Complexity of URLLC Scheduling and Efficient Approximation Schemes

Apostolos Destounis and Georgios S. Paschos
Mathematical and Algorithmic Sciences Lab, France Research Center, Huawei Technologies Co. Ltd.
email: firstname.lastname@huawei.com

***Abstract*—In this paper we address the problem of joint admission control and resource scheduling for *Ultra Reliable Low Latency Communications* (URLLC). We examine two models: (i) the *continuous*, where all allocated resource blocks contribute to the success probability, and (ii) a *binary*, where only resource blocks with strong signal are "active" for each user, and user k needs d_k active resource blocks for a successful URLLC transmission. In situations of congestion, we are interested in finding a subset of users that can be scheduled simultaneously. We show that finding a feasible schedule for at least m URLLC users is NP-complete in the (easier) binary SNR model, hence also in the continuous. Maximizing the reward obtained from a feasible set of URLLC users is NP-hard and inapproximable to within $(\log_2 d)^2/d$ of the optimal, where $d \doteq \max_k d_k$. On the other hand, we prove that checking a candidate set of users for feasibility and finding the corresponding schedule (when feasible) can be done in polynomial time, which we exploit to design an efficient heuristic algorithm for the general continuous SNR model. We complement our theoretical contributions with a numerical evaluation of our proposed schemes.**

I. Introduction

A. Motivation and Background

A key differentiator of upcoming 5G wireless networks is their ability to provide reliable low latency via the *Ultra Reliable Low Latency Communications* (URLLC) service class [1]. This capability is considered as an enabler for industrial automation [2], virtual reality applications [3], and control of vehicles [4]. Such applications require "live" wireless connections, where packets must be received within a very short time period since their creation. To effectively synchronize industrial machines and avoid car collisions, the URLLC requirement not only ensures that packets arrive in time, but also in a reliable manner, in the sense that the latency deadline may be violated only very rarely (e.g. once every 100k attempts). To achieve the URLLC requirement, 5G wireless networks will employ various intelligent techniques, including interface diversity [5], multi-path diversity [6], packet duplication [7] and short-packet communications [8]. *In this paper we focus on scheduling URLLC short packets.*

Previous wireless schedulers, designed for high bandwidth applications, assigned the *Resource Blocks* (RBs) opportunistically one-by-one to the user with the highest ratio of *instantaneous rate* over the *average throughput obtained thus far*. Such a simple and efficient algorithm achieves the optimal performance in that setting [9]. However, to optimally schedule a URLLC user, a radically different approach must be taken; the available resource blocks within a *Transmission Time Interval* (TTI) are proactively examined and an allocation is made such that the combination of the allocated blocks allow a URLLC user to achieve its reliability requirement. When multiple URLLC users are served by the same scheduler, a joint allocation of URLLC transmissions must be found on the available resource blocks such that the requirements of all users are satisfied. Therefore, URLLC scheduling consists in combinatorial allocation of resource blocks, which is a challenging setting for scheduling. This brings us to the natural complexity question: *are there efficient URLLC schedulers?*

A further complication arises when a scheduler must serve a set of URLLC users whose requirements are not simultaneously achievable. In this case, it is possible to reject some of the users, and then schedule the rest. However, identifying the optimal schedulable subset of users is shown in this paper to be an extremely difficult problem, impossible to resolve exactly under tight timing constraints.

More generally, this paper highlights a crucial consideration towards a theory of scheduling and admission control for guaranteed latency in wireless networks, that of complexity, which determines how feasible it is for a practical wireless system to operate with a given algorithm. The aim of this paper is thus to lay the foundations of understanding the complexity of URLLC communications. Specifically, our contributions include:

- We model URLLC scheduling at two different granularities, *(i)* the standard continuous Signal-to-Noise Ratio (SNR) model, and *(ii)* the binary SNR model, an approximation where each resource block is classified as active or inactive according to an SNR threshold.
- We show that the decision problem: *does there exist a URLLC schedule that satisfies $\geq m$ users within a TTI?* is NP-hard for both SNR models. The statement is proved by a reduction from the independent set problem, which allows us to characterize also the inapproximability of the corresponding optimization problem.
- For scheduling in the binary SNR model, however, we prove that given a set of URLLC users which is feasible, a schedule can be found in polynomial time solving a linear program. This remarkable simplification is due to the fact that the constraint matrix of the linear relaxation of our scheduling problem is shown to be *totally unimodular*.
- Regarding the admission control in the binary model, we show that the GREEDY algorithm provides a $1/(d+1)$ approximation to the original problem.
- Last, we propose the *Iterative Thresholding Algorithm* (ITA), which applies the above findings to the continuous SNR model. In our simulations ITA outperforms the

continuous greedy baseline by up to 30%.

B. Related Work

The quest for low latency wireless communications has gained significant attention in the literature. Works [10], [11] propose a queueing framework and model the URLLC reliability constraint as the probability that the delay of a packet exceeds a threshold. Their algorithms are based on converting the delay into throughput by the theory of *effective capacity*. The work [12] deals with obtaining bounds on delay violation probabilities for a single-user system, where the transmitter employs multiple antennas and short codewords by using *stochastic network calculus.* The above works assume that packets exceeding deadlines still count towards the system performance. When packets that arrive after the deadline are dropped, authors in [13] model the URLLC problem based on the *timely throughput* approach and focus on meeting a long term packet delivery rate.

Regarding scheduling in systems with multiple resources in the time-frequency domain, which is the focus of the current paper, authors in [14], [15] address scheduling in the frequency domain in a binary SNR model under a delay violation requirement. Regarding hard deadlines, authors in [16] examine the problem of maximizing the utility of *enhanced Mobile Broadband* (eMBB) users when URLLC transmissions are being *punctured* in resource blocks in the time-frequency grid of LTE. In addition, the work [17] examines the impact of resource allocation in the frequency domain coupled with *Hybrid Automatic Repeat reQuest* (HARQ) on how can a system support a load of URLLC users under queueing theoretic blocking models. Both these works assume that URLLC users need a fixed number of resources for successful transmission regardless of the actual realization of the channel in each resource block. Finally, [18] examines the impact of dynamically varying TTI length in order to serve URLLC before their deadline and still give enough utility to the eMBB users, assuming, however that wireless transmissions cannot fail, therefore not accounting for small blocklength transmissions.

Contrary to the aforementioned works, we examine joint admission control and scheduling of radio resource blocks where we take into account the realization of the wireless channel within each block and the transmission failure probabilities due to short block length transmissions. More importantly, our work is the first to characterize the computational complexity and examine approximation schemes for the problem of joint admission control and scheduling of URLLC users in a time-frequency resource grid with fixed resources.

