Dynamic Resource Trading in Sliced Mobile Networks

Özgür Umut Akgül, Ilaria Malanchini, and Antonio Capone

***Abstract*—Expanding the market of mobile network services and defining solutions that are cost efficient are the key challenges for next generation mobile networks. Network slicing is commonly considered to be the main instrument to exploit the flexibility of the new radio interface and core network functions. It targets splitting resources among services with different requirements and tailoring system parameters according to their needs. Regulation authorities also recognize network slicing as a way of opening the market to new players who can specialize in providing new mobile services acting as "tenants" of the slices. Resources can also be distributed between infrastructure providers and tenants so that they meet the requirements of the services offered. In this paper, we propose a model for dynamic trading of mobile network resources in a market that enables automatic optimization of technical parameters and of economic prices according to high level policies defined by the tenants. We introduce a mathematical formulation for the problems of resource allocation and price definition and show how the proposed approach can cope with quite diverse service scenarios presenting a large set of numerical results.**

***Index Terms*—Network slicing, infrastructure sharing, wireless market, pricing mechanism, dynamic resource sharing**

I. Introduction

THE traditional business model of mobile networks is centered on operators who acquire licenses for spectrum use, build their own infrastructure, and control the resource allocation according to their needs. This model is currently being challenged by a number of economic, regulatory, and technical circumstances, which are expected to change the mobile landscape in the near future.

The first well known factor that is challenging this model is the exponential growth of mobile traffic (cf. [1]) that is pushing operators to rapidly expand the capacity of their network with technology upgrades, coverage densification, and spectrum refarming. Unfortunately, the average revenues per user are not growing with the same pace (in some countries they are even decreasing) and the number of traditional users can no longer be increased. This is leading to an aggressive cost optimization and reduction that is not sustainable in the long run. A possible solution to the problem is the evolution of the technology towards supporting a larger set of applications beside the traditional mobile broadband. It is important, that not only the market expands but we use the network infrastructure intelligently as well to further stimulate the digital growth.

Research and standardization work items on 5G networks during the past few years have similarly been focusing on forming a new technology not only to be able to improve the performance of the previous network technologies, but also to support a wide range of vertical applications with diverse and stringent requirements in terms of throughput, delay, reliability and energy [2]. However, due to some fundamental technical limits, increasing the performance significantly, while satisfying all these heterogeneous constraints, is simply not possible, and the network must be optimized depending on the specific application domain. The concept of network slicing has been introduced with the goal of allowing resource allocation to different applications and traffic classes so that it meets the various quality requirements [3].

Even if network slicing can be seen as a precious tool for operators to manage their new generation networks, it poses new challenges as well. A straightforward way of allocating resources to different slices is through (almost) static partitioning, which can however lead to low efficiency. Dynamic resource allocation can be a solution, but it must accurately consider traffic evolution and performance constraints of all applications. Slicing the network might naturally generate new participants in the market. The operators of the network slices, named "tenants" in the 5G terminology, acquire resources from the traditional operators, who are turning into infrastructure providers in this changing environment. From the regulation authorities perspective, using slicing as a tool for infrastructure sharing, is a way of creating new market opportunities and exploring new spectrum licensing strategies.

The idea of infrastructure sharing among multiple virtual mobile operators has long been under considerations. Among the alternative sharing approaches listed by the Organization for Economic Co-operation and Development (OECD) report, active sharing is considered to be the most cost-efficient sharing approach [4]. Active sharing includes sharing both active network elements and spectrum resources. Virtual operators can then share resources with other operators and decrease costs [5]. Although a number of different sharing scenarios exist, the most common one includes a single infrastructure provider and a set of virtual mobile network operators (MVNOs) who acquire resources to serve their users. Note that MVNOs and tenants are similar in the sense that they both manage resources and can provide specialized services, the former in legacy networks while the latter as independent entities. For a given quality target, sharing allows saving resources by exploiting the multiplexing gain. The increased efficiency in resource usage and the adaptability to traffic conditions, are clear advantages [6] [7]. Infrastructure sharing has some similarities with resource sharing in Cognitive Radio Networks (CRNs) [8], but with the fundamental difference that

Ö. U. Akgül and A. Capone are with Dipartimento di Elettronica, Informazione e Bioingegneria (DEIB), Politecnico di Milano, Milano, Italy.

I. Malanchini is with Nokia Bell Labs, Stuttgart, Germany.

tenants (or MVNOs) have equal rights to access resources and, therefore, the problem is basically about resource negotiation rather than opportunistic access.

Most of the proposed sharing models rely on pre-negotiated service level agreements (SLAs) which regulate responsibilities of each party and define the fraction of resources to be assigned. Obviously, long term agreements with static resource assignments are not able to follow the fluctuations in the network demand [9]. Moreover, in wireless networks, there are some geographical areas that are not profitable for the virtual operators but still need to be covered by the infrastructure provider and the associated costs are hard to be mapped into SLAs. For these reasons, dynamic sharing of infrastructure resources is a more attractive alternative where virtual operators or tenants can negotiate resource allocation based on the needs following traffic and channel fluctuations [5] [10]. As argued in [7], the dynamic adjustment of the allocated resources, gives operators the possibility to take more business risks and thus, a dynamically shared wireless market tends to foster innovation. Considering all these aforementioned factors though, ensuring quality with heterogeneous traffic and with different performance parameters still need to be addressed in order to apply infrastructure sharing to network slicing scenarios.

Unlike infrastructure sharing, network slicing is a relatively new concept. Despite the commonly accepted definition of vertically grouped network resources, the specific negotiable attributes of each slice and the tools for service differentiation are still under discussion in the related literature and standardization bodies. In this work, we adopt the concept of a slice as a set of dedicated network resources assigned for specific services in a time interval. In order to assign resources to slices efficiently, the channel conditions, traffic characteristics and variations, and service heterogeneity must be considered [11]. The benefits of network slicing are investigated in [12]–[14] considering static SLAs without dynamic resource adaptation. The resource sharing among tenants in a sliced network is also investigated in [15] and [16]. However, built upon well-defined SLA shares, these works are unable to offer the needed flexibility in the next generation wireless networks. Moreover, they do not consider the long-term evolution of the infrastructure resources, which requires a dynamic resource pricing in line with the required capacity expansion. On a different note, [17] focus on the design of an optimum contract (i.e. SLA) among a set of infrastructure providers and a single MVNO. However, regardless of how well the agreed SLA is designed, the proposed over-restrictive structure prevents the exploitation of the dynamic network conditions (e.g. variations in the traffic demand or channel conditions). A virtualization framework is proposed in [18], where the resources are scaled according to tenants' dynamic needs and fairness is guaranteed not only between tenants, but also between users of different services. The model however does not consider adaptation to channel conditions and economic aspects of resource trading. In [19], we have proposed the first step towards dynamic network slicing in a shared network where tenants are able to renegotiate their slice sizes. In our proposed scheme, tenants retain service level guarantees, but they can revisit the agreements on the allocated resources in a very short time frame so that they can exploit fluctuations in traffic and channel condition and can efficiently control costs.

