Joint Scheduling of URLLC and eMBB Traffic in 5G Wireless Networks

Arjun Anand*, Gustavo de Veciana*, and Sanjay Shakkottai*

*Department of Electrical and Computer Engineering, The University of Texas at Austin

***Abstract*—Emerging 5G systems will need to efficiently support both broadband traffic (eMBB) and ultra-low-latency (URLLC) traffic. In these systems, time is divided into slots which are further sub-divided into minislots. From a scheduling perspective, eMBB resource allocations occur at slot boundaries, whereas to reduce latency URLLC traffic is pre-emptively overlapped at the minislot timescale, resulting in selective superposition/puncturing of eMBB allocations. This approach enables minimal URLLC latency at a potential rate loss to eMBB traffic.**

We study joint eMBB and URLLC schedulers for such systems, with the dual objectives of maximizing utility for eMBB traffic while satisfying instantaneous URLLC demands. For a linear rate loss model (loss to eMBB is linear in the amount of superposition/puncturing), we derive an optimal joint scheduler. Somewhat counter-intuitively, our results show that our dual objectives can be met by an iterative gradient scheduler for eMBB traffic that anticipates the expected loss from URLLC traffic, along with an URLLC demand scheduler that is oblivious to eMBB channel states, utility functions and allocations decisions of the eMBB scheduler. Next we consider a more general class of (convex) loss models and study optimal online joint eMBB/URLLC schedulers within the broad class of channel state dependent but time-homogeneous policies. We validate the characteristics and benefits of our schedulers via simulation.

***Index Terms*—wireless scheduling, URLLC traffic, 5G systems**

I. Introduction

An important requirement for 5G wireless systems is its ability to efficiently support both broadband and ultra-low-latency reliable communications. On one hand, broadband traffic – formally, enhanced Mobile Broadband (eMBB) – should support gigabit per second data rates (with a bandwidth of several 100 MHz) with moderate latency (a few milliseconds). On the other hand, Ultra Reliable Low Latency Communication (URLLC) traffic requires extremely low delays (0.25-0.3 msec/packet) with very high reliability (99.999%) [1]. To satisfy these heterogenous requirements, the 3GPP standards body has proposed an innovative *superposition/puncturing* framework for multiplexing URLLC and eMBB traffic in 5G cellular systems.

The proposed scheduling framework has the following structure [1]. As with current cellular systems, time is divided into slots, with proposed one millisecond (msec) slot duration. Within each slot, eMBB traffic can share the bandwidth over the time-frequency plane (see Figure 1). The sharing mechanism can be opportunistic (based on the channel states of various users); however, the eMBB shares are decided by the beginning, and fixed for the duration of a slot[1].

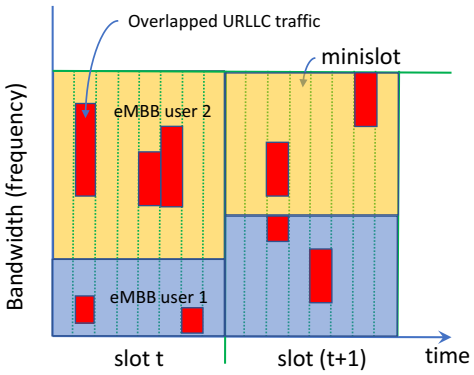

Fig. 1. Illustration of superposition/puncturing approach for multiplexing eMBB and URLLC: Time is divided into slots, and further subdivided into minislots. eMBB traffic is scheduled at the beginning of slots (sharing frequency across two eMBB users), whereas URLLC traffic can be dynamically overlapped (superpose/puncture) at any minislot.

URLLC downlink traffic may arrive during an ongoing eMBB transmission; if tight latency constraints are to be satisfied, they cannot be queued until the next slot. Instead each eMBB slot is divided into minislots, each of which has a 0.125 msec duration[2]. Thus upon arrival URLLC demand can be immediately scheduled in the next minislot *on top of the ongoing eMBB transmissions.* If the Base Station (BS) chooses non-zero transmission powers for both eMBB and overlapping URLLC traffic, then this is referred to as *superposition.* If eMBB transmissions are allocated zero power when URLLC traffic is overlapped, then it is referred to as *puncturing* of eMBB transmissions. The superposed/punctured URLLC traffic is sufficiently protected (through coding and HARQ if necessary) to ensure that it is reliably transmitted. At the end of an eMBB slot, the BS can signal the eMBB users the locations, if any, of URLLC superposition/puncturing. The eMBB user can in turn use this information to decode transmissions, with some possible loss of rate depending on the amount of URLLC overlaps. We refer to [1], [2] for additional details.

[1]The sharing granularity among various eMBB users is at the level of Resource Blocks (RB), which are small time-frequency rectangles within a slot. In LTE today, these are (1 msec $\times$ 180 KHz), and could be smaller for 5G systems.

[2]In 3GPP, the formal term for a 'slot' is eMBB TTI, and a 'minislot' is a URLLC TTI, where TTI expands to Transmit Time Interval.

A key problem in this setting is thus the *joint scheduling of eMBB and URLLC traffic over two time-scales.* At the slot boundary, resources are allocated to eMBB users based on their channel states and utilities, in effect, allocating long term rates to optimize high-level goals (e.g. utility optimization). Meanwhile, at each minislot boundary, the (stochastic) URLLC demands are overlapped (superposed/punctured) onto previously allocated eMBB transmissions. Decisions on the placement of such overlaps across scheduled eMBB user(s) will impact the rates they will see on that slot. Thus we have a coupled problem of jointly optimizing the scheduling of eMBB users on slots with the placement of URLLC demands across minislots.

A. Main Contributions

This paper is, to our knowledge, the first to formalize and solve the joint eMBB/URLLC scheduling problem described above. We consider various models for the eMBB rate loss associated with URLLC superposition/puncturing, for which we characterize the associated feasible throughput regions and online joint scheduling algorithms as detailed below.

(**Linear Model**): When the rate loss to eMBB is directly proportional to the fraction of superposed/punctured minislots, we show that the joint optimal scheduler has a nice decomposition: the stochastic URLLC traffic can be uniform-randomly scheduled in each minislot, and the eMBB scheduler can be scheduled via a greedy iterative gradient algorithm the only accounts for the expected rate loss due to the URLLC traffic.

(**Convex Model**): For more general models where the rate loss can be modeled through a convex function, we restrict to time homogeneous policies. In this setting, we characterize the capacity region and derive concavity conditions under which we can derive the effective rate seen by eMBB users (post-puncturing by URLLC traffic). We then develop a stochastic approximation algorithm jointly schedules eMBB and URLLC traffic, and show that it asymptotically maximizes utility for eMBB users while satisfying URLLC demands.

(**Threshold Model**): We finally consider a threshold model, where eMBB traffic is unaffected by puncturing until a threshold; beyond this threshold it suffers complete throughput loss (a 0-1 rate loss model). We consider two broad classes of time homogeneous policies, where the URLLC traffic is placed in minislots proportional to either the eMBB allocated bandwidths (Rate Proportional) or the eMBB thresholds (Threshold Proportional). We motivate these policies (e.g. minimizes probability of eMBB loss in any slot) and derive the associated throughput regions. Finally, we utilize the additional structure imposed by the RP and TP Placement policies along with the shape of the threshold loss function and derive fast gradient algorithms that converge and provably maximize utility.

B. Related Work

Resource allocation, utility maximization and opportunistic scheduling for downlink wireless systems has intensely studied for the last two decades, and has had a major impact on cellular standards. We refer to [3], [4] for a survey of the key results. In this paper, we focus on joint scheduling of URLLC and eMBB traffic. From an application point of view, there have been several studies arguing for the need for URLLC services (e.g. for industrial automation) [5], [6], [7].

With demand of both broadband and low-latency services growing, there has been rapid developments in the 5G standardization efforts in 3GPP. Of key relevance to this paper, the 3GPP RAN WG1 has focussed on standardizing slot structure for eMBB and URLLC, and have been evaluating signaling and control channels to support superposition and puncturing in recent meetings [1], [2]. We specifically refer the reader to Sections 8.1.1.3.4 – 8.1.1.3.6 in [2] for current proposals.

Beyond standards, recent work has focussed on system level design for such systems (overheads, packet sizes, control channel structure, etc.) [8], [9], [10]. Of particular note, [9] argues (based on system level simulation and queueing models) that statically partitioning bandwidth between eMBB and URLLC is very inefficient. There have also been several studies focussing on the physical layer aspects of URLLC (coding and modulation, fading, link budget) [11], [12]. However, to the best of our knowledge, our paper is the first to explore the resource allocation issues for joint scheduling of URLLC and eMBB traffic.

II. System Model

Traffic model. We consider a wireless system supporting a fixed set of backlogged eMBB users $\mathcal{U}$ and stationary URLLC traffic demands. eMBB scheduling decisions are made across slots while URLLC demands arrive and are immediately scheduled across minislots. Each eMBB slot has an associated set of minislots where $\mathcal{M} = \{1, \dots |\mathcal{M}|\}$ denotes there indices. URLLC demands across minislots are modeled as a independent and identically distributed (i.i.d.) random random process. We let the random variables $(D(m), m \in \mathcal{M})$ denote the URLLC demands per minislot for a typical eMBB slot. We let D be a random variable whose distribution is that of the aggregate URLLC demand per eMBB slot, i.e., $D \sim \sum_{m \in \mathcal{M}} D(m)$ with, cumulative distribution function $F_D()$ and mean $E[D] = \rho$. We assume demands have been normalized so the maximum URLLC demand per minislot is f and the maximum aggregate demands per eMBB slot is $f \times |\mathcal{M}| = 1$ i.e., all the frequency-time resources are occupied. URLLC demands per minislot exceeding the system capacity are blocked by URLLC scheduler thus $D \leq 1$ almost surely. As mentioned earlier the system is engineered so that blocked URLLC traffic on a minislot is a rare event, i.e., satisfies the desired reliability on such traffic.