II. System Model

We consider a system with K users, operating in frames. Each frame consists of R *Resource Blocks* (RBs) in the time - frequency domain. Let γ_r^k denote the user-k SNR in RB r within the current frame. Values γ_r^k are made known to the scheduler via measurements. The goal of the scheduler is to assign RBs to each user to satisfy their latency requirements.

The latency requirement of URLLC in the 5G specifications is $1ms$, and equal to the frame length [19]. Therefore, one way to satisfy the user-k latency requirement would be to correctly communicate L_k bits within each frame. However, due to unavoidable transmission errors, correct reception can not be ensured in a wireless system at all times. To address this inherent limitation of wireless systems, it is meaningful to consider a probabilistic *Service-Level Agreement* (SLA) in the following form:

Definition 1 (URLLC SLA)**.** *We say that the URLLC SLA of user k is satisfied in a given frame if:*

$$Pr\left(L_k \text{ bits correctly received}\right) \geq \theta_k.$$

In 5G specifications, $\theta_k = 0.99999$ and $L_k = 32$ Bytes [1].

In order to meet the above URLLC SLA, the scheduler assigns a set of RBs to each user using the scheduling variables $x_r^k \in \{0,1\}$, where $x_r^k = 1$ denotes that user k is scheduled to transmit in RB r. Assuming a user can exploit multiple assigned RBs to jointly encode messages, the frame error probability defined as

$$p_e^k(\boldsymbol{x}) \triangleq 1 - \Pr\left(L_k \text{ bits correctly received}\right),$$

depends on the assigned RBs $\boldsymbol{x} = (x_r^k)$: allocating more to user k will decrease its frame error probability. Specifically, an accurate estimate of user-k frame error probability for a schedule $\boldsymbol{x}$ can be found via the Polyanskiy bound [20], [21]:

$$p_e^k(\boldsymbol{x}) = Q\left(\frac{n\sum_r x_r^k \log_2(1+\gamma_r^k) - L_k + 0.5\log_2(n)\sum_r x_r^k}{\sqrt{n\sum_r x_r^k V(\gamma_r^k)}}\right) \quad (1)$$

where n is the number of channel uses, $Q(.)$ is the error function, and $V(\gamma) = 1 - \frac{1}{(1+\gamma)^2}$ is the dispersion of the *Additive White Gaussian Noise* channel with SNR γ [20]. Eq. (1) is significantly more accurate than the Shannon formula when the length of the transmitted packets is short, such as in our URLLC case.

In our system, the feedback about transmission failures is obtained at the end of the frame.[1] Therefore, to satisfy the user-k URLLC SLA, the scheduler must pro-actively schedule enough RBs to provide sufficiently low error probability, taking into account the user-specific SNRs $\gamma_r^k, r = 1, \ldots, R$. Assigning RBs to users is called *URLLC scheduling*, *and the focus of this paper is URLLC scheduling for K users.*

A. Binary SNR Model

To obtain insight into our scheduling problem, we insert numbers in the above formulas from the 5G specifications[2] and deduce the number of RBs required to satisfy the URLLC SLA for given L and γ (assuming $\gamma_r^k = \gamma, \forall r, k$), shown in Fig. 1. For 32 Bytes, as few as 3 RBs with $\gamma > 0dB$ are enough to guarantee the required 99.999% reliability. Many SNR values lead to the same result, and hence the exact value of the SNR may not be crucial. Instead, we will often require that each user is assigned a large enough number (here 3) of RBs with strong SNR (e.g. $> 0dB$).

[1] Exploiting feedback within frames is avoided in practical systems since it complicates decision making, and the available time between slots is minimal.

[2] We mention that the number of channel uses are computed based on taking half a 5G subframe as a scheduling unit in the time domain, i.e., 12 subcarriers and 7 symbols per resource block; this is an envisioned strategy for supporting low latency traffic in 5G [19].

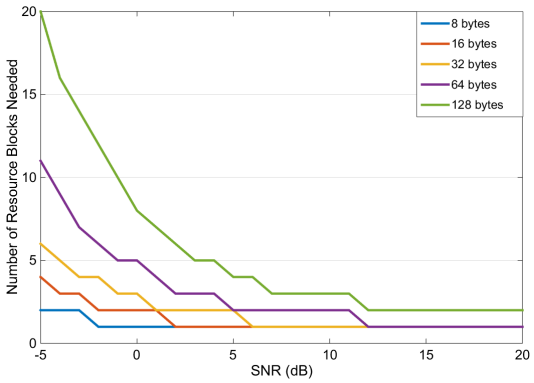


Fig. 1. Number of resource blocks needed for 99.999% reliability for different packet sizes in 5G NR. All resource blocks are assumed to have the same SNR.

The above motivates us to study the binary SNR model. We say RB r is *active* for user k (hence in state 1) if γ_r^k is larger than a threshold, and inactive (state 0) otherwise, and additionally d_k active RBs suffice for user k to achieve the required packet error probability. Essentially we have used a user-specific SNR threshold and treat all values below the threshold as zero. In the following, we set $\delta_r^k = 1$ if r is active for k, and 0 otherwise. This model is called the *binary SNR model*, while the previous one will be called continuous.

We show that *(i)* the joint URLLC admission control and scheduling problem remains complex under the binary SNR model, *(ii)* the difference between binary and continuous model is small in the LTE/5G specifications, and *(iii)* we may use the results of the binary model to obtain an efficient algorithm for the continuous model under any specifications.

B. URLLC Feasibility and Scheduling

An essential constraint for scheduling is to allocate each RB to at most one user, written as

$$\sum_k x_r^k \leq 1, \quad r = 1, \ldots, R. \tag{2}$$

Hereinafter, consider the set of *URLLC schedules*:

$$\mathcal{X} \triangleq \left\{ \boldsymbol{x} \in \{0,1\}^{K \times R} \;\middle|\; (2) \text{ satisfied} \right\}.$$

Next, we are interested in URLLC schedules $\boldsymbol{x} \in \mathcal{X}$ that also support the SLA of a set of users $\mathcal{K}$. Specifically, that the SLA of user k is satisfied in the continuous SNR model if

$$p_e^k(\boldsymbol{x}) \leq 1 - \theta, \tag{3}$$

and in the binary SNR model if

$$\sum_r x_r^k \geq d_k. \tag{4}$$

Definition 2 (Feasibility)**.** *We say that a URLLC schedule $\boldsymbol{x} \in \mathcal{X}$ is feasible for users $\mathcal{M}$ in the binary (continuous) SNR model if eq.* (4) *(eq.* (3)*) is satisfied for all $k \in \mathcal{M} \subseteq \mathcal{K}$. The set of all such feasible schedules is denoted with $\mathcal{X}(\mathcal{M}) \subseteq \mathcal{X}$.*

Given a set of users $\mathcal{K}$, their SLAs, and their SNRs $\boldsymbol{\gamma}$, an important question regards the URLLC scheduling feasibility:

Q1: is there a feasible schedule for $\mathcal{K}$?

