5G New Radio Resource Allocation Optimization for Heterogeneous Services

Nasim Ferdosian, Sara Berri, Arsenia Chorti

5G New Radio Resource Allocation Optimization for Heterogeneous Services

Nasim Ferdosian, Sara Berri, Arsenia Chorti
ETIS UMR 8051, CY-Tech, ENSEA, CNRS
Cergy 95000, France
nasim.ferdosian@ensea.fr

***Abstract*—5G new radio (NR) introduced flexible numerology to provide the necessary flexibility for multiplexing the communication of heterogeneous services on a shared channel. One of the fundamental challenges of 5G NR is to develop resource allocation schemes to efficiently exploit such flexibility to optimize resource allocation of ultra-reliable low-latency communications (URLLC) in coexistence with enhanced mobile broadband (eMBB) while ensuring their colliding performance requirements. This problem of 5G NR resource allocation to URLLC services coexisted with eMMB services is an NP-hard problem. We present a new formulation of this problem by considering the interplay of eMBB and URLLC services' constraints and targets. The objective of the formulated problem is to probabilistically meet quality of service (QoS) requirements of both eMBB and URLLC services when the optimal global solution is infeasible. To this end, we express the problem as an integer linear programming problem and consider two formulations with hard and soft URLLC throughput constraints. The overall problem is then treated as a specific instance of bin packing optimization, whose objective is to minimize the placements of URLLC services. We apply a greedy resource allocation algorithm to provide a near-optimal solution in polynomial-time to this problem. Finally, we numerically evaluate the optimal solutions of the two different formulations. Performance results demonstrate and confirm that the proposed 5G NR resource allocation solutions can provide efficient resource utilization while satisfying the performance requirements of the eMBB and URLLC services.**

***Index Terms*—Component; Formatting; Style; Styling; Insert**

I. Introduction

The emerging wide range of devices and services in a variety of application fields have introduced new performance and quality of service (QoS) requirements to be addressed by 5G mobile communication systems [1]. The main key enablers of 5G, as a service driven mobile communication technology is to address the requirements of the new era of heterogeneous services include flexible numerology, mini-slotting and optimized frame structure which have been defined by 5G New Radio (NR) [2]. Optimizing the allocation of 5G NR to support ultra-reliable and low latency communications (URLLC) and enhanced mobile broadband (eMBB) services with varying QoS requirements and characteristics remains a challenging issue [3]. URLLC services with extreme latency constraints coexist with eMMB services with very high data rate demands (Gigabits per second) and moderate latency requirement (a few milliseconds) [4].

The current research body has introduced different preemption and puncturing approaches to multiplex both URLLC and eMBB services [5]–[7]. In puncturing approaches, an arriving URLLC packet overtakes resources that have already been allocated to eMBB services, causes a throughput reduction to these kinds of data demanding services [8]. Therefore, while the current research provides a variety of resource allocation approaches, there is currently a gap for an efficient resource allocation solution to meet latency constraints and throughput requirements of URLLC services without compromising throughput of rate-hungry eMBB services.

Alternatively, the authors in [9] studied the allocation of 5G NR resources to eMBB and URLLC services, without using puncturing mechanisms, while exploiting the 5G NR flexible numerology and frame structure. 5G NR resource allocation to different services while ensuring their QoS requirements, was shown to be an NP-hard problem. They proposed different optimization methods to solve the problem and allocate resources to URLLC and eMBB services in two distinct and sequential steps respectively. However, these approaches identifying on the fly the optimal solution of the corresponding combinatorial optimization problem, have shown that depending on the size of the scheduling grid, the affordable latency and the throughput requirements of URLLC services, a solution might not always be feasible. In such infeasible cases, where available resources are not enough to fulfill all of the URLLC services' requirements, URLLC and eMBB services are totally dropped and lead to inefficient resource utilization. Moreover, these solutions allocate resources based on the utility values, computed using optimization solvers, demanding high computational complexities. Therefore, a trade-off between resource allocation efficiency and computational complexity is still a challenging issue.

