DYNAMIC POWER CONTROL FOR TIME-CRITICAL NETWORKING WITH HETEROGENEOUS TRAFFIC

Emmanouil Fountoulakis[1], Nikolaos Pappas[2], Anthony Ephremides[1,2]
[1]Department of Science and Technology, Linköping University, Sweden, [2]Electrical and Computer Engineering Department, University of Maryland, College Park, USA

NOTE: Corresponding author: Emmanouil Fountoulakis, emmanouil.fountoulakis@liu.se

Abstract – Future wireless networks will be characterized by heterogeneous traffic requirements. Examples can be low-latency or minimum-througput requirements. Therefore, the network has to adjust to different needs. Usually, users with low-latency requirements have to deliver their demand within a specific time frame, i.e., before a deadline, and they coexist with throughput oriented users. In addition, mobile devices have a limited-power budget and therefore, a power-efficient scheduling scheme is required by the network. In this work, we cast a stochastic network optimization problem for minimizing the packet drop rate while guaranteeing a minimum throughput and taking into account the limited-power capabilities of the users. We apply tools from Lyapunov optimization theory in order to provide an algorithm, named Dynamic Power Control (DPC) algorithm, that solves the formulated problem in real time. It is proved that the DPC algorithm gives a solution arbitrarily close to the optimal one. Simulation results show that our algorithm outperforms the baseline Largest-Debt-First (LDF) algorithm for short deadlines and multiple users.

Keywords – Deadline-constrained traffic, dynamic algorithms, heterogeneous traffic, Lyapunov optimization, power-efficient algorithms, scheduling.

1. INTRODUCTION

5G and beyond networks are poised to support a mixed set of applications that require different types of services. There are two main categories of applications. The first category includes applications that require bandwidth-hungry services and the second includes delay-sensitive applications. The second category differentiates the current networks from future networks. These applications require low-latency services and increase the need for time-critical networking. In time-critical networking, applications are required to deliver their demands within a specific time duration [1]. In other words, each packet or a batch of packets has a deadline within which data must be transmitted, otherwise, it is dropped and removed from the system [2]. This is connected with the notion of timely throughput. Timely throughput measures the long-term time average number of successful deliveries before the deadline expiration [3, 4]. Each time-critical application belongs to a different category. For example, motion control, smart grid control, and process monitoring belong to the industrial control category. Furthermore, the growing popularity of real-time media applications increases the need for designing networks that can offer services with low latency. Such applications are media production, interactive Virtual Reality (VR), cloud computing, etc, that are under the umbrella of the Tactile Internet [5].

With the pervasiveness of mobile communications, such applications need to perform over wireless devices. In order to achieve reliable communication, the devices have to adapt their power transmission according to channel conditions. However, many devices may have a limited power budget. Therefore, energy-efficient communications have become a very important issue. In this work, we propose a scheduling algorithm that handles a heterogeneous set of users with heterogeneous traffic. In particular, we consider a network with deadline-constrained users and users with minimum-throughput requirements, with a limited-power budget. We provide an algorithm that solves the scheduling problem in real time. We prove that the obtained solution is arbitrarily close to the optimal.

1.1 Related works

Delay-constrained network optimization and performance analysis have been extensively investigated [6]. A variety of approaches have been applied to different scenarios. There is a line of work that considers the control of the maximum number of retransmissions before the deadline expiration. In [7], the authors consider a user transmitting packets over a wireless channel to a receiver. An optimal scheduling scheme is proposed that provides the optimal number of retransmissions for a packet. In [8], the authors consider users with packets with deadlines in a random-access network. They show how the number of maximum retransmissions affects the packet drop rate. In [9], the authors consider a single transmitter that transmits symbols to a receiver. Each symbol has a deadline and a corresponding distortion function. The authors consider the distortion-minimization problem while fulfilling deadline constraints. In [10], the au-

thors consider a system with an Age-of-Information (AoI) oriented user and a deadline-constrained user. The authors provide the distribution of the AoI and the packet drop rate and they examine the interplay between them. Furthermore, energy and power efficient scheduling schemes for delay-constrained traffic have attracted a lot of attention over the last the few years [11–16]. In [11], the authors consider the minimization of drop rate for users with a limited power-budget. They propose an approximated algorithm that performs in real time. In [12], the authors propose an algorithm that minimizes the time average power consumption while guaranteeing minimum throughput and reducing the queueing delay. They also consider a hybrid multiple access system where the scheduler decides if the transmitter serves a user by orthogonal multiple access or non-orthogonal multiple access. In [13], [14], the authors utilize Markov decision theory to provide an optimal energy-efficient algorithm for delay-constrained users.

Lyapunov optimization theory has been widely applied for developing dynamic algorithms that schedule users with packets with deadlines. In [17–20], the authors consider the rate maximization under power and delay constraints. In [17], the authors consider the power allocation for users with hard-deadline constraints. In [18], the authors consider the rate maximization of non-real-time users while satisfying the packet drop rate for users with packets with deadlines. In [19, 20], consider packets with deadlines for scheduling real-time traffic in wireless environments. A novel approach for minimizing the packet drop rate while guaranteeing stability is provided in [21]. The authors combine tools from Lyapunov optimization theory and Markov decision processes in order to develop an optimal algorithm for minimizing the drop rate under stability constraints. However, the algorithm is able to solve small network scenarios because of the curse of dimensionality problem.

Besides delay-constrained traffic management, throughput-optimal algorithms have been developed over the years. Following the seminal work in [22], many researchers developed different solutions for the throughput-maximization problem by proposing a variety of approaches [23–25]. In [23], the authors consider the throughput-maximization while guaranteeing certain interservice times for all the links. They propose the time-since-last-service metric. They combine the last with the queue length of each user and they propose a max-weight policy based on Lyapunov optimization. In [24], [25], the authors consider the throughput-maximization in networks with dynamic flows. More specifically, in [24], the authors consider a hybrid system with both persistent and dynamic flows. They provide a queue-maximum-weight based algorithm that guarantees throughput-optimality while reducing the latency. In [25], the authors consider a network with dynamic flows of random size and they arrive in random size at the base station. The service times for each flow varies randomly because of the wireless chan-

nel. The authors provide a delay-MaxWeight scheduler that has proven its throughput is optimal. Research on scheduling heterogeneous traffic with Ultra-Reliable Low Latency (URLLC) users and enhanced Mobile Brodband (eMBB) has attracted a lot of attention by the community [26]- [33]. In [26], [27], the authors show the benefits of flexible Transmission Time Interval (TTI) for scheduling users with different types of requirements. In [28], the authors propose an algorithm that jointly schedules URLLC and eMBB traffic. They consider a slotted time system in which the slots are divided into mini slots. They consider the frequency and mini-slots allocation over one slot. In [29], the authors consider the resource allocation for URLLC users. They study resource allocation for different scenarios: i) OFDMA system, ii) system that includes retransmissions. In [30], [31], the authors propose a low-complexity algorithm for scheduling URLLC users. The authors in [32] consider the throughput maximization and HARQ optimization for URLLC users. Furthermore, reliable transmission is an important issue of URLLC communications. In [33], the authors consider a network in which multiple unreliable transmissions are combined to achieve reliable latency. The authors model the problem as a constrained Markov decision problem, and they provide the optimal policy that is based on dynamic programming.

1.2 Contributions

In this work, we consider two sets of users with heterogeneous traffic and a limited-power budget. The first set includes users with packets with deadlines and the second set includes users with minimum-throughput requirements. We provide a dynamic algorithm that schedules the users in real time and minimizes the drop rate while guaranteeing minimum throughput and limited-power consumption. The contributions of this work are the following.

- We formulate an optimization problem for minimizing the drop rate with minimum-throughput constraints and time average power consumption constraints.

- We provide a novel objective function for minimizing the drop rate. The objective function does not take into account only if a packet is going to expire or not, but also the remaining time of a packet before its expiration.

- We apply tools from the Lyapunov optimization theory to satisfy the time average constraints: throughput and power consumption.

- The proposed algorithm is proved to provide a solution arbitrarily close to the optimal.

- Simulation results show that our algorithm outperforms the baseline algorithm proposed in [3] for short deadlines and multiple users.