An important element for tenants and their business strategies (i.e. making long term plans, analyzing the possible risks and performing innovation) is a reasonable and predictable pricing model [7]. In the conventional network provisioning model, the infrastructure provider (whether it is a local operator or a specialized entity) charges tenants according to costs associated to the long-term infrastructure expansion strategy. This long-term strategy may not always be in line with the changes of the market and is definitely not able to meet all the tenants' interests [20]. The pricing model of infrastructure providers can therefore create barriers for the entrance of new players, as already shown for the traditional virtual mobile operator approach in [21]. The structure of the competition based on geographically distributed resources tends to favor a small number of major operators [22], eventually leading to a monopoly that can slow down innovation [7]. However, with dynamic infrastructure sharing, since the resources are pooled and tenants can adjust their shares dynamically, a more efficient and neutral pricing framework can be potentially achieved [23].

A reasonable approach is that of using variable market-driven prices and allowing tenants trading the resources based on needs and within short time frames. Unfortunately, it is not possible to understand the relationship between the economic aspects and the technical performance without a well-defined model. Such a model would also enable, tenants to exploit the full potential of dynamic sharing. Thus, a scheme able to automatically define prices and resource allocation based on high level tenant strategies and traffic estimation is of fundamental importance [9]. Even if there is extensive literature focused on the economic aspects (such as [21], [24]) and technical considerations (such as [25], [26]) separately, the definition of techno-economic models for resource sharing in sliced networks is still an uncovered area.

In this paper, we propose a dynamic wireless market model that can flexibly adjust the share of resources, assigned to network slices, to achieve the maximum utility for tenants. The contributions of this work can be summarized as follows. We propose:

- an enhanced wireless market model based on different services and quality requirements using dynamic pricing through the formulation (1a)-(1h) in Section III-A
- a two-step approach for adapting the network slices according to the fluctuations of the achievable rate and the variations of the traffic mix in short time scale in Section III-B
- a dynamic updating mechanism for optimizing the slice configuration based on the evolution of the resource distributions over time and the achieved spectral efficiency in Section III-C
- exploitation of the anticipatory information of the achievable rates for the resource allocation in Section III-D

The remainder of the paper is organized as follows: Section II contains the system model and the main assumptions. Following the system model, the optimization model is pre-

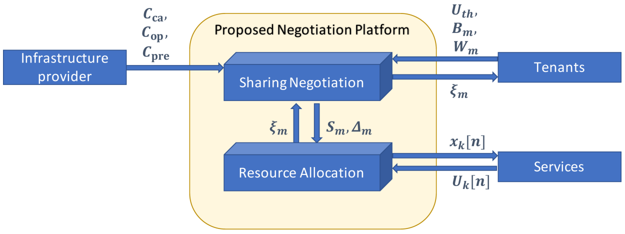


Fig. 1. Proposed negotiation platform

sented in Section III. In Section IV, the behavior and the validity of the optimization model are investigated through simulations. Section V concludes the paper and discusses possible extensions of the proposed approach.

II. System model

In order to provide a flexible and adaptive resource sharing algorithm for network slicing in a multi-tenant environment, we introduce a dynamic negotiation platform, shown in Fig. 1, which interacts with the different stakeholders and, based on the received inputs, allocates resources, assesses the performance and evaluates the corresponding costs. Table I summarizes the notation adopted in this work. In our system model, the stakeholders are as follows: a set of tenants M, with index m, sharing the downlink of a base station, an infrastructure provider (InP) who provides the shared base station, and a set of users K, who require heterogeneous services from their corresponding tenant. Also, let the set K_m be the set of users of tenant m, and thus $\sum_{m \in M} |K_m| = |K|$. In particular, we assume that each user requests only one type of service and the number of active users per tenant, i.e. the cardinality of K_m, is the same for all tenants (i.e. tenants have similar market shares). Note that such assumptions do not limit the generality of the proposed model, and they are made mainly for the sake of better understanding how the proposed framework is able to adapt the resource allocation to different slices based on different service requirements (and not due to the different traffic load of each slice). Generally speaking, our algorithm can cope with nonequivalent user distributions, which would lead to similar average achieved utilities among users of the same service type, but different resource distributions among tenants (scaled according to the total number of users). Time is discretized into slots, n, where N is the set of all time slots, i.e. simulation horizon.

Service level agreements regulate the sharing of resources between the InP and the tenants. We assume that the slice of tenant m is defined by three parameters, S_m, Δ_m and W_m. $S_m \in (0, 1)$, referred to as *guaranteed resource share*, indicates the ratio of resources that tenant m expects to receive on average. Furthermore, to guarantee flexibility, we assume that the resource allocation can deviate from the guaranteed resource share. In particular, the *maximum average allowed deviation* is denoted as Δ_m (as introduced in [27]). Namely, Δ_m sets the limit on the maximum deviation from S_m within a tenant-specific time window, W_m (over which the average is computed). Therefore, within each time window W_m, tenant m receives (on average) a fraction of resources between $(S_m - \Delta_m, S_m + \Delta_m)$. Note that, the time constraint imposed by the time window W_m can also be used to achieve differentiation among tenants and corresponding services. As opposed to [27], where sharing parameters were assumed to be constant, in this work S_m and Δ_m are periodically updated to fully exploit the advantages of dynamic trading. Namely, the period of such updates is set by the InP and is referred to as “renegotiation interval” (RI).

TABLE I
Summary of adopted notation

Symbol	Meaning
M	Set of tenants
m	Index of a specific tenant
K	Set of users
k	Index of a particular user
N	Total simulation horizon
n	Index of a particular time slot
S_m	Guaranteed resource share
Δ_m	Maximum average allowed deviation
W_m	Time window of tenant m
RI	Renegotiation interval
U_{th}	Utility target
B_m	Budget of tenant m
C_{op}	Operational expenses
C_{ca}	Capital expenses
C_{pre}	Pressure cost
$x_k[n]$	Assigned wireless resources to user k at n
$r_k[n]$	Achievable rate of user k at n
$\xi_m[n]$	Gap between the expected and achieved utility for tenant m
U_1	Utility of a not-activated service
R_1	The minimum rate for a service to be actived
U_2	Utility of a service that receives standard quality
R_2	Required achievable rate for standard quality
U_3	Maximum achievable utility for a service
R_3	Saturation point for a utility function

Furthermore, we assume that tenants set their utility targets[1], $U_{\text{th}} \in (0, 1)$ and their available budgets, B_m. In contrast, the InP is responsible for setting the respective costs of the wireless resources (c.f. Fig. 1). The total cost of the wireless resources consists of three parts, i.e., capital expenses, C_{ca}, operational expenses, C_{op} and pressure cost, C_{pre}. We assumed that the infrastructure provider does not have profit constraints and his main objective is to run a sustainable business model. Therefore, C_{ca} and C_{op} are scaling the cost of the conventional infrastructure and the operational cost of the resources. The pressure cost helps the regularization of the resource allocation. Similar to any demand based market, the pressure cost also regulates the resource consumption. For instance, if the system does not have sufficient resources to satisfy all the users, i.e. *resource scarcity*, the pressure cost is set to be greater than zero, so that tenants will have less incentive to buy resources (in terms of S_m), but more incentive to trade resources (via Δ_m). In contrast, in case the system has more than sufficient resources for all the users, i.e, *resource surplus*, the pressure cost is set to zero, reducing the overall cost and increasing the incentive to buy. Moreover, pressure cost can be seen as a way for the InP to collect the necessary

[1] In this work, we assume that all tenants select the same utility target, however, an analysis of the effects of choosing different utility targets, as means of differentiation for the tenants, has been proposed in [19].