In this article, we study the problem of 5G NR resource allocation to URLLC services coexisted with eMMB services. We formulate this problem as an integer linear programming problem, where the objectives is to determine resource allocation decisions that maximize the sum throughput of the eMBB services, subject to the throughput demands and latency constraints of the URLLC services. Moreover, we apply an alternative formulation of the problem with less stringent constraints to improve the feasibility of the problem. The overall problem is then treated as a specific instance of bin packing optimization, whose objective is to minimize the placements of URLLC services. We propose a greedy resource allocation algorithm as an approximate solution to the problem. Finally,

TABLE I: Resource Blocks in Flexible Numerology

	Shape 1	Shape 2	Shape 3	Shape 4
TTI duration (ms)	0.5	0.25	0.125	0.125
SCS (kHz)	15	30	60	60
Symbol duration (μs)	66.7	33.3	16.7	16.7
CP (μs)	4.7	2.3	1.2	4.17
Number of Symbols	7	7	7	6

we conduct a numerical analysis, and show that the proposed heuristic resource allocation algorithm provides a near-optimal solution in polynomial-time. In addition, we evaluate the optimal solutions of the two different formulations, and show that gains could be achieved by avoiding a close relationship between the eMBB and URLLC services.

The rest of the paper is organized as follows. We present an overview of the flexible numerology and frame structure in 5G NR in Section II. In Section III we give the problem, while in Section IV-B we introduce a re-formulation of the problem and a light-weight near-optimal heuristic. In Section V we present an extensive set of numerical results and related analysis. Finally, Section VI concludes the paper.

II. Flexible Numerology in 5G NR

5G NR Release-15 [2] defines a flexible numerology with subcarrier spacing (SCS) of 15, 30, and 60 kHz below 6 GHz, and 60 and 120 kHz above 6 GHz, compared to long-term evolution (LTE) which uses a fixed numerology with SCS of 15 kHz below 6 GHz. 5G NR also defines a 10 milliseconds (ms) frame, with each frame divided into 10 subframes of 1 ms, which are further divided into one or more mini-slots. A mini-slot comprises 14 OFDM symbols for a configuration using normal cyclic prefix, or 12 OFDM symbols for extended cyclic prefix. In 5G NR, the mini-slot size is defined according to the symbol duration, which is inverse to the SCS, to ensure the orthogonality of the subcarriers. By using higher SCS, the symbol duration decreases and hence also the mini-slot size, which is beneficial for lower latency [10].

In the present paper, we focus on a downlink resource scheduling scenario, where one base station (BS) serves both throughput hungry (eMBB) and ultra-low latency (URLLC) users [9]. A resource allocation to a user consists in assigning to it a set of adjacent subcarriers and mini-slots in the time-frequency grid, referred to as resource blocks. The permissible shapes of the resource blocks depend on the numerology and frame structure employed and can be either fixed or dynamically chosen; this latter case being dubbed as flexible numerology. Following the system model in [9], Table I presents the four most widely utilized resource block shapes determined by different configurations of numerology and frame structure, according to the 5G NR specifications. According to the flexible numerology principle, no restrictions are placed to eMBB and URLLC users with respect to the shape of resource blocks utilized to serve them.

In this framework, to optimize eMBB and URLLC co-existence, a joint scheduler allocates corresponding resource blocks in the time-frequency grid with the objective to maximize the sum throughput of the former, while satisfying the throughput demands and latency constraints of the latter.

In the following, we first present the original scheduling problem with hard throughput and latency constraints for URLLC users, followed by a re-formulation with soft throughput and latency constraints for URLLC users. Finally, we discuss a novel heuristic scheduler that solves the problem on a best effort approach for each individual URLLC user, i.e., near-optimal resource allocation is proposed whenever possible, otherwise the URLLC demands are partially satisfied to avoid the infeasible cases, which totally drop all URLLC demands.

III. Scheduling Problem Formulation with Hard Throughput Constraints

In the following, let us denote by $\mathcal{K}$ the set of all services, by $\mathcal{K}^{(c)}$ the set of eMBB services and by $\mathcal{K}^{(\ell)}$ the set of URLLC services. q_k and τ_k are respectively the throughput demand and maximum tolerant latency of service $k \in \mathcal{K}^{(\ell)}$. $\mathcal{B}$ denotes the set of all possible resource blocks according to the employed numerology and finally, $\mathcal{I}$ denotes the set of all mini-slots. We utilize the parameter $\alpha_{b,i}, b \in \mathcal{B}, i \in \mathcal{I}$ which indicates whether a block $b \in \mathcal{B}$ includes basic unit $i \in \mathcal{I}$, in which case $\alpha_{b,i} = 1$, otherwise $\alpha_{b,i} = 0$. Furthermore, we denote by $r_{b,k}, b \in \mathcal{B}, k \in \mathcal{K}$ the throughput of each resource block, under the constraint that the latency constraint is met, i.e.,

$$r_{b,k} = \{\text{Capacity of block } b \text{ for service } k\} \times \mathbf{1}_{\{\tau_k - t_b > 0\}} \quad (1)$$

where t_b is the end time of block b and $\mathbf{1}_{\{x\}}$ is the indicator function for the logical proposition x. Under the URLLC latency constraint, the resource block b cannot be assigned to $k \in \mathcal{K}^{(\ell)}$ if the end time of block b exceeds its latency constraint; this is embedded as a hard constraint by imposing $r_{b,k} = 0$ for respective resource blocks $b \in \mathcal{B}$ and service $k \in \mathcal{K}^{(l)}$. Finally, by $x_{b,k}$ we denote a binary variable that takes the value 1 if the resource block $b \in \mathcal{B}$ is assigned to service $k \in \mathcal{K}$, otherwise $x_{b,k} = 0$.