2.   SYSTEM MODEL

We consider N users transmitting packets to a single receiver over wireless fading channels. Let $\mathcal{N} \triangleq \{1, \ldots, N\}$ be the set of the total users in the system. Time is assumed to be slotted, and $t \in \mathbb{Z}$ denotes the tth slot.
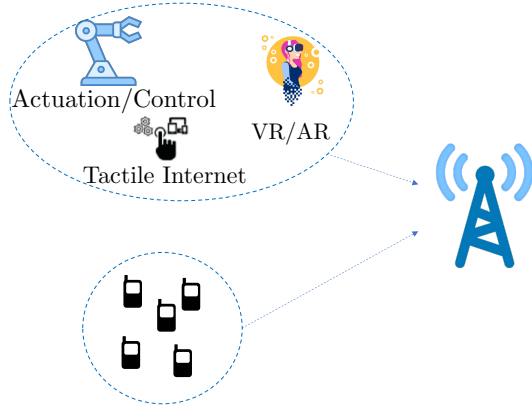

Fig. 1 – Example of the system model with time-critical users with packets with deadlines and users with minimum-throughput requirements.

We consider the users to be synchronized and controlled by a scheduler. In our system model, at most one user can transmit at each time slot. We consider two sets of users. The first set includes users that have arrivals of packets with deadlines. We denote the set of users with packets with deadlines by $\mathcal{R} \subseteq \mathcal{N}$. The second set of users, $\mathcal{U} \subseteq \mathcal{N}$, contains the users that have throughput requirements. We consider that each user $u \in \mathcal{U}$ is saturated and therefore, it has always packet to transmit. Note that $\mathcal{U} \cup \mathcal{R} = \mathcal{N}$ and $\mathcal{U} \cap \mathcal{R} = \emptyset$. An example of our system model is shown in Fig. 1.

For the deadline-constrained users, each packet that arrives in the queue of the users has a deadline by which the packet must be transmitted. Otherwise, it is dropped and removed from the system. We assume that the deadlines of the packets in the same queue are the same. However, deadlines of different queues may vary. We denote the packet deadline of the rth queue with $m_r \in \mathbb{Z}_+, \forall r \in \mathcal{R}$. Let $d_r(t)$ be the number of slots left in the tth slot before the packet that is at the head of queue r expires. Let $Q_r(t)$ be the number of packets in queue r in the tth slot. A packet arrives with probability π_r at every time slot in the queue of user r. Let $\boldsymbol{\alpha}(t) \triangleq \{\alpha_r(t)\}_{r \in \mathcal{R}}$, where $\alpha_r(t) \in \{0, 1\}$, represents the packet arrival process for each user r in the tth slot. The random variables of the packet arrival process are independent and identically distributed (i.i.d.). Let λ_r denote the arrival rate for user r and $\mathbb{E}\{\alpha_r(t)\} = \lambda_r$. Furthermore, at most one packet can be transmitted at each time slot and no collisions are allowed. In each queue of every user $r \in \mathcal{R}$, packets are served in the order that they arrive following the First In First Out (FIFO) discipline.

We assume that the channel state at the beginning of each time slot is known. The channel state remains constant within one slot but it changes from slot to slot. Let $\mathbf{S}(t) \triangleq$

$\{S_i(t)\}_{i \in \mathcal{N}}$ represent the channel state for each user i during slot t. Also, the channel can be either in "Bad" state (deep fading) or in "Good" state (mild fading). The possible channel states of each user i are described by the set $\mathcal{S} \triangleq \{\text{B}, \text{G}\}$, and $S_i(t) \in \mathcal{S}, \forall i \in \mathcal{N}$. For simplicity, we assume that the random variables of the channel process $\mathbf{S}(t)$ are i.i.d. from one slot to the next.

Let $\mathbf{p}(t) \triangleq [p_1(t), \ldots, p_N(t)]$ denote the power allocation vector in the tth slot. We consider a set of discrete power levels $\{0, P^{(\text{Low})}, P^{(\text{High})}\}$. The required power to have a successful transmission under "Bad" and "Good" channel conditions is denoted by P^{High} and P^{Low}, respectively. At each time slot, the set of selectable power levels $\mathcal{P}_i(t)$ for each user is conditioned on the channel state $S_i(t)$. For example, if the current channel state is "Bad", then $P^{(\text{Low})}$ cannot be selected. Thus, we have

$$p_i(t) \in \begin{cases} \left\{0, P^{(\text{High})}\right\}, \text{if } S_i(t) = \text{B} \\ \left\{0, P^{(\text{Low})}\right\}, \text{if } S_i(t) = \text{G} \end{cases}, \forall i \in \mathcal{N}. \quad (1)$$

Let $\mu_i(t)$ be the power allocation, or packet serving, indicator for the user i in the tth slot, we have

$$\mu_i(t) \triangleq \begin{cases} 1, \text{if } p_i(t) > 0 \\ 0, \text{otherwise} \end{cases}, \forall i \in \mathcal{N}. \quad (2)$$

At most one packet can be transmitted in a timeslot t, i.e., the vector $\mathbf{p}(t)$ has at most one non-zero element. The set of power constraints for $\mathbf{p}(t)$ is then defined by

$$\mathcal{P}(t) \triangleq \left\{ \mathbf{p}(t) : \sum_{i=1}^{N} \mathbf{1}_{\{\mu_i(t)=1\}} \leq 1 \right\}, \quad (3)$$

where $\mathbf{1}_{\{.\}}$ denotes the indicator function. For each user $r \in \mathcal{R}$, a packet is dropped if its deadline has expired. Since the queue follows the FIFO discipline, a packet is dropped under the following conditions: 1) it is at the head of the queue; 2) the remaining number of the slots to serve the packet is 1; and 3) power is not assigned to user r at the current slot. Let $D_r(t)$ be the indicator of the packet drop for user r at time t. The queue evolution for each user $r \in \mathcal{R}$ is described as

$$Q_r(t+1) = \max\left[Q_r(t) - \mu_r(t), 0\right] + \alpha_r(t) - D_r(t), \forall i \in \mathcal{R}. \quad (4)$$

We define the packet drop rate for each user $r \in \mathcal{R}$, the average power consumption for each user $i \in \mathcal{N}$, and the throughput for each user $u \in \mathcal{U}$ as

$$\overline{D}_r \triangleq \lim_{t \to \infty} \overline{D}_r(t), \forall r \in \mathcal{R}, \quad (5)$$

$$\overline{p}_i \triangleq \lim_{t \to \infty} \overline{p}_i(t), \forall i \in \mathcal{N}, \quad (6)$$

$$\overline{\mu}_u = \lim_{t \to \infty} \bar{\mu}_u(t), \forall u \in \mathcal{U}, \quad (7)$$

respectively, where, $\overline{D}_r(\tau) = \frac{1}{t} \sum_{\tau=0}^{t-1} D_r(t)$, $\overline{p}_i(t) = \frac{1}{t} \sum_{\tau=0}^{t-1} p_i(\tau)$, and $\bar{\mu}_u(t) = \frac{1}{t} \sum_{\tau=0}^{t-1} \mu_u(\tau)$. The packet drop

Table 1 – Notation Table.

$\mathcal{N}$	Set of the total users in the system	$\mathcal{P}_i(t)$	Set of selectable power levels of user i
$\mathcal{R}$	Set of the deadline-constrained users	$Z_u(t)$	Length of throughput-dept queue of user u
$\mathcal{U}$	Set of the minimum-throughput requirements users	$f_r(t)$	Cost function for user r
t	t^{th} slot	$\mu_r(t)$	Power allocation indicator of user i
$Q_r(t)$	Number of packets in queue r	$\mathcal{P}(t)$	Set of power constraints for $\mathbf{p}(t)$
π_r	Packet arrival probability of user i	$D_r(t)$	Packet drop indicator of user r
$\alpha_r(t)$	Packet arrival indicator of user r	$\overline{D}_r$	Packet drop rate of user r
m_r	Deadline of packet of user r	$\overline{p}_r$	Average power consumption of user r
$d_r(t)$	Number of slots left before the deadline of user r	$X_r(t)$	Length of virtual queue of user i
$\mathbf{S}(t)$	Channel states	$L(\cdot)$	Quadratic Lyapunov function
$\mathbf{p}(t)$	Power allocation vector	$\Delta(L(\cdot))$	Lyapunov drift
γ_i	Allowed average power consumption for user i	$\boldsymbol{\alpha}(t)$	Packet arrival indicator vector

rate represents the average number of dropped packets per time slot. The average power consumption represents the average of transmit power over all time slots. The throughput represents the average served packets per time slot for each user $u \in \mathcal{U}$.

