Joint α-Fair Allocation of RAN and Computing Resources to Vehicular Users with URLLC Traffic

Valentin Thomas Haider*†, Fidan Mehmeti*, Ana Cantarero† and Wolfgang Kellerer*
*Chair of Communication Networks, Technical University of Munich, Germany, {firstname.lastname}@tum.de
†BMW Group, Munich, Germany, ana.cantarero@bmw.de

***Abstract*—5G networks have emerged as the only viable solution to render a satisfying level of performance to different types of services, each of them with very stringent traffic requirements. One of those services are Ultra-Reliable Low-Latency Communications (URLLC). A use case where these services are especially sensitive are vehicular networks. Therefore, in order to satisfy their traffic requirements, adequate resource allocation schemes should be devised. However, the time-varying nature of the channel conditions in wireless networks renders this process challenging. In this paper, we consider the problem of jointly allocating Radio Access Network (RAN) resources and computing resources (to process the data from vehicles) such that all the traffic requirements of individual users are met and the utility is maximized for different types of fairness. We formulate an optimization problem for the general case of α-fairness, explore its characteristics, and consider in more detail the opposite sides of fairness; the case of *no fairness* provided ($\alpha = 0$) and the *max-min* fair allocation ($\alpha \to \infty$). For each of these problems, we propose polynomial-time assignment heuristics. Using data from real traces, we show that the performance achieved with our approaches is not more than 1% away from the optimum.**

***Index Terms*—5G, Vehicular networks, URLLC, α-fairness.**

I. Introduction

There are three service types provided by 5G networks [1]: enhanced mobile broadband (eMBB), massive machine-type communications (mMTC), and ultra-reliable low-latency communications (URLLC). The services that are of mMTC type require support to serve a large number of devices and low energy consumption. Very high data rates with high spectral efficiency are needed for eMBB. Our focus in this paper are URLLC type of services with the focus on vehicular users.

URLLC corresponds to applications like autonomous driving, remote surgery, and remote monitoring and control [2]. Their main requirements are to deliver packets with a very high reliability within a short time (on the order of ms), which is quite challenging. Furthermore, besides being transmitted, those data need to be processed as well, which complicates handling the data even further. The challenge is even more emphasized given the constrained network resources in the cell, and the ever increasing number of entities competing for those resources. The aforementioned URLLC services are not only sensitive to abiding by those stringent requirements, but given their nature, any failure to comply may bring a serious risk to human lives. Hence, the paramount importance of enabling their flawless operation.

Enabling this impeccable operation is especially challenging in cellular networks, where the channel characteristics of users exhibit dynamic behavior over time due to mobility and processes like shadowing. Thus, to provide a given data rate and the adequate amount of resources to process the data that will satisfy those stringent requirements, a proper resource allocation scheme has to be used on two levels: on the Radio Access Network (RAN) side and on the analyst side (e.g., edge cloud) for computing. Furthermore, due to the presence of multiple users in the cell, the operator needs to allocate those resources in an *efficient way* in order to satisfy those requirements for as many users as possible.

There are some important questions that arise related to joint network and edge cloud resource allocation that provides fairness among the users. Firstly, what is the policy that enables achieving different types of fairness with joint allocation of both RAN and computing resources in the use case of vehicular networks, while satisfying all the pertinent traffic requirements? Secondly, how does the requirement on the maximum allowed delay affect the overall utility?

To answer the aforementioned questions, in this paper, we formulate an optimization problem, with which the goal is to provide general α-fairness, after meeting the allowed maximum latency of all users, and given the constrained resources on both the RAN and edge side. We then look in more depth at two types of fairness, and propose polynomial-time algorithms which provide near-optimal results. The assumptions we make in this work are very realistic, like assuming a user experiences different channel gains over different channel resources (blocks), irrespective on how close they are in the frequency dimension. The results we provide in this paper are particularly important for the network operator, as they indicate how resources should be allocated to increase the total utility, while providing certain types of fairness, and also to get an idea on the inter-play of the assignment of different types of resources so that the delay guarantee is met. The main message of this paper is that the overall utility depends on the delay constraint for the no fairness case, while it depends on the number of users for the max-min fairness. Specifically, our principal contributions are:

- We formulate the problem of joint allocation of network and edge computing resources as a Network Utility Maximization (NUM) problem and solve the integer-relaxed version for general α.
- As the original problem is NP-hard, we propose polynomial-time algorithms for $\alpha = 0$ and $\alpha \to \infty$ that provide near-optimal performance.
- We evaluate our approach using real data from measurements and provide some interesting engineering insights.

978-1-6654-9734-3/23/$31.00 ©2023 IEEE

The remainder of this work is organized as follows. In Section II we discuss some related work. This is followed by the system model and problem formulation in Section III. In Section IV, we provide the analysis, whereas in Section V approximation algorithms for the two extremes of fairness are proposed. We evaluate the performance in Section VI. Finally, Section VII concludes this work.

II. Related Work

In [3], the authors consider the network slicing process for the three types of services in 5G to determine the optimal amount of slices for each service type in order to satisfy service requirements. However, it is not mentioned what is the required data rate that needs to be provided to URLLC users to satisfy their latency requirement. A paper concerned with the uplink of URLLC is [4]. However, the objective in [4] is not to increase the utility, nor to provide any type of fairness. Moreover, the processing part is not considered either.

Further, the work in [5] considers the optimal transmission and resource allocation for URLLC in cellular systems. The resource allocation is derived for fixed and adaptive transmission attempt assignments. While [5] is also concerned with reducing the required resources, the setup and the objective are different from our work, and providing fairness is not one of the aims. To meet the latency and reliability requirements in 5G networks, the authors in [6] propose a periodic resource allocation scheme. Packet sizes are constant. However, the scope of [6] is limited as the environment is a factory, and providing any sort of fairness is not the objective.

The problem of admission control for URLLC traffic has previously been considered in [7], where the focus is on two scenarios. In the first, all the users undergo homogeneous traffic and channel conditions and the maximum number of users that can be admitted is determined. In the second scenario in [7], the users are characterized by heterogeneous traffic and channel conditions, and an admission policy is provided whether to admit a new user in the network or not. As opposed to our work here, [7] does not optimize any function.

Another related work is [8], in which there are three objectives, similar to our work: maximize the total throughput in the network, provide proportional fairness, and attain max-min fairness. However, in [8] the first goal is to provide a given constant data rate to everyone and then reallocate the unused resources to the users according to the respective policies that lead to the aforementioned objectives. There are some important differences between our work and [8] though. While our setup is related to URLLC traffic, the target of [8] are users with eMBB traffic. Satisfying the requirements of URLLC users is more challenging.