The standard scheduling optimization problem is to maximize the sum throughput of $\mathcal{K}^{(c)}$ services under the constraint of satisfying the latency and throughput demands of the set $\mathcal{K}^{(\ell)}$, without any overlapping among the allocated resource blocks. The formal problem formulation is given as [9]:

$$\text{[P0]} \quad \max_{x_{b,k} \in \{0,1\}} \sum_{b \in \mathcal{B}} \sum_{k \in \mathcal{K}^{(c)}} r_{b,k} x_{b,k}, \quad (2)$$

$$\text{s.t.} \quad \sum_{b \in \mathcal{B}} r_{b,k} x_{b,k} \geq q_k, \quad k \in \mathcal{K}^{(\ell)}, \quad (3)$$

$$\sum_{b \in \mathcal{B}} \sum_{k \in \mathcal{K}} \alpha_{b,i} x_{b,k} \leq 1, \quad i \in \mathcal{I}. \quad (4)$$

In [9], it was proven that the combinatorial problem P0 is an NP-hard partition problem. Furthermore, the superior performance of scheduling using flexible as opposed to fixed numerology was demonstrated and established. Albeit, through

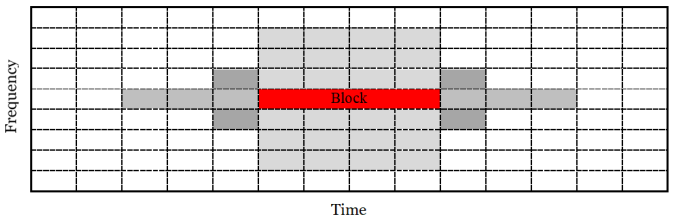

Fig. 1: Resource allocation of a candidate block and its corresponding conflicts; vertical blocks (light grey), horizontal blocks (grey) and square blocks (dark grey).

an extensive set of results the authors also showed that for certain sets of latency and throughput values, P0 becomes infeasible; in particular, as the latency constraints of URLLC services become more stringent and / or their throughput demands increase, the constraint (3) becomes more stringent and it might be impossible to find a solution in fixed size time-frequency grids. Such infeasible conditions lead to inefficient resource utilization, where the whole URLLC services' demands are ignored, whereas they could be partially covered. In the following section we address this issue by relaxing constraint (3).

On a different note, the authors in [11] investigated an alternative formulation of P0 by introducing an explicit description of the impact constraint (4), i.e., of the fact that the resource blocks are not allowed to overlap, to the optimal solution. To this end they introduced the concept of "conflict" that incurs by any specific URLLC or eMBB resource block placement to subsequent placements. To illustrate the idea, Fig. 1 depicts all the "conflicts" that arise from an arbitrary block placement, shown in red; the specific block allocation (in red) *forbids* any other block allocation in the sketched neighborhood (in grey), as constraint (4) does not allow for overlapping (partial or full) of block placements. In light of this, it is evident that even if a particular resource block might achieve maximum throughput, its allocation can be sub-optimal due to the losses because of the generated forbidden placements around it, i.e., the generated conflicts described by constraint (4) of P0 might outweigh the throughput gains. Leveraging this insight, a bin packing type of formulation of the P0 was proposed and related near-optimal heuristics were proposed.

In the following, we will first provide a reformulation of P0 that allows for probabilistic coverage of the URLLC demands in view of the fact that P0 is infeasible in certain scenarios, dubbed as P1. Subsequently, inspired by both P1 and the bin packing formulation proposed in [11], a novel heuristic scheduler is presented to accommodate URLLC service demands on a best effort approach.

IV. Problem Reformulation and Heuristic Scheduler

A. Reformulation of P0 with soft URLLC throughput constraints

The solution of the problem P0 mainly depends on the constraint (3), which requires that all the URLLC services' resource demands should be satisfied. Despite the desirable benefits brought by this formulation to the URLLC services, the constraint invoked raises the big issue of infeasibility, when available resources cannot cover the resource demands of all URLLC services. Such infeasible cases lead to allocate resources neither to URLLC services nor the eMBB services[1]. Indeed, it is important, and even necessary, to address the shortcoming of this formulation, by introducing the possibility to cover some of the URLLC services, instead of dropping all of them, whenever there is not enough resources to fully cover all URLLC services.