These metrics are connected and we will show in the following sections how the average power consumption affects the packet drop rate and the throughput.

3. PROBLEM FORMULATION

Our goal is to achieve the minimum drop rate for deadline-constrained users while providing a minimum throughput for each user $u \in \mathcal{U}$ and keeping the average power consumption for every user below a threshold. To this end, we provide the following stochastic optimization problem

$$\min_{\mathbf{p}(t)} \quad \sum_{r \in \mathcal{R}} \overline{D}_r \qquad (8)\text{a}$$

$$\text{s. t.} \quad \overline{p}_i \leq \gamma_i, \forall i \in \mathcal{N}, \qquad (8)\text{b}$$

$$\overline{\mu}_u \geq \delta_u, u \in \mathcal{U}, \qquad (8)\text{c}$$

$$\mathbf{p}(t) \in \mathcal{P}(t), \qquad (8)\text{d}$$

where $\gamma_i \in \left[0, P^{(\text{High})}\right]$ indicates the allowed average power consumption for each user i. Also, δ_u denotes the minimum throughput requirement for each user $u \in \mathcal{U}$. The constraint in *(8)*b ensures that average power consumption of each user i remains below γ_i power units.

The formulation above represents our intended goal which is the minimization of the packet drop rate. However, the objective function in *(8)*a makes the solution approach non-trivial. The decision variable, $\mathbf{p}(t)$ (power allocation), is optimized slot-by-slot for minimization of the objective function that is defined over an infinite time horizon. We have to cope with one critical point: we do not have prior knowledge about the future states of the channel and packet arrivals in the system. Therefore, we are not able to predict the values of the objective function in the future slots in order to decide on the power allocation that minimizes the cost. We aim to design a function whose future values are affected by the current decision and the remaining expiration time of the packets. To this end, we introduce a function incorporating the relative difference between the packet deadline m_r and the number of remaining future slots $(d_r(t) - 1)$ before its expiration as described below

$$f_r(t) \triangleq \frac{m_r - (d_r(t) - 1)}{m_r} \mathbf{1}_{\{\mu_r(t)=0\}}, \forall r \in \mathcal{R}. \qquad (9)$$

The function in *(9)* takes its extreme value, $f_r(t) = 0$, when a packet of user $r \in \mathcal{R}$ is served, or $f_r(t) = 1$ when a packet of user $r \in \mathcal{R}$ is dropped. Therefore, that function takes the same values with those of *(5)* in the extreme cases. In addition, the function in *(9)* assigns the cost according to the remaining time of a packet to expire in the intermediate states, i.e., when a packet is waiting in the queue. The cost increases when there is less time left for serving the packet with respect to the defined deadline. The time average of $f_r(t)$ is

$$\overline{f_r} \triangleq \lim_{t \to \infty} \overline{f}_r(t), \forall r \in \mathcal{R}, \qquad (10)$$

where $\overline{f}_r(t) \triangleq \frac{1}{t} \sum_{\tau=0}^{t-1} f_r(\tau)$ and

$$f = \sum_{r \in \mathcal{R}} f_r(t). \qquad (11)$$

Finally, we formulate a minimization problem by using (10) as shown below

$$\min_{\boldsymbol{p}(t)} \quad \sum_{r \in \mathcal{R}} \overline{f}_r \qquad (12)\text{a}$$

$$\text{s. t.} \quad \overline{p}_i \leq \gamma_i, \forall i \in \mathcal{N}, \qquad (12)\text{b}$$

$$\overline{\mu}_u \geq \delta_u, u \in \mathcal{U}, \qquad (12)\text{c}$$

$$\boldsymbol{p}(t) \in \mathcal{P}(t). \qquad (12)\text{d}$$

4. PROPOSED APPROXIMATE SOLUTION

The problem in *(12)* includes time average constraints. In order to satisfy these constraints, we aim to develop a policy that uses techniques different from standard optimization methods based on static and deterministic models. Our approach is based on Lyapunov optimization theory [34].

In particular, we apply the technique developed in [35] and further discussed in [34] and [36] in order to develop a policy that ensures that the constraints in *(12)*b and *(12)*c are satisfied.

Each inequality constraint in *(12)*b and *(12)*c is mapped to a virtual queue. We show below that the power constraint and minimum throughput constraints problems are transformed into queue stability problems.

Before describing the motivation behind the mapping of average constraints in *(12)*b and *(12)*c to virtual queues, let us recall one basic theorem that comes from the general theory of stability of stochastic processes [37]. Consider a system with K queues. The number of unfinished jobs of queue i is denoted by $q_k(t)$, and $\mathbf{q}(t) = \{q_k(t)\}_{k=1}^{K}$. The Lyapunov function and the Lyapunov drift are denoted by $L(\mathbf{q}(t))$ and $\Delta(L(\mathbf{q}(t))) \triangleq E\{L(\mathbf{q}(t+1)) - L(\mathbf{q}(t))|\mathbf{q}(t)\}$ respectively [37]. Below we provide the definition of the Lyapunov function [37].

Definition 1 (Lyapunov function): A function $L : \mathbb{R}^K \to \mathbb{R}$ is said to be a Lyapunov function if it has the following properties

- $L(\mathbf{x}) \geq 0, \forall \mathbf{x} \in \mathbb{R}^K$,

- It is non-decreasing in any of its arguments,

- $L(\mathbf{x}) \to +\infty$, as $||\mathbf{x}|| \to +\infty$.

Theorem 1 (Lyapunov Drift). *If there exist positive values B, ϵ such that for all time slots t we have $\Delta(L(\mathbf{q}(t)) \leq B - \epsilon \sum_{k=1}^{K} q_k(t)$, then the system $\mathbf{q}(t)$ is strongly stable.*

The intuition behind Theorem 1 is that if we have a queueing system, and we provide a scheduling scheme such that the Lyapunov drift is bounded and the sum of the length of the queues are multiplied by a negative value, then the system is stable. Our goal is to find a scheduling scheme for which the inequality of Theorem 1 holds for our application.

Let $\{X_i(t)\}_{i \in \mathcal{N}}$ and $\{Z_u(t)\}_{u \in \mathcal{U}}$ be the virtual queues associated with constraints *(12)*b and *(12)*c, respectively.

We update each virtual queue $X_i(t)$ at each time slot t as

$$X_i(t+1) = \max\left[X_i(t) - \gamma_i, 0\right] + p_i(t), \qquad (13)$$

and each virtual queue $Z_u(t)$ as

$$Z_u(t+1) = \max\left[Z_u(t) - \mu_u(t), 0\right] + \delta_u. \qquad (14)$$

Process $X_i(t)$ can be viewed as a queue with "arrivals" $p_i(t)$ and "service rate" γ_i. Process $Z_u(t)$ can be also viewed as a queue with "arrivals" δ_u and "service rate" $\mu_u(t)$.

We will show that the average constraints in *(12)*b and *(12)*c are transformed into queue stability problems. Then, we develop a dynamic algorithm and we prove that the algorithm satisfies Theorem 1 and achieves stability.

Lemma 1. *If $X_i(t)$ and $Z_u(t)$ are rate stable[1], then the constraints in (12)b and (12)c are satisfied.*

Proof. See Appendix B. □

Note that strong stability implies all of the other forms of stability [34, Chapter 2] including the rate stability. Therefore, the problem is transformed into a queue stability problem. In order to stabilize the virtual queues $X_i(t), \forall i \in \mathcal{N}$ and $Z_u(t), \forall u \in \mathcal{U}$, we first define the Lyapunov function as

$$L(\boldsymbol{\Theta}(t)) = \frac{1}{2} \sum_{i \in \mathcal{N}} X_i^2(t) + \frac{1}{2} \sum_{u \in \mathcal{R}} Z_u^2(t), \qquad (15)$$

where $\boldsymbol{\Theta}(t) = [\{X_i(t)\}_{i \in \mathcal{N}}, \{Z_u(t)\}_{u \in \mathcal{U}}]$, and the Lyapunov drift as

$$\Delta(\boldsymbol{\Theta}(t)) \triangleq \mathbb{E}\left\{L(\boldsymbol{\Theta}(t+1)) - L(\boldsymbol{\Theta}(t))|\boldsymbol{\Theta}(t)\right\}. \qquad (16)$$

The above conditional expectation is with respect to the random channel states and the arrivals.