Finally, the authors of [9] analyze different questions on URLLC RAN resource allocation. While they define an optimization problem where the sum over users satisfying their Service-Level-Agreement (SLA) is maximized, they do not provide a solution to the problem but just an analysis of its NP-hardness. Additionally, they cover the problem of deciding whether a given set of users can be scheduled such that their SLA is fulfilled. They provide a feasible resource allocation in polynomial time. However, the given solution is not optimal and per-block rates are either zero or a fixed number, which is a simplified approach compared to our assumptions.

III. Problem Formulation

A. System Model

The possibility of network slicing in 5G [10] enables assigning *dedicated* network resources to the same type of service, e.g., users with URLLC type of traffic that have the same reliability and latency requirements and are located in the area covered by the same gNodeB. Throughout this paper, we assume that the users of interest in the cell belong to the same use case, and hence require the same service quality. 5G uses *Physical Resource Blocks (PRBs)* as the unit of allocation on a per-slot basis, where slots are grouped into frames with a length of 10 ms. The number of slots per frame depends on the subcarrier spacing [11].

We consider vehicular users within the coverage area of a 5G macro base station (gNodeB) in the sub-6 GHz band (Fig. 1). The focus is on the uplink and the processing is done at the edge (which we assume is collocated with the Base Station (BS)). The system consists of a single BS, multiple vehicular users, and edge computing resources. There are N users simultaneously requesting a service by sending a packet to the BS, where the sent information is processed. To enable the communication, there are K PRBs available in the RAN. Additionally, L edge computing resources are available for the receiving entity to process the information.

Channel conditions vary from one frame to another. Users experience different channel conditions, i.e., different Channel Quality Indicator (CQI) values, across different PRBs even within the same frame. There are 15 possible values of the CQI [11]. Because of the user's mobility and the time-varying nature of the channels, the per-PRB CQI (which is a function of Signal-to-Interference-Plus-Noise-Ratio (SINR)) also changes from one frame to another, whose value depending on the Modulation and Coding Scheme (MCS) used sets the per-PRB rate per frame. Hence, scheduling has to be performed across two dimensions, *time* and *frequency*. Since the focus is on URLLC traffic, the procedure of sending and processing the information must be executed within a maximum time of T_{max}.[1] Therefore, at least one PRB and one edge computing resource must be assigned to every user, as otherwise the delay constraint cannot be fulfilled. A PRB and also each edge computing resource can only be allocated to one user and can either be fully allocated or unassigned.

Packet size: We assume that the packet sizes, Δ, are fixed [12]. This is reasonable as we are considering services in which the data are organized in small packets [12].

Packet generation: Packets are generated periodically on a per frame basis.

[1]While the requirement for ultra-high reliability for this type of traffic translates into transmitting more than 99% of the packets successfully within the maximum latency, here we are even more conservative and require that all the packets have to be transmitted and processed within the deadline.

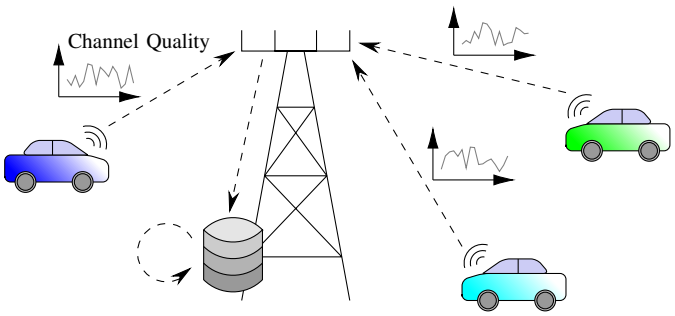

Fig. 1. Illustration of the system model.

B. Optimization Problem Formulation

In this paper, the goal is to maximize the utility over all users after satisfying their traffic requirements, taking into account the constrained RAN and computing resources. We focus on the general case of guaranteeing α-fairness, in the same spirit as the NUM approach [13]. We have the following optimization formulation:

$$\max_{\boldsymbol{I},\boldsymbol{m}} \quad \sum_{i=1}^{N} f_i^{\alpha}(\boldsymbol{I}_i, m_i) \tag{1a}$$

$$\text{s.t.} \quad \frac{\Delta}{\gamma_i} + \frac{\Delta}{m_i p} \leq T_{max}, \quad \forall i, \tag{1b}$$

$$\sum_{i=1}^{N} m_i \leq L, \tag{1c}$$

$$\sum_{i=1}^{N} I_{ij} \leq 1, \quad \forall j, \tag{1d}$$

$$\sum_{j=1}^{K} I_{ij} \geq 1, \quad \forall i, \tag{1e}$$

$$I_{ij} \in \{0,1\}, \quad \forall i, j, \tag{1f}$$

$$m_i \in \mathbb{N} \setminus \{0\}, \quad \forall i, \tag{1g}$$

where

$$f_i^{\alpha}(\boldsymbol{I}_i, m_i) = \\ = \begin{cases} \frac{1}{1-\alpha}\left((\gamma_i)^{1-\alpha} + (m_i p)^{1-\alpha}\right), & \alpha \neq 1 \\ \log(\gamma_i) + \log(m_i p), & \alpha = 1 \end{cases}, \tag{2}$$

and γ_i denotes $\sum_{j=1}^{K} I_{ij}\Phi_{ij}$. In the problem formulation, the decision variable $\boldsymbol{I} = \{I_{ij}\}$ denotes the $N \times K$ PRB allocation matrix in a given frame. Namely, if $I_{ij} = 1$, then PRB j is assigned to user i in that frame. The $N \times K$ matrix $\boldsymbol{\Phi} = \{\Phi_{ij}\}$ contains the data rates user i would experience when being allocated PRB j. It is derived from the CQI values that are reported for the users. The decision variable $\boldsymbol{m} = \{m_i\}$ is an $N \times 1$ vector consisting of the number of allocated edge computing resources per user i. The amount of data sent by each user at a time is Δ. Lastly, the parameter p denotes the processing rate that one edge computing resource can provide.

The objective (1a) maximizes the utility for general $\alpha \in [0, \infty)$. Note that $\alpha = 0$ corresponds to the case of *no-fairness*, whereas $\alpha \to \infty$ describes the *max-min* fairness. Apparently, as there are two types of resources to be allocated, they both affect the value of utility gained. In (2), the first term (both for $\alpha \neq 1$ and $\alpha = 1$) corresponds to the utility from assigning RAN resources to user i, while the second term denotes the utility after allocating a number of computing resources.

Constraint (1b) describes the maximum tolerable latency for every user. The finite amount of computing resources is captured by (1c). Constraint (1d) merely states that every block can be assigned to at most one user, whereas (1e) stipulates that every user needs to receive at least one PRB. Finally, (1f) and (1g) describe the integer nature of the decision variables, where the latter constraint includes the minimum number of one computing resource that must be assigned to every user.

IV. Analysis

The structure of the optimization problem described previously belongs to the class of Integer Nonlinear Programs, which are generally known to be NP-hard [14]. Therefore, we need some heuristics to obtain a solution to the aforementioned optimization problem.