Additionally, if the answer to Q1 is "yes" and hence $\mathcal{X}(\mathcal{K})$ is non-empty, then we would like to find an $\boldsymbol{x} \in \mathcal{X}(\mathcal{K})$, e.g., by solving the following feasibility problem:

URLLC Scheduling:

$$\min_{\boldsymbol{x} \in \mathcal{X}(\mathcal{K})} 0 \tag{5}$$

Both Q1 and (5) involve searching in a combinatorial space exponential to $K \times R$, and therefore are possibly complex to address. Unexpectedly, in Section IV we show that under the binary SNR model both questions can be addressed in polynomial time.

C. URLLC Admission Control

Next we consider the case that the answer to Q1 is "no", i.e., when scheduling all users in $\mathcal{K}$ is infeasible. In this case, we are interested in the following admission control question.

Q2: is there a schedule that satisfies at least m users?

We also consider a more general approach, where we assign to users non-negative utilities w^k, $k \in \mathcal{K}$, which are collected only for users with satisfied SLAs. We would like to choose the schedule $\boldsymbol{x} \in \mathcal{X}$ that collects the maximum total utility, which corresponds to ensuring the URLLC SLA for the most important users. The user-specific utilities can be tweaked according to the application, in order to provide preferential admission of users into the system; for instance, high utility users may correspond to remote controlled vehicles. We introduce the admission variable $z^k \in \{0,1\}$, which takes value 1 if user k will be served and 0 otherwise. Then we consider the following optimization:

URLLC Utility Maximization (UUM):

$$\text{maximize}_{\boldsymbol{x} \in \mathcal{X}, \boldsymbol{z}} \sum_{k=1}^{K} w^k z^k \tag{6}$$

$$\text{s.t.} \sum_r x_r^k \geq d_k z^k, \quad k = 1, \ldots, K \tag{7}$$

$$x_r^k \leq \delta_r^k, \qquad \forall (r,k) \tag{8}$$

$$x_r^k \in \{0,1\}, \quad \forall (r,k) \quad z^k \in \{0,1\}, \quad \forall k. \tag{9}$$

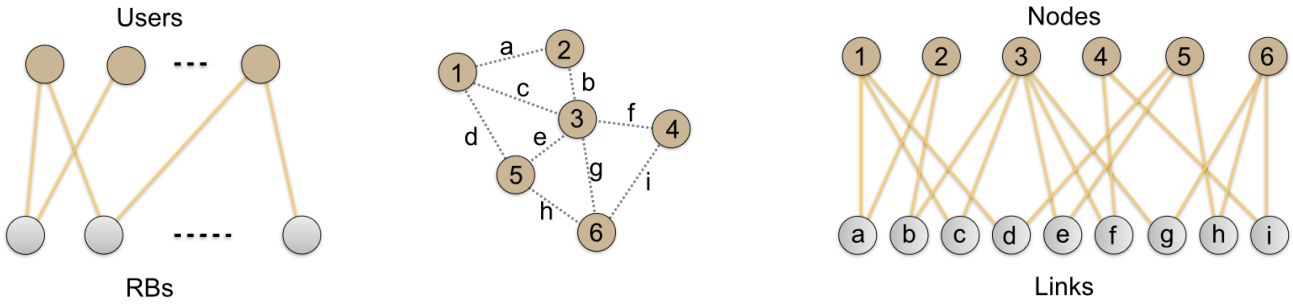
Note that the linear objective (6) drives the solution towards the $\mathcal{M}$-SLA feasible schedule that collects the highest utility. Constraint (7) ensures the SLA satisfaction of all selected users with $z^k = 1$ (it should be replaced with $p_e^k(\boldsymbol{x}) + \theta - 1 \leq 1 - z^k$ for the continuous SNR model). Constraint (8) restricts schedules on active resource blocks (should be omitted in continuous SNR), and (9) forces the variables to be integers.

For the binary model, (6)-(9) is an Integer Linear Program (ILP) of possibly large dimensions. In Section III we show that Q2 and the UUM problem are both complex to address exactly. Then in Sections V-VI we provide approximations for both models that are polynomial-time computable.

III. Complexity of URLLC Admission Control

Our analysis begins with the complexity of answering the question Q2: *is there a schedule that satisfies at least m users?*

To prove our complexity theorem we will make use of the notion of independent sets on graphs. Consider an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with vertices $\mathcal{V}$ and edges $\mathcal{E}$, such as the one in the example of Fig. 2-(b).



(a) URLLC graph (b) Abstract graph $\mathcal{G}$ (c) Bipartite graph $\mathcal{H}(\mathcal{G})$

Fig. 2. (a) Bipartite model of the URLLC scheduling problem. (b) Graph G on which we want to compute a maximum independent set. (c) Bipartite graph modeling the connectivity of G.

Definition 3. *A subset of nodes $I \subseteq \mathcal{V}$ is called an independent set on graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ if any two nodes $v, u \in I$ are not neighbors on $\mathcal{G}$, i.e. $\{v, u\} \notin \mathcal{E}$.*

For example, one may verify that the set of nodes $\{2, 4, 5\}$ forms an independent set in the graph of Fig. 2-(b). From the literature we know that finding the maximum independent set in a graph is NP-hard, and its decision version "*is there an independent set of size at least m?*" is NP-complete [22].

Further, the connectivity of a graph can be represented in an alternative way, shown in Fig. 2-(c). Specifically, we may consider a bipartite graph $\mathcal{H}(\mathcal{G}) = (\mathcal{V} \cup \mathcal{E}, \mathcal{L})$ which is built from $\mathcal{G}$ as follows. The left node partition is set to $\mathcal{V}$ and the right node partition is set to $\mathcal{E}$, hence the nodes of $\mathcal{H}$ is the set $\mathcal{V} \cup \mathcal{E}$. Then $(v, e) \in \mathcal{L}$ if and only if $v \in e$. An independent set on $\mathcal{G}$ is a selection of left nodes of $\mathcal{H}$ such that the induced graph (formed by keeping the selected left nodes, the right nodes, and the surviving links) has right degree at most 1.