Wireless channel variations. The wireless system experiences channel variations each eMBB slot which are modeled as an i.i.d. random process over set of channel states $\mathcal{S} = \{1, \dots, |\mathcal{S}|\}$. Let S be a random variable modeling the distribution over the states in a typical eMBB slot with probability mass function $p_S(s) = P(S = s)$ for $s \in \mathcal{S}$. For each channel state s eMBB user u has a known peak capacity $\hat{r}_u^s$. The wireless system can choose what proportions of the frequency-time resources to allocate to each eMBB user on

each minislot for each channel state. This is modeled by a matrix $\boldsymbol{\phi} \in \Sigma$ where

$$\Sigma := \Big\{ \mathbf{x} \in \mathbb{R}_+^{|\mathcal{U}|\times|\mathcal{M}|\times|\mathcal{S}|} \mid \sum_{u\in\mathcal{U}} x^s_{u,m} = f, \forall m \in \mathcal{M}, s \in \mathcal{S} \Big\} \quad (1)$$

and where the element $\phi^s_{u,m}$ represents the fraction of resources allocated to user u in mini slot m in channel state s. We also let $\phi^s_u = \sum_{m\in\mathcal{M}} \phi^s_{u,m}$, i.e., the total resources allocated to user u in an eMBB slot in channel state s. Now assuming no superposition/puncturing if the system is in channel state s and the eMBB scheduler chooses an allocation $\boldsymbol{\phi}$ the rate r_u allocated to user u would be given by $r_u = \phi^s_u \hat{r}^s_u$. The scheduler is assumed to know the channel state and can thus exploit such variations opportunistically in allocating resources to eMBB users. Note that for simplicity, we adopt a flat-fading model, namely, the rate achieved by an user is directly proportional to the fraction of bandwidth allocated to it (the scaling factor is the peak rate of the user for the current channel state).

Class of joint eMBB/URLLC schedulers. We consider a class of stationary joint eMBB/URLLC schedulers denoted by Π satisfying the following properties. A scheduling policy combines a possibly state dependent eMBB *resource allocation* $\boldsymbol{\phi}$ per slot with a URLLC *demand placement* strategy across minislots. The placement strategy may impact the eMBB users' rates since it affects the URLLC superposition/puncturing loads they will experience. As mentioned earlier in discussing the traffic model, in order to meet low latency requirements URLLC traffic demands are scheduled immediately upon arrival or blocked. The scheduler is assumed to be *causal* so it only knows the current (and past) channel states and achieved rates $\hat{r}^s_u, \forall, u \in \mathcal{U}, s \in \mathcal{S}$ but does not know the realization of future channels or URLLC traffic demands. In making superposition/puncturing decisions across minislots, the scheduler can use knowledge of the previous placement decisions that were made. In addition the scheduler is assumed to know (or can measure over time) the channel state distribution across eMBB slots and URLLC demand distributions per minislot i.e., that of $D(m)$, and per eMBB slot, i.e., D, and thus knows in particular $\rho = E[D]$.

In summary joint scheduling policy $\pi \in \Pi$ is thus characterized by the following:

- an eMBB resource allocation $\boldsymbol{\phi}^\pi \in \Sigma$ where $\phi^{\pi,s}_{u,m}$ denotes the fraction frequency-time slot resources allocated to eMBB user u on minislot m when the system is in state s.
- the distributions of URLLC loads across eMBB resources induced by its URLLC placement strategy, denoted by random variables $\mathbf{L}^\pi = (L^{\pi,s}_{u,m} | u \in \mathcal{U}, m \in \mathcal{M}, s \in \mathcal{S})$ where $L^{\pi,s}_{u,m}$ denotes the URLLC load superposed/puncturing the resource allocation of user u on minislot m when the channel is in state s.

The distributions of $L^{\pi,s}_{u,m}$ and their associated means $l^{\pi,s}_{u,m}$ depend on the joint scheduling policy π, but for all states, users and minislots satisfy

$$L^{\pi,s}_{u,m} \le \phi^{\pi,s}_{u,m} \quad \text{almost surely.}$$

In the sequel we let $L^{\pi,s}_u = \sum_{m\in\mathcal{M}} L^{\pi,s}_{u,m}$, i.e., the aggregate URLLC traffic superposed/puncturing user u in channel state s, and denote its mean by $l^{\pi,s}_u$ and note that

$$L^{\pi,s}_u \le \phi^{\pi,s}_u \quad \text{almost surely.}$$

We shall also $L^{\pi,s} = \sum_{u\in\mathcal{U}} L^{\pi,s}_u$ denote the aggregate induced load and note that any policy π and any state s we have that

$$\rho = \mathrm{E}[D] = \mathrm{E}[L^\pi] = \mathrm{E}[\sum_{u\in\mathcal{U}} L^{\pi,s}_u] = \sum_{u\in\mathcal{U}} l^{\pi,s}_u.$$

Modeling superposition/puncturing and eMBB capacity regions. Under a joint scheduling policy π we model the rate achieved by an eMBB user u in channel state s by a random variable

$$R^{\pi,s}_u \quad = \quad f^s_u(\phi^{\pi,s}_u, L^{\pi,s}_u) \quad (2)$$

where the *rate allocation function* $f^s_u(\cdot,\cdot)$ models the impact of URLLC superposition/puncturing – one would expect it to be increasing the first argument (the allocated resources) and decreasing in the second argument (the amount superposition/puncturing by URLLC traffic). One would also expect such functions to satisfy

$$f^s_u(\phi^s_u, l^s_u) = 0$$

if $\phi^s_u = l^s_u$, i.e., if superposition/puncturing occurs across all of an eMBB users resources no data is successfully transmitted, however, perhaps under the superposition some rate might still be extracted from the transmission. Also under our system model we have that

$$R^{\pi,s}_u \le f^s_u(\phi^{\pi,s}_u, 0) = \phi^{\pi,s}_u \hat{r}^s_u \text{ almost surely,}$$

with equality if there is no superposition/puncturing, i.e., when $l^s_u = 0$. We shall $r^{\pi,s}_u = E[R^{\pi,s}_u]$ denote the mean rates achieved by user u in state s under the URLLC superposition/puncturing distribution induced by scheduling policy π.

Models for Throughput Loss: In the sequel we shall consider specific forms of superposition/puncturing models: *(i)* linear, *(ii)* convex, and *(iii)* threshold models.

We rewrite the rate allocation function in (2) as the difference between the peak throughput and the loss due to URLLC traffic, and consider functions that can be decomposed as:

$$f^s_u(\phi^s_u, l^s_u) = \hat{r}^s_u \phi^s_u \left(1 - h^s_u\left(\frac{l^s_u}{\phi^s_u}\right)\right),$$

where $h^s_u : [0,1] \to [0,1]$ is the *rate loss function* and captures the relative rate loss due to URLLC overlap on eMBB allocations. The puncturing models we study now map directly to structural assumptions on the rate loss function $h^s_u(\cdot)$; namely it is a non-decreasing function, and is one of *linear, convex, or threshold* as shown in Figure 2.

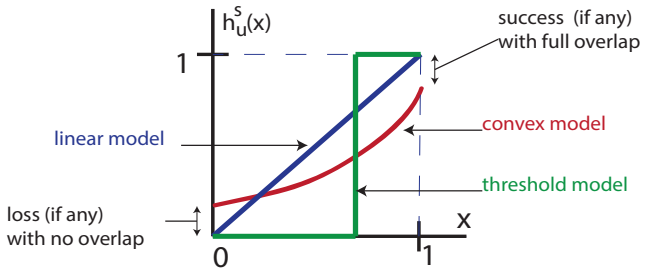

Fig. 2. The illustration exhibits the rate loss function for the various models considered in this paper, linear, convex and threshold.

Linear Model: Under the linear model, the expected rate for user u in channel state s for policy π is given by

$$r_u^{\pi,s} = \mathrm{E}[f_u^s(\phi_u^{\pi,s}, L_u^{\pi,s})] = \hat{r}_u^s(\phi_u^{\pi,s} - l_u^{\pi,s}),$$

i.e., $h_u^s(x) = x$, and the resulting rate to eMBB users is a linear function of both the allocated resources and mean induced URLLC loads. This model is motivated by basic results for the channel capacity of AWGN channel with erasures, see [13] for more details. Our system in a given network state can be approximated as an AWGN channel with erasures, when the slot sizes are long enough so that the physical layer error control coding of eMBB users use long code-words. Further, there is a dedicated control channel through which the scheduler can signal to the eMBB receiver indicating the positions of URLLC overlap. Indeed such a control channel has been proposed in the 3GPP standards [1]. Note that under this model the rate achieved by a given user depends on the aggregate superposition/puncturing it experiences, i.e., does not depend on which minislots and frequency bands it occurs. We discuss the policies for the linear model in Section III.

Convex Model: In the convex model, the rate loss function $h_u^s(\cdot)$ is convex (see Figure 2), and the resulting rate for eMBB user u in channel state s under policy π is given by

$$r_u^{\pi,s} = \mathrm{E}[f_u^s(\phi_u^{\pi,s}, L_u^{\pi,s})] = \hat{r}_u^s \phi_{\pi,s}^u \left(1 - E\left[h_u^s\left(\frac{L_u^{\pi,s}}{\phi_u^{\pi,s}}\right)\right]\right).$$

This covers a broad class of models, and is discussed in Section IV.

Threshold Model: Finally the threshold model is designed to capture a simplified packet transmission and decoding process in an eMBB receiver. The data is either received perfectly or it is lost depending on the amount of superposition/puncturing. With slight abuse of notation we shall let h_u^s also depend on both the relative URLLC load and the eMBB user allocation, i.e., $h_u^s(x, \phi_u^s) = \mathbf{1}(x \le t_u^s(\phi_u^s))$ where the threshold in turn is an increasing function $t_u^s()$ satisfying and satisfy $x \ge t_u^s(x) \ge 0$. Such thresholds might reflect various engineering choices where codes are adapted when users are allocated more resources, so as to be more robust to interference/URLLC superposition/puncturing. The resulting rate for eMBB user u in channel state s and policy π is then given by

$$r_u^{\pi,s} = \hat{r}_u^s \phi_u^{\pi,s} P(L_{\pi,u}^s \le \phi_u^{\pi,s} t_u^s(\phi_u^{\pi,s})).$$

While such a sharp falloff is somewhat extreme, it is nevertheless useful for modeling short codes that are designed to tolerate a limited amount of interference. In practice one might expect a smoother fall off, perhaps more akin to the convex model, e.g., when hybrid ARQ (HARQ) is used. We discuss polices under the threshold based model in Section V.