The approach we propose in this work consists of two steps. First, we relax the requirement on the decision variables to be integer and show that under those circumstances the transformed optimization problem is convex and solvable in polynomial time. Then, in Section V, we describe the second step of the method, which shows how to obtain the integer solutions using special algorithms.

We proceed with the first step, showing the convex nature of the problem (1), when $I_{ij} \in [0, 1]$ and $m_i \in [1, \infty)$. As the constraints (1c)-(1g) are linear, they are obviously convex. To show that the objective function is concave, it must be shown that the function $f_i^{\alpha}(\boldsymbol{I}_i, m_i)$ is concave, as the sum of concave functions is a concave function itself. We have:

Lemma 1. *The function $f_i^{\alpha}(\boldsymbol{I}_i, m_i)$ is concave.*

Proof. The gradient of $f_i^{\alpha}(\boldsymbol{I}_i, m_i)$ for $\alpha \neq 1$ is

$$\nabla f_i^{\alpha}(\boldsymbol{I}_i, m_i) = \begin{bmatrix} \Phi_{i1}\gamma_i^{-\alpha} & \dots & \Phi_{iK}\gamma_i^{-\alpha} & p(m_i p)^{-\alpha} \end{bmatrix}^T.$$

Then, the Hessian matrix of $f_i^{\alpha}(\boldsymbol{I}_i, m_i)$ for $\alpha \neq 1$ is

$$\nabla^2 f_i^{\alpha}(\boldsymbol{I}_i, m_i) = \\ = -\alpha\gamma_i^{-\alpha-1} \begin{bmatrix} \Phi_{i1}^2 & \dots & \Phi_{i1}\Phi_{iK} & 0 \\ \vdots & \ddots & \vdots & \vdots \\ \Phi_{iK}\Phi_{i1} & \dots & \Phi_{iK}^2 & 0 \\ 0 & \dots & 0 & \frac{p^2(m_i p)^{-\alpha-1}}{\gamma_i^{-\alpha-1}} \end{bmatrix}.$$

The characteristic polynomial of $\nabla^2 f_i^{\alpha}(\boldsymbol{I}_i, m_i)$ for $\alpha \neq 1$ is

$$\det(\nabla^2 f_i^{\alpha}(\boldsymbol{I}_i, m_i) - \lambda\mathbb{I}) = \\ = (-1)^{K-1}\lambda^{K-1}\left(\alpha p^2 (m_i p)^{-\alpha-1} + \lambda\right) * \\ \left(\alpha\gamma_i^{-\alpha-1}\Phi_{i1}^2 + \dots + \alpha\gamma_i^{-\alpha-1}\Phi_{iK}^2 + \lambda\right),$$

where $\mathbb{I}$ denotes the identity matrix in the corresponding dimension and λ are the eigenvalues of the Hessian $\nabla^2 f_i^{\alpha}(\boldsymbol{I}_i, m_i)$ for $\alpha \neq 1$, which can easily be found to be

$$\lambda_1, ..., \lambda_{K-1} = 0,$$

$$\lambda_K = -\alpha\gamma_i^{-\alpha-1}\left(\Phi_{i1}^2 + \cdots + \Phi_{iK}^2\right),$$
$$\lambda_{K+1} = -\alpha p^2 (m_i p)^{-\alpha-1}.$$

The proof for $\alpha = 1$ is omitted here, as the calculations follow the exact same procedure as for the case $\alpha \neq 1$. Since all eigenvalues of the Hessian $\nabla^2 f_i^\alpha(\boldsymbol{I}_i, m_i)$ (for any α) are smaller than or equal to 0, the Hessian is negative semidefinite, and thus the function $f_i^\alpha(\boldsymbol{I}_i, m_i)$ is concave $\forall\alpha$. □

We proceed with exploring the nature of (1b). We have:

Lemma 2. *Constraint* (1b) *is convex.*

Proof. Let us denote the left-hand side of (1b) as

$$t_i(\boldsymbol{I}_i, m_i) = \frac{\Delta}{\sum_{j=1}^{K} I_{ij}\Phi_{ij}} + \frac{\Delta}{m_i p} = \frac{\Delta}{\gamma_i} + \frac{\Delta}{m_i p}.$$

The gradient of $t_i(\boldsymbol{I}_i, m_i)$ is

$$\nabla t_i(\boldsymbol{I}_i, m_i) = \begin{bmatrix} \frac{-\Delta\Phi_{i1}}{\gamma_i^2} & \dots & \frac{-\Delta\Phi_{iK}}{\gamma_i^2} & \frac{-\Delta}{m_i^2 p} \end{bmatrix}^T.$$

Then, for the Hessian of $t_i(\boldsymbol{I}_i, m_i)$ we have

$$\nabla^2 t_i(\boldsymbol{I}_i, m_i) = \frac{2\Delta}{\gamma_i^3} * \begin{bmatrix} \Phi_{i1}^2 & \dots & \Phi_{i1}\Phi_{iK} & 0 \\ \vdots & \ddots & \vdots & \vdots \\ \Phi_{iK}\Phi_{i1} & \dots & \Phi_{iK}^2 & 0 \\ 0 & \dots & 0 & \frac{\gamma_i^3}{m_i^3 p}, \end{bmatrix}.$$

Computing the determinant of $\nabla^2 t_i(\boldsymbol{I}_i, m_i) - \lambda\mathbb{I}$, we obtain

$$\det(\nabla^2 t_i(\boldsymbol{I}_i, m_i) - \lambda\mathbb{I}) = \\ = (-1)^{K-1}\lambda^{K-1}\left(2\Delta m_i^{-3}p^{-1} - \lambda\right) * \\ \left(2\Delta\gamma_i^{-3}\Phi_{i1}^2 + \cdots + 2\Delta\gamma_i^{-3}\Phi_{iK}^2 - \lambda\right).$$

Hence, the eigenvalues of the Hessian $\nabla^2 t_i(\boldsymbol{I}_i, m_i)$ are

$$\lambda_1, ..., \lambda_{K-1} = 0,$$
$$\lambda_K = 2\Delta\gamma_i^{-3}\left(\Phi_{i1}^2 + \cdots + \Phi_{iK}^2\right),$$
$$\lambda_{K+1} = 2\Delta m_i^{-3}p^{-1}.$$

Since all the eigenvalues of the Hessian $\nabla^2 t_i(\boldsymbol{I}_i, m_i)$ are greater than or equal to zero, the Hessian is positive semidefinite and thus the function $t_i(\boldsymbol{I}_i, m_i)$ is convex. □

Theorem 3. *The relaxed-variable version of the optimization problem* (1) *is convex.*

Proof. Given Lemma 1, Lemma 2, and the fact that (1c)-(1g) are linear proves that (1) is a convex problem. □