To minimize the time average of the desired cost $f_r(t)$ while stabilizing the virtual queues $X_i(t), \forall i \in \mathcal{N}$, $Z_u(t), \forall u \in \mathcal{U}$, we use the *drift-plus-penalty* minimization approach introduced in [36]. The approach seeks to minimize an upper bound on the following drift-plus-penalty expression at every slot t

$$\Delta(\boldsymbol{\Theta}(t)) + V \sum_{r \in \mathcal{R}} \mathbb{E}\left\{f_r(t)|\boldsymbol{\Theta}(t)\right\}, \qquad (17)$$

where $V > 0$ is an "importance" weight to scale the penalty. An upper bound for the expression in *(17)* is shown below

$$\Delta(\boldsymbol{\Theta}(t)) + V \sum_{r \in \mathcal{R}} \mathbb{E}\{f_r(t)|\boldsymbol{\Theta}(t)\} \leq B$$

$$+ \sum_{i \in \mathcal{N}} \mathbb{E}\{X_i(t)(p_i(t) - \gamma_i)|\boldsymbol{\Theta}(t)\}$$

$$+ \sum_{r \in \mathcal{R}} \mathbb{E}\{Z_u(t)(\delta_u - \mu_u(t))|\boldsymbol{\Theta}(t)\}$$

$$+ V \sum_{r \in \mathcal{R}} \mathbb{E}\{f_r(t)|\boldsymbol{\Theta}(t)\}, \qquad (18)$$

---

[1]A discrete time process $Q(t)$ is *rate stable* if $\lim_{t \to \infty} \frac{Q(t)}{t} = 0$ with probability 1 [34].

where $B < \infty$ and $B \geq \frac{1}{2} \sum\limits_{i \in \mathcal{N}} \mathbb{E}\{p_i^2(t) + \gamma_i^2(t)|\Theta(t)\} + \frac{1}{2} \sum\limits_{r \in \mathcal{R}} \mathbb{E}\{\delta_r^2 + \mu_r^2(t)|\Theta(t)\} + \frac{1}{2} \sum\limits_{r \in \mathcal{R}} \mathbb{E}\{\alpha_r^2(t) + \mu_r^2(t)|\Theta(t)\}$. The complete derivation of the above bound can be found in Appendix A.

4.1 Min-Drift-Plus-Penalty Algorithm

We observe that the power allocation decision at each time slot does not affect the value of B. The minimum DPC algorithm observes the virtual queue backlogs of the virtual queues, the actual queue, and the channel states and makes a control action to solve the following optimization problem

$$
\min_{\boldsymbol{p}(t)} \quad \sum_{i \in \mathcal{N}} X_i(t)(p_i(t) - \gamma_i) + \sum_{r \in \mathcal{R}} Z_u(t)(\delta_u - \mu_u(t))
$$
$$
+ V \sum_{r \in \mathcal{R}} f_r(t) \tag{19a}
$$
$$
\text{s. t.} \quad \boldsymbol{p}(t) \in \mathcal{P}(t). \tag{19b}
$$

Lemma 2. *The optimal solution to problem (19) minimizes the upper bound of the drift-plus-penalty expression given in the right-hand-side of (18).*

Proof. See Appendix C. □

Theorem 2 (Optimality of DPC algorithm and queue stability). *The DPC algorithm guarantees that the virtual and the actual queues are strongly stable and therefore, according to Lemma 1, the time average constraints in (12)b and (12)c are satisfied. In particular, the time average expected value of the queues is bounded as*

$$
\lim_{t \to \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \left(\sum_{i \in \mathcal{N}} \mathbb{E}\{X_i(t)\} + \sum_{u \in \mathcal{U}} \mathbb{E}\{Z_u(t)\} \right) \leq
$$
$$
\frac{B + V\left(f^*(\epsilon) - f^{opt}\right)}{\epsilon}. \tag{20}
$$

In addition, the expected time average of function $f(t)$ is bounded as

$$
\lim_{t \to \infty} \sup \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\{f(\tau)\} \leq f^{opt} + \frac{B}{V}. \tag{21}
$$

Proof. See Appendix D. □

We summarize the steps of the DPC algorithm that solves the power control problem in *(19)* in Algorithm 1.

Algorithm 1: DPC

---
1 Input constant V, Initialization:
     $X_i(0) = 0, \gamma_i, \forall i \in \mathcal{N}, Z_u(0) = 0, \forall u \in \mathcal{U}.$
2 **for** $t = 1 : \ldots$ **do**
3 | $MinObj \leftarrow \infty$
4 | **for** $i = 1 : (|\mathcal{N}| + 1)$ **do**
5 | | $p_i(t) \in \mathcal{P}(t)$, Calculate $f_r(t), \forall r \in \mathcal{R}$
6 | | $Obj \leftarrow V \sum\limits_{r \in \mathcal{R}} f_r(t) + \sum\limits_{r \in \mathcal{R}} X_r(t)(p_i(t) - \gamma_r)$
   | | $+ \sum\limits_{u \in \mathcal{U}} Z_u(t)(\delta_u - \mu_i(t))$
7 | | **if** *MinObj>Obj* **then**
8 | | | $\mathbf{p}'(t) \leftarrow \mathbf{p}(t)$
9 | | | $MinObj \leftarrow Obj$
10 | $\mathbf{p}(t) \leftarrow \mathbf{p}'(t)$
11 | $X_j(t+1) \leftarrow \max\left[X_j(t) - \gamma_j, 0\right] + p_j(t), \forall j \in \mathcal{N}$
12 | $Z_u(t+1) = \max\left[Z_u(t) - \mu_u(t), 0\right] + \delta_u, \forall u \in \mathcal{U}$

In step 1, we initialize the importance factor V and the length of virtual queues $X_i(0), \forall i \in \mathcal{N}$, and $Z_u(0), \forall u \in \mathcal{U}$. We try all the possible power allocations from the set $\mathcal{P}$ in step 4, and we find the corresponding value of the objective function in step 6. In step 7, we check if the candidate power allocation gives a smaller value of the objective so far. In steps $11 - 12$, we updated the virtual queues. After the search, we obtain the solution, $\boldsymbol{p}'(t)$, for which the objective function takes its minimum value, $MinObj$.

5. SIMULATION RESULTS

In this section, we provide results that show the performance of the DPC algorithm regarding the packet drop rate and the convergence time for the time average constraints, i.e, throughput, and average power consumption constraints. We use a MATLAB environment to perform our simulations. For the time average performance, we run each algorithm for 10^6 slots.

We first show the results of a system with two users. Our goal is to show the performance of DPC for different values of the importance factor V. Second, we compare our proposed algorithm with a baseline algorithm called LDF algorithm proposed in [3].

5.1 Performance and convergence of DPC algorithm for different values of V

In Fig. 2, we provide results for a system with two users; user 1: deadline-constrained user, user 2: user with minimum-throughput requirements. The average power thresholds are $\gamma_1 = 0.7$ and $\gamma_2 = 0.65$ for user 1 and user 2, respectively. The minimum-throughput requirements for user 2 is $\delta_2 = 0.4$ packets/slot, and the deadlines of the packet of user 1 is $m = 10$ slots.

The probability of the channel being in "Good" state and in "Bad" state is 0.4 and 0.6, respectively. The high level power and the low level power is $P^{\text{High}} = 2$ power units

(a) Throughput-constraints convergence of user 2.