Capacity for eMBB traffic: We define the capacity $\mathcal{C} \subset \mathbb{R}_+^{|\mathcal{U}|}$ for eMBB traffic as the set of long term rates achievable under policies in Π. Let $\mathbf{c}^\pi = (c_u^\pi | u \in \mathcal{U})$ where

$$c_u^\pi = \sum_{s \in \mathcal{S}} r_u^{\pi,s} p_S(s).$$

Then the capacity is given by

$$\mathcal{C} = \{\mathbf{c} \in \mathbb{R}_+^{|\mathcal{U}|} \mid \exists\ \pi \in \Pi \text{ such that } \mathbf{c} \le \mathbf{c}^\pi\}.$$

Note that this capacity region depends on the scheduling policies under consideration as well as the distributions of the channel states and URLLC demands.

Scheduling objective: URLLC priority and eMBB utility maximization: As mentioned earlier, URLLC traffic is immediately placed upon arrival, at the minislot scale, i.e, no queueing is allowed. Thus if demands exceed the system capacity on a given minislot such traffic is lost. The system is engineered so that such URLLC overloads are extremely rare, and thus URLLC traffic can meet extremely low latency requirements with high reliability. For eMBB traffic we adopt a utility maximization framework wherein each eMBB user u has an associated utility function $U_u(\cdot)$ which is a strictly concave, continuous and differentiable of the average rate c_u^π experienced by the user. Our aim is to characterize optimal rate allocations associated with the utility maximization problem:

$$\max_{\mathbf{c}} \{\sum_{u \in \mathcal{U}} U_u(c_u) \mid \mathbf{c} \in \mathcal{C}\}, \tag{3}$$

and determine and associated scheduling policy π that will realize such allocations.

III. Linear Model for Superposition/Puncturing

As a thought experiment, consider a two-user system, with users having the same utility function (say square root function), but i.i.d. (across time and users) channel states. Suppose that a naive eMBB scheduler ignores channel states and statically partitions the bandwidth between these users (symmetry implies half the bandwidth to each user). In this case, it is clear that an optimal URLLC scheduler needs to be both channel-state and eMBB aware – at each minislot, depending on the instantaneous demand and the channel states, it needs to puncture the two users' shares of bandwidths differently. For instance at a certain minislot, if one user has a really poor channel state, then the URLLC traffic in that minislot would be mostly loaded onto the frequency resources occupied by this user (as the total rate loss to eMBB traffic will be minimal).

In this section, we show a surprising result – if the eMBB scheduler is intelligent, then the URLLC scheduler can be *oblivious to the channel states, utility functions and the actual rate allocations of the eMBB scheduler.*

A. Characterization of capacity region

Let us consider the capacity region for a wireless system based on linear superposition/puncturing model under a restricted class of policies Π^{LR} that combine feasible eMBB allocations $\boldsymbol{\phi} \in \Sigma$ with random placement of URLLC demands across minislots. For any $\pi \in \Pi^{LR}$ with eMBB allocation $\boldsymbol{\phi}^\pi$ the mean induced loads for such randomization for each state $s \in \mathcal{S}$ and minislot $m \in \mathcal{M}$ will satisfy $l^{\pi,s}_{u,m} = \rho\phi^{\pi,s}_{u,m}$. Indeed randomization clearly leads to an induced loads that are proportional to the eMBB allocations on a per mini-slot basis, but also per eMBB slot, i.e., $l^{\pi,s}_u = \rho\phi^{\pi,s}_u$. Thus for our linear superposition/puncturing model we have that

$$r^{\pi,s}_u = \hat{r}^s_u(\phi^{\pi,s}_u - l^{\pi,s}_u) = \hat{r}^s_u\phi^{\pi,s}_u(1-\rho).$$

Hence the overall user rates achieved under such a policy are given by $\mathbf{c}^\pi = (c^\pi_u | u \in \mathcal{U})$ where

$$c^\pi_u = \sum_{s\in\mathcal{S}} \hat{r}^s_u\phi^{\pi,s}_u(1-\rho)p_S(s).$$

The capacity region associated with policies that use URLLC randomization is thus given by

$$\begin{aligned}\mathcal{C}^{LR} &= \{\mathbf{c} \in \mathbb{R}^{|\mathcal{U}|}_+ \mid \exists \pi \in \Pi^{LR} \text{ s.t. } \mathbf{c} \le \mathbf{c}^\pi\} \\ &= \{\mathbf{c} \in \mathbb{R}^{|\mathcal{U}|}_+ \mid \exists \boldsymbol{\phi} \in \Sigma \text{ s.t. } \mathbf{c} \le \mathbf{c}^{\boldsymbol{\phi}}\},\end{aligned}$$

where we have used abused notation by using $\mathbf{c}^{\boldsymbol{\phi}}$ to represent the throughput achieved by a policy π that uses eMBB resource allocation $\boldsymbol{\phi}$ and randomized URLLC demand placement. Finally note that for any fixed $\rho \in (0,1)$, $\mathcal{C}^{LR}$ is a closed and bounded convex region. This is because an affine map of a convex region remains convex; hence multiplying the constraints on the capacity region defined by $\boldsymbol{\phi}$ by a constant $(1-\rho)$ preserves convexity of the rate region.

Theorem 1. *For a wireless system under the linear superposition/puncturing model we have that $\mathcal{C} = \mathcal{C}^{LR}$.*

The proof is deferred to the Appendix A. In other words the throughput $\mathbf{c}^\pi \in \mathcal{C}$ achieved by any feasible policy $\pi \in \Pi$ can also be achieved by policy π', with a possibly different eMBB resource allocation policy than π but utilizing random placement of URLLC demands across mini-slots.

B. Utility maximizing joint scheduling

Given the result in Theorem 1 we now restate the utility maximization problem as optimizing solely over joint scheduling policies that use URLLC random placement policies, as below.

$$\begin{aligned}\max_{\boldsymbol{\phi}\in\Sigma} \quad & \sum_{u\in\mathcal{U}} U_u(c^{\boldsymbol{\phi}}_u) \\ \text{s.t.} \quad & c^{\boldsymbol{\phi}}_u = \sum_{s\in\mathcal{S}} \hat{r}^s_u\phi^s_u(1-\rho)p_S(s), \quad \forall u \in \mathcal{U}.\end{aligned}$$

The above optimization problem has a strictly concave cost function, and convex constraints. Thus, at face-value, it appears that we can immediately apply the gradient scheduler introduced in [15], which is an online algorithm that converges and solves the optimization problem. This intuition is approximately correct, but subject to two modifications.

First, the setting in [15] has deterministic rates in each channel state. However, in our case, in each channel state, the rates are stochastic due to i.i.d. puncturing due to URLLC traffic (which accounts for the $(1-\rho)$ correction). This can be easily addressed by modifying the setting in [15]; the finite state and i.i.d. nature of puncturing implies that the proofs in [15] hold with minor modifications; we skip the details.

The second issue is somewhat more nuanced. In current wireless systems (e.g. LTE) and proposals for 5G systems, a slot is partitioned into a collection of Resource Blocks (RB), where each RB is a time-frequency rectangle (1 msec $\times$ 180 KHz in LTE). Importantly, these RBs can be individually allocated to different eMBB users. If we now apply the gradient scheduler in [15] to our setting, the result will be that all RBs in a slot will be allocated to the same user. While this is no-doubt asymptotically optimal, it seems intuitive that sharing RBs across users even within a slot will lead to better short-term performance. Indeed this intuition has been explored in the context of iterative MaxWeight algorithms to provide formal guarantees, see [16], [17]. The high level idea is that even within a slot, RB allocations are iterative, where future RB allocation need to account for prior rate allocations even within the same slot. This is formalized below, where we have fully described the joint eMBB-URLLC scheduler.

The URLLC scheduler: As explained in the previous section, the URLLC scheduler places the URLLC traffic uniformly at random over the minislots.

The eMBB scheduler: Let there be B resource blocks available for allocation every eMBB slot, indexed by $1, 2, \ldots, B$. Let $\overline{R}_u(t-1)$ be the random variable denoting the average rate received by eMBB user up to eMBB slot $t-1$. In any eMBB slot t we schedule an user $u(b)$ in RB b such that

$$u(b) \in \text{argmax}\left\{\hat{r}^s_u U_u'\left(\overline{r}^\epsilon_u(b-1,t)\right), u = 1,2,\ldots,\mathcal{U}\right\}, \quad (4)$$

where $\overline{r}^\epsilon_u(b-1,t)$ is an estimate of the average rate received by eMBB user u till slot t which is iteratively updated as follows:

$$\overline{r}^\epsilon_u(b,t) = \begin{cases}\overline{R}_u(t-1), & b = 0, \\ (1-\epsilon)\,\overline{r}^\epsilon_u(b-1,t) & \\ +\epsilon\left(\hat{r}^s_u\frac{1}{B}(1-\rho)\mathbb{1}(i=u(b))\right), & b \neq 0.\end{cases} \quad (5)$$

In the above equation, ϵ is a small positive value. At the end of eMBB slot t, the eMBB scheduler receives feedback from the eMBB receivers indicating the actual rates received by the eMBB users due to allocations through (5). We denote this rate received eMBB user u in slot by the random variable $R_u(t)$. We finally update $\overline{R}_u(t)$ as follows:

$$\overline{R}_u(t) = (1-\epsilon)\,\overline{R}_u(t-1) + \epsilon R_u(t). \quad (6)$$

This update is analogous to the gradient algorithm [15] (see also iterative algorithms in [16], [17]). The optimality proof

of this algorithm follows (with minor modifications) from the analysis in [15]; we skip the details.

Remarks: (i) A natural decomposition of the joint eMBB+URLLC scheduling is now apparent. On one hand, the eMBB scheduler maximizes utilities based on the *expected* channel rates stemming from uniformly random puncturing of minislots (accounted for through the $(1-\rho)$ multiplicative factor), and does so using the iterative gradient scheduler. The URLLC scheduler, on the other-hand, is completely agnostic to either the channel state or the actual eMBB allocations and simply punctures minislots based on the current instantaneous demand.

(ii) The fact that the URLLC traffic is completely agnostic to the channel state and eMBB utilities/allocation is surprising. Intuitively it seems plausible that one could load an eMBB user with a lower marginal utility with more URLLC traffic, while protecting a eMBB user with a higher marginal utility and achieve a better sum utility. Further, it seems reasonable that eMBB users with a worse channel state (and thus lower rate) could be loaded with additional URLLC traffic. However, Theorem. 1 implies that there exists an optimal solution that is achieved by channel and utility oblivious, uniform loading of URLLC traffic, thus providing a very simple algorithm for URLLC scheduling.