For the purpose of proving the polynomial-time solvability of the relaxed optimization, the problem is reformulated into a convex optimization problem with generalized inequality constraints in the following. For the subsequent derivations, we define the n-dimensional quadratic cone as

$$\mathcal{Q}^n = \left\{\boldsymbol{x} \in \mathbb{R}^n \,|\, x_1 \geq \sqrt{x_2^2 + \cdots + x_n^2}\right\},$$

the n-dimensional power cone parameterized by a real number $\zeta \in [0, 1]$ as

$$\mathcal{P}_\zeta^n = \left\{\boldsymbol{x} \in \mathbb{R}^n \,|\, x_1^\zeta x_2^{1-\zeta} \geq \sqrt{x_3^2 + \cdots + x_n^2}, x_1, x_2 \geq 0\right\},$$

and the exponential cone as

$$\mathcal{K}_{exp} = \left\{\boldsymbol{x} \in \mathbb{R}^3 \,|\, x_1 \geq x_2 e^{x_3/x_2}, x_1, x_2 > 0\right\}.$$

First, we write the relaxed optimization problem in epigraph form and introduce the slack variables s_{ki}, $k \in \{1, 2\}$, $i \in \{1, ..., N\}$, so that the problem transforms into

$$\min_{g, \boldsymbol{I}, \boldsymbol{m}, \boldsymbol{s}} \quad g$$
$$\text{s.t.} \quad -\sum_{i=1}^{N} h_i^\alpha(s_{1i}, s_{2i}, g) \leq 0, \tag{3a}$$
$$\frac{\Delta}{s_{1i}} + \frac{\Delta}{s_{2i}} - T_{max} \leq 0, \quad \forall i, \tag{3b}$$
$$\text{(1c), (1d), (1e)},$$
$$0 \leq I_{ij} \leq 1, \quad \forall i, j, \tag{3c}$$
$$1 - m_i \leq 0, \quad \forall i, \tag{3d}$$
$$s_{1i} = \sum_{j=1}^{K} I_{ij}\Phi_{ij}, \quad \forall i, \tag{3e}$$
$$s_{2i} = m_i p, \quad \forall i, \tag{3f}$$

where

$$h_i^\alpha(s_{1i}, s_{2i}, g) = \begin{cases} \frac{1}{1-\alpha}\left(s_{1i}^{1-\alpha} + s_{2i}^{1-\alpha}\right) + g, & \alpha \neq 1 \\ \log(s_{1i}) + \log(s_{2i}) + g, & \alpha = 1 \end{cases}.$$

Next, we introduce conic reformulations for the constraints (3a) and (3b).

Lemma 4. *The constraint* (3b) *can be written as*

$$\left(s_{1i} + s_{2i} - \frac{\Delta}{T_{max}}; s_{1i}, s_{2i}, \frac{\Delta}{T_{max}}\right) \in \mathcal{Q}^4. \tag{4}$$

Proof. By definition, (4) transforms into

$$\sqrt{s_{1i}^2 + s_{2i}^2 + \frac{\Delta^2}{T_{max}^2}} \leq s_{1i} + s_{2i} - \frac{\Delta}{T_{max}}.$$

Squaring both sides and subtracting everything below the square root on both sides leads to

$$0 \leq 2s_{1i}s_{2i} - 2\frac{\Delta}{T_{max}}(s_{1i} + s_{2i}),$$

which is easily transformed into

$$\frac{\Delta}{s_{1i}} + \frac{\Delta}{s_{2i}} - T_{max} \leq 0,$$

by dividing by $-2s_{1i}s_{2i}$, and multiplying by T_{max}. □

For the cases $\alpha \in (0, 1)$ and $\alpha \in (1, \infty)$, we convert constraint (3a) to the constraints (6) by setting $\beta = 1 - \alpha$ and introducing the slack variable u_{ki}: Bringing the sum over s_{ki}^β to the right side of the inequality results in

$$-g \leq \frac{1}{\beta}\sum_{k=1}^{2}\sum_{i=1}^{N} s_{ki}^\beta, \tag{5}$$

which is transformed into

$$(5) = \begin{cases} -g\beta \leq \sum_{k=1}^{2}\sum_{i=1}^{N} u_{ki}, & \text{(6a)} \\ u_{ki} \leq s_{ki}^{\beta}, \quad \forall k,i;\ \alpha \in (0,1) & \text{(6b)} \\ g\,|\beta| \geq \sum_{k=1}^{2}\sum_{i=1}^{N} u_{ki}, & \text{(6c)} \\ u_{ki} \geq s_{ki}^{\beta}, \quad \forall k,i;\ \alpha \in (1,\infty) & \text{(6d)} \end{cases}.$$

Now, we consider the case $\alpha \in (0,1)$, which implies that $\beta \in (0,1)$.

Lemma 5. *The constraint* (6b) *can be written as*

$$(s_{ki}, 1; u_{ki}) \in \mathcal{P}^3_{\beta}. \tag{7}$$

Proof. By definition, (7) is equivalent to

$$s_{ki}^{\beta} 1^{1-\beta} \geq \sqrt{u_{ki}^2}, s_{ki} \geq 0,$$

which simplifies to

$$s_{ki}^{\beta} \geq u_{ki}, s_{ki} \geq 0.$$

The constraint $s_{ki} \geq 0$ that is introduced with this reformulation is fulfilled due to constraints (3c) and (3d). □

Next, consider $\alpha \in (1,\infty)$, which implies that $\beta \in (-\infty, 0)$.

Lemma 6. *The constraint* (6d) *can be written as*

$$(u_{ki}, s_{ki}; 1) \in \mathcal{P}^3_{1/(1-\beta)}. \tag{8}$$

Proof. By definition, (8) transforms into

$$u_{ki}^{1/(1-\beta)} s_{ki}^{-\beta/(1-\beta)} \geq \sqrt{1^2}, u_{ki} \geq 0, s_{ki} \geq 0,$$

which simplifies to

$$u_{ki} \geq s_{ki}^{\beta}, u_{ki} \geq 0, s_{ki} \geq 0,$$

when taking everything to the power of $(1-\beta)$ and multiplying both sides by s_{ki}^{β}. The additional constraints $u_{ki} \geq 0$ and $s_{ki} \geq 0$ that are introduced with this reformulation are met due to the positiveness of s_{ki} implied by (3c) and (3d). □

Lastly, consider the case of $\alpha = 1$. In this scenario, the constraint (3a) must be rewritten to

$$-g \leq \sum_{k=1}^{2}\sum_{i=1}^{N} \log s_{ki}, \tag{9}$$

by adding the sum of the logarithms on both sides. Using again the slack variables u_{ki}, (9) can be expressed as

$$-g \leq \sum_{k=1}^{2}\sum_{i=1}^{N} u_{ki}, \tag{10a}$$

$$u_{ki} \leq \log s_{ki}, \quad \forall k,i. \tag{10b}$$

Lemma 7. *Constraint* (10b) *can be rewritten as*

$$(s_{ki}, 1, u_{ki}) \in \mathcal{K}_{exp}. \tag{11}$$

Proof. By definition, (11) is equivalent to

$$s_{ki} \geq 1 * e^{u_{ki}/1}, s_{ki} > 0,$$

which can be written as

$$\log s_{ki} \geq u_{ki}, s_{ki} > 0$$

when taking the logarithm of both sides. The additional constraint $s_{ki} > 0$ that is introduced with this reformulation is fulfilled due to the constraints defined in (1e) and (3d). □

