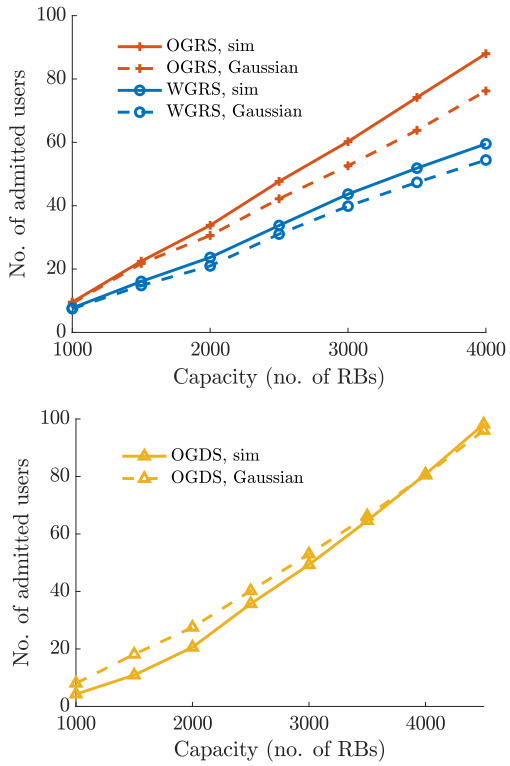


Fig. 7. Admission control assuming CLT approximation on the total RBs required for all admitted URLLC users.

WGRS busy cycle, i.e., $27\%, 47\%, 22\%$ for strong, medium, and weak users, respectively. Similarly, OGDS required fewer resources than WGRS during a WGRS busy cycle, i.e., $43\%, 62\%, 54\%$ for strong, medium, and weak users, respectively.

D. Practical Considerations

In this section, we consider some additional practical considerations affecting our proposed schedulers including HARQ, user mobility, and how confidence levels on the measured resource usage statistics impact the effectiveness of our proposed admission control strategy.

1) Transmission Error: So far we have discussed delay constrained scheduling based on the *connectivity* assumption that a user's packet transmissions are successful at all times. In practice, user transmissions are bound to see errors due to the nature of wireless channel uncertainty. The probability with which errors in wireless transmission occur depends on the size of the data packets, channel strength, and the amount of redundancy added to the original data for error detection and/ or correction. In this subsection, we evaluate the probability of error for all algorithms over various transport block lengths (which is higher for larger packets).

Fig. 8 shows the packet transmission error probability of our proposed algorithms for various user channel strengths as a function of the transport block length. The probability of transmission error ϵ was modeled based on the Polyanski bound [39], [40]. For transmission of block length m, coding rate r and channel SNR γ, the error bound is given by,

$$\epsilon = Q\left(\sqrt{\frac{m}{(\log_2 e)^2\left(1-\frac{1}{1+\gamma^2}\right)}}\left(\log_2(1+\gamma)-r\right)\right). \quad (18)$$

Recall that the modulation and coding scheme is chosen using the 3GPP MCS table ([37, Table 5.2.2.1-2]) based on the instantaneous channel strength. It can be seen that the performance of all our proposed algorithms results in a ten-fold decrease in the probability of transmission errors as compared to the WGRS, with the OGDS algorithm being the best among all feasible (future agnostic) algorithms considered. The proposed algorithms schedule transmissions when the channel has a higher probability of being better than future channels before the deadline expires, which in turn leads to a lower probability of transmission errors.

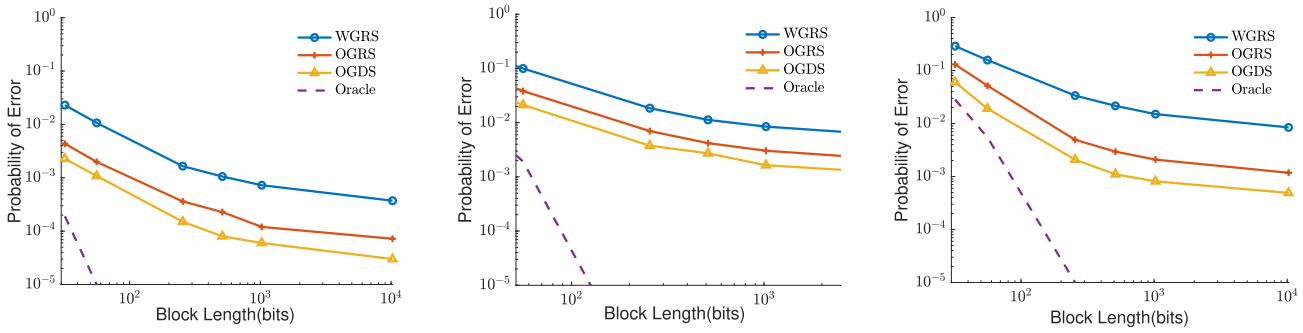
2) HARQ: Wireless networks typically use automatic retransmit requests whenever there are transmission errors that lead to packet losses. A one-shot retransmission would typically suffice if the modulation and coding scheme were carefully chosen so as to maximize the likelihood of successful retransmission. So a simple modification to the proposed algorithms is to reduce the target delay deadline by one slot and perform a one-shot retransmission of packets lost due to transmission error. Note that the proposed modification assumes that the delay constraints exceed several time slots (more than 2 slots). Incorporating the modified target deadline to allow for HARQ one-shot retransmissions leads to successful packet deliveries for all types of users with an associated loss in spectral efficiency. Setting an earlier deadline for strong, medium, and weak users leads to a loss in spectral efficiency of at most $9\%, 14\%$, and 15%, respectively.

3) User Mobility: The adaptive rate thresholds (refer to Equation (9)) depend on the accuracy with which the distribution of the channel rate variations can be empirically estimated. A sufficient number of past channel values are required to estimate the channel variation statistics, but short enough to track non stationary changes and exclude obsolete channel data. This calls for selecting the number of past channel samples that are used to determine the channel rate distribution, which could potentially depend upon the user's mobility speed.