Theorem 1. *It is NP-complete to determine whether there exists a feasible URLLC schedule for $|\mathcal{M}| \geq m$ users (i.e. to answer Q2) in the binary SNR model.*

Proof: We will establish a reduction from the decision problem *is there an independent set of size at least m?*, which is NP-complete.

For an instance of Q2 in the binary model we may construct the following URLLC bipartite graph connecting users and RBs, where we draw a link from user k to RB r if $\delta_r^k = 1$ in the binary model (i.e. the RB is active for this user), cf. Fig. 2-(a). In order to perform the reduction, we first assume that there exists a *URLLC oracle* algorithm that given input $(\mathcal{K}, \mathcal{R}, \boldsymbol{\delta}, d_k, m)$[3] it provides the answer to Q2 in polynomial time. The reduction is to show that we may use this algorithm to solve every instance of the independent set decision problem.

Consider any graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, and the decision problem *"is there an independent set of size at least m?"* on the corresponding bipartite connectivity graph $\mathcal{H}(\mathcal{G}) = (\mathcal{V} \cup \mathcal{E}, \mathcal{L})$. Run the URLLC oracle with $\mathcal{K} = \mathcal{V}$, $\mathcal{R} = \mathcal{E}$, $\delta_e^v = 1$ if and only if $v \in e$ (e is incident to $v \in \mathcal{V}$ in $\mathcal{G}$), and $d_v = M_v$, where M_v is the degree of node v in $\mathcal{G}$. Report the answer of the oracle as the answer to our decision problem.

First, the above procedure runs in polynomial steps by the hypothesis of the oracle. To prove the correctness, we work as follows. Suppose that the URLLC oracle encounters a subset of users $\mathcal{U} \subseteq \mathcal{V}$ that are found SLA feasible, then there exists an activation of links (feasible schedule) such that *(i)* for every $v \in \mathcal{U}$, all M_v links are activated (i.e. all d_v transmissions are scheduled), and *(ii)* the number of activated links incident to any right node is ≤ 1 (since the schedule is feasible); it follows that $\mathcal{U}$ is an independent set on $\mathcal{G}$. We conclude that *the URLLC oracle finds a set $\mathcal{U}$ to be SLA feasible if and only if $\mathcal{U}$ forms and independent set on $\mathcal{G}$*. Hence, if the oracle returns "yes", then we know there exists an independent set of size m in $\mathcal{G}$. Conversely, if it returns "no", we know that there is no independent set of size m or larger in $\mathcal{G}$. This completes the reduction. The problem is NP-complete because, as we will show next, given a candidate solution $\mathcal{M}$ we may verify its feasibility in polynomial time, hence our problem is in NP. ■

Corollary 2. *The UUM problem in* (6) *is NP-hard.*

Corollary 2 can be proven by a reduction from Q2. Suppose UUM can be solved in polynomial time for any instance, then select an instance with $w^k = 1,\ \forall k$, and run the UUM oracle. The obtained maximum utility can be used to directly determine the answer to Q2. That is, UUM is no easier than Q2, which, in turn, is no easier than the independent set decision problem.

Corollary 3. *In the continuous SNR model obtained by replacing* (7) *with $p_e^k(\boldsymbol{x}) + \theta - 1 \leq 1 - z^k$ and omitting* (8)*, Q2 and UUM are NP-hard.*

In the continuous SNR model, consider instances that have only two possible values for SNRs, 0 and the threshold used for the binary SNR model. The arising instances coincide with respective instances in the binary SNR model, hence all instances of the binary model also appear in the continuous, hence also the hard ones.

Finally, we can use the independent set construction above to bound the approximation ratio of the UUM problem in the binary SNR model:

Proposition 4. *Denote $d = \max_k[d_k]$, and consider the case $w^k = 1,\ \forall k$. If $P \neq NP$ and the "Unique Games Conjecture"*[4] *holds, the UUM problem under the binary SNR model admits*

[3]We remind that, $\mathcal{K}$ stands for the set of all URLLC users, $\mathcal{R}$ stands for the set of all resource blocks, $\boldsymbol{\delta}$ is a matrix with $\delta_r^k = 1$ denoting that block r is "active" for user k,d_k is the number of required successful transmissions for the satisfaction of the URLLC constraint and m is the number of users we want to satisfy.

[4]For more details about this conjecture cf. [23]

no polynomial time algorithm with approximation ratio better than $\frac{(\log_2 d)^2}{d}$.

Proof: Indeed, notice that $d = \max_k[d_k]$ is the maximum left degree of the bipartite graph, which is $d_{max}(\mathcal{G})$ when transforming a general graph $\mathcal{G}$ to it, as described in the proof of Theorem 1. Therefore, if a better approximation was possible in polynomial time, that oracle could be used to obtain in polynomial time an independent set that approximates the optimal better than $\frac{(\log_2 d)^2}{d}$, which, if the Unique Games Conjecture holds, is not possible unless $P=NP$ [22]. ■

IV. Optimal URLLC Scheduling in Binary SNR

Having established that answering Q2 and solving the UUM problem are both very complex, in this Section we shift our attention to the scheduling problems. Surprisingly, we will prove in the binary SNR model that given a designated set of users $\mathcal{M}$, we can answer if the set is schedulable (Q1) and find a feasible schedule (when the answer is "yes") in polynomial time. In turn, this result is very important as *(i)* it allows us to achieve maximum URLLC scheduling performance with low-complexity algorithms, and *(ii)* will lead us to obtain an efficient admission control algorithm.

Our analysis is based on the concept of Total Unimodularity of a matrix. Formally we have:

Definition 4 (Total Unimodularity)**.** *A (square) matrix* $\boldsymbol{B}$ *is called unimodular if det(*$\boldsymbol{B}$*)* $\in \{-1, 0, 1\}$*. A matrix* $\boldsymbol{A}$ *is called totally unimodular if every square submatrix of* $\boldsymbol{A}$ *is unimodular.*

This concept is of great importance in Linear Programming. Let the polyhedron $\mathcal{P}(\boldsymbol{A}, \boldsymbol{b}) = \{\boldsymbol{x} : \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b}, \boldsymbol{x} \geq 0\}$ be the feasible set of a Linear Program (LP). If a matrix $\boldsymbol{A}$ is totally unimodular and $\boldsymbol{b}$ a vector of integers, then the vertices of $\mathcal{P}(\boldsymbol{A}, \boldsymbol{b})$ have all integral elements [24, Theorem 13.2]. This implies that any LP of the form $\max_{\boldsymbol{x}\in\mathcal{P}(\boldsymbol{A},\boldsymbol{b})} \boldsymbol{c}^T\boldsymbol{x}$ has at least one integral solution, and if its solution is unique, then it is necessarily integral.