Theorem 8. *The relaxed-variable version of the optimization problem* (1) *can be written as a convex optimization problem with generalized inequality constraints.*

Proof. Given Lemmas 4, 5, 6, and 7 and the fact that (3a) is linear for $\alpha = 0$ concludes the proof. □

Problem (1) reads in a relaxed form, written as a convex optimization problem with generalized inequality constraints for any $\alpha \in [0,\infty)$:

$$\min_{g,\boldsymbol{I},\boldsymbol{m},\boldsymbol{s},\boldsymbol{u}} \ g \tag{12a}$$

$$\text{s.t. } -\sum_{i=1}^{N} e_i^{\alpha}(s_{1i}, s_{2i}, u_{1i}, u_{2i}, g) \leq 0, \tag{12b}$$

$$\text{(1c), (1d), (1e), (3c), (3d), (3e), (3f), (4)}, \tag{12c}$$

where

$$(12b) = \begin{cases} \text{(3a)}, & \alpha = 0 \\ \text{(6a), (7)}, \quad \forall k,\ i, & 0 < \alpha < 1 \\ \text{(10a), (11)}, \quad \forall k,\ i, & \alpha = 1 \\ \text{(6c), (8)}, \quad \forall k,\ i, & \alpha > 1 \end{cases}.$$

For the final verification of the polynomial-time solvability of the optimization problem stated in (12), we define the following generalized logarithms and note their degrees. More details on the generalized logarithm can be found in Section 11.6 in [15]. The generalized logarithm for the n-dimensional quadratic cone $\mathcal{Q}^n$ can be designed as [15]

$$\Gamma_{\mathcal{Q}}(\boldsymbol{x}) = \log\left(x_1^2 - \sum_{i=2}^{n} x_i^2\right). \tag{13}$$

The degree of a generalized logarithm can be calculated as $\theta_\Gamma = \nabla\Gamma(\boldsymbol{x})^T\boldsymbol{x}$, cf. [15]. The degree of the function $\Gamma_{\mathcal{Q}}(\boldsymbol{x})$ is therefore 2. Additionally, we define the generalized logarithm for the n-dimensional power cone $\mathcal{P}^n_{\zeta}$ as

$$\Gamma_{\mathcal{P}}(\boldsymbol{x}) = \log\left(x_1^{2\zeta} x_2^{(2-2\zeta)} - \sum_{i=3}^{n} x_i^2\right) + (1-\zeta)\log(x_1) + \zeta\log(x_2), \tag{14}$$

as introduced in [16]. The degree of the function $\Gamma_{\mathcal{P}}(\boldsymbol{x})$ is calculated as 3. Finally, we define the generalized logarithm for the exponential cone $\mathcal{K}_{exp}$ as [16]

$$\Gamma_{\mathcal{K}_{exp}}(\boldsymbol{x}) = \log\left(x_2 \log\left(\frac{x_1}{x_2}\right) - x_3\right) + \log x_1 + \log x_2, \tag{15}$$

and again note its degree as 3. For any linear inequality constraint, a slack variable can be attached to the system of equality constraints and the generalized logarithm for the slack variable has degree 1, as the slack variable needs to be in $\mathbb{R}_+$. Using these definitions of the generalized logarithms, a logarithmic barrier function $\Lambda(\boldsymbol{w})$ can be defined as

$$\Lambda(\boldsymbol{w}) = -\sum_{c=1}^{Z} \Gamma_c(\boldsymbol{w}),$$
$$\mathbf{dom}\,\Lambda = \{\boldsymbol{w} \mid f_c(\boldsymbol{w}) \prec_{K_c} 0, c = 1, ..., Z\},$$

where $Z = (3 + 2K)N + 2 + K$ for $\alpha = 0$ and $Z = (5 + 2K)N + 2 + K$ for $\alpha \neq 0$. $\boldsymbol{w}$ is composed of the vectorized matrix $\boldsymbol{I}$ as well as the vectors $\boldsymbol{m}$, $\boldsymbol{s} = \{s_{ki}\}$, and $\boldsymbol{u} = \{u_{ki}\}$. $\Gamma_c(\boldsymbol{w})$ are the generalized logarithms defined above for each generalized inequality constraint $f_c(\boldsymbol{w})$ in the convex optimization problem with generalized inequalities defined in (12). This implies that the barrier method can be applied in order to solve this optimization problem. The subsequent complexity analysis is based on the property of self-concordance.

Lemma 9. *The logarithmic barrier function $\Lambda(\boldsymbol{w})$ is self-concordant.*

Proof. The logarithmic barrier for the positive orthant defined by the slack variables of all linear inequalities is a self-concordant function as $-\log x$ is self-concordant and the sum of self-concordant functions is self-concordant [15]. The logarithmic barriers established using the generalized logarithms defined in (13)-(15) are self-concordant as well [16]. □

Lemma 10. *The number of total Newton steps excluding the initial centering step for solving* (12) *using the Barrier method can be bounded by [15]*

$$T_{Barrier} = \left\lceil \frac{\log(\bar{\theta}/(t^{(0)}\epsilon))}{\log \mu} \right\rceil * \left(\frac{\bar{\theta}(\mu - 1 - \log \mu)}{\chi} + \log_2 \log_2(1/\epsilon) \right). \quad (16)$$

Proof. Given Lemma 9 and the fact that (12a) is linear, the function $tg + \Lambda(\boldsymbol{w})$, which is the objective of the Barrier method, is self-concordant. Additionally, given that this function is closed and the sublevel sets of the optimization problem (12) are bounded leads to (16), cf. [15]. □

The parameter $\mu > 1$ is an algorithm parameter of the barrier method, $t^{(0)} > 0$ is the initial value of the parameter t of the barrier method, and $\epsilon > 0$ is the specified tolerance of the barrier method [15]. The parameter χ is a constant that depends on the backtracking parameters introduced in Alg. 9.2 in [15], which is used for line search in Newton's method. The last parameter $\bar{\theta}$ is the sum of the degrees of the generalized logarithms $\Gamma_c(\boldsymbol{w})$, which for the considered problem is computed as

$$\bar{\theta} = \begin{cases} (4 + 2K)N + 2 + K, & \alpha = 0 \\ (10 + 2K)N + 2 + K, & \alpha \neq 0 \end{cases}. \quad (17)$$