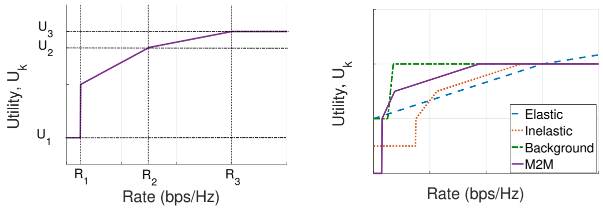
revenue in order to upgrade or expand the existing network capacity (in case of resource scarcity). The pricing mechanism is further explained in Section III.

Based on all the inputs described above as well as the users' channel conditions, the proposed negotiation platform optimally allocates the resources to the different slices. Namely, let $x_k[n]$ be the wireless resources allocated to user k at time slot n, and $r_k[n]$ the achievable rate for user k at time slot n. The actual achieved rate of user k at time slot n is then given by $r_k[n]x_k[n]$. Furthermore, we assume that each user k produces a utility $U_k[n]$ that depends on the achieved rate as well as the requested service type. The average achieved utility of tenant m at n is the average achieved utility over all its users, i.e. $\sum_{k\in K_m}\frac{U_k[n]}{|K_m|}$. The difference between the utility target U_{th} and the average achieved utility is defined as the tenant's gap and denoted by $\xi_m[n]$. Such gap is used to measure the performance of the proposed resource sharing algorithm, where the best possible operating point is the one for which the gap is equal to zero.

A. Utility Functions

Even if the quality perceived by the users depends on several elements, we assume in this paper that it can be quantified by using the achieved rate. We therefore consider a generic continuous utility function $U_k(R_k[n])$, function of the average achieved rate $R_k[n]$, as shown in Fig. 2(a). This function is used in our framework to model different utilities for heterogeneous services. Namely, each specific service function is determined by varying six parameters, i.e. U_1, U_2, U_3, R_1, R_2 and R_3. The minimum rate, required to consider a service as active, is assumed to be R_1. When the average achieved rate is lower than R_1, i.e. $R_k[n] < R_1$, the utility function returns the utility value $U_1 \leq 0$. In case the service achieves the average rate of R_1 than the utility returns zero. R_2 represents the standard quality for the services where the utility function provides a utility value equal to U_2. Finally, R_3 indicates the saturation point for the utility function, after which the function becomes non-increasing. The maximum utility for the service type, that is achieved at $R_k[n] = R_3$, is given by U_3. Note that the modeling of the function by using four regions (hence six parameters) reflects the idea that the service quality can fall in either one of the following categorizes: no service, low quality, high quality, maximum quality (above which no further advantage is perceived by the user). Furthermore, the choice of piecewise linear functions is mainly due to mathematical tractability, but this does not limit the validity of the proposed sharing platform, which can incorporate also more complex functions.

Using the generic utility function presented above, we defined the specific utility functions for four service types envisioned for 5G: elastic services, inelastic services, machine to machine (M2M) services and background services. In particular, prioritization (or fairness) among services (and in particular between critical and non-critical services) can be set by using different (or equal) slopes of the utility functions (e.g. between $R_1 - R_2$ and $R_2 - R_3$). A detailed description of the specific utility functions chosen for the four different services is provided here and presented in Fig. 2(b):



(a) Generic utility function

(b) Exemplary utility functions

Fig. 2. Generic utility function (left) and exemplary utility functions per service type (right)

1) Elastic traffic: By definition, elastic services, do not have strict rate or delay constraints. Thus, we consider them to be active as soon as the average achieved rate is greater than zero, $R_k[n] > 0$, meaning $R_1 = 0$ and $U_1 = 0$. Moreover, for elastic users we do not set any upper bound on their rate expectations, meaning $R_3 \to \infty$ and $U_3 \to \infty$. Since the service requirements are quite flexible, the utility function has a smaller slope compared to the ones of the other services in any of the same regions.

2) Inelastic traffic: Inelastic services, being a demanding service type, require a minimum rate to provide service availability, e.g. as in case of video streaming. For this reason, we set R_1 relatively high, e.g. to provide a continuous service experience for the users. Similar to video streaming, the utility of inelastic services (i.e. perceived quality) is highly affected by the fluctuations of the achieved rate (e.g., the variations in the video quality between 144p and 720p). Therefore, we impose a steep slope between R_1 and R_2 to force a quick increase in the utility as a function of the average achieved rate. However, after reaching a certain quality, the increase in the average achieved rate is less noticeable, and therefore, we choose a lower slope between R_2 and R_3. As mentioned above, to enforce fairness, the slope of inelastic services between R_2 and R_3 is equal to the one of elastic services between R_1 and R_2.

3) Background traffic: These services usually run in the background and require relatively low rate. As soon as this is achieved, the utility function reaches its saturation point, i.e. $R_2 = R_3$. Furthermore, since they do not have a strict delay constraint, the minimum utility is considered to be zero, i.e. $U_1 = 0$.

4) Machine to machine (M2M) traffic: We group M2M services envisioned in 5G into three main categories and model the M2M requests as a mixture of all three service types. Namely, the M2M utility function represents three types of services, i.e. emergency, low-rate-delay-sensitive and rate sensitive. We assume that M2M incorporates all three services but how the tenant-specific resource distribution is handled within M2M is not in the scope of this work. However, we assume that tenants will prioritize their M2M services and assign resources accordingly. The emergency services, which require low rate but with high priority, are modeled with the R_1 rate. Since not achieving this rate can have a dramatic impact on the system, we set U_1 to a negative value.

Consequently, not serving the emergency services results in a big gap for the tenants. The low rate and delay-sensitive M2M applications are modeled between R_1 and R_2. As shown in Fig. 2(b), since there is a delay constraint, the utility function characteristic has a relatively large slope for these types of services. Finally, for the rate constrained services, as the name suggests, achieving higher rates has higher priority than having a low delay. Therefore they are modeled between R_2 and R_3 with a relatively smaller slope.

III. Scheduling problem and analysis

A. Mathematical programming formulation

The scheduler of the shared base station allocates resources by using the optimization model formulated in (1a)–(1h). The proposed techno-economic model runs in real time and controls both the resource allocation and the respective price negotiations in an online manner. Namely, the resource shares of the tenants are dynamically chosen based on their Quality of Service (QoS) expectations (i.e. the achieved rate per user and tenant's time window, W_m), the channel conditions and tenant's market power (i.e. their budget, number of users and traffic mix). The optimizer dynamically assigns resources to each slice per service type and per tenant to minimize the total gap, i.e., as in (1a), $\sum_{m \in M} \xi_m$. By jointly optimizing the resource allocations for all tenants, the scheduler has the flexibility to prioritize the users with the best channel conditions and therefore maximize the utilization of the resources and spectral efficiency.

$$\min_{x_k[n]} \sum_{m \in M} \xi_m[n] \tag{1a}$$

$$\text{s.t.} \;\; U_{\text{th}} - \sum_{k \in K_m} U_k(R_k[n]) \leq \xi_m, \; \forall m \in M, \tag{1b}$$

$$\epsilon_m[n] = \left(\frac{1}{(a_m+1)} \sum_{i=n-a_m}^{n} \sum_{k \in K_m} x_k[i] \right) - S_m, \; \forall m \in M, \tag{1c}$$

$$|\epsilon_m[n]| \leq \Delta_m, \quad \forall m \in M, \quad \forall n \in N, \tag{1d}$$

$$\begin{aligned} \sum_{i=n-a_m}^{n} & (S_m(C_{\text{ca}} + C_{\text{op}}) + \epsilon_m[i] C_{\text{op}} + f_{\text{pre}}(C_{\text{pre}}, \; \xi_m)) \\ & \leq B_m(a_m+1), \forall m \in M, \end{aligned} \tag{1e}$$

$$0 \leq \Delta_m \leq \frac{1}{a_m+1} \sum_{i=n-a_m}^{n} \sum_{k \in K_{m,elastic}} x_k[i], \quad \forall m \in M, \tag{1f}$$

$$\sum_{k \in K} x_k[n] \leq 1, \quad x_k[n] \geq 0, \quad \forall k \in K, \tag{1g}$$

$$\sum_{m \in M} S_m \leq 1 \;, \quad S_m \geq 0, \quad \forall m \in M, \tag{1h}$$