To evaluate this we use the Random Way Point model (RWP) to model user mobility with a constant speed in the range of $5-40$m/s. Using the user's location given by RWP model, the wireless channel variations are modeled based on the 3GPP channel model with correlated log-normal Shadow fading based on the distance traveled. Specifically, the correlation factor for shadowing is given by, $R_x(\cdot) = \exp\left(-\frac{\text{distance}}{d_{corr}}\right)$, where the parameter d_{corr} depends on the presence or absence of Line of Sight (LoS) signal, i.e.,

1) LoS shadowing : Log $\mathcal{N}(0,4)$, $\quad d_{corr} = 10$m.
2) nLoS shadowing: Log $\mathcal{N}(0,7.82)$, $d_{corr} = 13$m.

Fig. 9 shows the resource requirement for various algorithms as a function of the number of past samples used to estimate the CDF. We find that for various mobility speeds in the range of 5-40 m/s, the number of samples (channel



(a) Strong user (b) Medium user (c) Weak user

Fig. 8. Probability of packet transmission error for various user types.

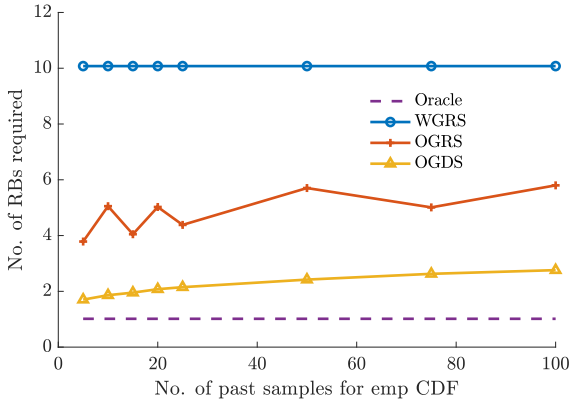


Fig. 9. Resource requirement for a mobile user moving at a speed of 40 m/s.

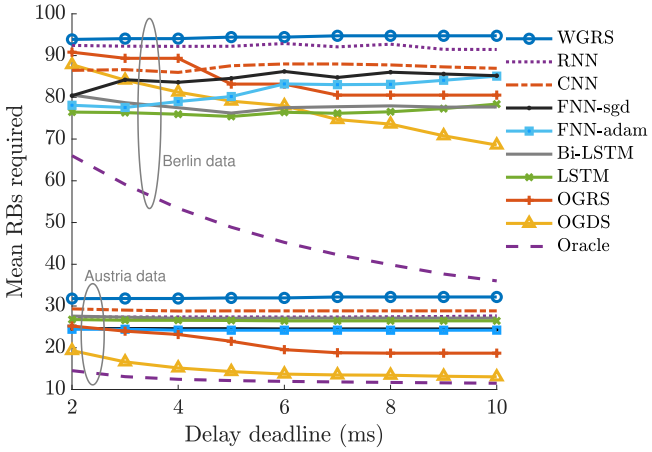


Fig. 10. Resource requirement across proposed scheduling algorithms for non-stationary wireless trace data in [42] marked as "Berlin data" and in [43] marked as "Austria data".

history) required to efficiently track the user's nonstationary channel distribution is 5. As can be seen, using more samples leads to obsolete channel information being included in the empirical estimate, resulting in the *adaptive rate* threshold being irrelevant to the current wireless channel – and hence a loss in spectral efficiency.

4) Delay Constrained Schedulers Based on Based Channel Prediction: Recall that in Section IV-A we proposed an offline/ genie based policy that given *perfect* knowledge of future channel realizations achieved the best possible spectral efficiency subject to the packet delay constraints. It is thus natural to attempt to implement such a policy based on predicted future channel rates given the observed channel history. To explore the potential of this approach we considered various possible predictors introduced in [41] and followed their methodology for training our predictions for the wireless channel data sets[2] in [42] and [43] using Adam/ Stochastic Gradient Descent optimization based Feed-forward Neural Networks (FNN) and state-of-the-art machine learning architectures including Long Short-Term Memory (LSTM) [44], Recurrent Neural Network (RNN) [45], Convolutional Neural Network (CNN) [46]. We evaluated the spectral efficiency of the prediction based delay constrained schedulers, for traffic having deadlines from 2 to 10ms. In this study, bursty traffic arrivals with packets of size 1024 bits were considered, which arrive according to an ON and OFF process with a duration of 10 and 5 time slots respectively, with an ON rate of 10 pkts/slot.

Fig. 10 shows the spectral efficiency (as measured by the mean resource requirements to support the delay constrained traffic) that the prediction based schedulers and our proposed schedulers achieve for the wireless trace data in [42] and [43]. The normalized Root Mean Square Error (RMSE) for neural network predictions lies in the range of $0.081 - 0.137$ for "Berlin data" in [42] and $0.057 - 0.127$ for "Austria data" in [43]. Later on, we will see how this small difference in prediction accuracy leads to large variations in performance.

The overall improvement in spectral efficiency saturates beyond a certain delay deadline (see Fig. 10) due to the diminishing value of the flexibility that each additional time slot provides. Further, to exploit the additional flexibility of more relaxed deadlines, one needs to have accurate predictions of the channel capacity over longer horizons, which in practice is not possible. To see this clearly, we provide the prediction accuracy results for neural network prediction for both "Berlin data" and "Austria data". Figures 11, 12 show the Root Mean Square Error (RMSE) and 95% confidence interval for channel rate predictions for the wireless trace data given in [42]

[2]Data trace of wireless channel strength in terms of SINR captured for vehicular users in cities Berlin and Austria.