Our strategy will be (i) to show that an appropriate relaxation of the UUM is an LP with a totally unimodular constraint matrix and (ii) to construct a mock objective function such that the corresponding LP has unique solution.

Recall that we are given a set of users $\mathcal{M}$ and we want to decide if there exists an $\mathcal{M}$-SLA feasible schedule, that is if set $\mathcal{X}(\mathcal{M})$ is nonempty. As a first step, starting from the feasibility space of (6), we can fix $z^k = 1$, if $k \in \mathcal{M}$ and 0 otherwise, to arrive at an expression for $\mathcal{X}(\mathcal{M})$:

$$\mathcal{X}(\mathcal{M}) = \left\{ \boldsymbol{x} \in \{0,1\}^{R\times|\mathcal{M}|} \;\middle|\; \begin{array}{ll} \sum_r x_r^k \geq d_k, & k \in \mathcal{M} \\ \sum_k x_r^k \leq 1, & r \in \mathcal{R} \\ x_r^k \leq \delta_r^k, & \forall (r,k) \end{array} \right\}$$

The idea is that we will relax the scheduling variables to $\boldsymbol{x} \in [0,1]^{R\times|\mathcal{M}|}$ and obtain an LP. In order to force the LP to have a unique solution, we introduce random costs c_r^k, which are drawn uniformly from $[0, 1]$. We then have:

Relaxed LP

$$\min_{\boldsymbol{x}\in\mathcal{X}(\mathcal{M})} \sum_{r=1}^{R}\sum_{k=1}^{|\mathcal{M}|} c_r^k x_r^k \tag{10}$$

Lemma 5. *The constraint matrix of the relaxed LP* (10) *is totally unimodular.*

Proof: First, let us stack all variables in vector

$$\boldsymbol{\xi} = [x_1^1, x_1^2, \ldots, x_1^M, x_2^1, \ldots, x_2^M, \ldots, x_R^1, \ldots x_R^M]^T$$

The constraints $\boldsymbol{x} \in \mathcal{X}(\mathcal{M})$ of the relaxed LP then have the form $\boldsymbol{A}\boldsymbol{\xi} \leq \boldsymbol{b}, \boldsymbol{\xi} \geq \boldsymbol{0}$, with constraint matrix[5]

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{A}_1^T & \boldsymbol{A}_2^T & \boldsymbol{I}_{RM} \end{bmatrix}^T$$

, where:

- $\boldsymbol{A}_1$ is the $M \times RM$ matrix corresponding to the first set of constraints, i.e. $\boldsymbol{A}_1 = -[\boldsymbol{I}_M|\boldsymbol{I}_M|...|\boldsymbol{I}_M]$, where the identity matrix of size M appears R times.
- $\boldsymbol{A}_2$ is the $R \times RM$ matrix corresponding to the second set of constraints, i.e. its r-th row has elements in columns $\{(r-1)M+1, (r-1)M+2, ..., (r-1)M+M\}$ equal to one and the rest zero.

We can thus observe that all of the following are true for the matrix $\boldsymbol{A}_3 = [-\boldsymbol{A}_1^T \quad \boldsymbol{A}_2^T]^T$:

1) Every entry is either 0 or 1.
2) At each column, there two nonzero elements, both taking value 1.
3) Sets $\mathcal{S}_1, \mathcal{S}_2$ that have as elements the rows of $\boldsymbol{A}_1$ $\boldsymbol{A}_2$, respectively are disjoint.
4) For every column one of the rows with a nonzero element belongs to $\mathcal{S}_1$ and the other belongs to $\mathcal{S}_2$.

It then follows [24, Th. 13.3] that the matrix $\boldsymbol{A}_3$ is totally unimodular. Since multiplying rows of a totally unimodular matrix by -1 results in a totally unimodular matrix, the matrix $\boldsymbol{A}_4 = \begin{bmatrix} \boldsymbol{A}_1^T & \boldsymbol{A}_2^T \end{bmatrix}^T$ is totally unimodular, therefore the constraint matrix $\boldsymbol{A} = \begin{bmatrix} \boldsymbol{A}_4^T & \boldsymbol{I}_{RM} \end{bmatrix}^T$ is totally unimodular as well, completing the proof. ■

We now present Algorithm 1, which uses the relaxed LP to answer Q1. It has to be noted that solving the relaxed LP here means to run a procedure which returns an optimal solution if the LP is feasible and an indication that is infeasible otherwise, which can be done, for example, using the Ellipsoid algorithm, see for example [24, Chapter 8].

Algorithm 1 Check feasibility in the binary SNR model

```
Construct the relaxed problem.
Select c_{r,k} ∈ [0,1] uniformly at random.
Solve the relaxed LP (10).
if The LP is feasible then
    Return: (yes) M is feasible
    Return: The solution x* of the LP as a schedule.
else
    Return: (no) M is not feasible
end if
```

Theorem 6. *Algorithm 1 always returns a correct answer to Q1, and with probability* 1 *a feasible schedule if the answer is "yes".*

Proof: If the relaxed LP is infeasible, we may immediately conclude that there is no feasible schedule for $\mathcal{M}$ users. On the other hand, if the LP is feasible then since from Lemma 5 its

[5]We use the notation $\boldsymbol{I}_N$ for the identity matrix of size N.

corresponding constraint matrix is totally unimodular and the right hand sides of the constraints are integers, at least one solution should be integral (see [24, Theorem 13.2]), therefore a feasible schedule for $\mathcal{M}$ users. We can then conclude that Algorithm 1 returns a correct answer to Q1.

Assume now that the relaxed LP is feasible. Since the cost vector $\boldsymbol{c}$ is chosen uniformly at random, the probability that the hyperplane $\boldsymbol{c}^T\boldsymbol{x} = 0$ is parallel with any of the facets of the LP polyhedron is zero, therefore the relaxed LP has a unique solution with probability one. Due to total unimodularity of the constraint matrix, the unique solution obtained in this way is necessarily integral, therefore a feasible schedule. ∎

An immediate corollary is that Q1 can be answered in polynomial time:

Corollary 7. *Question Q1 can be answered in polynomial time (in $R, |\mathcal{M}|$ and N) for the binary SNR model.*

Proof: The relaxed problem has $R|\mathcal{M}|$ variables and $R + |\mathcal{M}| + 2R|\mathcal{M}|$ constraints, which are both polynomial in R and $|\mathcal{M}|$. Hence Q1 can be answered by checking the feasibility and finding the solution of an LP with size polynomial to R and $\mathcal{M}|$, which can be done in polynomial time, e.g. with the Ellipsoid algorithm [24, Chapter 8]. ∎

V. Approximate Admission Control in Binary SNR

In this section, we prove that the GREEDY algorithm guarantees $1/(d+1)$ of the maximum in the joint admission control and scheduling problem, where d refers to the maximum number of active RBs required among users. For example, in case $d_k = d = 3$ (as in the introduction), then this shows that GREEDY achieves at least 25% of the optimal. Following proposition 4, we may conclude that GREEDY achieves the optimal approximation up to poly-logarithmic terms. Hence, we can not hope for a much better approximation guarantee.