Constraint (1b) sets the gap of tenant m as the difference between its target utility (i.e. U_{th}) and the sum of the achieved utility over its users (i.e. sum of $U_k(R_k[n])$). Note that within each time window, of length W_m, we evaluate the average by considering the values from the beginning of the time window to the current time slot n, i.e. over $a_m + 1$ time slots, where $a_m \equiv n - 1 \bmod W_m$. Therefore, the average achieved rate for user k at time slot n is

$$R_k[n] = \frac{1}{(1+a_m)} \left(\sum_{i=n-a_m}^{n} x_k[i] r_k[i] \right).$$

Furthermore, we assume that all the users have the same importance to the tenants, thus, $U_3 = U_{\text{th}}/K_m \; \forall k \in K_m$. By selecting the same value of maximum utility, U_3, for all the users, the tenants also guarantee neutrality in their provided services. However, depending on the agreements between the service providers and the tenants, as well as in accordance with regulatory constraints, this value can be changed, thus allowing our model to include also non-neutral services.

The instantaneous average deviation from the guaranteed resource share, $\epsilon_m[n]$, is given in (1c). Namely, the instantaneous deviation at n for tenant m is given by subtracting the guaranteed resource share S_m from the average assigned resource to the users of m, where the average, as done for the average achieved rate, is evaluated from the beginning of the current time window till time slot n.

Constraint (1d) ensures that $\epsilon_m[n]$ is not larger than Δ_m, which by definition is the tenant-specific maximum allowed deviation. Note that ϵ_m can either be positive or negative, i.e. $\epsilon_m \in [-\Delta_m, \Delta_m]$. The former case indicates that the tenant has received – on average and within the current time window – more resources than S_m, while the latter case corresponds to the opposite.

Furthermore, constraint (1e) sets the budget constraint per tenant. The first term of the left-hand side scales both CAPEX and OPEX according to S_m, which means that in case of no sharing (when $\Delta_m = 0$), the tenant will have to pay for the requested resources. The second term, i.e. $\epsilon_m[n]C_{\text{op}}$, allows tenants to dynamically adjust their total costs according to their resource usage and budget. Namely, if a tenants' actual resource usage is less than the guaranteed resource share (i.e. $\epsilon_m[n] < 0$), then the tenant will not pay for the OPEX cost of the unused resources. The third term of the budget constraint is a function, $f_{\text{pre}}(C_{\text{pre}}, \xi_m)$, of the pressure cost unit C_{pre}, defined by the InP, and of the tenant's gap ξ_m. Namely, the gap considered for the evaluation of the pressure cost is the one obtained at the end of the previous time window (i.e. it varies at every time window, but kept constant within the same time window). The effects of the pressure cost term are evident when, e.g., there is a resource demand that exceeds the available resources. In this case, since the resources are limited, the tenants face non-zero gaps, $\xi_m > 0$, which corresponds to an increase of the pressure cost as well as of the total cost of resources. This increase in the cost pushes tenants to increase their Δ_m and decrease S_m. In the extreme case, tenants opt for full sharing, i.e. $\Delta_m = 1$, which allows the scheduler to provide the most spectrum efficient and cost efficient allocation. Moreover, the pressure cost allows the infrastructure provider to accumulate additional revenues not directly used for the current infrastructure, but envisioned to support capacity expansion to meet the tenants' quality requirements. In this respect, scaling the pressure cost by the gap provides an accurate estimation of the capacity needed to satisfy all the tenants.

Constraint (1f) forces the maximum deviation Δ_m to be at maximum, equal to the resources assigned to the elastic users of tenant m, which implies that tenants are not willing to trade resources used for critical, i.e. non-elastic, services. By setting $\Delta_m = 0$, tenants indicate that their services are non-elastic and they require the resources they stated by S_m. However, in this case, they also lose the flexibility to adapt to traffic dynamics. Finally, (1g) ensures that the assigned resources do not exceed the total available resources in the system and, similarly, (1h) limits the sum of all S_m to the total amount of resources.

B. Two-step approach

The formulation presented in the previous section is able to capture the dynamics of the resource negotiation, considering both the scheduling aspects and the economical constraints (prices and budgets). However, due to its computational complexity, it is not suitable to be used in real-time. Therefore, we decided to split the problem into two, namely the decision on the real time resource allocation and the decision on the negotiations of the sharing parameters (S_m, Δ_m).

In particular, we separate our model into two sub-problems, P_1 and P_2. The first problem, P_1, focuses on the real time resource allocation with the objective of minimizing the total gap and it is solved at every time slot n. During P_1, the sharing parameters (S_m, Δ_m) are assumed to be constant and, therefore, the constraints that regulates the sharing (i.e. (1f) and (1h)) are inactive. The outcome of P_1 is then given by the allocated resources and corresponding tenants' gaps. The second problem, P_2, is solved at the end of each time window, by the update of the sharing parameters according to the channel conditions of the users at that time and the tenants' targets (i.e. in terms of U_{th}). In this case, the objective is to find the best sharing parameters so that the total gap of the previous time window is minimized. Namely, P_2 receives the achievable rates from the previous time window as input and derives the optimum sharing parameters S_m^{opt} and Δ_m^{opt} by solving (1a)−(1h).

Note that even if both problems, P_1 and P_2 are derived from the same formulation (1a)−(1h), they are actually separate and different problems since the active variables (and constraints) are different. Formally, P_1 and P_2 are defined as follows:

$$\begin{aligned} P1 :=& \min_{x_k[n]} \xi_m[n] \\ & \text{s.t. (1b), (1c), (1d), (1e), (1g)} \\ P2 :=& \min_{x_k[n], S_m, \Delta_m} \xi_m[n] \\ & \text{s.t. (1b), (1c), (1d), (1e), (1f), (1g), (1h)} \end{aligned}$$

C. Update mechanism

As described above, P_2 derives the optimum sharing parameters, i.e. S_m^{opt} and Δ_m^{opt}, for all the tenants, in order to achieve the minimum total gap $\sum\limits_{m \in M} \xi_m^{\text{opt}}$. However, it is important to remember that the optimization problem is solved by using the achievable rates of the previous time window only, meaning that S_m^{opt} and Δ_m^{opt} are optimal only with respect to the previous window. Therefore, to capture the statistic nature of the channel over a longer time span, the sharing parameters are updated with a weighted approach. Namely, the new values for the sharing parameters, S_m^{new} and Δ_m^{new} to be used in the upcoming time window, are derived as:

$$S_m^{\text{new}} = \alpha_m S_m^{\text{opt}} + (1 - \alpha_m) S_m^{\text{old}}, \tag{2}$$

$$\Delta_m^{\text{new}} = \alpha_m \Delta_m^{\text{opt}} + (1 - \alpha_m) \Delta_m^{\text{old}}. \tag{3}$$

where the feature scaling coefficient, α_m, is calculated as:

$$\alpha_m = \frac{\xi_m - \xi_m^{\text{opt}}}{\xi_m + \xi_m^{\text{opt}}}. \tag{4}$$

By definition α_m measures the difference between the achievable optimum gap and the actual gap observed by the tenant. For instance, when $\xi_m = \xi_m^{\text{opt}} = 0$, the feature scaling coefficient is also 0, which means that the most recently calculated sharing parameters are the optimum values and are therefore also used for the upcoming time window without scaling. In general, with the proposed update mechanism, our framework is able to adapt to the varying channel conditions in a reactive manner. The sharing parameters are automatically updated to provide service quality which is satisfying the tenants' requirements while maintaining proportional fairness among them. A thorough study of the α_m selection and its effects on the model's adaptability has been proposed in [9].