Theorem 11. *The complexity of the solution of the optimization problem* (12) *in terms of Newton steps is*

$$T_{Barrier} = \mathcal{O}\left(\log\left(KN/\epsilon\right)\left(KN + \log_2 \log_2\left(1/\epsilon\right)\right)\right). \quad (18)$$

Proof. Using (17) plugged into (16) and simplifying this expression leads to (18). □

V. Conversion Algorithms

In the previous section, it was shown that an optimal solution to the relaxed optimization problem can be found in polynomial time. However, this solution is a continuous solution, which violates the natural restriction that only integer fractions of RAN and computing resources can be allocated. Therefore, for the values of $\alpha = 0$ and $\alpha \to \infty$, specific algorithms for the conversion of the continuous solution to an integer resource allocation were developed. First, the conversion algorithm for the edge computing resource allocation is introduced, which is used for both fairness cases. Afterwards, the algorithms for the specific cases of α are presented. Thereby, $\boldsymbol{J} = \{J_{ij}\}$ denotes the $N \times K$ integer RAN allocation matrix and $\boldsymbol{n} = \{n_i\}$ is the $N \times 1$ integer edge computing resource allocation vector. $\boldsymbol{I}$ and $\boldsymbol{m}$ are their continuous equivalents.

When applying the approximation algorithms, it is assumed that an admission control was performed before accepting users to the network to ensure the availability of enough resources. The general operating principle of the algorithms can be explained as follows: First, the continuous edge computing allocation is converted to an integer assignment as described in Subsection V-A.[2] Next, enough RAN resources are allocated such that every user fulfills its latency requirement. Finally, the remaining RAN resources are allocated with the aim of meeting the specific fairness criterion.

A. Conversion Algorithm for Edge Computing Resources

The continuous to integer conversion of the edge computing resource allocation is done by simple mathematical rounding. As this procedure can lead to the assignment of more than L edge computing resources, a limit check is conducted after the rounding. If more than L computing resources are allocated, then the user with a continuous allocation value closest above $\star.5$, where $\star$ denotes an arbitrary integer, is assigned one resource less than it would have received by strict mathematical rounding. This is done until L edge computing resources are allocated. Similarly, if less than L resources are allocated, the users closest below $\star.5$ will receive one more resource until L resources are assigned. The described procedure is summarized in Alg. 1. Its complexity is $\mathcal{O}(N)$.

B. No Fairness

If all constraints were neglected, the case $\alpha = 0$ would lead to an allocation where each PRB j is allocated to the user who is experiencing the highest CQI value for that PRB. The allocation of the edge computing resources could be done randomly, as each edge computing resource offers the same processing rate and thus contributes in the same way to the objective no matter to which user the resource is assigned. However, each user's packet must be handled within a given time. Therefore, the edge computing resources are allocated

[2]Note that for solving the continuous optimization only $L - N$ computing resources are used. Afterwards, one "extra" resource is distributed to each user during the conversion process, cf. Algs. 2, 3, such that the integer edge computing resource allocation per user is at least as high as the continuous allocation, which ensures the feasibility of the integer solution.

Algorithm 1 Integer Edge Computing Resource Allocation

Input: $N, L, \boldsymbol{m}$
Output: $\boldsymbol{n}$

function ECRALLOC($N, L, \boldsymbol{m}$)
  for all m_i **do**
    $n_i = \lfloor m_i + 0.5 \rfloor$
  end for
  Create empty lists w and z.
  while $\sum_{i=1}^{N} n_i > L$ **do**
    $l = 1$, $k = 0$
    for $i = 1$ to N **do**
      if $i \notin w$ **then**
        $r_i = m_i \mod \lfloor m_i \rfloor - 0.5$
        if $0 < r_i < l$ **then**
          $l = r_i$, $k = i$
        end if
      end if
    end for
    $n_k = \lfloor m_i \rfloor$, attach k to list w.
  end while
  while $\sum_{i=1}^{N} n_i < L$ **do**
    $a = -1$, $b = 0$
    for $i = 1$ to N **do**
      if $i \notin z$ **then**
        $r_i = m_i \mod \lfloor m_i \rfloor - 0.5$
        if $a < r_i < 0$ **then**
          $a = r_i$, $b = i$
        end if
      end if
    end for
    $n_b = \lceil m_i \rceil$, attach b to list z.
  end while
  return $\boldsymbol{n}$
end function

such that users with worse channel conditions get more edge computing resources in order to minimize the number of required PRB allocations for that user, as allocations of PRBs to users with lower CQI values have a negative impact on the maximization of the objective. Once all users are assigned enough edge computing and RAN resources to fulfill their delay constraints, the remaining PRBs are allocated to the users experiencing the best channel conditions. Alg. 2 summarizes the previous explanations. Its complexity is $\mathcal{O}(N + K)$.

C. Max-Min Fairness

When $\alpha \to \infty$, the pure objective is characterized as the minimization of the sum of the reciprocals of the data and processing rates raised to a large positive number. This minimization is achieved once the data and processing rates of each user are equal. Hence, the edge computing resources are split evenly among the users and the PRBs are allocated such that the difference between the users' data rates is minimized while maximizing the minimum data rate any user is experiencing.

Algorithm 2 Integer Resource Allocation for $\alpha = 0$

Input: $N, K, L, \boldsymbol{m}, \boldsymbol{I}, \boldsymbol{\Phi}$
Output: $\boldsymbol{n}, \boldsymbol{J}$

function ALLOCA0($N, K, L, \boldsymbol{m}, \boldsymbol{I}, \boldsymbol{\Phi}$)
  $\boldsymbol{n} =$ ECRALLOC$(N, L - N, \boldsymbol{m}) + \mathbf{1}$, $\boldsymbol{J} = 0$.
  for $i = 1$ to N **do**
    Calculate $w_i = \sum_{j=1}^{K} I_{ij}\Phi_{ij}$.
  end for
  Create list z with users i ordered
  s.t. Δ/w_i is decreasing.
  while list z is non-empty **do**
    for user i in list z **do**
      Find $\arg\max_j I_{ij}\Phi_{ij}$.
      if $\exists$ more than one j maximizing $I_{ij}\Phi_{ij}$ **then**
        Choose randomly between those j.
      end if
      Alloc. PRB j to user i.
      Update $\boldsymbol{J}_j$ and set $\boldsymbol{I}_j = \mathbf{0}$.
      Calculate delay δ_i using n_i and $\boldsymbol{J}_i$.
      if $\delta_i \leq T_{max}$ **then**
        Remove user i from list z.
      end if
    end for
  end while
  for all non-allocated PRBs k **do**
    Find $\arg\max_i \Phi_{ik}$.
    Alloc. PRB k to user i and update $\boldsymbol{J}_k$.
  end for
  return $\boldsymbol{n}$, $\boldsymbol{J}$
end function

This kind of allocation is also done when a delay constraint is introduced. However, the edge computing resources and PRBs are then allocated such that the delay constraint can be fulfilled for all users. This implies that the differences between the users' data rates might increase and the minimum data rate achieved by any user might decrease. If needed, also the computing resources might not be split equally. Thus, in the heuristic, after the delay constraints are fulfilled, the remaining PRBs are assigned such that the minimum data rate is maximized. Alg. 3 aggregates the outlined procedure. Its computational complexity is $\mathcal{O}(N + K)$.