To this end, we propose in the following an alternative formulation that aims at maximizing the joint overall sum throughput, i.e., including both URLLC and eMBB services, with relaxed constraint for fully satisfying the resource demands of URLLC services. The problem re-formulation, dubbed as P1 is given below:

$$\text{[P1]} \quad \max_{x_{b,k}\in\{0,1\}} \sum_{b\in\mathcal{B}}\sum_{k\in\mathcal{K}} r_{b,k}x_{b,k}, \tag{5}$$

$$\text{s.t.} \quad \sum_{b\in\mathcal{B}} r_{b,k}x_{b,k} \leq q'_k, \quad k\in\mathcal{K}^{(\ell)}, \tag{6}$$

$$\sum_{b\in\mathcal{B}}\sum_{k\in\mathcal{K}} \alpha_{b,i}x_{b,k} \leq 1, \quad i\in\mathcal{I}, \tag{7}$$

where, $q'_k = q_k + u_k, k\in\mathcal{K}^{(l)}$ and $u_k \geq 0$. The parameter u_k is introduced to assign more or less stringency to the URLLC demands constraint. Moreover, it might depend on the throughput (and by extension to the latency) tolerance. Note that, the constraint (3) is substituted by (6) to separate the feasibility of the two services and the different demands could be treated independently; this would make it possible to perform resource allocation to the eMBB services without having as a prerequisite to satisfy all of the constraints of the URLLC services, as required by (3) in P0. Furthermore, the new objective function, namely (5), is compromising the URLLC services' demands as well to avoid the trivial solution[2] $x^{\star}_{b,k} = 0, k\in\mathcal{K}^{(l)}, \forall b\in\mathcal{B}$, which is not beneficial.

This approach moves towards the direction of assigning differentiated services (DiffServ) type of *throughput QoS guarantees* to layer 2 radio access network (RAN) scheduling, a task typically performed at layer 3. The motivation behind this undertaking is that as network slicing and network function virtualization will be handled by the virtualization orchestrator, flexible QoS guarantees will be jointly handled by a unique virtualization layer.

B. Heuristic Scheduler

In [11], it has been demonstrated that to maximize the throughput of $\mathcal{K}^{(c)}$ services while accommodating the data rate demands and latency constraints of $\mathcal{K}^{(\ell)}$ users, it is necessary,

[1]Following the strict formulation of P0, when the URLLC resource allocations are infeasible, the scheduler does not allocate any resources at all.

[2]This solution became feasible in P1 whatever the throughput demand q_k.

Algorithm 1 Bin Packing Resource Allocation Algorithm [12]

Input: throughput matrix $\mathbf{r} = [r_{b,k}]$, $b \in \mathcal{B}, k \in \mathcal{K}$, aggregated-throughput-loss vector $\mathbf{e}$, demand vector of URLLC services $\mathbf{q}$, set of all available resource blocks $\mathcal{B}$.
Output: Block-service assignment $\mathbf{s}$.

```
for k = 1 to |q| do
    create the following categories:
    for i = 1 to M do
        Cat^i U^k = all resource blocks b ∈ B where
        ⌈q_k / r_{b,k}⌉ = i;
        Check pairwise conflicts among categorized blocks
        and remove the blocks with the higher aggregated-
        throughput-loss;
    end for
end for
Phase 1 (K^(ℓ) resource allocation):
for i = 1 to M do
    select the Cat^i U^k which has the least number of blocks;
    if (|Cat^i U^k| ≥ i and q_k is not already met) then
        B' ← (select i number of blocks in Cat^i U^k with the
        least aggregated-loss-value);
        s ← s ∪ (b', k') , k' = i, ∀b' ∈ B';
        Remove from B the blocks in s and those overlapping
        with the blocks in s;
        if q_k is met then
            K^(ℓ) ← K^(ℓ) \ {k'};
        end if
    end if
end for
Phase 2 (K^(c) resource allocation):
repeat
    (b', k') ← arg max_{b∈B, k∈K^(c)} r_{b,k};
    s ← s ∪ (b', k');
    Remove from B the blocks in s and those overlapping
    with the blocks in s;
until B = ∅
```