In summary, the following Algorithm 1 is used to solve the dynamic network slicing and resource trading problem introduced in (1a)−(1h).

Algorithm 1 Two-step algorithm with update mechanism

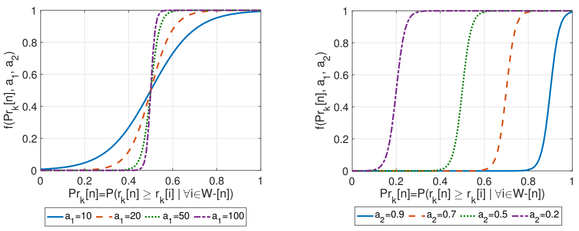
Input: $C_{\text{ca}}, C_{\text{op}}, C_{\text{pre}}, U_{\text{th}}, U_k(R_k), B_m, r_k, N, W_m, RI$

```
for Every Renegotiation Interval $RI$ do
    for Every time slot in $RI$, $n \in RI$ do
        $x_k[n], \xi_m[n] \leftarrow P_1(r_k[n], S_m, \Delta_m)$
    $S_m^{\text{opt}}, \Delta_m^{\text{opt}}, x_k^{\text{opt}}, \xi_m^{\text{opt}} \leftarrow P_2(r_k[n - RI : n - 1])$
    $\alpha_m \leftarrow \frac{\xi_m - \xi_m^{\text{opt}}}{\xi_m + \xi_m^{\text{opt}}}$
    $S_m^{\text{new}} \leftarrow \alpha_m S_m^{\text{opt}} + (1 - \alpha_m) S_m^{\text{old}}$
    $\Delta_m^{\text{new}} \leftarrow \alpha_m \Delta^{\text{opt}} + (1 - \alpha_m) \Delta^{\text{old}}$
```

D. Exploiting the channel information

The real-time scheduling problem, P_1, exclusively focuses on the optimization of the current time slot n without taking into account the upcoming slots. Thus, it is incapable of fully exploiting the transmission opportunities. As a result, P_1 requires a larger amount of resources compared to the one estimated by P_2 in order to provide comparable performance. As a matter of fact, P_2 derives the minimum values of S_m and Δ_m, in order to minimize the gap, which are however too restricting for P_1. Therefore, to improve the performance of P_1, a channel-aware filter is designed to exploit the statistical information of the channel.

Specifically, we design a channel-aware filter to evaluate the rate expectations for the upcoming time slots of each user, while scheduling the resources for the given time slot n. Even



(a) Varying a_1, when $a_2 = 0.5$

(b) Varying a_2, when $a_1 = 50$

Fig. 3. Variation of the sigmoid function for different a_1 (left) and a_2 (right) values

though prediction techniques of the channel characteristics are out of scope of this paper, we assume that the infrastructure provider can learn a statistical profile of the channel behaviors. Therefore, we assume that there is available data on the probable density rate on each available user, $k \in K$, for the infrastructure providers. Based on this data the probability can be evaluated for each specific user within the given time window, whether the user is in the "best" time slot to assign resources, $Pr_k[n] = P(r_k[n] \geq r_k[i] \;\forall i \in W) \in [0,1]$, i.e. the slot with best channel conditions compared to the other time slots. In particular, a probability value of 0 indicates that the channel condition at slot n is the worst that can ever be observed, thus, the scheduler should avoid assigning resources, while a value of 1 means that the current channel condition is the best possible and therefore as many resources as possible should be assigned. However, we do not use this probability directly, but we filter it as described below before passing it as input to P_1.

We design a two-step filtering function to map the statistical information onto the assignment decisions. As a first step, the statistical information is scaled using a sigmoid function, i.e. $f(Pr_k[n], a_1, a_2) = 1/(1+e^{-a_1(Pr_k[n]-a_2)})$, as presented in Fig. 3. The characteristic of the sigmoid function can be controlled by using two parameters, i.e. $[a_1, a_2]$ (cf. Fig. 3(a) and Fig. 3(b)). The former parameter, a_1, controls the slope of the linear region of the sigmoid and indirectly controls the resource efficiency. Namely, assuming that the number of users is low, decreasing the slope of the linear region leads to a situation where unassigned resources exist while the tenants cannot achieve their goals. In contrast, increasing a_1 results in assigning resources also with bad channel conditions, thus decreasing the efficiency of the channel utilization. The latter parameter, a_2, allows the shift of the sigmoid function (c.f. Fig 3(b)). In this case, choosing large values of a_2 gives advantages only to the users with high probabilities. However, when tenants select small time windows, it leads to unassigned resources even in the presence of gaps. In contrast, small values of a_2 equalizes all users making the filter ineffective.

The output of the sigmoid function, $f(Pr_k[n], a_1, a_2)$, provides an understanding on how good the channel conditions for a specific user are with respect to what the certain user can achieve in the given time window. However, $f(Pr_k[n], a_1, a_2)$ does not give information on how good the channel is with respect to the other users in that time slot. Therefore, this first step of the filtering process might not be sufficient to guide the scheduler when there is a significant difference among the distributions of the users' channel.

Consequently, an additional filtering step is introduced to capture these variations among the users' channel conditions. More specifically, taken the output of the sigmoid function, $f(Pr_k[n], a_1, a_2)$, the second step outputs $f(Pr_k[n], a_1, a_2)^p$, where p is scalar. If the variations in the achievable rates among users are negligible, e.g. the users have similar pathlosses, the p value can be set to 1. In contrast, if the difference is not negligible, a larger value of p should be chosen.

The output of the filter function, referred to as "priority coefficient" and indicated by $\beta_k[n]$, is then used by the scheduler to give priority to the users with the best channel condition (i.e. $\beta_k[n] = 1$) and to discard the users with the worst channel conditions (i.e. $\beta_k[n] = 0$). In order to incorporate this information into P_1, the constraint (1b) is updated as

$$U_{\text{th}} - \sum_{k \in K_m} \beta_k[n] U_k(R_k[n]) \leq \xi_m, \;\forall m \in M. \qquad (5)$$

Since the channel information is used to guide the real-time scheduling algorithm, the gap values calculated by P_2 are then derived without priority coefficients, as given in (1b).

Note that the specific values chosen for $[a_1, \; a_2]$ as well as p, combined with the channel conditions, affect the resource allocation. Hereafter, we do not discuss the policies used by the tenants to select those values, but assumed they are given (i.e. we empirically derived those used for the numerical evaluation).

IV. Simulation Results

In this section, we first present the parameters and the simulation setup used for the evaluation and then show the effectiveness of the proposed algorithms with some numerical results.