The GREEDY algorithm works as follows: *(i)* The users are ordered in decreasing utilities $w^{(1)} \geq \cdots \geq w^{(K)}$ with ties broken randomly, *(ii)* starting from highest utilities we allocate to k user d_k RBs at random, *(iii)* if there are not enough RBs, then the user is rejected altogether.

Proposition 8. *Let $d = \max_{k\in\mathcal{K}} d_k$, $w^k = 1, \forall k$. Then* GREEDY *guarantees an approximation ratio of at least $\frac{1}{d+1}$ for the binary SNR model.*

Proof: We assume, without loss of generality, that $\sum_r \delta_r^k \geq d_k$, $\forall k \in \mathcal{K}$, i.e. each user has enough active resource blocks (if not we may eliminate those users and redefine $\mathcal{K}$). Let $\hat{\boldsymbol{x}}$ denote the schedule returned by GREEDY, $\hat{\boldsymbol{z}}$ the corresponding admission and $\boldsymbol{z}_*$ the optimal admission. We note that if user k is admitted in GREEDY it gets allocated exactly the minimum required number of resource blocks and if not it is allocated no resource blocks at all.

We may partition the sets of users and resource blocks into two (disjoint) subsets: $\mathcal{K}_1, \mathcal{R}_1$ are the admitted users and assigned resource blocks by GREEDY respectively, while $\mathcal{K}_0 = \mathcal{K} \setminus \mathcal{K}_1, \mathcal{R}_0 = \mathcal{R} \setminus \mathcal{R}_1$. The following are true: *(i)* $|\mathcal{R}_1| \leq d|\mathcal{K}_1|$, since each user needs at most d resource blocks and *(ii)* no user in $\mathcal{K}_0$ can be scheduled successfully with only RBs from $\mathcal{R}_0$ (by the premise that they are not scheduled by GREEDY).

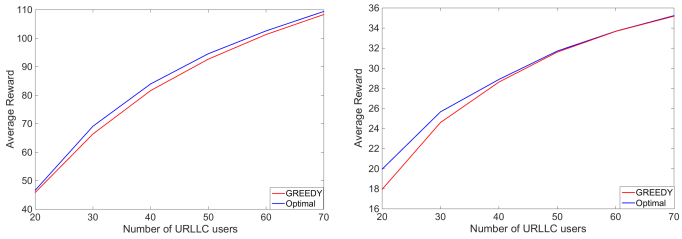


Fig. 3. Performance comparison (each point is the average over 10000 trials) between GREEDY and optimal in the binary SNR model. (left) Each user has a random utility in $[0, 5]$. (right) All users have the same utility.

Since no user in $\mathcal{K}_0$ can be scheduled with RBs exclusively from $\mathcal{R}_0$, we observe that an upper bound on the admissible users (hence on $\sum_k z_*^k$) is $|\mathcal{R}_1| + |\mathcal{K}_1|$, which is attained if all users in $\mathcal{K}_1$ can be scheduled exclusively with $\mathcal{R}_0$ RBs and $|\mathcal{R}_1|$ users from $\mathcal{K}_0$ can be scheduled with 1 RB from $\mathcal{R}_1$ and the rest from $\mathcal{R}_0$. Therefore, we have

$$\sum_k z_*^k \leq |\mathcal{R}_1| + |\mathcal{K}_1| \leq d|\mathcal{K}_1| + |\mathcal{K}_1| = (d+1)\sum_k \hat{z}^k,$$

finishing the proof. ∎

We further mention that in case $d = 1$, and hence $d_k = 1$, $\forall k$, then the URLLC joint admission control and scheduling simplifies to a maximum weighted matching problem, which can be solved optimally in polynomial time, see e.g. [24, Chapter 10]. Furthermore, if there exists a $\mathcal{K}$-SLA feasible schedule, we can find it in polynomial time by running Algorithm 1 before GREEDY, thus the performance is optimal in this case as well.

Figure 3 shows a performance comparison between GREEDY and the optimal solution of the UUM problem, which was obtained via an ILP solver. We perform the comparison for different simulation settings. Users are placed at random in an area such that their mean SNRs computed using path loss models are between 0 and 20 dB. The system has 50 resource blocks, each experiencing $i.i.d.$ Rayleigh fading. The thresholds are chosen such that $d_k = 1$ if the average SNR of user k is over $12.5dB$, $d_k = 3$ if it is lower than $4dB$ and $d_k = 2$ in between. In the left case, each user has a random utility in $[0, 5]$, while in the right all users have the same utility.

We can observe that in all cases the gap between the performance of the two algorithms is very small (in fact GREEDY performs much closer to optimal than the predicted guarantee), therefore validating our intuition that GREEDY is an effective solution for the admission control problem the binary SNR model. In addition, GREEDY is a fast algorithm compared to the one that solves the ILP optimally; on average, the runtime of GREEDY was about 25 times faster than solving the ILP exactly.

Next, we will use the GREEDY algorithm together with the result of Section III that the feasibility problem can be solved in polynomial time for the binary SNR model to provide an admission controller for the continuous SNR model.

VI. Admission Control in Continuous SNR

We now shift our attention to the continuous SNR model, where joint coding is performed over the resource blocks that belong to the same user. As we discussed in Section III, the binary SNR model is a special instance of this general case,

therefore all hardness results hold here as well. In addition, even the fractional relaxation of the problem is difficult to address, since it is in fact a non-convex problem. To see this, recall that the corresponding probability of incorrect decoding is given by (1). In order to satisfy the SLA constraint we should have, using also (7) that

$$\sum_r (n\log_2(1+\gamma_r^k) + 0.5\log_2(n))x_r^k - Q^{-1}(1-\theta)\sqrt{n\sum_r V(\gamma_r^k)x_r^k} - L_k \geq 0, \forall k.$$

Since the left hand side is a convex function of $\boldsymbol{x}$ and is required to be greater than 0, the points satisfying these constraints form a non-convex set.

A baseline greedy approach to solve this problem is to order the users in decreasing utilities, and then place one by one at random RBs if their constraint is satisfied. Below we propose a binary SNR-inspired heuristic, called *Iterative Thresholding Algorithm* (ITA) and show via experimentation that outperforms the mentioned baseline.