to minimize the number of conflicts and subsequently minimize the number of URLLC placements. By considering this scheduling problem as a bin packing optimization problem, we use the linear complexity greedy heuristic approach proposed in [12], as a computationally efficient resource allocation solution, summarized in Algorithm 1. It performs allocation of blocks to $\mathcal{K}^{(\ell)}$ services based on the minimization of the number of conflicts which is achieved by the minimizing the number of $\mathcal{K}^{(\ell)}$ resource allocations (placements). To this end, Algorithm 1 finds an allocation that satisfies the $\mathcal{K}^{(\ell)}$s' demands by using the least number of resource blocks. The resource allocation for $\mathcal{K}^{(\ell)}$ services and $\mathcal{K}^{(c)}$ services are treated in two separate phases. The former is performed first because of the latency requirement.

For each $k \in \mathcal{K}^{(\ell)}$ we generate M categories with decreasing fractional sizes with respect to $q_k, k \in \mathcal{K}^{(\ell)}$, i.e., category $i \in \{1, \ldots, M\}$ is defined as the set of all resource blocks $b \in \mathcal{B}$ for which the ceiling of the ratio of the service demand over the throughput of block b is equal to i, or equivalently, category $Cat^i U^k$ contains the available resource blocks which satisfy at least $1/i$-th of the service demand q_k.

For example, $Cat^1 U^1$ is the category of the blocks which individually satisfy the whole service 1's demand. Then, in $\mathcal{K}^{(\ell)}$ resource allocation phase, the minimization of the number of $\mathcal{K}^{(\ell)}$ placements is achieved by starting from the categories of largest blocks, i.e., blocks in $Cat^1 U^k$ are allocated first and then we move to $Cat^2 U^k$, and so on. The elements of each selected category $Cat^i U^k$, are then ordered with decreasing aggregate loss for the eMBB users, on operation of complexity $|Cat^i U^k| \log |Cat^i U^k|$ and the first i of them are selected for the placement of URLLC user k. If there are not enough elements for the allocation, then the respective blocks are moved the next in chain category.

After each placement the allocated blocks are removed from $\mathcal{B}$ and all other categories. In the case that a URLLC user cannot be accommodated, then no blocks are allocated to it.

Once the set of URLLC users has been serviced, we move to the last phase of the algorithm for the resource allocation to $\mathcal{K}^{(c)}$ services. Algorithm 1 selects the block-service pairs with the highest throughput $r_{b,k}, k \in \mathcal{K}^{(c)}$ from the remaining available resource blocks and continues until no more blocks are available. The ordering of the utilities has a complexity of $\mathcal{O}\big(\max_{i,k}\{|Cat^i U^k| \log(|Cat^i U^k|)\}\big)$, which is the overall complexity of Algorithm 1.

V. Numerical Results

To showcase the effectiveness of the proposed reformulation of the resource allocation as P1 and of Algorithm 1, we present numerical results for the sum bit rate of the eMBB services and the percentage of covered URLLC services using the same simulation setup as in [9][3]. This simulation environment was implemented based on the control channel overhead model for supporting the flexible numerology defined in [13] and considers the effect of guard band (i.e., of the cyclic prefix) on the achievable data rate by resource blocks as modeled in [14]. The global optimum solution of P0 is calculated by employing Gurobi optimization solvers. This optimal solution of P0, presented in [9], is non-scalable with high complexity and is used here for benchmarking. In addition, the global optimum solution of P1 is obtained by using the optimization solver IBM ILOG CPLEX.

Different 5G numerologies and URLLC configurations in terms of data rate demands and latency tolerances are considered. In more detail, we considered URLLC latency constraints $\tau = \{0.25, 0.5, 1, 1.5, 2\}$ msec and bit rate demands $q_k = \{16, 32, 64, 128, 256, 512, 1024\}$ kbps for a set of $|\mathcal{K}^{(\ell)}| = 5$ services $k \in \mathcal{K}^{(\ell)}$. The parameter u_k is varying with respect to the latency tolerance as follows: in case of URLLC bit rate demands 64 kbps and 128 kbps it is set to 136, 116 and 96 for latency $\in \{0.25, 1\}$, $\in \{0.5\}$, and $\in \{1.5, 2\}$, respectively;

[3]We thank the authors of [9] for kindly sharing their simulation codes in IEEE DataPort.

in case of URLLC bit rate demands 256 kbps it is set to 244, and 124 for latency $\in \{0.25, 0.5, 1\}$, and $\in \{1.5, 2\}$, respectively; in case of URLLC bit rate demands 512 kbps it is set to 158, and 138 for latency $\in \{0.25, 0.5, 1, 1.5\}$, and $\in \{2\}$, respectively; in case of URLLC bit rate demands 1024 kbps it is set to 176 for latency $\in \{0.25, 0.5, 1, 1.5, 2\}$.