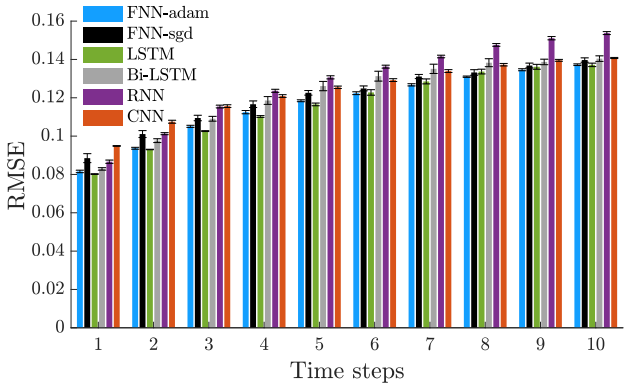


Fig. 11. RMSE across various neural network architectures for channel rate prediction [42].

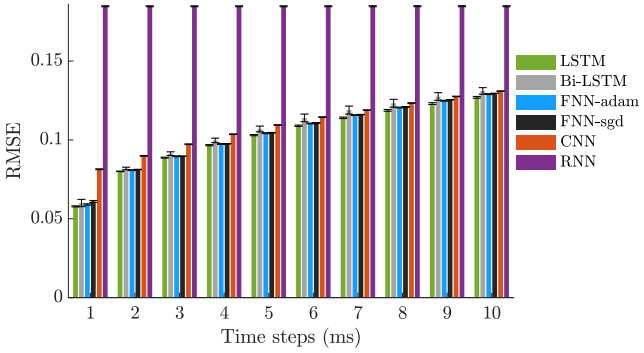


Fig. 12. RMSE across various neural network architectures for channel rate prediction [43].

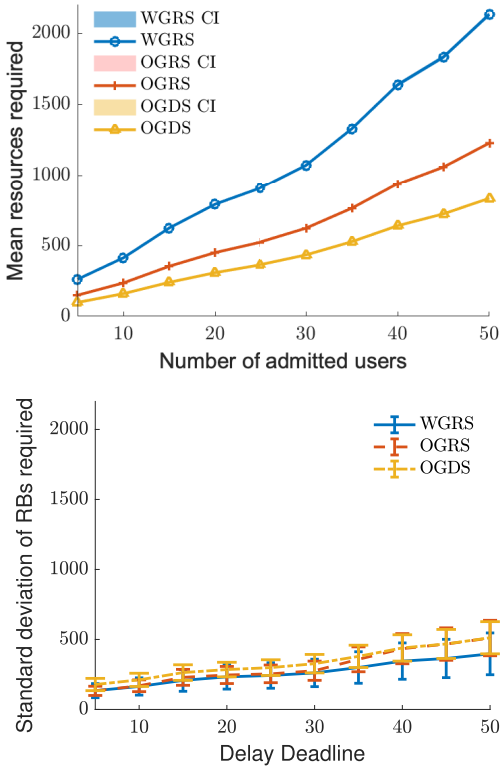


Fig. 13. Confidence interval for the mean and standard deviation of the total number of RBs required for heterogeneous users.

and [43], respectively. As one would expect, the architecture delivering the lowest prediction error, FNN-Adam for "Berlin data" in [42] and LSTM for "Austria data" in [43], leads to the most spectrally efficient scheduling for prediction based delay constrained schedulers. Our proposed measurement-based scheduler, OGDS outperforms all the prediction based schedulers for "Berlin data", with OGRS only slightly worse than FNN-Adam based scheduler when delay deadlines are less than 2 ms. However, for "Austria data", a higher prediction accuracy for shorter delays ($\leq$ 6 ms) leads to better spectral efficiency for prediction based schedulers than both OGDS and OGRS. It appears that the proposed ML-based schedulers are more sensitive to prediction errors when deadlines are relaxed ($>$ 6ms), resulting in lesser spectral efficiency than OGDS.

As one would expect, the architecture delivering the lowest prediction error, FNN-Adam for "Berlin data" in [42] and LSTM for "Austria data" in [43], leads to the most spectrally efficient scheduling for prediction based delay constrained schedulers. Our proposed measurement-based scheduler, OGDS outperforms all the prediction based schedulers for "Berlin data", with OGRS only slightly worse than FNN-Adam based scheduler when delay deadlines are less than 2 ms. However, for "Austria data", a higher prediction accuracy for shorter delays ($\leq$ 6 ms) leads to better spectral efficiency for prediction based schedulers than both OGDS and OGRS. It appears that the proposed ML-based schedulers are more sensitive to prediction errors when deadlines are relaxed ($>$ 6ms), resulting in lesser spectral efficiency than OGDS.

It should be noted that all neural networks were trained using thousands of samples of data before being deployed on test data. This would mean the neural network has adequate training on actual rate variations in the wireless environment and a prediction phase where the user remains stationary, which is unrealistic! One could in principle use a *"meta"* scheduler that uses neural network prediction-based schedulers for low delay deadlines and utilizes OGRS/ OGDS algorithms for higher delay deadlines. However developing such a framework is outside of the scope of this paper, but could perhaps draw on the ideas of a meta scheduler discussed in [47]. In summary, we have proposed measurement-based schedulers that appear robust for real-world traces and have much lower computational complexity than schedulers that use NNs. *Online* time series predictors that can learn to predict with fewer learning samples and quickly adapt to non-stationary variations could be a promising avenue for improving spectral efficiency.

5) Measurement Error: The admission control strategies discussed in [3] and [4], rely on accurate knowledge of the users' aggregate resource usage statistics. In practice, however, one would need to measure the system's resource usage – which could potentially evolve based on the overall network traffic and user channel dynamics. Consequently, it is of interest to know the measurement errors in the mean and variance of aggregate resource usage that would impact performance.

We estimate the measurement error for the mean and standard deviation of the resources required by constructing a confidence interval based on 100 samples of the total RBs

required for servicing the heterogeneous users. We determine the confidence interval for the mean, using the Gaussian confidence interval, $CI = \hat{\mu} \pm z_\alpha \hat{\sigma}$, where $z_\alpha \in \mathbb{R}$ is the value such that $\mathbb{P}(X \geq z_\alpha) \leq \alpha$. The confidence interval for the true standard deviation σ_{true} was obtained using the formula,

$$\sqrt{\frac{(k-1)\hat{\sigma}^2}{\chi^2_{\alpha/2}}} \leq \sigma_{\text{true}} \leq \sqrt{\frac{(k-1)\hat{\sigma}^2}{\chi^2_{1-\alpha/2}}} \tag{19}$$

where k is the number of samples used to obtain the sample variance and $\chi^2_\alpha = x : \mathbb{P}(X \geq x) = \alpha$, with X being χ^2 distributed with $k-1$ degrees of freedom.