A. Parameters and simulation setup

We consider the downlink of a single base station, shared among $|M|$ tenants. Unless specified otherwise, each tenant serves $|K_m| = 4$ users and each user is associated with a specific traffic type, i.e. elastic, inelastic, M2M or background. The total set of users, $K = \cup_m K_m$, is distributed homogeneously in the coverage area of the base station and considered

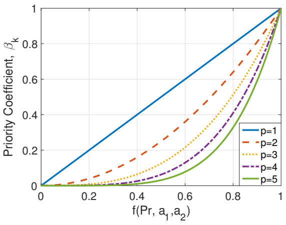


Fig. 4. Changes in the characteristic of the filter function according to the variations of p

TABLE II
SERVICE SPECIFIC PARAMETERS AND THEIR VALUES

Parameter	Elastic	Inelastic	M2M	Background
R_1 (bps/Hz)	0	0.1	0.01	0.05
R_2 (bps/Hz)	1.083	0.225	0.075	0.07
R_3 (bps/Hz)	∞	0.55	0.4	0.07
U_1	0	-0.5	-1	0
U_2	1	0.7	0.7	1
U_3	∞	1	1	1

to be active during the whole simulation, which is set to $N = 5000$ transmission time intervals (TTIs). Depending on the considered technology, capability of the base station and physical constraints, the TTI can be translated into a specific time duration such that enough time is given to the proposed mechanism to converge to the optimum solution. The presented results are averaged over 100 independently generated instances.

The parameters that are used for the utility functions, presented in Fig. 2, are given in Table II. The utility target is $U_{\text{th}} = |K_m|, \ \forall m \in M$. Unless specified otherwise, the length of the time window, W_m, is considered to be equal for all tenants. The time window for the renegotiation interval is assumed to be 80 TTI long. The values used for the costs and budgets are $C_{\text{ca}} = 20, \ C_{\text{op}} = 20, \ B_m = 100, \forall m \in M$. As proposed in [9], when tenants have the same budget, the pressure cost is evaluated as C_{ca} scaled by the number of tenants, i.e. $C_{\text{pre}} = \frac{C_{\text{ca}}}{|M|}$ and $f(C_{\text{pre}}, \xi_m[|W_m|]) = \xi_m[|W_m|] \times C_{\text{pre}}/|W_m|$.

A frequency flat fading channel is assumed between the base station and the users. This is model by using i.i.d. Rayleigh coefficients, which lead to exponential channel gains, $|h_k[n]|^2$. Based on that, the Signal to Interference-plus-Noise Ratio (SINR) is calculated for each user k at each time slot n as:

$$\gamma_k[n] = |h_k[n]|^2 \frac{P d_k^{-\alpha}}{\sigma^2 + I_0}, \tag{6}$$

where P is the transmit power (in Watts), d_k is the distance between the user k and the base station (in meters) and α is the path-loss exponent. In this work, the interference is modeled as the sum of the thermal noise, σ^2 and the average interference, I_0. Therefore, by using (6), the achievable rate of user k at time slot n is expressed by

$$r_k[n] = \log_2(1 + \gamma_k[n]). \tag{7}$$

Finally, the considered filter values (introduced in Section III-D) are set to $a_1 = 10, \ a_2 = 0.5, p = 3$.

B. Time complexity analysis

As briefly analyzed in [9], the renegotiation interval, which is set by the InP, affects the time complexity of the algorithm. Table III depicts the variation of average computation time of P_1 and P_2 depending on the renegotiation interval in a scenario with $|M| = 3$, $|K| = 12$. The simulations are run in Matlab, whereas the optimization problems P_1 and P_2 are solved by the Gurobi commercial solver [28]. The simulations are run on a Intel 2.4 GHz PC with 6 GB of RAM.

The results show that the longer the renegotiation interval is, the longer it takes to solve P_2. This is reasonable though since the algorithm has to find the optimal sharing parameters over a longer time interval. In contrast, the duration of solving the real time scheduler, P_1, is not heavily affected by the length of the renegotiation interval.

TABLE III
EFFECTS OF RENEGOTIATION INTERVAL ON COMPUTATION TIME

Renegotiation Interval	P_1 duration (sec)	P_2 duration (sec)
5 TTIs	0.0015	0.0431
25 TTIs	0.0012	0.1923
50 TTIs	0.0016	0.5069
80 TTIs	0.0011	1.4832
100 TTIs	0.0015	2.4412

Note that both P_1 and P_2 have time constraints dictated by the system model we proposed. Namely, we need to run P_1 every time slot and P_2 every time window. In order to obtain acceptable computation time for real time implementation, two different approaches could be used. On one hand, P_1 could be run using more powerful machine to reduce the computation time. On the other hand, for cases where the computational time of P_2 becomes too large, an alternative heuristic approach could be proposed, which is, however, out of the scope of this paper.

C. Value of channel information

In Sec. III-D, we introduce a channel-aware filter to integrate the statistical channel information into the real time scheduler. Basically, we propose to replace constraint (1b) with constraint (5). The proposed channel-aware approach is a simple prediction algorithm, that evaluates current channel conditions taking into account past observations and future expectations.

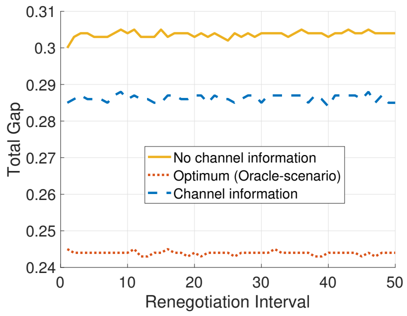


Fig. 5. Effects of integrating channel information on the total gap, for $|M| = 2, \ |K| = 8$

Hereafter, we want to show the effects of exploiting such channel information on the total achieved gap with respect to: (1) the case without channel information (P_1 solved using constraint (1b)) and (2) the case with perfect knowledge of the future channel conditions (Oracle scenario). Fig. 5 shows the results for $|M| = 2$ tenants and $|K| = 8$ users. Our observation is that feeding the model with an estimation of the channel allows the scheduler to better detect the instantaneous opportunities and to increase resource and cost efficiency, by decreasing the total gap.

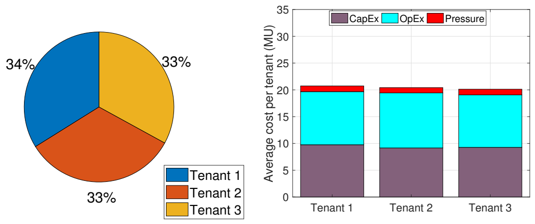
TABLE IV
CHANNEL INFORMATION'S IMPROVEMENT ON THE TOTAL GAP WITH RESPECT TO NO-CHANNEL INFORMATION CASE

$\|K\|$	Improvement of total ξ_m
8	33.2%
16	38.5%
24	38.6%

Table IV shows the effect of increasing the number of users $|K|$ on the total gap, as percentage improvement with respect to no-channel information case. Increasing $|K|$ gives the scheduler a higher flexibility in exploiting the transmission opportunities and also higher probability to detect good time slots. In contrast, when $|K|$ is small, the scheduler needs higher accuracy to detect transmission opportunities. However, we can also observe that the performance improvement saturates when further increasing the number of users. This indicates a limit in the improvement that can be achieved by using this approach.

D. Symmetric traffic scenarios

In this section, we report results for a case in which $|M| = 3$ tenants have symmetric traffic (the same amount of users per service type).