A. Performance Comparison of P0, P1 and of the Heuristic Scheduler in Terms of eMBB Sum Bit Rate

The obtained results are presented in Figs. 2-6. The bit rate per user in $k \in \mathcal{K}^{(c)}$ for all the examined algorithms in case of URLLC bit rate demands 16 kbps and 32 kbps are almost same as the ones in case of bi rate demand 64 kbps. Therefore, for the sake of brevity, we omit the presentation of this set of results.

The numerical results, through Figs. 2-6, show that as the latency tolerance decreases and the bit rate demands of $\mathcal{K}^{(\ell)}$ services increase, in contrast to P0 which becomes infeasible, the formulation P1 always provides a solution irrespective of the parameters. For example in case of bit rate demand 1024 kbps, Fig. 6, where P0 has no feasible solution for non of the URLLC latency tolerance values, P1 and the proposed heuristic algorithm provides feasible solutions. This is thanks to the fact that P1 aims at a global sum bit rate optimization with relaxed constraints for the URLLC bit rate coverage. However, in some cases (e.g $\tau = 0.5$ ms and bit rate 256 kbps), P1 fails to satisfy the URLLC demands whereas this is feasible using P0. In these cases however, a higher sum bit rate for $K^{(c)}$ services is proposed. Interestingly, when the bit rate demand is small (e.g., 64 kbps), all the URLLC demands are satisfied and the gap between the bit rate of the set of $\mathcal{K}^{(c)}$ services provided by the two formulations, P0 and P1, is relatively small when the latency is either $\tau = 1.5$ ms or 2 ms and 0 otherwise. These results suggest that, it is possible to consider either P1 or P0 without performance loss even when P0 is feasible. Furthermore, the formulation P1 could provide a better trade-off.

With respect to the proposed heuristic scheduler, as can be seen in Figs. 2-6, it incurs negligible performance loss when P0 is feasible and succeeds in providing an allocation in all scenarios. Comparing the heuristic scheduler to P1, the sum bit rate of eMBB services is lower in most of the cases. However, as will be discussed next, it manages to cover the full URLLC demands for wider palette of latency and bit rate parameters for the URLLC services.

B. Performance Comparison of P0, P1 and of the Heuristic Scheduler in Terms of URLLC Coverage

In this subsection, we compare the results of P0, P1 and the heuristic scheduler in terms of URLLC coverage. For this, we show in Table II, the percentage of covered URLLC users for all the considered approaches. From the table, we can see that the heuristic scheduler covers all the URLLC users most of the time, and always provides better results than the formulations P0 and P1. This is because, the heuristic schedules two types of users sequentially, and the URLLC resource allocation is performed first, which explains consequently its lowest performance in terms of sum throughput of the eMBB users in comparison with other approaches, when the percentage of covered URLLC is not zero (all the considered cases except for: $\tau = 0.25$ and bit rate 1024).

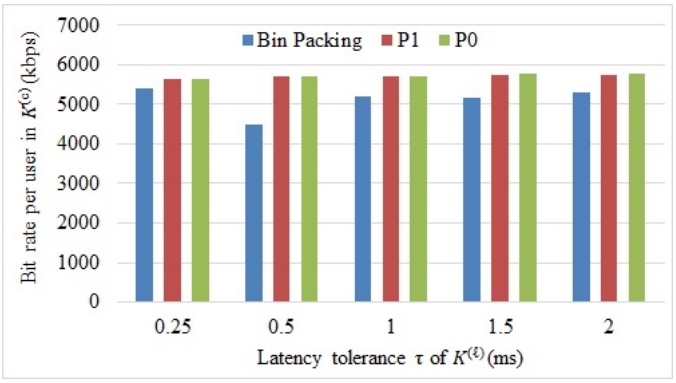

Fig. 2: Bit rate of $\mathcal{K}^{(c)}$ with respect to latency tolerance of $\mathcal{K}^{(\ell)}$ for the bit rate demand of $\mathcal{K}^{(\ell)}$ equals 64 kbps.