Fig. 13 shows the confidence interval for the mean and standard deviation for a 99.9999% confidence interval on the resource requirement based on all three scheduling policies. The confidence interval around the mean is quite narrow and re-emphasizes the certainty equivalence of the measured mean. As for the standard deviation, Fig. 13 shows that one needs to account for the confidence interval of the variance to provide robust admission control to URLLC users.

VI. Conclusion

We have proposed two classes of opportunistic delay constrained wireless schedulers, which can meet heterogeneous users' strict delay deadlines while being spectrally efficient, i.e., minimizing the resources required, thus permitting the system to achieve additional eMBB user throughput. The underlying design principle for OGRS policies is to ensure that the wireless scheduler meets or exceeds the service that a fixed rate scheduler designed based on leaky bucket constrained delay analysis would assign. Thus by design, OGRS policies can also be used to efficiently deliver a Guaranteed Bit Rate (GBR) service. Our proposed OGDS policies allow for more aggressive opportunistic scheduling which depending on the delay constraints can achieve within 10% of the spectral efficiency of optimal offline scheduling. Both policies use dynamic opportunistic thresholds to leverage the knowledge of the user's marginal channel quality rate distribution which in practice would be measured and/ or tracked, based on a limited number, say 10, of the previous channel realizations. In this study, it was considered that channels were independent and identically distributed (i.i.d.) over time. Consequently, the potential for enhancing spectral efficiency through statistical prediction of forthcoming channel realizations could be investigated when there is a correlation across channels over time.

Appendix

Let M_n^{WGRS} and M_n^{OGRS} denote the (possibly fractional) number of resource blocks used to serve the user queue at time n, under WGRS(s) and OGRS(s)-DTE scheduling policies, respectively. Without loss of generality, let the system start with an empty queue and let $(0, N]$ denote a busy cycle of the WGRS policy. We compare the performance of WGRS and OGRS schedulers under a coupled queueing system, where both queues see the same arrival and channel rate processes but one is serviced by scheduling policy and the other by OGRS. First, we will show that in any WGRS busy cycle, the resource requirement for WGRS stochastically dominates that of OGRS, i.e.,

$$\sum_{n=1}^{N} M_n^{\text{WGRS}} \geq_{st} \sum_{n=1}^{N} M_n^{\text{OGRS}}. \tag{20}$$

Then we will prove that in a steady state, the average resource requirement under WGRS is greater than that required by OGRS using the stochastic dominance result.

As long as the user queue is sufficiently backlogged, WGRS provides a deterministic service rate s throughout it's busy cycle. The only nondeterministic part of the WGRS scheduling policy is at the end of it's busy cycle N when there might not be enough data in the queue to utilize service rate s fully. Let us partition the interval $(0, N]$ based on time instants when the channel rate exceeds the adaptive threshold. Define $T_1 \in (0, N]$ as the first time the channel rate exceeds the threshold γ_{T_1} of the OGRS policy, i.e.,

$$T_1 = \min\left(N, \min_{t>0}(t : C_t > \gamma_t)\right), \tag{21}$$

where γ is the OGRS-DTE threshold as previously defined in III-B.3. Note by definition, the user queue length and the number of RBs utilized to schedule data under both WGRS and OGRS policies will be the same until T_1, i.e.,

$$\begin{aligned} M_n^{\text{WGRS}} &= M_n^{\text{OGRS}} \text{ a.s. }, \quad \forall n \in (0, T_1), \text{ and} \\ Q_n^{\text{WGRS}} &= Q_n^{\text{OGRS}} \text{ a.s. }, \quad \forall n \in (0, T_1]. \end{aligned} \tag{22}$$

Consider a particular realization of $T_1 = t_1$. Denote by Q_{t_1} (same for WGRS and OGRS) the amount of data available to be transmitted at time t_1 and note that this is the same for both policies.

Now we shall compare the number of RBs that will be used by both the policies to service Q_{t_1}. Since the channel rate exceeds the DTE threshold at time t_1, OGRS will use $M_{t_1}^{\text{OGRS}} = \dfrac{Q_{t_1}}{C_{t_1}}$ RBs to clear the entire queue. However, the WGRS policy will require $\Delta_1 = \left\lceil \frac{Q_{t_1}}{s} \right\rceil$ time slots to service the same amount of bits in the queue Q_{t_1}. Clearly, $Q_{t_1} \leq s\Delta_1$, therefore, one can conclude that,

$$M_{t_1}^{\text{OGRS}} \leq s\frac{\Delta_1}{\gamma_{t_1}} \text{ a.s.} \tag{23}$$

We will show that the number of RBs allocated by OGRS-DTE policy in the interval $(0, t_1]$ is stochastically dominated by that allocated by WGRS policy in the interval $(0, t_1 + \Delta_1)$. Since the number of RBs utilized by both the WGRS and OGRS policies is the same in the interval $(0, t_1)$, it is sufficient to compare their resource allocations in the interval $[t_1, t_1 + \Delta_1)$.