To design ITA, we have used the following insights gained from the analysis of the binary SNR model. Namely, we leverage the facts that *(i)* only a few resource blocks are needed for each user as illustrated by Fig. 1, *(ii)* if a feasible schedule exists for the binary SNR model, we can obtain it in polynomial time as we showed in Section V and *(iii)* GREEDY gets close to the optimal as we showed in Section V. Our idea is to start with a high SNR threshold, use the corresponding binary SNR model for this threshold to assign RBs via Alg. 1 if all users are schedulable (or via GREEDY otherwise), fix the satisfied users and their assigned RBs, and then progressively lower the SNR threshold repeating the same procedure. The proposed algorithm is detailed as Algorithm 2.

We compare Alg. 2 to the baseline explained above. Results regarding the problem of maximizing the number of admitted users (i.e. $w^k = 1$ for every k) are shown in Fig. 4. We examine three cases regarding user placement: (i) users are placed at random in the cell with mean SNRs (due to large scale fading) between $0dB$ and $20dB$ and the cases where users have the same mean SNR with a (ii) relatively low ($5dB$) and (iii) relatively high ($15dB$) value. We can observe that where the SNR is high (i.e. users are placed close to the Base Station) the performance of the two algorithms is almost the same. Moreover, the two algorithms admit similar number of users in all cases where the number of RBs is much lower than the total number of URLLC users. These results were to be expected since in the latter case most resource blocks are good for every user and in the former there are enough resources to find good RBs for each user. More interestingly though, when the number of RBs is comparable and/or lower to the number of URLLC users and the users are placed at random or have the same and relatively low mean SNRs–the regime where admission control really becomes an important aspect of the problem–*ITA outperforms the baseline algorithm by a margin that increases with the number of users and can reach up to around* 30%. The reason behind this gain is that ITA schedules efficiently groups of users for each threshold, while the baseline algorithm may use a resource block that is more useful to some other user.

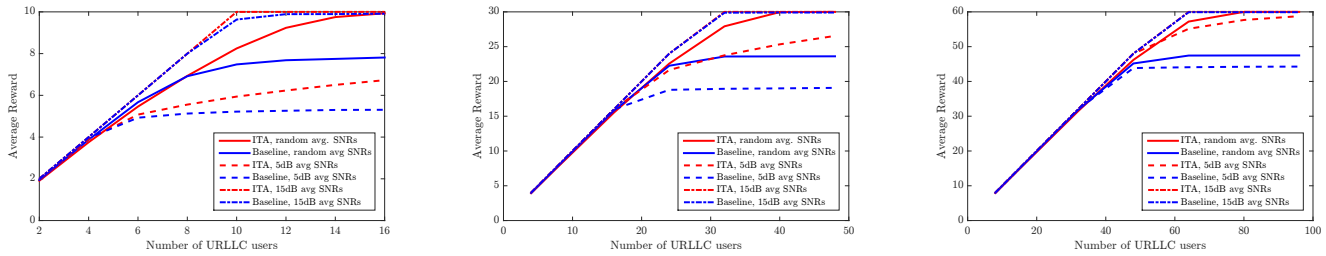
Algorithm 2 Iterative Thresholding Algorithm (ITA)

Initialization: $\mathcal{M} \leftarrow \mathcal{K}, \mathcal{S} \leftarrow \emptyset, \boldsymbol{x} \leftarrow \mathbf{0}$.
for $d = 1, 2, ..., D_{max}$ **do**
  Find the minimum SNR $s(d)$ such that d resource blocks of SNR $s(d)$ are sufficient.
  For each user $k \in \mathcal{M}$ and resource block $r \in \mathcal{R}$, put $\delta_{(r,k)} = \mathbb{1}_{\{\gamma_r^k \geq s(d)\}}$
  Run Algorithm 1 for users in $\mathcal{M}$ and blocks in $\mathcal{R}$.
  if the problem is feasible **then**
    Return $\boldsymbol{x}(d)$ as the feasible schedule.
  else
    if $d = 1$ **then**
      Run a maximum weighted matching algorithm in the resulting connectivity graph, return $\boldsymbol{x}(d)$ as the resulting schedule.
    else
      Run GREEDY for users in $\mathcal{M}$ and blocks in $\mathcal{R}$, return $\boldsymbol{x}(d)$ as the resulting schedule.
    end if
  end if
  Update the schedule: $\boldsymbol{x} \leftarrow \boldsymbol{x} + \boldsymbol{x}(d)$.
  Update the set of scheduled users: $\mathcal{S} \leftarrow \mathcal{S} \cup \left\{k \in \mathcal{M} : \sum_r \boldsymbol{x}_r^k(d) \geq d\right\}$
  Update the set of remaining users: $\mathcal{M} \leftarrow \mathcal{K} \setminus \mathcal{S}$.
  Update the set of available resources: $\mathcal{R} \leftarrow \mathcal{R} \setminus \{r \in \mathcal{R} : \sum_k x_r^k > 0\}$
  if $\mathcal{R} = \emptyset$ **then**
    Break from the loop
  end if
end for
Return the schedule $\boldsymbol{x}$.

Finally, we examine a setting where users are placed randomly in a cell but the reward a user will bring if admitted is proportional to the logarithm of its mean SNR. The rationale here is that an operator may want to prioritize admitting URLLC users with good general channel conditions, since it is more sustainable to serve these users for their whole session duration. Results are presented in Fig. 5. Same as before, ITA brings significant benefits over the baseline algorithm when the number of RBs becomes comparable or less than the number of users.

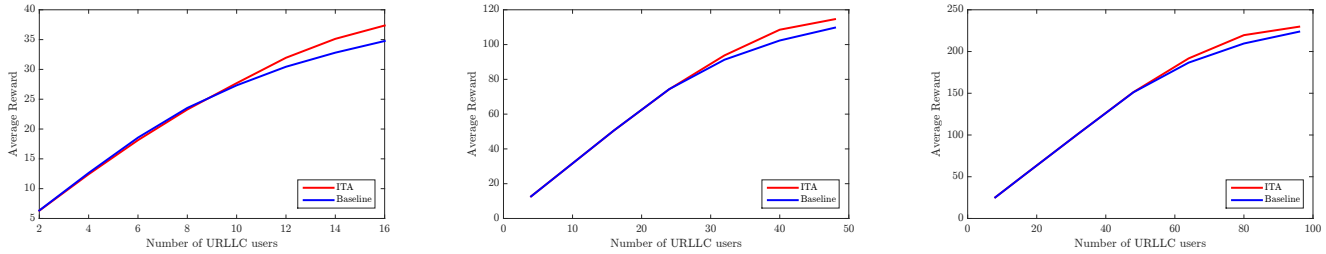
VII. Conclusion

In this paper, we proved, via a reduction to the maximum weighted independent set problem, that joint admission control and scheduling of URLLC users with time-frequency resource blocks is NP-hard even in a simplified problem, where resource blocks have binary SNRs and user k requests at least M_k resource blocks at state 1 to satisfy her reliability constraints. However, for this binary SNR problem, checking if a set of users is feasible can be done in polynomial time via a proper LP. This implies that hardness comes mainly from the admission control part of the problem, and we proposed a