(a) Average resource distribution

(b) Average total cost

Fig. 6. Average resource distribution and average total cost per tenant for $|K = 12|$

Due to the symmetry among tenants, we can observe an equivalent resource and cost distribution, as shown in Fig. 6. This proves that, as desired, in symmetric cases our model behaves perfectly fair among tenants. Furthermore, Fig. 7 reports the average utility per tenant per service and, as above, we can observe that there is a symmetric behavior among tenants, but different prioritization among slices, i.e., services. Namely, due to the utility based prioritization, when the system does not have sufficient resources to fully satisfy all of them, the elastic users are penalized and reach lower utility compared to the other services. Moreover, both inelastic and M2M services are achieving an average utility less than 1 due to the utility function used (c.f. Fig. 2(b)). To be more precise, after reaching the utility value of U_2, all the services have the same slope, that provides fairness between elastic service and the rest of the services.

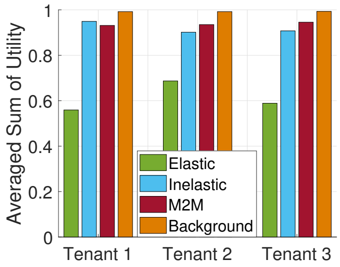
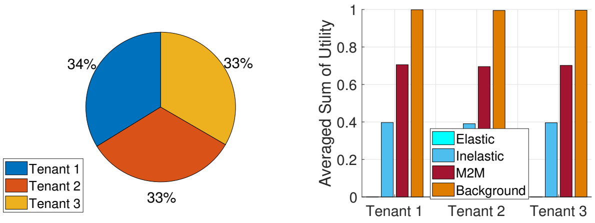


Fig. 7. Average utility per service per tenant

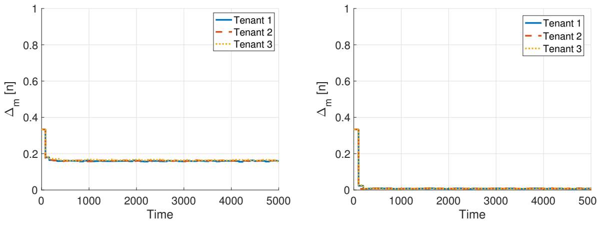
Now, we show how the proposed framework reacts to load changes. In particular, we increase the number of users of each tenant to $|K_m| = 16$ users (i.e. total number of users $|K| = 48$), while keeping fixed the system capacity and utility function parameters. As shown in Fig. 8, despite the strong competition for resources, fairness among tenants is still achieved. Moreover, in Fig. 8(b) even more emphasis is shown on the prioritization given to different services. As expected, the elastic traffic, which has the lowest priority, is being affected mostly from the resource scarcity. In contrast, such prioritization guarantees that the emergency and low-rate-delay-sensitive M2M traffic (i.e. defined in Section II-A4) can achieve the service expectations even in such an extreme scenario (which is proved by the fact that for this service type at least utility equal to U_2 is achieved).



(a) Average resource distribution

(b) Average utility per service

Fig. 8. Average resource distribution and average utility per service per tenant for $|K| = 48$



(a) Δ_m variation for $|K| = 12$

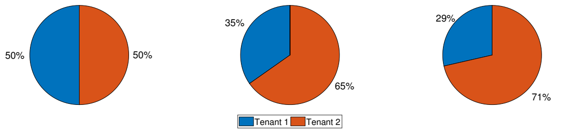
(b) Δ_m variation for $|K| = 48$

Fig. 9. Adaptation of Δ_m to the increasing traffic

Another interesting effect of the increasing load is shown in Fig. 9. We can observe that resource scarcity affects the tenant's willingness of trading resources. As a matter of fact, when $|K| = 12$, Δ_m converges to a non-zero value, which guarantees a certain level of flexibility in resource allocations (c.f. Fig. 9(a)). This flexibility allows the scheduler, and tenants, to adopt an opportunistic behavior thus enhancing cost and resource efficiency. On the other hand, when the load drastically increases (c.f. Fig. 9(b)), the inability of serving elastic users forces $\Delta_m = 0,\ \forall m \in M$, thus reducing the flexibility of sharing.

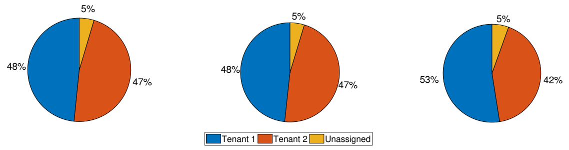
E. Impact of time window

In this section, we analyze the impact of time window differentiation among tenants. Fig. 10 and Fig. 11 show the effects of varying the time window length on the resource distribution between $|M| = 2$ tenants in case of resource scarcity and resource surplus, respectively.



(a) $W_1 = 80,\ W_2 = 80$ (b) $W_1 = 80,\ W_2 = 40$(c) $W_1 = 80,\ W_2 = 20$

Fig. 10. Effects of window differentiation on average resource distribution per tenant in resource scarcity scenario

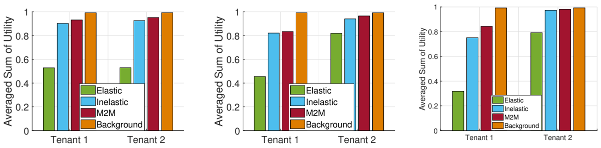


(a) $W_1 = 80,\ W_2 = 80$ (b) $W_1 = 80,\ W_2 = 40$(c) $W_1 = 80,\ W_2 = 20$

Fig. 11. Effects of window differentiation on average resource distribution per tenant in resource surplus scenario

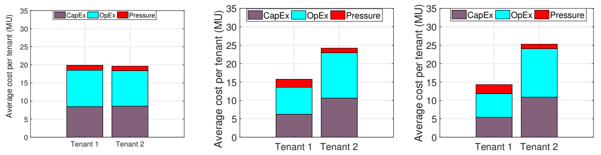
Generally speaking, smaller time windows indicate that the tenant's requirements need to be satisfied with higher frequency (i.e. within a shorter time frame). Therefore, due to the more stringent delay constraints, the InP has to prioritize the tenant with smaller W_m in order to be able to satisfy its utility target. On one hand, this prioritization does not affect the resource distribution between the two tenants, whenever there are sufficient resources to satisfy all the tenants, i.e. resource surplus (cf. Fig. 11). On the other hand, however, in case of resource scarcity (cf. Fig. 10), the priority given to the tenant with smaller time window (Tenant 2 in this example) causes an imbalance in the resource allocation, which increases proportionally the difference between the window lengths. Since choosing a smaller time window corresponds to potentially getting more resources, the selection of this parameter has to be monitored by the InP or a regulatory body.

Fig. 12 shows the effects of time window differentiation on the average utility per tenant per service in case of resource scarcity. As expected, the tenant with smaller time window receives a higher priority in the scheduler, which corresponds to a higher average utility with respect to the one achieved by the other tenant. Furthermore, results show that the service which is most penalized by the prioritization is the elastic one.