Recall Lemma 2 to obtain,

$$\mathbb{P}(C_{t_1} > C_{t_1+i} | C_{t_1} \geq \gamma_{t_1}) \geq \mathbb{P}(C_{t_1} < C_{t_1+i} | C_{t_1} \geq \gamma_{t_1}), \tag{24}$$

where $i = 1, \ldots, \Delta_1 - 1$. Now consider the distribution of the number of RBs that WGRS requires to clear the same queue. The only way that WGRS could require fewer RBs than $s\Delta_1, 1 \leq \Delta_1 \leq d$ is if all the channel realizations between $[t_1 + 1, t_1 + \Delta_1)$ are greater than C_{t_1}, i.e., for any particular

realization of the random variable $\Delta_1 = k$,

$$\begin{aligned}
&\mathbb{P}\left(\sum_{n=0}^{k-1} M_{t_1+n}^{\text{WGRS}} \leq s\frac{k}{\gamma}\middle| C_{t_1} \geq \gamma_{t_1}\right)\\
&= \mathbb{P}\left((C_{t_1+1} \geq C_{t_1}) \cap \ldots (C_{t_1+k} \geq C_{t_1} | C_{t_1} \geq \gamma_{t_1})\right),\\
&= \prod_{i=1}^{k-1} \mathbb{P}\left(C_{t_1+i} \geq C_{t_1} | C_{t_1} \geq \gamma_{t_1}, \Delta_1 = k\right),\\
&\overset{(a)}{\leq} \prod_{i=1}^{k-1} \mathbb{P}\left(C_{t_1+i} < C_{t_1} | C_{t_1} \geq \gamma_{t_1}, \Delta_1 = k\right),\\
&\overset{(b)}{=} q^{k-1} < 1 = \mathbb{P}\left(M_{t_1}^{\text{OGRS}} \leq s\frac{k}{\gamma}\middle| C_{t_1} \geq \gamma_{t_1}\right), \qquad (25)
\end{aligned}$$

where inequality (a) follows from equation (24). By definition of t_1 note that $C_{t_1} \geq \gamma_{t_1}$, where the channel rate threshold is at least as large as the median, i.e., $\gamma_{t_1} \geq F_C^{-1}(1/2)$ always, since $\gamma_{t_1} = F_C^{-1}(1 - \frac{1}{\Delta_1+2})$ and $\frac{1}{x+2} \leq \frac{1}{2}, \forall x \geq 0$. So it follows that each of the probabilities in the product of step (a) has a value of $q \leq \frac{1}{2}$. Therefore, based on equations (22) and (25), we have, $\forall t > 0$,

$$\mathbb{P}\left(\sum_{n=0}^{\Delta_1-1} M_{t_1+n}^{\text{WGRS}} \leq t\middle| \Delta_1 = k\right) \leq \mathbb{P}\left(M_{t_1}^{\text{OGRS}} \leq t\middle| \Delta_1 = k\right). \qquad (26)$$

Consequently, we draw the following conclusion applying theorem [48, Theorem 1.2.15] about the preservation of the stochastic order of two random variables, if there exists an order when conditioned on a dependent random variable, i.e.,

$$M_{t_1}^{\text{OGRS}} \leq_{st} \sum_{n=0}^{\Delta_1-1} M_{t_1+n}^{\text{WGRS}}. \qquad (27)$$

Note that the above equation holds for any realization t_1 of the random variable T_1, so it must hold for all realizations of T_1. Moreover, recall (22) where the number of resources utilized by both policies are equal until T_1, so summarizing the previous equation and (22) we have,

$$\sum_{n=0}^{T_1} M_n^{\text{OGRS}} \leq_{st} \sum_{n=0}^{T_1+\Delta_1-1} M_n^{\text{WGRS}}. \qquad (28)$$

Finally, note that if $T_1 = N$, we are done with the proof. Otherwise, we shall partition the WGRS busy cycle $(0, N]$ for each of the scheduling policies based on the number of occurrences of the channel rate exceeding the threshold,

$$\begin{aligned}
&\text{OGRS:}(0, T_1], (T_1, T_2] \ldots (T_P, N],\\
&\text{WGRS :}(0, T_1 + \Delta_1), [t_1 + \Delta_1, T_2 + \Delta_2) \ldots [T_P + \Delta_p, N].
\end{aligned} \qquad (29)$$

Here the times T_i are defined as follows,

$$T_i = \min\left(N, \min_{t > T_{i-1}} (t : C_t > \gamma_t)\right). \qquad (30)$$

For the subsequent WGRS cycle $(t_1, T_2]$, if $T_2 > t_1 + \Delta_1 - 1$ then by OGRS algorithm design, the number of RBs allocated by OGRS in the interval $(t_1, t_1 + \Delta_1)$ is zero and we can repeat the same analysis as we did for interval $(0, t_1]$ to establish stochastic dominance. In case $T_2 \leq t_1 + \Delta_1 - 1$, we know that $\gamma_t \geq F_C^{-1}(1 - \frac{1}{3}), \forall t \in [t_1 + 1, t_1 + \Delta_1)$. Therefore we have $\gamma_{T_2} > \gamma_{t_1}$, and the occurrence of any future rate realizations for WGRS to be better than C_{T_2} will be governed by,

$$q = \mathbb{P}\left(C_{T_2+i} > C_{T_2} \middle| C_{T_2} > \gamma_{T_2}\right) < 1/3, \qquad (31)$$

which has to be satisfied by WGRS over multiple time slots according to $\Delta_2 = \left\lceil \frac{Q_{T_2}}{s} \right\rceil$ a.s., in order to utilize lesser resources than OGRS-DTE. Stochastic dominance of the number of resources allocated by WGRS over that allocated by OGRS can be derived as in equation (25), now using q as in (31).

Finally, to complete the proof let us define the following random variables,

$$X_i = \sum_{n=1+T_i}^{T_{i+1}} M_n^{\text{OGRS}} \text{ and } Y_i = \sum_{n=T_i+\Delta_i}^{T_{i+1}+\Delta_{i+1}-1} M_n^{\text{OGRS}}, \qquad (32)$$

where $T_0 = 0$ and $\Delta_0 = 0$. We have shown that $X_i \leq_{st} Y_i, 1 \leq i \leq I$, where I is the number of time instances that the channel rate exceeds the adaptive threshold over the interval $(0, N]$. The result in (20) follows from [48, Theorem 1.2.17], because

$$\sum_{i=1}^{I} X_i = \sum_{n=1}^{N} M_n^{\text{WGRS}} \text{ and } \sum_{i=1}^{I} Y_i = \sum_{n=1}^{N} M_n^{\text{OGRS}}. \qquad (33)$$

We provide simulation results in Sec.V-C that establish the stochastic dominance of the number of resource blocks allocated by both OGRS-DTE policy over WGRS, during a WGRS busy cycle.