(b) Average power consumption convergence of user 1.
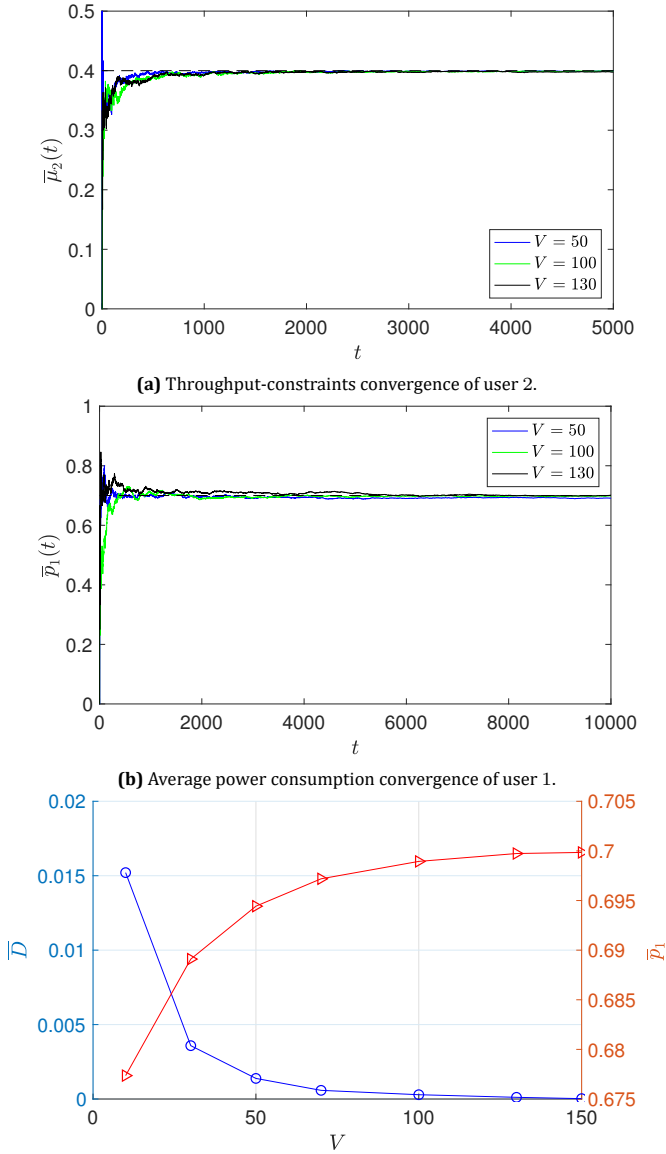
(c) Trade-off between packet drop rate and average power consumption of user 1 for different values of V.

Fig. 2 – DPC algorithm performance for different values of V. $\lambda_1 = 0.5$, $\gamma_1 = 0.7, \gamma_2 = 0.65, \delta_2 = 0.4, m = 10$. and $P^{\text{Low}} = 1$ power units, respectively.

In Fig. 2a, we show the convergence of the algorithm regarding the minimum-throughput constraints for different values of the importance factor V. We observe that as the value of V increases, the time convergence increases as well. However, we observe that after approximately 2500 slots, the algorithm converges and the minimum-throughput requirements are satisfied. In Fig. 2b, we provide results for the converge of the DPC algorithm regarding the average power consumption constraints of user 1. The algorithm converges after approximately 8000 slots for each value of V. The probability of the channel of user 1 being in "Good" state is 0.4. Therefore, the user needs to transmit with a high power level for a large portion of the time and that affects the average power consumption. Thus, the average power consumption constraint with $\gamma_2 = 0.7$ is a tight constraint and the algorithm needs

more time to converge.

In Fig. 2c, we show the trade-off between the packet drop rate and average power consumption of user 1. We observe that as the value of V increases the average power consumption increases and approaches threshold γ_1. For $V = 10$, we observe that the average power consumption is far from the threshold. In this case, the value of the virtual queue that corresponds to the average power consumption is larger than the cost function for a large period of time because the importance factor is relatively small. Therefore, the DPC seeks to minimize the larger term of the objective function that is the value of the virtual queue. On the other hand, as we increase V, the cost function is weighted more and therefore, the DPC algorithm seeks to minimize the cost function which is the most weighted term in the objective function. However, the average power consumption remains always below threshold γ_1.
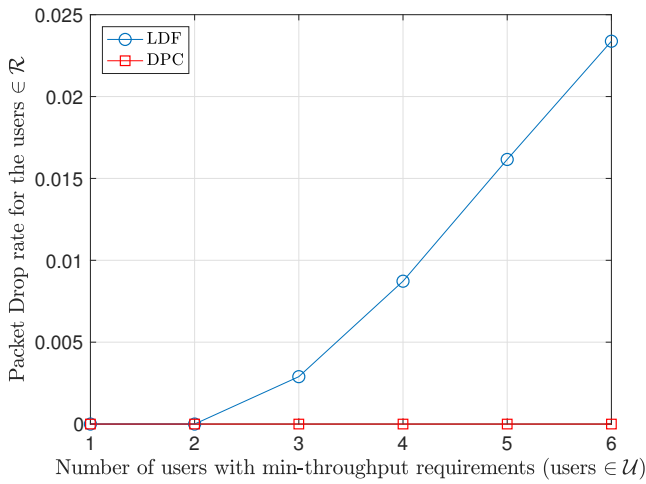
5.2 DPC vs LDF

In this subsection, we compare the performance of our algorithm with that of LDF. The LDF algorithm allocates power to the user with the *largest-debt*. Algorithm LDF selects, at each time slot t, the node with the highest value of $y_i(t)$, where $y_i(t)$ is the throughput debt and is defined as

$$y_i(t+1) = tq_i - \sum_{\tau}^{t} \mu_i(t), \qquad (22)$$

where q_i is the throughput requirements for user i. In our case, for the users with throughput requirements, $q_i = \delta_i$. For the deadline-constrained users, q_i will be equal to the percentage of the desired served packets for users with deadlines. For example, if our goal is to achieve a zero drop rate we set $q_i = \lambda_i$. However, this is not always feasible, i.e., zero drop rate and satisfaction of the throughput constraints. Therefore, in this case, we get a higher drop rate and lower throughput. Note that the LDF algorithm does not account for the average power constraints. It was shown in [3] that LDF is throughput optimal when the problem is feasible for systems with users with throughput requirements.

In this set-up, we consider one user with packets with deadlines and a set with multiple users with minimum-throughput requirements. The probability that the channel is in "good" state is equal to 0.9 for all the users. In order to observe a fair comparison between the algorithm, we consider that the average power threshold is 2 for all the users. Therefore, the average power constraint for every user is always satisfied. The arrival rate for user 1 is $\lambda_1 = 0.35$ packets/slot.

In Fig. 3, we compare the performance of the algorithms regarding the packet drop rate as the number of users with minimum-throughput requirements increases. In Fig. 3a, the deadline for the packets of user 1 is $m = 10$. We observe that the DPC algorithm outperforms the
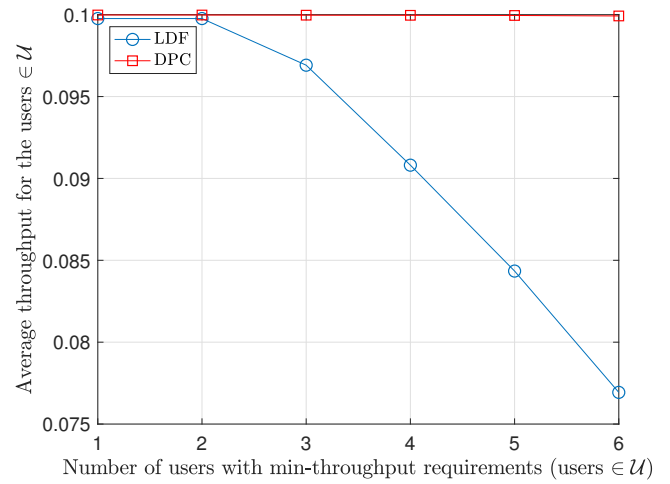
(a) Packet deadline: $m = 10$.
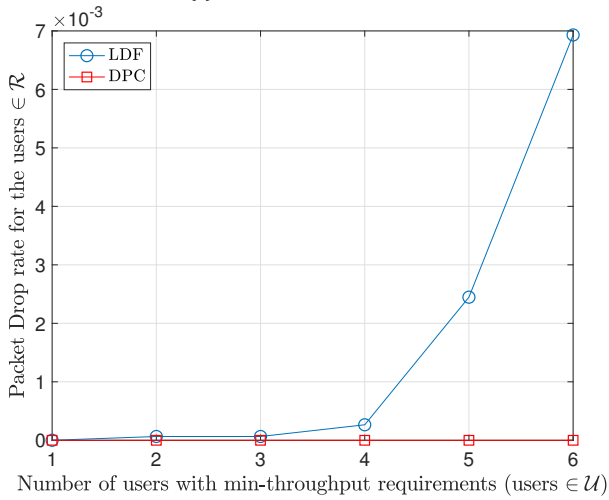
(b) Packet deadline: $m = 30$.