VI. Performance Evaluation

A. Simulation Setup

For input parameters we have used a 5G trace with data measured in the Republic of Ireland. These traces are described in detail in [17], and a statistical analysis is given in [18]. The parameter of interest from the trace is the CQI with 15 levels, which serves to determine the per-block rate of a user in a frame. These measurements were conducted for one user, but at different days, for different applications, and when the user was static and moving around. To mimic the dynamic nature of the users in the simulations, we have picked

TABLE I
PER-PRB RATES FOR DIFFERENT CQI

CQI	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R (kbps)	48	73.6	121.9	192.2	282	378	474.2	712	772.2	874.8	1063.6	1249.6	1448.4	1640.6	1778.4

Algorithm 3 Integer Resource Allocation for $\alpha \to \infty$

Input: $N, K, L, \boldsymbol{m}, \boldsymbol{I}, \boldsymbol{\Phi}$
Output: $\boldsymbol{n}, \boldsymbol{J}$

function ALLOCAINF($N, K, L, \boldsymbol{m}, \boldsymbol{I}, \boldsymbol{\Phi}$)
  Follow lines 2 to 21 from Alg. 2.
  for $i = 1$ to N **do**
    Calculate $w_i = \left(\sum_{j=1}^{K} J_{ij}\Phi_{ij}\right)^{|1-\alpha|}$.
  end for
  Create list z with users i ordered s.t. w_i is increasing.
  for all non-allocated PRBs k **do**
    Take $z(1)$, find $\arg\min_k \left(\max_i(\Phi_{ik}) - \Phi_{z(1)k}\right)$.
    Allocate PRB k to user $z(1)$ and update $\boldsymbol{J}_k$.
    Set $w_{z(1)} = \left(\sum_{j=1}^{K} J_{z(1)j}\Phi_{z(1)j}\right)^{|1-\alpha|}$.
    Reorder list z with users i s.t. w_i is increasing.
  end for
  return $\boldsymbol{n}$, $\boldsymbol{J}$
end function

only measurements where the user was moving around. The subcarrier spacing is 30 KHz, making the PRB width 360 KHz and a frame consisting of 20 slots [11]. The total number of PRBs is $K = 120$. The corresponding data rates per CQI are given in Table I. The number of edge computing resources is $L = 120$ and the processing rate per resource is 500 kbps. The packet size of a packet is 5 kbit. For both types of fairness, simulation data are gathered for $N = \{5, 8, 10\}$ and for $T_{max} = \{3, 5, 10\}$ ms using MATLAB R2021b together with CVX [19] and Mosek [20]. As the integer allocation problem is NP-hard, it was not possible to obtain optimal integer solutions satisfying our accuracy demands. Hence, we compare our approaches to the optimal continuous solution.

B. Benchmark (Round-Robin)

The benchmark allocation against which the special approximation algorithms are compared is the Round-Robin principle. This means that all users are allocated one computing resource and one PRB in each iteration. Once a user fulfills its delay constraint, it will not be assigned any more resources until every user complies with its latency target. Thereafter, the remaining computing and RAN resources are allocated one by one to all users, until no resources are available anymore.

C. Results for No Fairness

Various measurement points for different CQI inputs are depicted in Fig. 2 for two selected scenarios (left two figures). For a better visibility, the benchmark objective values are only shown once. They are in the same range for the other depicted scenario. It is observable that the heuristic outperforms the benchmark algorithm by far. Additionally, the solution retrieved using Alg. 2 is, independent of N and T_{max}, almost always attaining the optimal continuous solution. The largest deviation from the heuristic solution to the optimum was 1% among all the scenarios in 100 data points, while the overall average deviation for $\alpha = 0$ was only 0.24%. The good performance evaluation is supported by Fig. 3, where the average objective value from the heuristic is very close or equal to the average optimal continuous objective value (left three figures). The average is taken over 100 measurement points. Another observation from Fig. 3 is that the average objective value increases when loosening the delay constraint. The reason for this behavior is that more PRBs can be allocated to users experiencing the best channel conditions, as in general a user does not need that many resources to fulfill its delay constraint when T_{max} is increased.

D. Results for Max-Min Fairness

For the max-min fairness, $\alpha = 13$ was used to mimic the behavior of $\alpha \to \infty$. Due to numerical reasons during the optimization, a higher α-value could not be used. A similar output to the no fairness case can be observed for the max-min fairness (right two/three figures) in Figs. 2, 3. For a better comparability, the benchmark results are again only shown once in Fig. 2, as they are in the same range for the other shown scenario. The benchmark is again outperformed by the heuristic, and the heuristic solution is almost attaining the continuous optimum (not more than 0.0617% away), indicating that the continuous optimum is almost an integer optimum. The average deviation of the heuristic solution from the continuous optimum is 0.0006%. In Fig. 3 it can be observed that the objective value is always very close to the optimum and gets worse the higher the number of users is, since then the resources must be split among more users and the reciprocals of the RAN data and the edge processing rates get larger. The benchmark averages are largely influenced by some outliers, where few users are experiencing very bad channel conditions (measurments 15 and 16, see Fig. 2).

VII. CONCLUSION

In this paper, we considered the problem of jointly allocating RAN and computing resources to vehicular users so that their latency requirement is met, while simultaneously providing α-fairness. For the special cases $\alpha = 0$ and $\alpha \to \infty$, we provided algorithms with polynomial-time complexity and have shown that their performance is very close to the optimum, and that they considerably outperform other well-known resource allocation algorithms. The results were obtained with input parameters taken from real datasets. In the future, we plan to also consider the joint allocation of downlink RAN resources as well as the cases $\alpha = 1$ and $\alpha = 2$.