(a) 10 Resource Blocks (b) 30 Resource Blocks (c) 60 Resource Blocks

Fig. 4. Average number of admitted URLLC users (here admitting each user has unit reward) with (a) low (b) medium and (c) high number of resources allocated to URLLC traffic. Each user has a packet size of 32 Bytes and requests 99.999% reliability. The results shown are averages over 5000 trials with generated Rayleigh fading.



(a) 10 Resource Blocks (b) 30 Resource Blocks (c) 60 Resource Blocks

Fig. 5. Average reward gathered from admitting URLLC users with (a) low (b) medium and (c) high number of resources allocated to URLLC traffic. Each user has a packet size of 32 Bytes and requests 99.999% reliability. The results shown are averages over 5000 trials with generated Rayleigh fading.

greedy algorithm with a provable approximation ratio. Finally, for the problem with continuous SNRs, we proposed a heuristic that iteratively selects thresholds for the SNRs and iteratively solves a binary SNR problem. Our heuristic outperforms the baseline in simulations.

REFERENCES

[1] 3GPP, "Study on Scenarios and Requirements for Next Generation Access Technologies (Release 15)," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) Group Radio Access Network 38.913, 06 2018, version 15.2.0.

[2] B. Holfeld, D. Wieruch, T. Wirth, L. Thiele, S. A. Ashraf, J. Huschke, I. Aktas, and J. Ansari, "Wireless Communication for Factory Automation: an opportunity for LTE and 5G systems," *IEEE Commun. Mag.*, vol. 54, no. 6, pp. 36–43, Jun 2016.

[3] H. Zhang, N. Liu, X. Chu, K. Long, A. H. Aghvami, and V. C. M. Leung, "Network Slicing Based 5G and Future Mobile Networks: Mobility, Resource Management, and Challenges," *IEEE Commun. Mag.*, vol. 55, no. 8, pp. 138–145, 2017.

[4] C. Campolo, A. Molinaro, A. Iera, and F. Menichella, "5G Network Slicing for Vehicle-to-Everything Services," *IEEE Wireless Commun. Mag.*, vol. 24, no. 6, pp. 38–45, Dec 2017.

[5] J. J. Nielsen, R. Liu, and P. Popovski, "Ultra-Reliable Low Latency Communication Using Interface Diversity," *IEEE Trans. Commun.*, vol. 66, no. 3, pp. 1322–1334, Mar 2018.

[6] R. Kotaba, C. N. Manchón, T. Balercia, and P. Popovski, "Uplink transmissions in URLLC systems with shared diversity resources," *IEEE Wireless Commun. Lett.*, 2018.

[7] J. Rao and S. Vrzic, "Packet Duplication for URLLC in 5G: Architectural Enhancements and Performance Analysis," *IEEE Netw.*, vol. 32, no. 2, pp. 32–40, Mar 2018.

[8] G. Durisi, T. Koch, and P. Popovski, "Toward Massive, Ultra-reliable, and Low-Latency Wireless Communication With Short Packets," *Proc. IEEE*, vol. 104, no. 9, pp. 1711–1726, Sept 2016.

[9] J. Huang, V. G. Subramanian, R. Agrawal, and R. A. Berry, "Downlink scheduling and resource allocation for OFDM systems," *IEEE Trans. Wireless Commun.*, vol. 8, no. 1, pp. 288–296, Jan 2009.

[10] C. She, C. Yang, and T. Q. S. Quek, "Radio Resource Management for Ultra-Reliable and Low-Latency Communications," *IEEE Commun. Mag.*, vol. 55, no. 6, 2017.

[11] ——, "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.

[12] J. Arnau and M. Kountouris, "Delay performance of MISO wireless communications," in *WiOpt*, May 2018.

[13] A. Destounis, G. S. Paschos, J. Arnau, and M. Kountouris, "Scheduling URLLC users with reliable latency guarantees," in *WiOpt*, May 2018.

[14] M. Sharma and X. Lin, "Ofdm downlink scheduling for delay-optimality: Many-channel many-source asymptotics with general arrival processes," in *ITA Workshop*, 2011.

[15] S. Bodas, S. Shakkottai, L. Ying, and R. Srikant, "Low-complexity scheduling algorithms for multichannel downlink wireless networks," *IEEE/ACM Trans. Netw.*, vol. 20, no. 5, pp. 1608–1621, Oct 2012.

[16] A. Anand, G. de Veciana, and S. Shakkottai, "Joint Scheduling of URLLC and eMBB Traffic in 5G Wireless Networks," in *IEEE INFOCOM*, 2018.

[17] A. Anand and G. de Veciana, "Resource Allocation and HARQ Optimization for URLLC Traffic in 5G Wireless Networks," *IEEE J. Sel. Areas Commun*, vol. 36, no. 11, Nov 2018.

[18] E. Fountoulakis, N. Pappas, Q. Liao, V. Suryaprakash, and D. Yuan, "An examination of the benefits of scalable TTI for heterogeneous traffic management in 5G networks," in *RAWNET*, 2017.

[19] 3GPP, "NR; Physical channels and modulation (Release 15)," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) Group Radio Access Network 38.211, 06 2018, version 15.2.0.

[20] Y. Polyanskiy, H. V. Poor, and S. Verdu, "Channel coding rate in the finite blocklength regime," *IEEE Trans. Inf. Theory*, vol. 56, no. 5, pp. 2307–2359, May 2010.

[21] W. Yang, G. Durisi, T. Koch, and Y. Polyanskiy, "Quasi-static multiple-antenna fading channels at finite blocklength," *IEEE Trans. Inf. Theory*, vol. 60, no. 7, pp. 4232–4265, Jul. 2014.

[22] P. Austrin, S. Khot, and M. Safra, "Inapproximability of vertex cover and independent set in bounded degree graphs," *Theory Comput.*, vol. 7, pp. 27–43, 2011.

[23] S. Khot, "On the unique games conjecture," in *FOCS*, 2005.

[24] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*. Mineola, New York: Dover Publications Inc., 1998.