Fig. 3 – DPC vs LDF. Drop rate comparison for different values of packet deadline m.
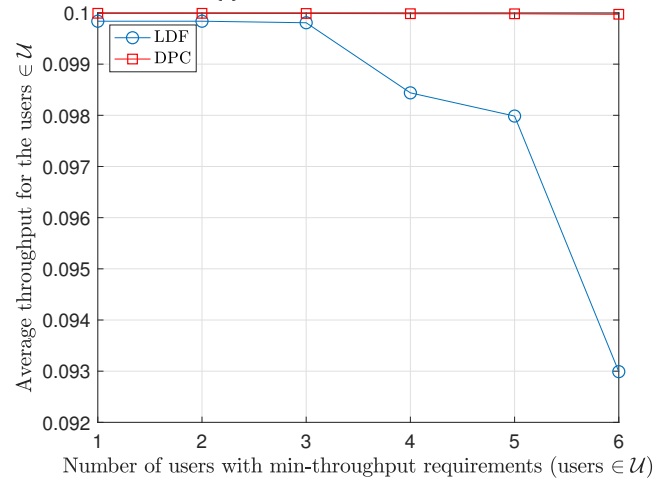

(a) Packet deadline: $m = 10$.

(b) Packet deadline: $m = 30$.

Fig. 4 – DPC vs LDF. Throughput comparison for different values of packet deadline m.
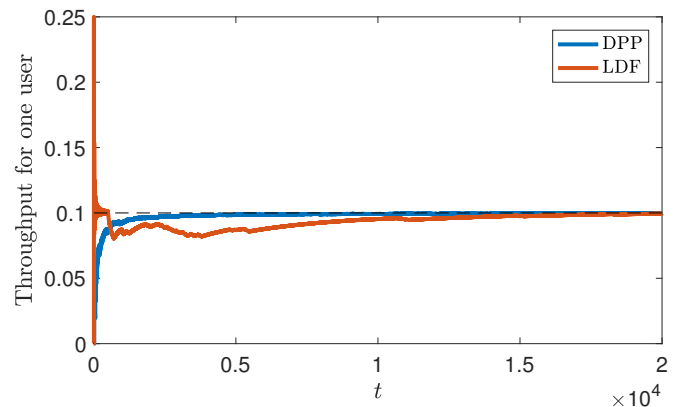

Fig. 5 – DPC vs LDF. Min-throughput requirements convergence. Packets deadline: $m = 100$ slots.

LDF algorithm in terms of drop rate as the number of minimum-throughput requirements increases. In Fig. 3b, the deadline for the packets of user 1 is $m = 30$. We observe that the performance of LDF has been improved. However, the DPC outperforms LDF in this case as well. We observe that LDF is more sensitive on the size of the deadline of the packets.

In Fig. 4, we compare the performance of the algorithm regarding the average total throughput of users with minimum-throughput requirements. In Fig. 4a, we show results for packets deadline that is $m = 10$. Also in this case, we observe that the DPC algorithm outperforms the LDF. However, for larger deadlines, the performance of the LDF is improved, as shown in Fig. 4b.

In Fig. 5, we provide results for the convergence time of throughput requirements of one user. In this set-up, we consider a system with one user with packets with deadlines and six users with minimum-throughput requirements. In the previous results that are shown in Figures 4 and 3, we observe that as we increase the value of m, we get a better performance of the LDF algorithm. In the system of which the results are shown in Fig. 5, we set a very large value to the deadlines of

user 1. We observe that the DPC algorithm converges much earlier than the LDF algorithm. However, both algorithms converge after many slots. This explains the phenomenon of worst performance of LDF for small values of m. Since the LDF algorithm allocates power to the users with largest throughput-debt, it does not consider the remaining slots for each packet. Therefore, when the deadline, m, is small, the packets expire before the algorithm

converges in terms of throughput requirements and the drop rate increases.

6. CONCLUSIONS AND FUTURE WORK

In this work, we considered heterogeneous traffic with two sets of users. The first set contains users with packets with deadlines, and the second contains users with minimum-throughput requirements, all with a limited power budget. We considered the packet drop rate minimization with minimum-throughput guarantees. A dynamic algorithm was provided that solves the scheduling problem in real time. We proved that our scheduling scheme provides a solution arbitrarily close to the optimal. Simulation results show that the proposed algorithm outperforms the baseline algorithm, LDF, when the deadlines are short, and it is faster in terms of convergence. An interesting direction, for future work, would be the assumption that packets of the same user can have different deadlines, and there are multiple power levels.

7. ACKNOWLEDGMENTS

Appendices

A. UPPER BOUND ON THE LYAPUNOV DRIFT OF DPC

Using the fact that $(\max[Q - b, 0] + A)^2 \leq Q^2 + A^2 + b^2 + 2Q(A - b)$, we rewrite *(13)*, *(14)*, as

$$X_i^2(t+1) \leq X_i^2(t) + p_i^2 + \gamma_i^2 + 2X_i(t)(p_i(t) - \gamma_i), \quad (23)$$

$$Z_u^2(t+1) \leq Z_u^2(t) + \delta_u^2 + \mu_u^2(t) + 2Z_u(t)(\delta_u - \mu_u(t)), \quad (24)$$

respectively. Rearranging the terms in *(23)* and *(24)*, dividing them by two, and taking the summations, we obtain

$$\sum_{i \in \mathcal{N}} \frac{X_i^2(t+1) - X_i^2(t)}{2} \leq \sum_{i \in \mathcal{R}} \frac{p_i^2(t) + \gamma_i^2}{2} + \sum_{i \in \mathcal{N}} X_i(t)(p_i(t) - \gamma_i), \quad (25)$$

$$\sum_{u \in \mathcal{U}} \frac{Z_u^2(t+1) - Z_u^2(t)}{2} \leq \sum_{u \in \mathcal{U}} \frac{\delta_u^2 + \mu_u^2(t)}{2} + \sum_{u \in \mathcal{U}} Z_u(t)(\delta_u - \mu_u(t)). \quad (26)$$

Taking the conditional expectations in *(25)* and *(26)*, and adding them together, we obtain the bound for the Lyapunov drift in *(18)*. To prove that B is bounded, we have to find an example and a scheduling scheme for which B

takes its maximum value that is bounded. We consider the following set-up:

- The scheduler allocates power at every time slot with the maximum power level,

- $\gamma_i = P^{\text{High}}, \forall i \in \mathcal{N}$,

- $\delta_u = 1, \forall u \in \mathcal{U}$,

- $\lambda_i = 1, \forall i \in \mathcal{N}$.

Then, for the above scheduling scheme, we set $B = \frac{1}{2} \sum_{i \in \mathcal{N}} \mathbb{E}\{p_i^2(t) + \gamma_i^2(t)|\boldsymbol{\Theta}(t)\} + \frac{1}{2} \sum_{r \in \mathcal{R}} \mathbb{E}\{\delta_r^2 + \mu_r^2(t)|\boldsymbol{\Theta}(t)\} + \frac{1}{2} \sum_{r \in \mathcal{R}} \mathbb{E}\{\alpha_r^2(t) + \mu_r^2(t)|\boldsymbol{\Theta}(t)\} = \frac{1}{2}|\mathcal{N} + 1|(P^{\text{High}})^2 + \frac{1}{2}|\mathcal{R} + 1| + \frac{1}{2}|\mathcal{R} + 1| < \infty$. We observe that even in the scenario in which we take the maximum values of γ_i, δ_u, and λ_i, B is bounded.