IV. Convex Model – Time-Homogenous Policies

In this section we shall consider joint scheduling for wireless systems for a general superposition/puncturing model. This is a somewhat complex problem, whence we will focus our attention on a restricted, but still rich, class of scheduling policies which we refer to as time-homogeneous eMBB/URLLC schedulers. We identify a key concavity requirement in Condition 1 (that is satisfied by convex loss functions) that enables a stochastic approximation approach for utility maximization.

A. Time-homogeneous eMBB/URLLC Scheduling policies

We shall define time-homogeneous eMBB/URLLC schedulers as follows. First, feasible eMBB allocations $\boldsymbol{\phi} \in \Sigma$ will be restricted such that for any eMBB slot in channel state $s \in \mathcal{S}$ allocations are *time-homogeneous* across minislots across the slot, i.e., $\phi^s_{u,1} = \phi^s_{u,m}, \forall m \in \mathcal{M}$ and its overall allocation for the slot is given by $\phi^s_u = |\mathcal{M}|\phi^s_{u,1}$. The set of time-homogeneous eMBB allocations is thus given by

$$\Sigma^U := \left\{ \mathbf{x} \in \Sigma \mid \forall s \in \mathcal{S}, u \in \mathcal{U},\ x^s_{u,m} = x^s_{u,1} \quad \forall m \in \mathcal{M} \right\}.$$

Second, URLLC demand placement per minislot are done proportionally to pre-specified weights, and these weights are assumed to be time-homogeneous across minislots. In particular such policies are parametrized by a weight matrix $\gamma \in \Sigma^U$, where induced load on user u under channel state s and slot m is given by

$$L^s_{u,m} = \frac{\gamma^s_{u,m}}{\sum_{u' \in \mathcal{U}} \gamma^s_{u',m}} D(m) = \frac{\gamma^s_{u,1}}{f} D(m).$$

The eMBB and URLLC allocations are however coupled together since it must be the case that for all $u \in \mathcal{U}$ $L^s_{u,m} \le \phi^s_{u,m} = \phi^s_{u,1}$ almost surely, i.e., one can not induce more superposition/puncturing on a user than the resources it has been allocated on that slot. so the following condition must be satisfied. Thus we must have that for all $u \in \mathcal{M}$

$$D(m) \le \min_{u \in \mathcal{U}} \frac{\phi^s_{u,1}}{\gamma^s_{u,1}} f.$$

Note we have assumed that $D(m) \le f$ almost surely, thus if $\frac{\phi^s_{u,1}}{\gamma^s_{u,1}} \ge 1$ this may not hold.

Assumption 1. *We say a system satisfies a $(1-\delta)$ URLLC sharing factor per minislot if $D(m) \le f(1-\delta)$ almost surely for all $m \in \mathcal{M}$.*

Under a $(1-\delta)$ URLLC demand backoff a time-homogeneous eMBB resource allocation $\boldsymbol{\phi}$ and URLLC allocation $\boldsymbol{\gamma}$ is will be feasible if for all $s \in \mathcal{S}$ we have

$$(1-\delta) \le \min_{u \in \mathcal{U}} \frac{\phi^s_{u,1}}{\gamma^s_{u,1}},$$

which is satisfied as long as $(1-\delta)\gamma^s_{u,1} \le \phi^s_{u,1}$ for all $u \in \mathcal{U}$. This motivates the following definition.

Definition 1. *Under a $(1-\delta)$ sharing factor, the feasible time-homogeneous eMBB/URLLC scheduling policies are parameterized by $\boldsymbol{\phi}, \boldsymbol{\gamma} \in \Sigma^U$ such that $(1-\delta)\boldsymbol{\gamma} \le \boldsymbol{\phi}$. We shall denote the set of such policies as follows:*

$$\Pi^{U,\delta} := \{(\boldsymbol{\phi}, \boldsymbol{\gamma}) \mid \boldsymbol{\phi}, \boldsymbol{\gamma} \in \Sigma^U \text{ and } (1-\delta)\boldsymbol{\gamma} \le \boldsymbol{\phi}\},$$

where $\Pi^{U,\delta}$ is a convex set.

B. Characterization of throughput region

In this section we characterize the throughput regions achievable under time-homogeneous scheduling.

Theorem 2. *Under a $(1-\delta)$ sharing factor and time-homogeneous scheduler $\boldsymbol{\pi} = (\boldsymbol{\phi}^{\boldsymbol{\pi}}, \boldsymbol{\gamma}^{\boldsymbol{\pi}}) \in \Pi^{U,\delta}$ the probability of induced throughput for user r $u \in \mathcal{U}$ in channel state $s \in \mathcal{S}$ is given by*

$$r^{\boldsymbol{\pi},s}_u = \mathrm{E}[f^s_u(\phi^{\boldsymbol{\pi},s}_u, \gamma^{\boldsymbol{\pi},s}_u D)],$$

and the overall user throughputs are given by $\mathbf{c}^{\boldsymbol{\pi}} = (c^{\boldsymbol{\pi}}_u : u \in \mathcal{U})$ where $c^{\boldsymbol{\pi}}_u = \sum_{u \in \mathcal{U}} r^{\boldsymbol{\pi},s}_u p_S(s)$.

The proof is available in Appendix B. Based on the above we can define feasible throughput region constrained to the time-homogeneous policies in $\Pi^{U,\delta}$. First let us define

$$\mathcal{C}^{U,\delta} \quad = \quad \{\mathbf{c} \in \mathbb{R}^{|\mathcal{U}|}_+ \mid \exists \boldsymbol{\pi} \in \Pi^{U,\delta} \text{ s.t. } \mathbf{c} \le \mathbf{c}^{\boldsymbol{\pi}}\}.$$

We shall let $\hat{\mathcal{C}}^{U,\delta}$ denote the convex hull of $\mathcal{C}^{U,\delta}$. Note that throughputs rates in the convex hull are achievable through policies that do time sharing/randomization amongst time-homogeneous scheduling policies in $\Pi^{U,\delta}$.

Condition 1. *For all $s \in \mathcal{S}$ and $u \in \mathcal{U}$ the functions $g^s_u(,)$ given by*

$$g^s_u(\phi^s_u, \gamma^s_u) = \mathrm{E}[f^s_u(\phi^s_u, \gamma^s_u D)], \tag{7}$$

are jointly concave on $\Pi^{U,\delta}$.

Lemma 1. *Condition 1 is satisfied for systems where superposition/puncturing of each user is modelled via either a*

1) *Convex loss function,*
2) *Threshold loss function with fixed relative thresholds, i.e.,* $t_u^s(\phi_u^s) = \alpha_u^s$ *for* $\phi \in [0,1]$ *and the URLLC demand distribution* $F_D(\cdot)$ *is such that* $F_D(\frac{1}{x})$ *is concave in* x *(satisfied by the truncated Pareto distribution).*

The proof is available in Appendix B. With this condition in place, we now describe the throughput region.

Theorem 3. *Suppose that Condition 1 holds. then* $\mathcal{C}^{U,\delta} = \hat{\mathcal{C}}^{U,\delta}$, *i.e., there is no need to consider time-sharing/randomization amongst time-homogeneous eMBB/URLLC policies.*

The proof is available in Appendix B. Thus, with time-homogeneous policies and imposing concavity of from Condition 1, the above result sets up a convex optimization problem in (ϕ, γ), i..e, we have a concave cost function with convex constraints. Thus, by iteratively updating (ϕ, γ), we can develop an online algorithm that asymptotically maximizes utility. Below, we formally develop a stochastic approximation algorithm to achieve this objective.

C. Stochastic approximation based online algorithm

We first restate the utility maximization problem for time-homogeneous URLLC/eMBB scheduling policies:

$$\max_{\boldsymbol{\phi},\boldsymbol{\gamma}\in\Pi^{U,\delta}} \quad \sum_{u\in\mathcal{U}} U_u\left(\sum_{s\in\mathcal{S}} p_{\mathcal{S}}(s) g_u^s\left(\phi_u^s, \gamma_u^s\right)\right). \tag{8}$$

Observe that the objective function consists of a sum of compositions of non-decreasing concave function ($U_u(\cdot)$), and supposing Condition 1 holds, a concave function $g_u^s(\cdot,\cdot)$ in $\boldsymbol{\phi}$ and $\boldsymbol{\gamma}$. Further, the constraint set is convex. Therefore, the above problem fits in the framework of standard convex optimization problems. However, solving the above problem requires the knowledge of all possible network states and its probability distribution, resulting in an *offline* optimization problem. In this section, we develop a stochastic approximation based online algorithm to solve the above problem.

Online algorithm: Let $\overline{R}_u(t-1)$ be the random variable denoting the average rate received by eMBB user up to eMBB slot $t-1$. Let s be the network state in slot t. Define vectors $\phi^s := \{\phi_u^s, \mid u \in \mathcal{U}\}$ and $\gamma^s := \{\gamma_u^s \mid u \in \mathcal{U}\}$. At the beginning of eMBB slot t, we compute the vectors $\left(\tilde{\phi}(t), \tilde{\gamma}(t)\right)$ as the solution to the following optimization problem.

$$\max_{\phi^s,\gamma^s} \quad \sum_{u\in\mathcal{U}} U_u^{'}\left(\overline{R}_u(t-1)\right) g_u^s(\phi_u^s, \gamma_u^s), \tag{9}$$

$$\text{s.t.} \quad \phi^s \geq (1-\delta)\,\gamma^s, \tag{10}$$

$$\sum_{u\in\mathcal{U}} \phi_u^s = 1 \text{ and } \sum_{u\in\mathcal{U}} \gamma_u^s = 1, \tag{11}$$

$$\phi^s \in [0,1]^{|\mathcal{U}|} \text{ and } \gamma^s \in [0,1]^{|\mathcal{U}|}. \tag{12}$$

This optimization problem is a convex optimization problem and can be solved numerically using standard convex optimization techniques. Using $\left(\tilde{\phi}(t), \tilde{\gamma}(t)\right)$, we schedule URLLC and eMBB traffic as follows:

The eMBB scheduler: For notational ease, we fluidize the bandwidth. Specifically, we assume that the bandwidth of a resource block is very small when compared to the total bandwidth available. Hence, the bandwidth can be split into arbitrary fractions and we allocate $\tilde{\phi}_u(t)$ fraction of the total bandwidth to eMBB user u.