If we let the time $n \to \infty$, and $(N_i)_{i \in \mathbb{N}}$ denote the WGRS busy cycle lengths, then the average number of RBs per time slot required by WGRS is given by,

$$\begin{aligned}
\mathbb{E}[M^{\text{WGRS}}] &= \lim_{n\to\infty} \frac{1}{n} \sum_{k=1}^{n} M_k^{\text{WGRS}}\\
&= \lim_{n\to\infty} \frac{1}{n} \sum_{i:N_i \leq n} \sum_{n=1}^{N_i} M_n^{\text{WGRS}}\\
&\geq_{st} \frac{1}{n} \sum_{i:N_i \leq n} \sum_{n=1}^{N_i} M_n^{\text{OGRS}} = \mathbb{E}[M^{\text{OGRS}}]. \qquad (34)
\end{aligned}$$

Lemma 2: For any two positive random variables X_1, X_2 and a constant $\gamma > \text{median}(X_2)$, the following inequality holds,

$$\mathbb{P}(X_1 \geq X_2 | X_1 \geq \gamma) \geq \mathbb{P}(X_1 < X_2 | X_1 \geq \gamma), \qquad (35)$$

as long as X_1 has a non-zero probability of taking values higher than γ, i.e., $\mathbb{P}(X_1 \geq \gamma) > 0$.

Proof: The Left Hand Side (LHS) of equation (35) can be written as,

$$\mathbb{P}(X_1 \geq X_2 | X_1 \geq \gamma) = \mathbb{E}[\mathbb{1}_{(X_1 \geq X_2)} | X_1 \geq \gamma], \qquad (36)$$

where $\mathbb{1}_E$ is the indicator function of the event E. Similarly expressing the right-hand side of the equation as an

expectation and then finding the difference yields,

$$\begin{aligned}&\mathbb{P}(X_1 \geq X_2 | X_1 \geq \gamma) - \mathbb{P}(X_1 < X_2 | X_1 \geq \gamma) \\ &= \mathbb{E}[\mathbb{1}_{(X_1 \geq X_2)} | X_1 \geq \gamma] - \mathbb{E}[\mathbb{1}_{(X_1 < X_2)} | X_1 \geq \gamma] \\ &= \mathbb{E}[\mathbb{1}_{(X_1 \geq X_2)} - \mathbb{1}_{(X_1 < X_2)} | X_1 \geq \gamma]. \end{aligned} \tag{37}$$

Whenever a realization of X_1 is below γ, the RHS above is 0. Otherwise, the right hand side of (37) becomes,

$$\begin{aligned}\mathbb{E}[\mathbb{1}_{(x \geq X_2)} - \mathbb{1}_{(x < X_2)}] &= \mathbb{E}[\mathbb{1}_{(x \geq X_2)}] - \mathbb{E}[\mathbb{1}_{(x < X_2)}] \\ &= \underbrace{\mathbb{P}[(x \geq X_2)]}_{\geq \frac{1}{2}} - \underbrace{\mathbb{P}[(x < X_2)]}_{< \frac{1}{2}} \geq 0.\end{aligned} \tag{38}$$

From (37) and (38), it is clear that for any realization of X_1, the LHS of (37) is greater than or equal to 0, leading to the result in (35). It should be noted that while we have provided proof for only OGRS-DTE, the same proof would hold for any fixed percentile threshold $\alpha \geq 0.5$. □

REFERENCES

[1] *5G Study on Channel Models for Frequencies From 5 to 100 GHz Release 14*, document (TR) 38.901, 3GPP, Apr. 2018.

[2] *New Services & Applications with 5G Ultra-Reliable Low Latency Communications*, 5G Americas, 3G Americas LLC, Bellevue, WA, USA, White Paper, 2018.

[3] G. Chandrasekaran, G. D. Veciana, V. Ratnam, H. Chen, and C. Zhang, "Spectrally efficient guaranteed rate scheduling for heterogeneous QoS constrained wireless networks," in *Proc. 21st Int. Symp. Modeling Optim. Mobile, Ad Hoc, Wireless Netw. (WiOpt)*, Aug. 2023, pp. 1–8.

[4] G. Chandrasekaran, G. D. Veciana, V. Ratnam, H. Chen, and C. Zhang, "Delay and jitter constrained wireless scheduling with near-optimal spectral efficiency," in *Proc. IEEE 34th Annu. Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2023, pp. 1–7.

[5] A. L. Stolyar, "Maximizing queueing network utility subject to stability: Greedy primal-dual algorithm," *Queueing Syst.*, vol. 50, no. 4, pp. 401–457, Aug. 2005.

[6] A. Jalali, R. Padovani, and R. Pankaj, "Data throughput of CDMA-HDR a high efficiency-high data rate personal communication wireless system," in *Proc. IEEE 51st Veh. Technol. Conf. (VTC-Spring)*, vol. 3, May 2000, pp. 1854–1858.

[7] A. Eryilmaz and I. Koprulu, "Discounted-rate utility maximization (DRUM): A framework for delay-sensitive fair resource allocation," in *Proc. 15th Int. Symp. Modeling Optim. Mobile, Ad Hoc, Wireless Netw. (WiOpt)*, May 2017, pp. 1–8.

[8] I.-H. Hou and P. Kumar, "Utility-optimal scheduling in time-varying wireless networks with delay constraints," in *Proc. 11th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, New York, NY, USA, 2010, pp. 31–40.

[9] I. Hou and P. R. Kumar, "Real-time communication over unreliable wireless links: A theory and its applications," *IEEE Wireless Commun.*, vol. 19, no. 1, pp. 48–59, Feb. 2012.