B. PROOF OF LEMMA 1

Proof. Using the basic sample property [34, Lemma 2.1, Chapter 2], we have

$$\frac{X_i(t)}{t} - \frac{X_i(0)}{t} \geq \frac{1}{t}\sum_{\tau=0}^{t-1} p_i(\tau) - \frac{1}{t}\sum_{\tau=0}^{t-1} \gamma_i, \quad (27)$$

$$\frac{Z_u(t)}{t} - \frac{Z_u(0)}{t} \geq \frac{1}{t}\sum_{\tau=0}^{t-1} \delta_u - \frac{1}{t}\sum_{\tau=0}^{t-1} \mu_u(t). \quad (28)$$

Therefore, if $X_i(t)$ and $Z_u(t)$ are rate stable [2], so that $\frac{X_i(t)}{t} \to 0, \forall i \in \mathcal{N}$, and $\frac{Z_u(t)}{t} \to 0, \forall u \in \mathcal{U}$, with probability 1, then constraints *(12)*b and *(12)*c are satisfied with probability 1 [38]. $\square$

C. PROOF OF LEMMA 2

Proof. Let $\mathbf{p}(t)$ represent any, possibly randomized, power allocation decision made at slot t. Suppose that $\mathbf{p}^*(t)$ is the optimal solution to problem *(19)*, and under action $\mathbf{p}^*(t)$ the value of $f_i(t)$ yields $f_i^*(t)$ and that of $\mu_u(t), \mu_u^*(t)$. Then, we have

$$V f^*(t) + \sum_{i \in \mathcal{N}} X_i(t)(p^*(t) - \gamma_i) + \sum_{u \in \mathcal{U}} Z_u(t)(\delta_u - \mu_u^*(t))$$
$$\leq V f(t) + \sum_{i \in \mathcal{N}} X_i(t)(p(t) - \gamma_i) + \sum_{u \in \mathcal{U}} Z_u(t)(\delta_u - \mu_u(t)). \quad (29)$$

---

[2] A discrete time process $Q(t)$ is strongly stable if $\limsup_{t \to \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\{|Q(\tau)|\} < \infty$, [34].

Taking the conditional expectations of *(29)*, we have the result as

$$V\mathbb{E}\{f^*(t)|\Theta(t)\} + \sum_{i\in\mathcal{N}} \mathbb{E}\{X_i|\Theta(t)\}(t)(p_i^*(t)-\gamma_i)$$

$$+ \sum_{u\in\mathcal{U}} \mathbb{E}\{Z_u(t)|\Theta(t)\}(\delta_u - \mu_u^*(t)) \leq$$

$$V\mathbb{E}\{f(t)|\Theta(t)\} + \sum_{i\in\mathcal{N}} \mathbb{E}\{X_i|\Theta(t)\}(t)(p_i(t)-\gamma_i)$$

$$+ \sum_{u\in\mathcal{U}} \mathbb{E}\{Z_u(t)|\Theta(t)\}(\delta_u - \mu_u(t)).$$

$\square$

D. PROOF OF THEOREM 2

Proof. Suppose that a feasible policy ω exists, i.e., constraints *(12)*b and *(12)*c are satisfied. Furthermore, assume that, for the ω policy, the following holds

$$\mathbb{E}\{p_i(t)-\gamma_i\} \leq -\epsilon, \qquad (30)$$

$$\mathbb{E}\{\delta_u - \mu_u(t)\} \leq -\epsilon, \qquad (31)$$

$$\mathbb{E}\{f^*(\epsilon)\} = f^*(\epsilon), \qquad (32)$$

where $f^*(\epsilon)$ is a sub-optimal solution. Applying *(30)* and *(31)* into *(18)*, we obtain

$$\mathbb{E}\{L(\Theta(t+1))\} - \mathbb{E}\{L(\Theta(t))\} + V\mathbb{E}\{f(t)\} \leq$$

$$B - \epsilon\left(\sum_{i\in\mathcal{N}} \mathbb{E}\{X_i(t)\} + \sum_{u\in\mathcal{U}} \mathbb{E}\{Z_u(t)\}\right) + Vf^*(\epsilon),$$

taking $\epsilon \to 0$ and the sum over $\tau = 0, \ldots, t-1$ we obtain

$$\frac{1}{t}\sum_{\tau=0}^{t-1} \mathbb{E}\{f(\tau)\} \leq \frac{-\mathbb{E}\{L(\Theta(t))\} + \mathbb{E}\{L(\Theta(0))\} + Bt}{Vt}$$

$$+ f^{\text{opt}}, \qquad (33)$$

taking $t \to \infty$, we obtain

$$\lim_{t\to\infty} \sup \frac{1}{t}\sum_{\tau=0}^{t-1} \mathbb{E}\{f(\tau)\} \leq f^{\text{opt}} + \frac{B}{V}. \qquad (34)$$

That concludes the second part of Theorem 2. In order to prove the stability of the queues, we manipulate *(33)*

$$\left(\sum_{i\in\mathcal{N}} \mathbb{E}\{X_i(t)\} + \sum_{u\in\mathcal{U}} \mathbb{E}\{Z_u(t)\}\right) \leq$$

$$\frac{B}{\epsilon} - \frac{\mathbb{E}\{L(\Theta(t+1))\} - \mathbb{E}\{L(\Theta(t))\}}{\epsilon} - \frac{V(f^*(\epsilon)-f(t))}{\epsilon}. \qquad (35)$$

By taking the sum over $\tau = 0, \ldots, t-1$ and divide by t, we obtain

$$\frac{1}{t}\sum_{\tau=0}^{t-1}\left(\sum_{i\in\mathcal{N}} \mathbb{E}\{X_i(t)\} + \sum_{u\in\mathcal{U}} \mathbb{E}\{Z_u(t)\}\right) \leq$$

$$\frac{B}{\epsilon} - \frac{\mathbb{E}\{L(\Theta(t))\} - \mathbb{E}\{L(\Theta(0))\}}{t\epsilon} + \frac{V(f^*(\epsilon)-f(t))}{\epsilon}, \qquad (36)$$

neglecting the negative term and taking $t \to \infty$, we have

$$\lim_{t\to\infty} \frac{1}{t}\sum_{\tau=0}^{t-1}\left(\sum_{i\in\mathcal{N}} \mathbb{E}\{X_i(t)\} + \sum_{u\in\mathcal{U}} \mathbb{E}\{Z_u(t)\}\right) \leq$$

$$\frac{B + V(f^*(\epsilon)-f(t))}{\epsilon}. \qquad (37)$$

Consider that $\mathbb{E}\{f(t)\} \geq f^{\text{opt}}$, we obtain the final result as

$$\lim_{t\to\infty} \frac{1}{t}\sum_{\tau=0}^{t-1}\left(\sum_{i\in\mathcal{N}} \mathbb{E}\{X_i(t)\} + \sum_{u\in\mathcal{U}} \mathbb{E}\{Z_u(t)\}\right) \leq$$

$$\frac{B + V(f^*(\epsilon)-f^{\text{opt}})}{\epsilon}. \qquad (38)$$

This shows that the queues are strongly stable for $\epsilon > 0$.

$\square$

REFERENCES

[1] F. Alriksson, L. Boström, J. Sachs, E. Wang, and A. Zaidi, "Critical IoT connectivity," Ericsson review, vol. 102, pp. 52–64, 2020.

[2] I. Hou and P. R. Kumar, "Packets with deadlines: A framework for real-time wireless networks," Synthesis Lectures on Communication Networks, vol. 6, no. 1, pp. 1–116, 2013.

[3] I. Hou, V. Borkar, and P. R. Kumar, "A theory of QoS for wireless," in Proc. IEEE INFOCOM, pp. 486–494, 2009.

[4] S. Lashgari and A. S. Avestimehr, "Timely throughput of heterogeneous wireless networks: Fundamental limits and algorithms," IEEE Trans. Inf. Theory, vol. 59, no. 12, pp. 8414–8433, 2013.

[5] O. Holland, E. Steinbach, R. V. Prasad, Q. Liu, Z. Dawy, A. Aijaz, N. Pappas, K. Chandra, V. S. Rao, S. Oteafy et al., "The IEEE 1918.1 "tactile internet" standards working group and its standards," Proc. of the IEEE, vol. 107, no. 2, pp. 256–279, 2019.

[6] Y. Cui, V. K. N. Lau, R. Wang, H. Huang, and M. Shunqing Zhang, "A survey on delay-aware resource control for wireless systems–large deviation theory, stochastic lyapunov drift, and distributed stochastic learning," IEEE Trans. Inf. Theory, vol. 58, no. 3, pp. 1677–1701, Mar. 2012.

[7] N. Master and N. Bambos, "Power control for packet streaming with head-of-line deadlines," Performance Evaluation, vol. 106, pp. 1–18, 2016.

[8] N. Nomikos, N. Pappas, T. Charalambous, and Y.-A. Pignolet, "Deadline-constrained bursty traffic in random access wireless networks," in IEEE Proc. SPAWC, pp. 1–5, 2018.

[9] A. Faridi and A. Ephremides, "Distortion control for delay-sensitive sources," IEEE Trans. Inf. Theory, vol. 54, no. 8, pp. 3399–3411, 2008.