The URLLC Scheduler: We load different eMBB users with URLLC traffic according to the vector $\tilde{\gamma}(t)$.

At the end of eMBB slot t, the eMBB scheduler receives feedback from the eMBB receivers indicating the rates received by the eMBB users. Let us denote the rate received eMBB user u in slot by the random variable $R_u(t)$. We update $\overline{R}_u(t)$ as follows:

$$\overline{R}_u(t) = (1-\epsilon_t)\,\overline{R}_u(t-1) + \epsilon_t R_u(t), \tag{13}$$

where $\{\epsilon_t \mid t = 1,2,3,\ldots\}$ is a sequence of positive numbers which satisfy the following (standard) condition:

Condition 2. *The averaging sequence* $\{\epsilon_t\}$ *satisfies:*

$$\sum_{t=1}^{\infty} \epsilon_t = \infty \quad \textit{and} \quad \sum_{t=1}^{\infty} \epsilon_t^2 < \infty$$

.

Finally, we state the main result of this section, which is the optimality of the stochastic approximation based online algorithm.

Theorem 4. *Let* $\boldsymbol{r}^*$ *be the optimal average rate vector received by eMBB users under the solution to the offline optimization problem. Suppose that Conditions 1 and 2 hold. Then we have that:*

$$\lim_{t\to\infty} \overline{\boldsymbol{R}}(t) = \boldsymbol{r}^* \quad \textit{almost surely}. \tag{14}$$

The proof is available in the Appendix B.

V. Threshold Model and Placement Policies

In the previous section, we developed a stochastic approximation algorithm for time-homogeneous policies. This algorithm iteratively solves an optimization problem described in (9). This optimization problem jointly optimizes over a pair of row vectors (ϕ^s, γ^s). While this convex optimization problem can be solved using standard methods, it could become computationally challenging as the number of users scale up.

In this section, we shall restrict our attention to a threshold model for superposition/puncturing, and look at policies that impose structural conditions on the puncturing matrix $\boldsymbol{\gamma}$. We will show that the resulting class of policies have nice theoretical properties that lead to simpler online algorithms (solving (4), which is an one-dimensional search).

We consider two types of structural conditions on the puncturing matrix $\boldsymbol{\gamma}$, resource proportional and threshold proportional placement policies, described below.

(i) Resource Proportional (RP) Placement: The first is based on allocating URLLC demands in proportion to eMBB user slot allocations, i.e., $\gamma_u^s = \phi_u^s$. We refer to this as Resource Proportional (RP) Placement and denote such policies by

$$\Pi^{RP,\delta} := \{(\boldsymbol{\phi}, \boldsymbol{\gamma}) \in \Pi^{U,\delta} \mid \boldsymbol{\gamma} = \boldsymbol{\phi}\},$$

and define the associated achievable throughput region

$$\mathcal{C}^{RP,\delta} = \{\mathbf{c} \in \mathbb{R}_+^{|\mathcal{U}|} \mid \exists \boldsymbol{\pi} \in \Pi^{RP,\delta} \text{ s.t. } \mathbf{c} \leq \mathbf{c}^{\boldsymbol{\pi}}\}.$$

The motivation for RP Placement comes from the optimality of random placement for the linear model in Section III. Observe that if puncturing occurs uniformly randomly, then the expected number of punctures is directly proportional to the fraction of bandwidth allocated to an eMBB user. Thus, RP Placement has the interpretation of a *determinized version* of the policy we previously studied with linear loss functions.

(ii) Threshold Proportional (TP) Placement: The second policy allocates URLLC demands in proportion to the eMBB users associated loss thresholds so as to avoid losses,

$$\gamma_u^s = \frac{\phi_u^s t_u^s(\phi_u^s)}{\sum_{u' \in \mathcal{U}} \phi_{u'}^s t_{u'}^s(\phi_{u'}^s)}.$$

We refer to this as Threshold Proportional (TP) Placement and denote such policies by

$$\Pi^{TP,\delta} := \{(\boldsymbol{\phi}, \boldsymbol{\gamma}) \in \Pi^{U,\delta} \mid \gamma_u^s = \frac{\phi_u^s t_u^s(\phi_u^s)}{\sum_{u' \in \mathcal{U}} \phi_{u'}^s t_{u'}^s(\phi_{u'}^s)} \forall s \in \mathcal{S}, u \in \mathcal{U}\}.$$

The associated achievable throughput region is denoted

$$\mathcal{C}^{TP,\delta} = \{\mathbf{c} \in \mathbb{R}_+^{|\mathcal{U}|} \mid \exists \boldsymbol{\pi} \in \Pi^{TP,\delta} \text{ s.t. } \mathbf{c} \leq \mathbf{c}^{\boldsymbol{\pi}}\}.$$

The following theorem provides a formal motivation for TP Placement,. The main takeaway here is that the *probability of any loss in an eMBB slot under TP Placement policy is a lower bound over all other strategies.*

Theorem 5. *Consider a system with $(1-\delta)$ sharing factor. Consider a joint scheduling policy based on the TP URLLC placement i.e, $\boldsymbol{\pi} = (\boldsymbol{\phi}^{\boldsymbol{\pi}}, \boldsymbol{\gamma}^{\boldsymbol{\pi}}) \in \Pi^{TP,\delta}$. Then $\boldsymbol{\pi}$ achieves the minimum probability of eMBB loss amongst all joint scheduling policies using the same eMBB resource allocation $\boldsymbol{\phi}^{\boldsymbol{\pi}}$.*

The proofs (along with characterizations of the capacity region for RP and TP Placement policies) are available in Appendix C.

A. Online scheduling for RP and TP Placement

In this section, we consider online algorithms that implement the RP and TP Placement policies. While the stochastic approximation algorithm developed in Section IV-C can clearly be used, the additional structure imposed by the RP and TP Placement policies, and the shape of the threshold loss function (discussed below) can result in much simpler algorithms (with optimality guarantees).

We consider the case where $t_u^s(\phi)$ is a (state dependent but ϕ independent) constant, i.e., $t_u^s(\phi) = \alpha^s$, where $\alpha^s \in (0,1)$.

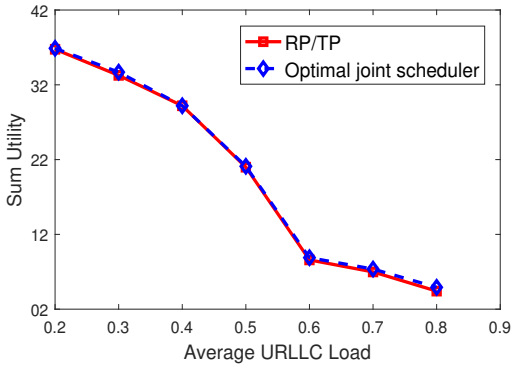

Fig. 3. Sum utility as a function of URLLC load ρ for the optimal and TP Placement policies under threshold model ($\delta = 0.1$).

Intuitively, this means that eMBB traffic which has a higher share of the bandwidth is more resilient to losses (e.g. through coding over larger fraction of resources). Then, by substituting this loss function in (25) and (28) (where we also use the fact that $\sum_{u \in \mathcal{U}} \phi_u^s = 1$), we have that

$$r_u^{\boldsymbol{\pi},s} = \hat{r}_u^s \phi_u^s F_D(\alpha^s).$$

Comparing with the development in Section III-B, we observe that the cost and constraints are identical if $F_D(\alpha^s)$ replaces $(1-\rho)$. Note that a small difference is that $F_D(\alpha^s)$ is state and user dependent, whereas $(1-\rho)$ does not depend on either; however, it is easy to see that the development in Section III-B immediately generalizes to this setting. Hence, we can interpret $F_D(\alpha^s)$ as the state and user dependent average rate loss due to puncturing via the RP or TP Placement policies.

We can now employ the rate-based iterative gradient scheduler developed in Section III-B (by replacing $(1-\rho)$ in (5) by a user-dependent $F_D(\alpha^s)$), and the theoretical guarantees directly carry over. As this algorithm only minimizes over users at each slot in (4), this is easier to implement when compared to the stochastic approximation algorithm developed in Section IV-C.

VI. Simulations

We consider a system with a total of 100 RBs available per eMBB slot, with 8 minislots per eMBB slot. In an eMBB slot, $\hat{r}_u^s$ for an eMBB user is drawn from the finite set $\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$ Mbps with equal probability and i.i.d. across users and slots. Our system consists of 20 users, and with 100 channel states (all equally likely). The (20 users $\times$ 100 states) rate matrix is one-time synthesized by independently and uniformly sampling a rate from the finite rate set for each matrix element.

We first consider a threshold model with $\alpha^s = 0.3$ for 50% of eMBB states and $\alpha^s = 0.7$ for the rest. We use the utility function $U_u(r) = \log(r) + 6.5$ for all eMBB users, where r is measured in Mbps (constant added to ensure non-negativity of the sum utility). URLLC load in an eMBB slot (D) is generated form the truncated Pareto distribution with

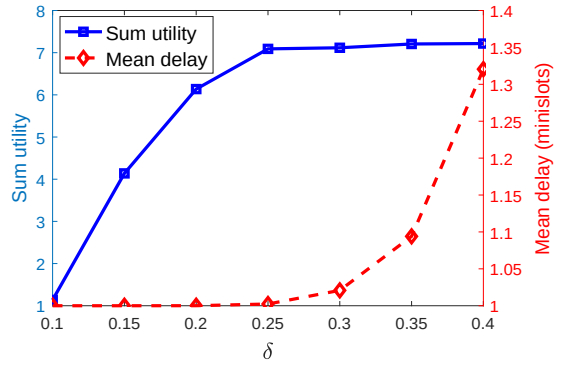

Fig. 4. Sum utility and mean URLLC delay as a function of δ.