[10] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," in *Proc. 29th IEEE Conf. Decis. Control*, Dec. 1990, pp. 2130–2132.

[11] S. Shakkottai and A. L. Stolyar, "Scheduling for multiple flows sharing a time-varying channel: The exponential rule," *Transl. Amer. Math. Soc.*, vol. 207, pp. 185–202, Dec. 2002.

[12] B. Sadiq, S. J. Baek, and G. de Veciana, "Delay-optimal opportunistic scheduling and approximations: The log rule," *IEEE/ACM Trans. Netw.*, vol. 19, no. 2, pp. 405–418, Apr. 2011.

[13] S. Shakkottai and A. L. Stolyar, "Scheduling algorithms for a mixture of real-time and non-real-time data in HDR," in *Teletraffic Science and Engineering*, vol. 4. Amsterdam, The Netherlands: Elsevier, 2001, pp. 793–804.

[14] M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, P. Whiting, and R. Vijayakumar, "Providing quality of service over a shared wireless link," *IEEE Commun. Mag.*, vol. 39, no. 2, pp. 150–154, Feb. 2001.

[15] A. Eryilmaz and R. Srikant, "Fair resource allocation in wireless networks using queue-length-based scheduling and congestion control," *IEEE/ACM Trans. Netw.*, vol. 15, no. 6, pp. 1333–1344, Dec. 2007.

[16] J.-Y. L. Boudec and P. Thiran, *Network Calculus: A Theory of Deterministic Queuing Systems for the Internet* (Lecture Notes in Computer Science). Berlin, Germany: Springer, 2003.

[17] C.-S. Chang, *Performance Guarantees in Communication Networks*. Berlin, Germany: Springer-Verlag, 2000.

[18] L. B. Le, E. Hossain, and A. S. Alfa, "Service differentiation in multirate wireless networks with weighted round-robin scheduling and ARQ-based error control," *IEEE Trans. Commun.*, vol. 54, no. 2, pp. 208–215, Feb. 2006.

[19] P. Lin, B. Benssou, Q. L. Ding, and K. C. Chua, "CS-WFQ: A wireless fair scheduling algorithm for error-prone wireless channels," in *Proc. 9th Int. Conf. Comput. Commun. Netw.*, vol. 3, 2000, pp. 276–281.

[20] S. Patil and G. de Veciana, "Managing resources and quality of service in heterogeneous wireless systems exploiting opportunism," *IEEE/ACM Trans. Netw.*, vol. 15, no. 5, pp. 1046–1058, Oct. 2007.

[21] J. J. Jaramillo and R. Srikant, "Optimal scheduling for fair resource allocation in ad hoc networks with elastic and inelastic traffic," in *Proc. IEEE INFOCOM*, Mar. 2010, pp. 1–9.

[22] C. Tsanikidis and J. Ghaderi, "Near-optimal packet scheduling in multi-hop networks with end-to-end deadline constraints," *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 52, no. 1, pp. 33–34, Jun. 2024.

[23] S. Cayci and A. Eryilmaz, "Learning for serving deadline-constrained traffic in multi-channel wireless networks," in *Proc. 15th Int. Symp. Modeling Optim. Mobile, Ad Hoc, Wireless Netw. (WiOpt)*, May 2017, pp. 1–8.

[24] S. Chilukuri, G. Piao, D. Lugones, and D. Pesch, "Deadline-aware TDMA scheduling for multihop networks using reinforcement learning," in *Proc. IFIP Netw. Conf. (IFIP Netw.)*, Jun. 2021, pp. 1–9.

[25] C. Li, W. Chen, and H. V. Poor, "Diversity enabled low-latency wireless communications with hard delay constraints," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 7, pp. 2107–2122, Jul. 2023.

[26] R. Singh and P. Kumar, "Throughput optimal decentralized scheduling of multihop networks with end-to-end deadline constraints: Unreliable links," *IEEE Trans. Autom. Control*, vol. 64, no. 1, pp. 127–142, Jan. 2019.

[27] Z. Yu, Y. Xu, and L. Tong, "Deadline scheduling as restless bandits," *IEEE Trans. Autom. Control*, vol. 63, no. 8, pp. 2343–2358, Aug. 2018.

[28] R. Singh and P. R. Kumar, "Adaptive CSMA for decentralized scheduling of multi-hop networks with end-to-end deadline constraints," *IEEE/ACM Trans. Netw.*, vol. 29, no. 3, pp. 1224–1237, Jun. 2021.

[29] R. J. Gibbens and F. P. Kelly, "Measurement-based connection admission control," in *Proc. 15th Int. Teletraffic Congr.*, vol. 2, 1997, pp. 879–888.

[30] M. Grossglauser and D. Tse, "A framework for robust measurement-based admission control," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 27, no. 4, pp. 237–248, Oct. 1997.

[31] D. Tse and M. Grossglauser, "Measurement-based call admission control: Analysis and simulation," in *Proc. INFOCOM*, vol. 3, 1997, pp. 981–989.

[32] L. Breslau, S. Jamin, and S. Shenker, "Comments on the performance of measurement-based admission control algorithms," in *Proc. 19th Annu. Conf. IEEE Comput. Commun. Societies Conf. Comput. Commun.*, vol. 3, Mar. 2000, pp. 1233–1242.

[33] G. S. Kesava and N. B. Mehta, "Multi-connectivity for URLLC and coexistence with eMBB in time-varying and frequency-selective fading channels," *IEEE Trans. Wireless Commun.*, vol. 22, no. 6, pp. 3599–3611, Nov. 2023.

[34] S. Patil, "Measurement-based opportunistic scheduling for heterogenous wireless systems," *IEEE Trans. Commun.*, vol. 57, no. 9, pp. 2745–2753, Sep. 2009.

[35] D. I. Shuman and M. Liu, *Opportunistic Scheduling With Deadline Constraints in Wireless Networks*. New York, NY, USA: Springer, 2011, pp. 127–155.