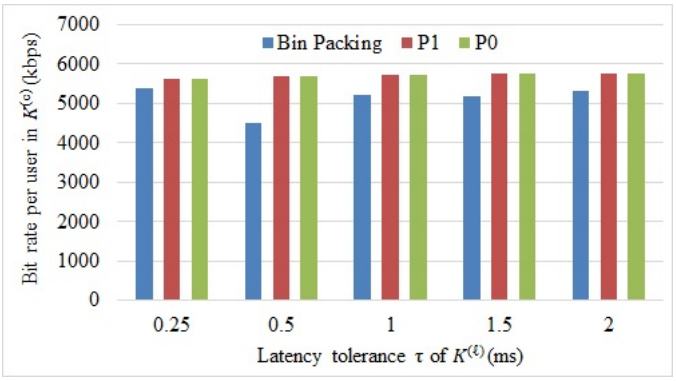

Fig. 3: Bit rate of $\mathcal{K}^{(c)}$ with respect to latency tolerance of $\mathcal{K}^{(\ell)}$ for the bit rate demand of $\mathcal{K}^{(\ell)}$ equals 128 kbps.

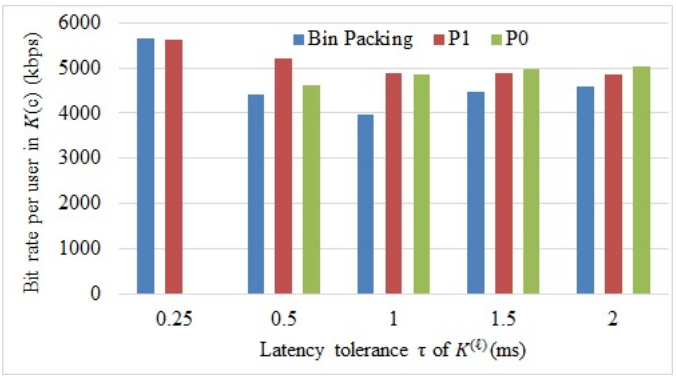

Fig. 4: Bit rate of $\mathcal{K}^{(c)}$ with respect to latency tolerance of $\mathcal{K}^{(\ell)}$ for the bit rate demand of $\mathcal{K}^{(\ell)}$ equals 256 kbps.

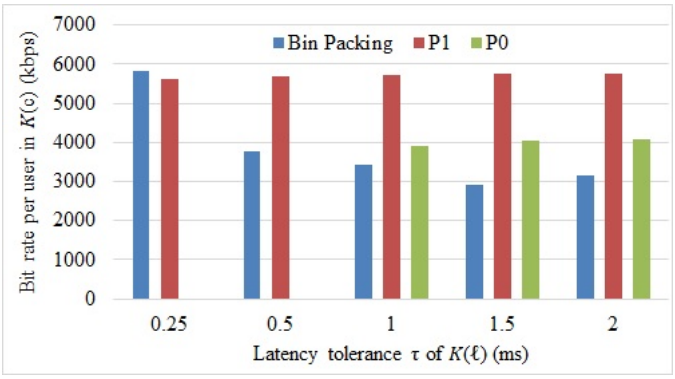

Fig. 5: Bit rate of $\mathcal{K}^{(c)}$ with respect to latency tolerance of $\mathcal{K}^{(\ell)}$ for the bit rate demand of $\mathcal{K}^{(\ell)}$ equals 512 kbps.

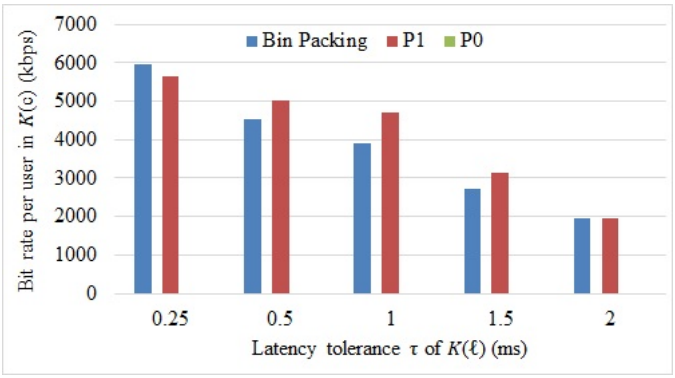

Fig. 6: Bit rate of $\mathcal{K}^{c}$ with respect to latency tolerance of $\mathcal{K}^{(\ell)}$ for the bit rate demand of $\mathcal{K}^{(\ell)}$ equals 1024 kbps.

TABLE II: Satisfied URLLC demands ratio in %.