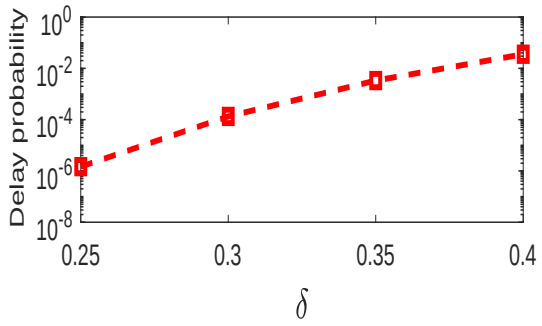

Fig. 5. Log-scale plot of the probability that URLLC traffic is delayed by more than two minislots (0.25 msec) for various values of δ.

tail exponent $\eta = 2$. We compare the optimal policy (stochastic approximation algorithm, see Section IV-C) with that from the TP Placement policy (the simpler gradient algorithm in Section V-A). In this case, as the threshold functions are (state-dependent) constants, the RP and TP Placement policies are the same. As we can see in Figure 3, the TP Placement policy tracks the optimal policy very well.

In Figure 5, we study the trade-off between achieving a higher eMBB utility and lowering the mean delay of URLLC traffic for different values of the sharing factor $1-\delta$. Figure 5 plots the corresponding probability that the URLLC traffic delay exceeds two minislots ($0.125 \times 2 = 0.25$ msec). To study this trade-off we generate URLLC arrivals in each minislot from an uniform distribution between $[0, 1/8]$ (recall there are 8 minislots). In each minislot, we can serve at most $\frac{1-\delta}{8}$ units of URLLC traffic. If the URLLC load in a given minislot is more than $\frac{1-\delta}{8}$, the remaining URLLC traffic is queued and served in the next minislot on a FCFS basis. For the eMBB users we use a convex model with $h_u^s(s) = e^{\kappa_u(x-1)}$ where κ_u determines the sensitivity of an eMBB user to an URLLC load. We have chosen $\kappa = 0.2$ for 50 % of the users and $\kappa = 0.7$ for the rest. We also set $\forall u \; U_u(x) = \log(x) + 4.2$ (constant added to ensure positive sum utility). In summary, a larger value of δ limits the amount of URLLC traffic than can be served in a minislot. However, a larger δ enlarges the constraint set $\Pi^{U,\delta}$ in the eMBB utility maximization problem, and hence we get higher eMBB utility.

VII. Conclusion

In this paper, we have developed a framework and algorithms for joint scheduling of URLLC (low latency) and eMBB (broadband) traffic in emerging 5G systems. Our setting considers recent proposals where URLLC traffic is dynamically multiplexed through puncturing/superposition of eMBB traffic. Our results show that this joint problem has structural properties that enable clean decompositions, and corresponding algorithms with theoretical guarantees.

Acknowledgements

The work of Arjun Anand was supported by FutureWei Technologies, Gustavo de Veciana was partially supported by NSF grant CNS-1343383 and FutureWei Technologies, and Sanjay Shakkottai was partially supported by NSF grant CNS-1343383 and the US DoT D-STOP Tier 1 University Transportation Center.

References

[1] 3GPP TSG RAN WG1 Meeting 87, November 2016.
[2] Chairman's notes 3GPP: 3GPP TSG RAN WG1 Meeting 88bis, Available at http://www.3gpp.org/ftp/TSG_RAN/WG1_RL1/TSGR1_88b/Report/, April 2017.
[3] R. Srikant and L. Ying, *Communication Networks: An Optimization, Control, and Stochastic Networks Perspective*. Cambridge University Press, 2014.
[4] L. Georgiadis, M. J. Neely, and L. Tassiulas, "Resource allocation and cross-layer control in wireless networks," *Foundations and Trends in Networking*, vol. 1, no. 1, 2006.
[5] 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 Communications Magazine*, vol. 54, no. 6, pp. 36–43, June 2016.
[6] O. N. C. Yilmaz, Y. P. E. Wang, N. A. Johansson, N. Brahmi, S. A. Ashraf, and J. Sachs, "Analysis of ultra-reliable and low-latency 5G communication for a factory automation use case," in *2015 IEEE International Conference on Communication Workshop (ICCW)*, June 2015, pp. 1190–1195.
[7] M. Gidlund, T. Lennvall, and J. Akerberg, "Will 5G become yet another wireless technology for industrial automation?" in *2017 IEEE International Conference on Industrial Technology (ICIT)*, March 2017, pp. 1319–1324.
[8] K. I. Pedersen, G. Berardinelli, F. Frederiksen, P. Mogensen, and A. Szufarska, "A flexible 5G frame structure design for frequency-division duplex cases," *IEEE Communications Magazine*, vol. 54, no. 3, pp. 53–59, March 2016.
[9] C.-P. Li, J. Jiang, W. Chen, T. Ji, and J. Smee, "5G ultra-reliable and low-latency systems design," in *2017 European Conference on Networks and Communications (EuCNC)*, June 2017, pp. 1–5.
[10] G. Durisi, T. Koch, and P. Popovski, "Toward massive, ultrareliable, and low-latency wireless communication with short packets," *Proceedings of the IEEE*, vol. 104, no. 9, pp. 1711–1726, Sept 2016.
[11] G. Durisi, T. Koch, J. Ostman, Y. Polyanskiy, and W. Yang, "Short-packet communications over multiple-antenna rayleigh-fading channels," *IEEE Trans. on Comm.*, vol. 64, no. 2, pp. 618–629, Feb 2016.
[12] B. Singh, Z. Li, O. Tirkkonen, M. A. Uusitalo, and P. Mogensen, "Ultra-reliable communication in a factory environment for 5G wireless networks: Link level and deployment study," in *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Sept 2016, pp. 1–5.
[13] D. Julian, "Erasure networks," in *Proceedings IEEE International Symposium on Information Theory,*, Jul. 2002.
[14] Anonymized extended draft with Appendix available at: https://1drv.ms/b/s!AhCjqMQUmYUiaydaM3WtBJlLaYE.

[15] A. L. Stolyar, "On the asymptotic optimality of the gradient scheduling algorithm for multiuser throughput allocation," *Operations Research*, vol. 53, no. 1, pp. 12–25, 2005.
[16] S. Bodas, S. Shakkottai, L. Ying, and R. Srikant, "Low-complexity scheduling algorithms for multi-channel downlink wireless networks," in *Proceedings of IEEE Infocom*, 2010.
[17] ——, "Scheduling for small delay in multi-rate multi-channel wireless networks," in *Proceedings of IEEE Infocom*, 2011.
[18] S. Boyd and L. Vandenberge, *Convex Optimization*. Cambridge University Press, 2003.
[19] H. J. Kushner and G. G. Yin, *Stochastic Approximation Algorithms and Applications*. Springer, 1997.

APPENDIX

A. Proofs from Section III

Theorem 1. *For a wireless system under the linear superposition/puncturing model we have that $\mathcal{C} = \mathcal{C}^{LR}$.*

Proof. Clearly since $\Pi^{LR} \subset \Pi$ we have that $\mathcal{C}^{LR} \subset \mathcal{C}$

Now consider any policy $\pi \in \Pi$ with eMBB user allocations $\boldsymbol{\phi}^{\pi}$ and URLLC loads $\mathbf{l}^{\pi}$ and associated long term throughput is $\mathbf{c}^{\pi}$ given by

$$c_u^{\pi} = \sum_{s \in \mathcal{S}} \hat{r}_u^s(\phi_u^{\pi,s} - l_u^{\pi,s}) p_S(s).$$

Let us define a π' based on π to have per mini-slot eMBB user allocations given by

$$\phi_{u,m}^{\pi',s} = \frac{\phi_u^{\pi,s} - l_u^{\pi,s}}{\sum_{u' \in \mathcal{U}} \phi_{u'}^{\pi,s} - l_{u'}^{\pi,s}} f = \frac{\phi_u^s - l_u^{\pi,s}}{1-\rho} f,$$

for $s \in \mathcal{S}$, $u \in \mathcal{U}$ and $m \in \mathcal{M}$. Since induced mean loads on an eMBB user can not exceed its allocation we have that $\boldsymbol{\phi}^{\pi} \geq \mathbf{l}^{\pi}$ so the above allocations are positive. Note also that this allocation is not mini-slot dependent, but normalized so that per mini-slot they sum to f and over the whole eMBB slot sum to 1, i.e., $\boldsymbol{\phi}^{\pi'} \in \Sigma$. Thus for such an allocation we have that

$$\phi_u^{\pi',s} = \frac{\phi_u^s - l_u^{\pi,s}}{1-\rho}.$$

Also suppose that π' uses randomized URLLC placement across mini-slots which induces mean URLLC loads proportional to the allocations, i.e., $l_u^{\pi',s} = \rho \phi_u^{\pi',s}$. It follows that

$$\begin{aligned} \phi_u^{\pi',s} - l_u^{\pi',s} &= \phi_u^{\pi',s} - \rho\phi_u^{\pi',s} \\ &= (1-\rho)\phi_u^{\pi',s} \\ &= \phi_u^{\pi,s} - l_u^{\pi,s}, \end{aligned}$$

and so $c_u^{\pi,s} = c_u^{\pi',s}$ for all $s \in \mathcal{S}$ and $u \in \mathcal{U}$. Thus for any policy π there is a policy π' which uses randomized URLLC placement and achieves the same long term throughputs. It follows that $\mathcal{C} \subset \mathcal{C}^{LR}$ and so $\mathcal{C} = \mathcal{C}^{LR}$. □

B. Proofs from Section IV

Theorem 2. *Under a $(1-\delta)$ sharing factor and time-homogeneous scheduler $\boldsymbol{\pi} = (\boldsymbol{\phi}^{\boldsymbol{\pi}}, \boldsymbol{\gamma}^{\boldsymbol{\pi}}) \in \Pi^{U,\delta}$ the probability of induced throughput for user r $u \in \mathcal{U}$ in channel state $s \in \mathcal{S}$ is given by*

$$r_u^{\boldsymbol{\pi},s} = \mathrm{E}[f_u^s(\phi_u^{\boldsymbol{\pi},s}, \gamma_u^{\boldsymbol{\pi},s} D)].$$

and the overall user throughputs are given by $\mathbf{c}^{\boldsymbol{\pi}} = (c_u^{\boldsymbol{\pi}} : u \in \mathcal{U})$ where $c_u^{\boldsymbol{\pi}} = \sum_{u \in \mathcal{U}} r_u^{\boldsymbol{\pi},s} p_S(s)$.