[36] A. Destounis, G. S. Paschos, J. Arnau, and M. Kountouris, "Scheduling URLLC users with reliable latency guarantees," in *Proc. 16th Int. Symp. Modeling Optim. Mobile, Ad Hoc, Wireless Netw. (WiOpt)*, 2018, pp. 1–8.

[37] *5G NR Physical Layer Procedures for Data*, document (TS) 38.214, 3GPP, Apr. 2018.

[38] A. Dembo and O. Zeitouni, *Large Deviations Techniques and Applications* (Stochastic Modelling and Applied Probability). Berlin, Germany: Springer, 2009.

[39] G. Ozcan and M. C. Gursoy, "Throughput of cognitive radio systems with finite blocklength codes," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 11, pp. 2541–2554, Nov. 2013.
[40] C. She, C. Yang, and T. Q. Quek, "Joint uplink and downlink resource configuration for ultra-reliable and low-latency communications," *IEEE Trans. Commun.*, vol. 66, no. 5, pp. 2266–2280, May 2018.
[41] R. Chandra, S. Goyal, and R. Gupta, "Evaluation of deep learning models for multi-step ahead time series prediction," *IEEE Access*, vol. 9, pp. 83105–83123, 2021.
[42] R. Hernangomez et al., *Berlin V2X*, IEEE Dataport, 2022, doi: 10.21227/8cj7-q373.
[43] S. Farthofer, M. Herlich, C. Maier, S. Pochaba, J. Lackner, and P. Dorfinger, "CRAWDAD srfg/lte-4g-highway-drive-tests-salzburg," IEEE Dataport, 2022, doi: 10.15783/6gc4-y070.
[44] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[45] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with LSTM," *Neural Comput.*, vol. 12, no. 10, pp. 2451–2471, Oct. 2000.
[46] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1–9.
[47] J. Song, G. de Veciana, and S. Shakkottai, "Meta-scheduling for the wireless downlink through learning with bandit feedback," *IEEE/ACM Trans. Netw.*, vol. 30, no. 2, pp. 487–500, Apr. 2022.
[48] M. A. Mccomb, "Comparison methods for stochastic models and risks," *Technometrics*, vol. 45, no. 4, pp. 370–371, Nov. 2003.

Geetha Chandrasekaran (Member, IEEE) received the B.E. degree from CEG Anna University, the M.S. degree from IIT Madras, and the Ph.D. degree from UT Austin, all in electronics and communication engineering. She is currently a Faculty Member with the Department of Computational Engineering and Mathematical Sciences, Texas A&M-San Antonio. Drawing on recent developments in machine learning algorithms and wireless communication networks, her Ph.D. thesis focuses on improving quality of service (QoS) for next-generation wireless applications. She worked on several algorithms for the physical and MAC layers of the wireless communication protocol. She also has over five years of experience in the field of consumer electronics and telecommunications, including design, development, innovation, and maintenance of embedded firmware for various consumer electronic products at Honeywell and Motorola. A key focus of her research is improving the performance of next-generation wireless communication networks using statistical inference and machine learning.

Gustavo de Veciana (Fellow, IEEE) received the Ph.D. degree in electrical engineering from UC Berkeley in 1993. He was the Director and the Associate Director of the Wireless Networking and Communications Group (WNCG) from 2003 to 2007. He is currently a Professor and the Associate Chair of the Department of Electrical and Computer Engineering. His research focuses on the design, analysis, and control networks, information theory, and applied probability. His current research interests include measurement, modeling, and performance evaluation; wireless and sensor networks; and architectures and algorithms to design reliable computing and networked systems. He is a recipient of the Cockrell Family Regents Chair in Engineering at U.T. Austin and the NSF CAREER Award 1996 and a co-recipient of seven best paper awards, including the 2021 IEEE Communication Society W. Bennett Prize. He also serves on the board of trustees for IMDEA Networks Madrid. He is the Editor at Large of IEEE/ACM TRANSACTIONS ON NETWORKING.

Vishnu V. Ratnam (Senior Member, IEEE) received the B.Tech. degree (Hons.) in electronics and electrical communication engineering from IIT Kharagpur, Kharagpur, India, in 2012, where he graduated as the Salutatorian for the class of 2012, and the Ph.D. degree in electrical engineering from the University of Southern California, Los Angeles, CA, USA, in 2018. He is currently a Staff Research Engineer II with the Standards and Mobility Innovation Laboratory, Samsung Research America, Plano, TX, USA. His research interests include Wi-Fi standards, wireless sensing, AI for wireless, mm-Wave, and terahertz communication. He was a recipient of the IIT Kharagpur Young Alumni Achiever Award in 2024 and the Best Student Paper Award with the IEEE International Conference on Ubiquitous Wireless Broadband (ICUWB) in 2016 and is a member of the Phi-Kappa-Phi Honor Society.

Hao Chen received the B.S. and M.S. degrees in information engineering from Xi'an Jiaotong University, Shaanxi, in 2010 and 2013, respectively, and the Ph.D. degree in electrical engineering from The University of Kansas, Lawrence, KS, USA, in 2017. He is currently a Senior Staff Engineer with the Standards and Mobility Innovation Laboratory, Samsung Research America, where he is working on algorithm design and prototyping of AI for wireless communication, wireless sensing, and localization. His research interests include network optimization, machine learning, and 5G cellular systems.

Charlie Zhang (Fellow, IEEE) received the Ph.D. degree from the University of Wisconsin, Madison. He worked with the Nokia Research Center and Motorola Mobility, for six years, before joining Samsung, in 2007. He is currently a SVP with Samsung Research America, where he leads research, prototyping, and standardization for 5G/6G and other wireless systems. He is also the Corporate VP and the Head of the Global 6G Team, Samsung Research America. He is serving as the ATIS North America Next-G Alliance Full Member Group Vice Chair. He was the Board Chair of the FiRa Consortium from May 2019 to May 2023 and the Vice Chair of the 3GPP RAN1 working group from 2009 to 2013, where he led the development of LTE and LTE-advanced technologies.