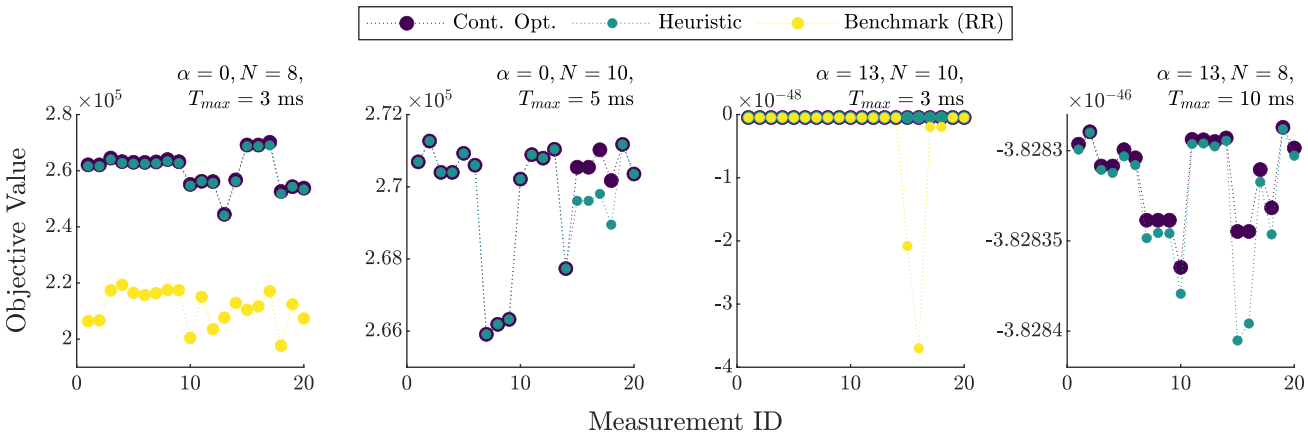

Fig. 2. Objective values for different CQI inputs for selected scenarios.

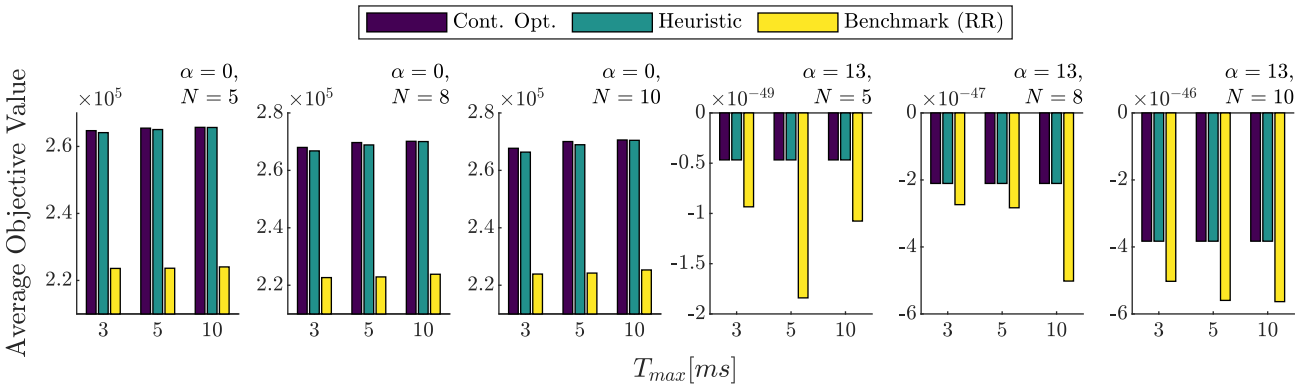

Fig. 3. Average objective values for all considered scenarios.

ACKNOWLEDGEMENT

This work was supported in part by the BMW Group and in part by the German Federal Ministry of Education and Research (BMBF) as part of the project "6G-ANNA" with project identification number 16KISK107.

REFERENCES

[1] P. Popovski, K. F. Trillingsgaard, O. Simeone, and G. Durisi, "5G wireless network slicing for eMBB, URLLC, and mMTC: A communication-theoretic view," *IEEE Access*, vol. 6, 2018.

[2] M. Bennis, M. Debbah, and H. V. Poor, "Ultra-reliable and low-latency wireless communication: Tail, risk, and scale," *Proceedings of the IEEE*, vol. 106, no. 10, 2018.

[3] H. Chien, Y. Lin, C. Lai, and C. Wang, "End-to-end slicing with optimized communication and computing resource allocation in multi-tenant 5G systems," *IEEE Trans. on Veh. Tech.*, vol. 69, no. 2, 2020.

[4] M. Centenaro, L. Vangelista, and S. Saur, "Analysis of 5G radio access protocols for uplink URLLC in a connection-less mode," *IEEE Transactions on Wireless Communications*, vol. 19, no. 5, 2020.

[5] H. Shariatmadari, S. Iraji, Z. Li, M. A. Uusitalo, and R. Jäntti, "Optimized transmission and resource allocation strategies for ultra-reliable communications," in *Proc. of IEEE PIMRC*, 2016.

[6] Y. Han, S. E. Elayoubi, A. Galindo-Serrano, V. S. Varma, and M. Messai, "Periodic radio resource allocation to meet latency and reliability requirements in 5G networks," in *Proc. of IEEE VTC-Spring*, 2018.

[7] F. Mehmeti and T. La Porta, "Admission control for URLLC users in 5G networks," in *Proc. of ACM MSWiM*, 2021.

[8] F. Mehmeti and T. La Porta, "Reducing the cost of consistency: Performance improvements in next generation cellular networks with optimal resource reallocation," *IEEE Tran. on Mobile Computing*, vol. 21, no. 7, 2022.

[9] A. Destounis and G. S. Paschos, "Complexity of urllc scheduling and efficient approximation schemes," *arXiv preprint arXiv:1904.11278*, 2019.

[10] S. E. Elayoubi, S. B. Jemaa, Z. Altman, and A. Galindo-Serrano, "5G RAN slicing for verticals: Enablers and challenges," *IEEE Communications Magazine*, vol. 57, no. 1, 2019.

[11] ETSI, "5G NR overall description: 3GPP TS 38.300 version 15.3.1 release 15." www.etsi.org, 2018. Technical specification.

[12] J. Navarro-Ortiz, P. Romero-Diaz, S. Sendra, P. Ameigeiras, J. J. Ramos-Munoz, and J. M. Lopez-Soler, "A survey on 5G usage scenarios and traffic models," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 2, 2020.

[13] R. Srikant, *The Mathematics of Internet Congestion Control*. Birk., 2004.

[14] J. Lee and S. Leyffer, *Mixed integer nonlinear programming*, vol. 154. Springer Science & Business Media, 2011.

[15] S. Boyd and L. Vandenberghe, *Convex optimization*. CUP, 2004.

[16] R. Chares, *Cones and interior-point algorithms for structured convex optimization involving powers and exponentials*. PhD thesis, Université Catholique de Louvain Louvain-la-Neuve, Louvain, Belgium, 2009.

[17] D. Raca, D. Leahy, C. J. Sreenan, and J. J. Quinlan, "Beyond throughput, the next generation: A 5G dataset with channel and context metrics," in *Proc. of ACM MMSys*, 2020.

[18] F. Mehmeti and T. La Porta, "Analyzing a 5G Dataset and Modeling Metrics of Interest," in *Proc. of IEEE MSN*, 2021.

[19] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1." http://cvxr.com/cvx, Mar. 2014.

[20] MOSEK ApS, *The MOSEK optimization toolbox for MATLAB manual. Version 10.0.16(BETA).*, 2022.