Proof. Under a policy $\boldsymbol{\pi} = (\boldsymbol{\phi}^{\boldsymbol{\pi}}, \boldsymbol{\gamma}^{\boldsymbol{\pi}}) \in \Pi^{U,\delta}$ we have that the induced loads are given by

$$L_{u,m}^{\boldsymbol{\pi},s} = \frac{\gamma_{u,1}^{\boldsymbol{\pi},s}}{f} D(m),$$

so we have that

$$L_u^{\boldsymbol{\pi},s} = \sum_{u \in \mathcal{U}} L_{u,m}^{\boldsymbol{\pi},s} = \frac{\gamma_{u,1}^{\boldsymbol{\pi},s}}{f} \sum_{u \in \mathcal{U}} D(m) = \frac{\gamma_{u,1}^{\boldsymbol{\pi},s}}{f} D = \gamma_u^{\boldsymbol{\pi},s} D.$$

where the last equality follows from the uniformity of URLLC splits and normalization it follows that

$$r_u^{\boldsymbol{\pi},s} = \mathrm{E}[f_u^s(\phi_u^{\boldsymbol{\pi},s}, L_u^{\boldsymbol{\pi},s})] = \mathrm{E}[f_u^s(\phi_u^{\boldsymbol{\pi},s}, \gamma_u^{\boldsymbol{\pi},s} D)].$$

□

Lemma 1. *Condition 1 is satisfied for systems where superposition/puncturing of each user is modelled via either a*

1) *convex loss function,*
2) *threshold-based loss function with fixed relative thresholds, i.e., $t_u^s(\phi_u^s) = \alpha_u^s$ for $\phi \in [0.1]$ and the URLLC demand distribution F_D is such that $F_D(\frac{1}{x})$ is concave in x (satisfied by the truncated Pareto distribution).*

Proof. Recall that convex loss functions are specified as follows

$$f_u^s(\phi_u^s, l_u^s) = \hat{r}_u^s \phi_u^s (1 - h_u^s(\frac{l_u^s}{\phi_u^s})),$$

with $h_u^s : [0,1] \to [0,1]$ a convex increasing function. For time-homogenous policies we have defined

$$\begin{aligned} g_u^s(\phi_u^s, \gamma_u^s) &= \mathrm{E}[f_u^s(\phi_u^s, \gamma_u^s D)] \\ &= \hat{r}_u^s E[\phi_u^s - \phi_u^s h_u^s(\frac{\gamma_u^s}{\phi_u^s} D)]. \end{aligned}$$

Recall that convex function $h()$ one can define a function $l(\phi, \gamma) = \phi h(\frac{\gamma}{\phi})$ known as the perspective of $h()$ which is known to be jointly convex in its arguments. It follows that $\phi - \phi h(\frac{\gamma}{\phi})$ is jointly concave, and so is $g_u^s()$ since it is a weighted aggregation of jointly concave functions.

For threshold-based loss functions where $t_u^s(\phi_u^s) = \alpha_u^s$ we have that

$$\begin{aligned} g_u^s(\phi_u^s, \gamma_u^s) &= \mathrm{E}[f_u^s(\phi_u^s, \gamma_u^s D)] \\ &= \hat{r}_u^s \phi_u^{\pi,s} P(\gamma_u^s D \leq \phi_u^{\pi,s} \alpha_s^u) \\ &= \hat{r}_u^s \phi_u^{\pi,s} F_D(\frac{\phi_u^{\pi,s} \alpha_s^u}{\gamma_u^s}). \end{aligned}$$

Now using the same result on the perspective functions of variables the result follows. The truncated Pareto case can be easily verified by taking derivatives. □

Theorem 3. *Suppose that Condition 1 holds. then $\mathcal{C}^{U,\delta} = \hat{\mathcal{C}}^{U,\delta}$, i.e., there is no need to consider time-sharing/randomization amongst time-homogeneous eMBB/URLLC policies.*

Proof. Clearly $\mathcal{C}^{U,\delta} \subset \mathcal{C}^{U,\delta}$. We will show that $\mathbf{c} \in \hat{\mathcal{C}}^{U,\delta}$ then their exists $\boldsymbol{\pi} = (\boldsymbol{\phi}^{\boldsymbol{\pi}}, \boldsymbol{\gamma}^{\boldsymbol{\pi}}) \in \Pi^{U,\delta}$ such that $\mathbf{c} \leq \mathbf{c}^{\boldsymbol{\pi}}$ from which it follows that $\mathcal{C}^{U,\delta} \subset \mathcal{C}^{U,\delta}$.

Suppose $\mathbf{c} \in \hat{\mathcal{C}}^{U,\delta}$, then it can be represented as a convex combination of policies $\Pi^{U,\delta}$, in each channel state. For example suppose for simplicity that for that in channel state $s \in \mathcal{S}$ we have that $\lambda \in [0,1]$ one time shares between two policies $\boldsymbol{\pi}_1$ and $\boldsymbol{\pi}_2$ to achieve throughputs for $u \in \mathcal{U}$ given by

$$r_u^s = \lambda r_u^{\boldsymbol{\pi}_1,s} + (1-\lambda) r_u^{\boldsymbol{\pi}_2,s}.$$

Consider u we have

$$\begin{aligned} r_u^s &= \lambda r_u^{\boldsymbol{\pi}_1,s} + (1-\lambda) r_u^{\boldsymbol{\pi}_2,s} \\ &= \lambda g_u^s(\phi_u^{\boldsymbol{\pi}_1,s}, \gamma_u^{\boldsymbol{\pi}_1,s}) + (1-\lambda) g_u^s(\phi_u^{\boldsymbol{\pi}_2,s}, \gamma_u^{\boldsymbol{\pi}_2,s}) \\ &\leq g_u^s(\lambda\phi_u^{\boldsymbol{\pi}_1,s} + (1-\lambda)\phi_u^{\boldsymbol{\pi}_2,s},\ \lambda\gamma_u^{\boldsymbol{\pi}_1,s} + (1-\lambda)\phi_u^{\gamma_2,s}) \\ &= g_u^s(\phi_u^{\boldsymbol{\pi},s}, \gamma_u^{\boldsymbol{\pi},s}), \end{aligned}$$

where $\phi_u^{\boldsymbol{\pi},s} = \lambda\phi_u^{\boldsymbol{\pi}_1,s} + (1-\lambda)\phi_u^{\boldsymbol{\pi}_2,s}$ and $\gamma_u^{\boldsymbol{\pi},s} = \lambda\gamma_u^{\boldsymbol{\pi}_1,s} + (1-\lambda)\gamma_u^{\boldsymbol{\pi}_2,s}$. Clearly $\boldsymbol{\phi}^{\boldsymbol{\pi}}, \boldsymbol{\gamma}^{\boldsymbol{\pi}}$ as given above correspond to a policy $\boldsymbol{\pi}$ such that $\boldsymbol{\pi} \in \Pi^{U,\delta}$ since the set is convex. It also follows that $r_u^s \leq r_u^{\boldsymbol{\pi},s}$, so $c_u^s \leq c_u^{\boldsymbol{\pi},s}$ and so $\mathbf{c} \leq \mathbf{c}^{\boldsymbol{\pi}}$. □

Theorem 4. *Let $\boldsymbol{r}^*$ be the optimal average rate vector received by eMBB users under the solution to the offline optimization problem. Suppose that Conditions 1 and 2 hold. Then we have that:*

$$\lim_{t\to\infty} \overline{\boldsymbol{R}}(t) = \boldsymbol{r}^* \quad \textit{almost surely}. \tag{15}$$

The proof requires intermediate lemmas, detailed below. For the ease of exposition, let us define $U(\boldsymbol{r}) := \sum_{u\in\mathcal{U}} U_u(r_u)$ and $\nabla U(\boldsymbol{r}) := \left[\frac{\partial U_1(x)}{\partial x}\Big|_{x_1=r_1}, \frac{\partial U_2(x)}{\partial x}\Big|_{x_2=r_2}, \ldots, \frac{\partial U_1(x)}{\partial x}\Big|_{r_{x_{|\mathcal{U}|}=|\mathcal{U}|}}\right]^T$. First we have the following important lemma regarding the stochastic approximation algorithm.

Lemma 2. *$\boldsymbol{R}(t) = \left[R_1(t), R_2(t), \ldots, R_{|\mathcal{U}|}\right]^T$ is an unbiased estimator of argmax: $\nabla U\left(\overline{\boldsymbol{R}}(\boldsymbol{t})\right)^T \boldsymbol{c}$, i.e.,*

$$\mathbb{E}\left[\boldsymbol{R}(t)\right] = \underset{\boldsymbol{c}\in\mathcal{C}^{\mathcal{U},\delta}}{\textit{argmax:}}\ \nabla U\left(\overline{\boldsymbol{R}}(\boldsymbol{t})\right)^T \boldsymbol{c}. \tag{16}$$

Proof. Based on the definition of $\mathcal{C}^{\mathcal{U},\delta}$ we can re-write $\underset{\boldsymbol{c}\in\mathcal{C}^{\mathcal{U},\delta}}{\max:}\ \nabla U\left(\overline{\boldsymbol{R}}(\boldsymbol{t})\right)^T \boldsymbol{c}$ as follows:

$$\max_{\boldsymbol{\phi},\boldsymbol{\gamma}} \quad \sum_{u\in\mathcal{U}} U_u^{'}\left(\overline{R}_u(t)\right)\left(\sum_{s\in\mathcal{S}} p_{\mathcal{S}}(s) g_u^s\left(\phi_u^s, \gamma_u^s\right)\right) \tag{17}$$

$$\text{s.t.} \quad \boldsymbol{\phi} \geq (1-\delta)\,\boldsymbol{\gamma}, \tag{18}$$

$$\boldsymbol{\phi},\ \boldsymbol{\gamma} \in \Pi^{U,\delta}. \tag{19}$$

Observe that the above optimization problem can be solved separately for each network state $s \in \mathcal{S}$. The de-coupled problem for any state s is same as the optimization problem (9) in our online algorithm. With a slight abuse of notation, let $\left(\tilde{\phi}(s), \tilde{\gamma}(s)\right)$ be the optimal solution to the online problem when $S(t) = s$. Conditioned on $S(t) = s$, we have that:

$$\begin{aligned} \mathbb{E}\left[R_u(t) \mid S(t) = s\right] &= \mathbb{E}\left[f_u^s\left(\tilde{\phi}_u^s, \tilde{\gamma}_u^s D\right) \mid S(t) = s\right] \\ &= g_u^s\left(\tilde{\phi}_u^s, \tilde{\gamma}_u^s\right) \quad \forall u \in \mathcal{U}. \end{aligned} \tag{20}$$

Computing $\mathbb{E}\left[\mathbb{E}\left[R_u(t) \mid S(t)\right]\right]$ gives the desired result (16). □

The main intuition behind the proof of optimality is that for large t, the trajectories of $\overline{\boldsymbol{R}}(t)$ can be approximated by the solution to the following differential equation in $\boldsymbol{x}(t)$ with continuous time t:

$$\frac{d\boldsymbol{x}(t)}{dt} = \underset{\boldsymbol{c}\in\mathcal{C}^{\mathcal{U},\delta}}{\text{argmax:}}\ \nabla U\left(\boldsymbol{x}(t)\right)^T \boldsymbol{c} - \boldsymbol{x}(t). \tag{21}$$

Let us define $q(\boldsymbol{x}) := \underset{\boldsymbol{c}\in\mathcal{C}^{\mathcal{U},\delta}}{\text{argmax:}}\ \nabla U\left(\boldsymbol{x}\right)^T \boldsymbol{c}$. To show the optimality of our online algorithm, we shall also require the following result on the above differential equation.