[10] E. Fountoulakis, T. Charalambous, N. Nomikos, A. Ephremides, N. Pappas, "Information Freshness and Packet Drop Rate Interplay in a Two-User Multi-Access Channel", to appear, in Proc. IEEE ITW, 2021.

[11] E. Fountoulakis, N. Pappas, Q. Liao, A. Ephremides, and V. Angelakis, "Dynamic power control for packets with deadlines," in IEEE Proc. GLOBECOM, pp. 1–6, 2018.

[12] M. Choi, J. Kim, and J. Moon, "Dynamic power allocation and user scheduling for power-efficient and delay-constrained multiple access networks," IEEE Trans. Wireless Commun, vol. 18, no. 10, pp. 4846–4858, 2019.

[13] N. Salodkar, A. Bhorkar, A. Karandikar, and V. S. Borkar, "An on-line learning algorithm for energy efficient delay constrained scheduling over a fading channel," IEEE J. Sel. Areas Commun., vol. 26, no. 4, pp. 732– 742, May 2008.

[14] M. Goyal, A. Kumar, and V. Sharma, "Power constrained and delay optimal policies for scheduling transmission over a fading channel," in Proc. IEEE INFOCOM, vol. 1, May 2003, pp. 311–320.

[15] A. Dua and N. Bambos, "Downlink wireless packet scheduling with deadlines," IEEE Trans. Mobile Comput., vol. 6, no. 12, pp. 1410–1425, Dec. 2007.

[16] A. Fu, E. Modiano, and J. N. Tsitsiklis, "Optimal transmission scheduling over a fading channel with energy and deadline constraints," IEEE Trans. Wireless Commun., vol. 5, no. 3, pp. 630–641, Mar. 2006.

[17] A. Ewaisha and C. Tepedelenlioglu, "Power control and scheduling under hard deadline constraints for on-off fading channels," in Proc. IEEE WCNC, March 2017, pp. 1–6.

[18] A. E. Ewaisha and C. Tepedelenlioglu, "Optimal power control and scheduling for real-time and non-real-time data," IEEE Trans. Vehic. Tech, vol. 67, no. 3, pp. 2727–2740, 2017.

[19] S. ElAzzouni, E. Ekici, and N. Shroff, "Is deadline oblivious scheduling efficient for controlling real-time traffic in cellular downlink systems?" in Proc. IEEE INFOCOM, pp. 49–58, 2020.

[20] K. S. Kim, C.-P. Li, I. Kadota, and E. Modiano, "Optimal scheduling of real-time traffic in wireless networks with delayed feedback," Allerton, pp. 1143–1149, 2015.

[21] M. J. Neely and S. Supittayapornpong, "Dynamic Markov decision policies for delay constrained wireless scheduling," IEEE Trans. Automatic Control, vol. 58, no. 8, pp. 1948–1961, 2013.

[22] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," in Proc. IEEE CDC, pp. 2130–2132, 1990.

[23] B. Li, R. Li, and A. Eryilmaz, "Throughput-optimal scheduling design with regular service guarantees in wireless networks," IEEE/ACM Trans. Net., vol. 23, no. 5, pp. 1542–1552, 2014.

[24] X. Lan, Y. Chen, and L. Cai, "Throughput-optimal H-QMW scheduling for hybrid wireless networks with persistent and dynamic flows," IEEE Trans. Wireless Commun., vol. 19, no. 2, pp. 1182– 1195, 2019.

[25] B. Sadiq and G. De Veciana, "Throughput optimality of delay-driven maxweight scheduler for a wireless system with flow dynamics," Allerton, pp. 1097–1102, 2009.

[26] L. You, Q. Liao, N. Pappas, and D. Yuan, "Resource optimization with flexible numerology and frame structure for heterogeneous services," IEEE Commun. Letters, vol. 22, no. 12, pp. 2579–2582, 2018.

[27] 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 Proc. WiOpt, May 2017, pp. 1–6.

[28] A. Anand, G. De Veciana, and S. Shakkottai, "Joint scheduling of urllc and embb traffic in 5G wireless networks," IEEE/ACM Trans. Net., vol. 28, no. 2, pp. 477–490, 2020.

[29] 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, pp. 2411–2421, 2018.

[30] A. Karimi, K. I. Pedersen, and P. Mogensen, "Low-complexity centralized multi-cell radio resource allocation for 5g urllc," in Proc. IEEE WCNC, 2020, pp. 1–6.

[31] N. B. Khalifa, V. Angilella, M. Assaad, and M. Debbah, "Lowcomplexity channel allocation scheme for urllc traffic," IEEE Trans. Commun., 2020.

[32] A. Avranas, M. Kountouris, and P. Ciblat, "Throughput maximization and ir-harq optimization for urllc traffic in 5G systems," in Proc. IEEE ICC , 2019, pp. 1–6.

[33] A. Destounis, G. S. Paschos, J. Arnau, and M. Kountouris, "Scheduling urllc users with reliable latency guarantees," in Proc. WiOpt, 2018, pp. 1–8.

[34] M. J. Neely, Stochastic Network Optimization with Application to Communication and Queueing Systems. Morgan & Claypool, 2010.

[35] M. J. Neely, "Energy optimal control for time-varying wireless networks," IEEE Trans. Inf. Theory, vol. 52, no. 7, pp. 2915–2934, July 2006.

[36] 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.

[37] S. Meyn and R. L. Tweedie, Markov Chains and Stochastic Stability. 2nd ed. New York, NY, USA: Cambridge University Press, 2010.

[38] M. J. Neely, "Queue Stability and Probability 1 Convergence via Lyapunov Optimization," ArXiv e-prints, Aug. 2010. [Online]. Available: https://arxiv.org/abs/1008.3519

AUTHORS

Emmanouil Fountoulakis (Graduate Student Member, IEEE) received a Diploma degree in electronic and computer engineering from the Technical University of Crete, Chania, Greece, in 2016.

He is currently pursuing a Ph.D. degree with the Department of Science and Technology, Linköping University, Sweden. His research interests include age-of-information, low-latency communications, energy-efficient resource allocation, applied queueing theory, and stochastic control.

Nikolaos Pappas (Member, IEEE) received a B.Sc. degree in computer science, an M.Sc. degree in computer science, a B.Sc. degree in mathematics, and a Ph.D. degree in computer science from the University of Crete, Greece, in 2005, 2007, 2012, and 2012, respectively.

From 2005 to 2012, he was a Graduate Research Assistant with the Telecommunications and Networks Laboratory, Institute of Computer Science, Foundation for Research and Technology-Hellas, and a Visiting Scholar with the Institute of Systems Research, University of Maryland at College Park, College Park, MD, USA. From 2012 to 2014, he was a Post-Doctoral Researcher with the Department of Telecommunications, Supélec, France.

Since 2014, he has been with Linköping University, Norrköping, Sweden, as a Marie Curie Fellow (IAPP), where he is currently an Associate Professor in mobile telecommunications with the Department of Science and Technology. His main research interests include the field of wireless communication networks with emphasis on the stability analysis, energy harvesting networks, network-level cooperation, age-of-information, network coding, and stochastic geometry.

From 2013 to 2018, he was an Editor of IEEE COMMUNICATIONS LETTERS. He is currently an Editor of IEEE TRANSACTIONS ON COMMUNICATIONS, IEEE/KICS JOURNAL OF COMMUNICATIONS AND NETWORKS, and IEEE OPEN JOURNAL OF COMMUNICATIONS SOCIETY. He is also a Guest Editor of IEEE INTERNET OF THINGS JOURNAL for the Special Issue Age of Information and Data Semantics for Sensing, Communication and Control Co-Design in IoT.

Anthony Ephremides (Life Fellow, IEEE) received a Ph.D. degree in electrical engineering from Princeton University in 1971. Since 1971, he has been with the University of Maryland, College Park, MD, USA. He holds the Cynthia Kim Professorship of information technology at the Electrical and Computer Engineering Department, University of Maryland, where he is currently a Distinguished University Professor.

He has a joint appointment at the Institute for Systems Research, of which he was among the founding members, in 1986. He has authored several hundred articles, conference presentations, and patents.

His research interests lie in the areas of communication systems and networks and all related disciplines, such as information theory, control and optimization, satellite systems, queueing models, signal processing, and so on. He is especially interested in wireless networks, energy efficient systems, and the new notion of age of information.