τ	Problem	64 kbps	256 kbps	512 kbps	1024 kbps
0.25 ms	Bin Packing	100%	20%	0%	0%
	P1	100%	0%	0%	0%
	P0	100%	0%	0%	0%
0.5 ms	Bin Packing	100%	40%	60%	0%
	P1	100%	40%	20%	0%
	P0	100%	100%	0%	0%
1 ms	Bin Packing	100%	100%	80%	40%
	P1	100%	80%	40%	0%
	P0	100%	100%	100%	0%
1.5 ms	Bin Packing	100%	100%	100%	60%
	P1	100%	100%	100%	20%
	P0	100%	100%	100%	0%
2 ms	Bin Packing	100%	100%	100%	80%
	P1	100%	100%	100%	60%
	P0	100%	100%	100%	0%

VI. Conclusion

In this paper, we have studied the joint URLLC and eMBB resource allocation, over the envisioned flexible physical layer architecture for 5G, by accounting for the conflicts. Our goal is to determine the resource allocation to meet data rate demands and latency constraints of URLLC services without compromising eMBB data rate. We formulated the global problem as an integer linear program with hard URLLC data rate constraints, and then proposed to re-formulate the considered problem with less stringent constraints to improve the feasibility of the problem. Moreover, a heuristic solution with a near-linear complexity is proposed which solves the problem by disassociating the URLLC and eMBB services and scheduled the former first to meet their latency requirements. Numerical results have demonstrated that, *i)* the new re-formulation with soft URLLC data rate constraints is always feasible which is not the case when we consider hard constraints, and, *ii)* in terms of URLLC coverage, the proposed heuristic performs better than the optimal solutions which cannot always produce feasible resource allocation. In future work, other heuristic solutions that meet probabilistic QoS requirements of other kinds of services, such as massive machine-type communications, at high load conditions of network traffic, will be investigated.

References

[1] ITU, "5G Overview," Setting the Scene for 5G: Opportunities and Challenges, Geneva: International Telecommunication Union (ITU), 2018.

[2] 3GPP, "NR; Physical channels and modulation," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 38.211, 03 2020, version 16.1.0.

[3] Y. Zhao, X. Chi, L. Qian, Y. Zhu, and F. Hou, "Resource allocation and slicing puncture in cellular networks with embb and urllc terminals co-existence," *IEEE Internet of Things Journal*, 2022.

[4] M. Darabi, V. Jamali, L. Lampe, and R. Schober, "Hybrid puncturing and superposition scheme for joint scheduling of urllc and embb traffic," *IEEE Communications Letters*, vol. 26, no. 5, pp. 1081–1085, 2022.

[5] Y. Huang, Y. T. Hou, and W. Lou, "Deluxe: A dl-based link adaptation for urllc/embb multiplexing in 5g nr," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 1, pp. 143–162, 2021.

[6] H. Yin, L. Zhang, and S. Roy, "Multiplexing urllc traffic within embb services in 5g nr: Fair scheduling," *IEEE Transactions on Communications*, vol. 69, no. 2, pp. 1080–1093, 2020.

[7] Y. Huang, Y. T. Hou, and W. Lou, "A deep-learning-based link adaptation design for embb/urllc multiplexing in 5g nr," in *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*. IEEE, 2021, pp. 1–10.

[8] J. Li and X. Zhang, "Deep reinforcement learning-based joint scheduling of embb and urllc in 5g networks," *IEEE Wireless Communications Letters*, vol. 9, no. 9, pp. 1543–1546, 2020.

[9] L. You, Q. Liao, N. Pappas, and D. Yuan, "Resource Optimization with Flexible Numerology and Frame Structure for Heterogeneous Services," *IEEE Communications Letters*, vol. 22, no. 12, pp. 2579–2582, 2018.

[10] O. Semiari, W. Saad, M. Bennis, and M. Debbah, "Integrated Millimeter Wave and Sub-6 GHz Wireless Networks: A Roadmap for Joint Mobile Broadband and Ultra-Reliable Low-Latency Communications," *IEEE Wireless Communications*, vol. 26, no. 2, pp. 109–115, 2019.

[11] S. Skaperas, N. Ferdosian, A. Chorti, and L. Mamatas, "Scheduling optimization of heterogeneous services by resolving conflicts," *arXiv preprint arXiv:2103.01897*, 2021.

[12] N. Ferdosian, S. Skaperas, A. Chorti, and L. Mamatas, "Conflict-aware multi-numerology radio resource allocation for heterogeneous services," in *2021 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE, 2021, pp. 1–6.

[13] H. Miao and M. Faerber, "Physical downlink control channel for 5g new radio," in *2017 European conference on networks and communications (EuCNC)*. IEEE, 2017, pp. 1–5.

[14] A. Yazar and H. Arslan, "A Flexibility Metric and Optimization Methods for Mixed Numerologies in 5G and Beyond," *IEEE Access*, vol. 6, pp. 3755–3764, 2018.