Lemma 3. *The differential equation* (21) *is globally asymptotically stable. Furthermore, for any initial condition $\boldsymbol{x}(0) \in C^{\mathcal{U},\delta}$, we have that $\lim_{t\to\infty} \boldsymbol{x}(t) = \boldsymbol{r}^*$.*

Proof. To prove this lemma it is enough to show that there exists a Lyapunov function $L(\boldsymbol{x}(t))$ such that it has a negative drift when $x(t) \neq \boldsymbol{r}^*$ and has zero drift when $x(t) = \boldsymbol{r}^*$. Define $L(\boldsymbol{x}) = U(\boldsymbol{r}^*) - U(\boldsymbol{x})$. Observe that under our assumption of strictly concave $U_u(\cdot)$, the offline optimization problem is guaranteed to have an unique optimal solution, which is $\boldsymbol{r}^*$. Therefore, $\forall \boldsymbol{x} \in C^{\mathcal{U},\delta}$ and $\boldsymbol{x} \neq \boldsymbol{r}^*$ $L(\boldsymbol{x}) > 0$. Next we will compute the drift of $L(\boldsymbol{x}(t))$ with respect to time.

$$\frac{dL(\boldsymbol{x}(t))}{dt} = -\nabla U\left(\boldsymbol{x}(t)\right)^T \frac{d\boldsymbol{x}(t)}{dt}, \tag{22}$$

$$= -q\left(\boldsymbol{x}(t)\right) + \nabla U\left(\boldsymbol{x}(t)\right)^T \boldsymbol{x}(t), \tag{23}$$

$$< 0 \qquad \forall \boldsymbol{x}(t) \neq \boldsymbol{r}^*. \tag{24}$$

To get inequality (24), first observe that from the definition of $q(\boldsymbol{x}(t))$ and (23), we get that $\frac{dL(\boldsymbol{x}(t))}{dt} \leq 0$. However, we have to show that this inequality is strict for $\boldsymbol{x}(t) \neq \boldsymbol{r}^*$. Observe that $q(\boldsymbol{x}) = \boldsymbol{x}$ is a necessary and sufficient condition for optimality of the offline optimization problem, see [18] for more details. From strict concavity of the utility functions, we have an unique optimal point $\boldsymbol{r}^*$. Therefore, $\frac{dL(\boldsymbol{x}(t))}{dt} < 0$ for $\boldsymbol{x}(t) \neq \boldsymbol{r}^*$ and $\frac{dL(\boldsymbol{x}(t))}{dt} = 0$ at $\boldsymbol{x}(t) = \boldsymbol{r}^*$. □

To conclude the proof, Lemmas 2 and 3 along with the condition 2 satisfy all the conditions necessary to apply Theorem 2.1 in Chapter 5, [19] which states that $\overline{\boldsymbol{R}}(t)$ converges to $\boldsymbol{r}^*$ almost surely.

C. Proofs and Additional Results from Section V

First we state is a corollary to Theorem 2 for systems having threshold model for superposition/puncturing.

Corollary 1. *Under a $(1-\delta)$ sharing factor and time-homogeneous scheduler $\boldsymbol{\pi} = (\boldsymbol{\phi}^{\boldsymbol{\pi}}, \boldsymbol{\gamma}^{\boldsymbol{\pi}}) \in \Pi^{U,\delta}$ the probability of induced eMBB loss for user $u \in \mathcal{U}$ in channel state $s \in \mathcal{S}$ is given by*

$$\epsilon_u^{\boldsymbol{\pi},s} = 1 - F_D(\frac{\phi_u^{\boldsymbol{\pi},s} t_u^s(\phi_u^{\boldsymbol{\pi},s})}{\gamma_u^{\boldsymbol{\pi},s}}).$$

where F_D denotes the cumulative distribution function of the URLLC demands on a typical eMBB slot. Then the associated user throughput is given by

$$r_u^{\boldsymbol{\pi},s} = \hat{r}_u^s \phi_u^{\boldsymbol{\pi},s} F_D(\frac{\phi_u^{\boldsymbol{\pi},s} t_u^s(\phi_u^{\boldsymbol{\pi},s})}{\gamma_u^{\boldsymbol{\pi},s}}).$$

and the overall user throughputs are given by $\mathbf{c}^{\boldsymbol{\pi}} = (c_u^{\boldsymbol{\pi}} : u \in \mathcal{U})$ where

$$c_u^{\boldsymbol{\pi}} = \sum_{u \in \mathcal{U}} \hat{r}_u^s \phi_u^s F_D(\frac{\phi_u^{\boldsymbol{\pi},s} t_u^s(\phi_u^{\boldsymbol{\pi},s})}{\gamma_u^{\boldsymbol{\pi},s}}) p_S(s).$$

The following two corollaries are direct consequences of Corollary 1 and Theorem 3 restricted to RP and TP Placement strategies, and characterize the throughput regions under these policies.

Corollary 2. *Consider a wireless system with full sharing factor and time-homogeneous scheduler based on the RP URLLC Placement policy $\boldsymbol{\pi} = (\boldsymbol{\phi}^{\boldsymbol{\pi}}, \boldsymbol{\gamma}^{\boldsymbol{\pi}}) \in \Pi^{RP,\delta}$. Then any eMBB resource allocation $\boldsymbol{\phi}$ combined with a RP URLLC demand placement policy, $\boldsymbol{\gamma} = \boldsymbol{\phi}$ is feasible. The probability of loss for user $u \in \mathcal{U}$ in channel state $s \in \mathcal{S}$ is given by*

$$\epsilon_u^{\boldsymbol{\pi},s} = 1 - F_D(t_u^s(\phi_u^{\boldsymbol{\pi},s})),$$

with associated user throughput

$$r_u^{\boldsymbol{\pi},s} = \hat{r}_u^s \phi_u^s F_D(t_u^s(\phi_u^{\boldsymbol{\pi},s})). \tag{25}$$

Further if for all $s \in \mathcal{S}$ and $u \in \mathcal{U}$ the functions $g_u^s(,)$ given by

$$g_u^s(\phi_u^s) = \phi_u^s F_D(t_u^s(\phi_u^{\boldsymbol{\pi},s})), \tag{26}$$

are concave then $\mathcal{C}^{RP,\delta} = \hat{\mathcal{C}}^{RP,\delta}$.

Corollary 3. *Under a $(1-\delta)$ sharing factor and jointly uniform scheduler based on the TP URLLC Placement policy $\boldsymbol{\pi} = (\boldsymbol{\phi}^{\boldsymbol{\pi}}, \boldsymbol{\gamma}^{\boldsymbol{\pi}}) \in \Pi^{TP,\delta}$, the probability of induced eMBB loss user $u \in \mathcal{U}$ in channel state $s \in \mathcal{S}$ is given by*

$$\epsilon_u^{\boldsymbol{\pi},s} = 1 - F_D(\sum_{u \in \mathcal{U}} \phi_u^{\boldsymbol{\pi},s} t_u^s(\phi_u^{\boldsymbol{\pi},s})), \tag{27}$$

with associated user throughput

$$r_u^{\boldsymbol{\pi},s} = \hat{r}_u^s \phi_u^s F_D(\sum_{u \in \mathcal{U}} \phi_u^{\boldsymbol{\pi},s} t_u^s(\phi_u^{\boldsymbol{\pi},s})). \tag{28}$$

Further if for all $s \in \mathcal{S}$ and $u \in \mathcal{U}$ the functions $g_u^s(,)$ given by

$$g_u^s(\phi_u^s, \gamma_u^s) = \phi_u^s F_D(\sum_{u \in \mathcal{U}} \phi_u^{\boldsymbol{\pi},s} t_u^s(\phi_u^{\boldsymbol{\pi},s})), \tag{29}$$

are jointly concave then $\mathcal{C}^{TP,\delta} = \hat{\mathcal{C}}^{TP,\delta}$.

Finally, using the above corollary, we show the optimality of TP Placement with respect to probability of loss on a given eMBB slot.

Theorem 5. *Consider a system with $(1-\delta)$ sharing factor. Consider a joint scheduling policy based on the TP URLLC Placement i.e, $\boldsymbol{\pi} = (\boldsymbol{\phi}^{\boldsymbol{\pi}}, \boldsymbol{\gamma}^{\boldsymbol{\pi}}) \in \Pi^{TP,\delta}$. Then $\boldsymbol{\pi}$ achieves the minimum probability of eMBB loss amongst all joint scheduling policies using the same eMBB resource allocation $\boldsymbol{\phi}^{\boldsymbol{\pi}}$.*

Proof. Clearly the probability of loss depends on the minislot demands and the users thresholds. If one relaxes the sequential constraint on URLLC allocations, one can consider aggregating the the minislot demands and pooling together the users superposition/puncturing thresholds. The probability of loss for this relaxed system is simply the probability the demand exceeds the size of the superposition/puncturing pool, i.e., The probability of loss under the pooled resources is given by

$$P(D \geq \sum_{u \in \mathcal{U}} \phi_u^s t_u^s(\phi_u^s)).$$

This is clearly a lower bound for any placement policy. Note however that the threshold proportional strategy meets this bound from Corollary 3 (see Equation 27) so it indeed minimizes the probability of loss on a given eMBB slot. □