(a) $W_1 = 80,\ W_2 = 80$ (b) $W_1 = 80,\ W_2 = 40$(c) $W_1 = 80,\ W_2 = 20$

Fig. 12. Effects of window differentiation on average utility per service in resource scarcity scenario

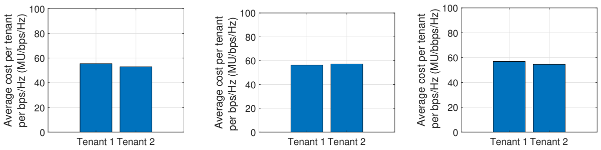


(a) $W_1 = 80,\ W_2 = 80$ (b) $W_1 = 80,\ W_2 =$(c) $W_1 = 80,\ W_2 = 20$
40

Fig. 13. Effects of window differentiation on average total cost per tenant in resource scarcity scenario

In contrast, non-elastic services are preserved by the utility based prioritization (i.e. the slopes of the utility functions shown in Fig. 2(b)) and experience only marginal decrease in the achieved utility. On the other hand, the tenant with smaller time window perceives an increase in utility for all the services, for critical as well as for elastic services. Note that this implies the negative effect of reducing the efficiency in resource usage, since more resources are assigned to either of the two tenants, independent of the channel conditions of its users.

Finally, Fig. 13 and Fig. 14 report the economic effects of window differentiation. Fig. 13 shows that, according to the resource distribution, the tenant with smaller W_m pays a higher cost, on average, while the tenant with larger time window length decreases the total costs. On the other hand, Fig. 14 reveals that the tenants' actual average cost per bps/Hz is similar for all cases. This confirms that the costs paid by the tenants is actually proportional to the resources they get.



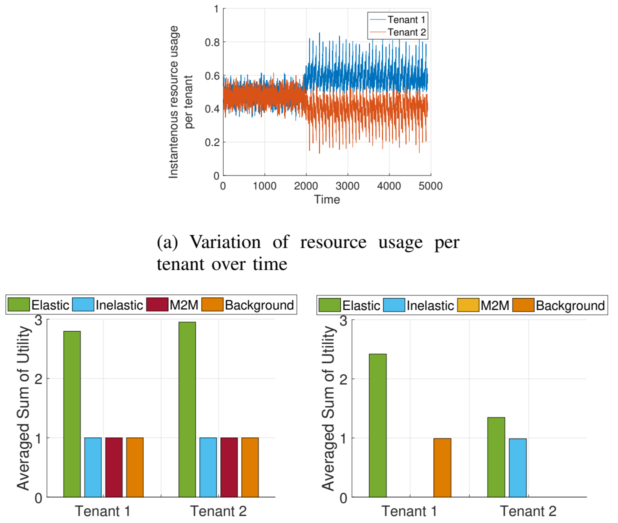
(a) $W_1 = 80,\ W_2 = 80$ (b) $W_1 = 80,\ W_2 = 40$(c) $W_1 = 80,\ W_2 = 20$

Fig. 14. Effects of window differentiation on average total cost per bps/Hz in resource scarcity scenario

F. Adaptation to changes in traffic mix

In [19], we analyze the ability of the proposed model to adapt to the changes of the wireless environment. Also we conclude that, in case of resource scarcity, such changes mainly affect the elastic services and our model is able to converge to a new optimal state adapting to the new conditions.

In other words, we investigate a resource surplus scenario, and analyze the reaction time and the effects of varying the traffic mix of the tenants.



(a) Variation of resource usage per tenant over time

(b) Average sum of utility before the change of traffic mix

(c) Average sum of utility after the change of traffic mix

Fig. 15. Adaptation to the variations in the traffic mixture

Fig. 15 shows the adaptation to the changes in the traffic mix. In particular, we assume that till $n = 1920$, the two tenants have symmetric traffic, i.e. 1 user per service type and a total of $|K| = 8$ users. At $n = 1920$, the traffic mix of the tenants changes as follows: the first tenant retains only non-critical services (i.e. it has 2 users with elastic services and 2 users with background services) while the second tenant specializes on critical services (i.e. 3 users with inelastic services and 1 user with elastic service). In Fig. 15(a), we can observe, between $n = 1920$ and $n = 2000$, a gradual change in the instantaneous assigned resources. After at least one renegotiation interval, the tenant's sharing parameters are updated, and this leads to a converge of the resource assignment. In Fig. 15(b) and Fig. 15(c), the average utility per service per tenant is shown before and after the traffic mix change, respectively. Note that, after the change, the elastic services achieve smaller utility on average. This is due to the fact that the number of users per service increases, which means that the resources requested by the non-elastic service (background for tenant 1 and inelastic for tenant 2) also increase.

G. Service specialized tenants

This section investigates the effects of service specialization on the proposed model. More specifically, we analyze the coexistence of four tenants with only one service type and one tenant with multiple service types. This also helps us addressing the question on whether our framework motivates tenants to enter the sharing market as specialized tenants or, in contrast, it is neutral to this choice. Therefore, we consider the scenario with $|M| = 2$ tenants, where the first tenant enters the market as virtually 4 different tenants (one per type). Also, we assume $|K| = 16$ users in total (2 users per service per tenant).

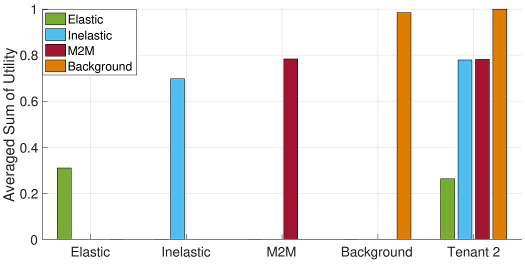


Fig. 16. The average utility per services per tenant

Fig. 16 clearly shows that entering the market as specialized tenant does not provide any advantages in terms of average achieved utility. Furthermore, Fig. 17 shows that a symmetry between the specialized tenants and the tenant with multiple services also exists in terms of the total average costs. Finally, Fig. 18 reports the resource distribution among tenants, which clearly indicates that also resources are split equally (i.e. each tenant gets approximately half of the available resources).

We can conclude that the proposed framework and the corresponding pricing mechanism are neutral to service specialization. Also, service prioritization (defined in Section II-A) is preserved and fairness is achieved in terms of both resource allocation and costs.

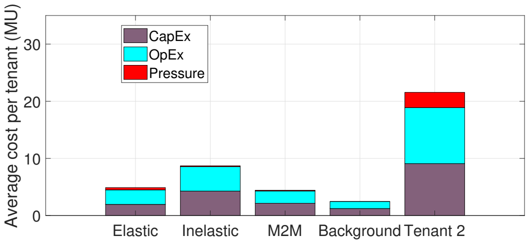


Fig. 17. The average total cost per service per tenant

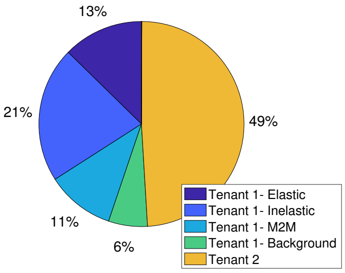
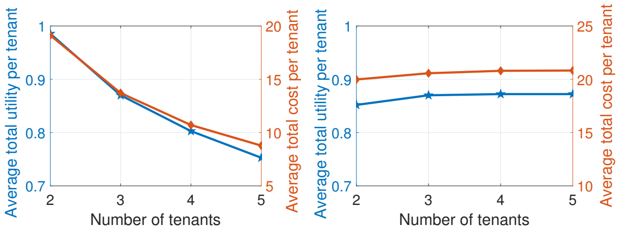


Fig. 18. Effects of service specialization on resource distribution

H. Costs and utility in different sharing scenarios

In this subsection, the effects of the number of tenants $|M|$ on the average cost per tenant and the average utility per tenant



(a) Short term

(b) Long term

Fig. 19. Effects of increasing number of tenants on the average utility and average costs per tenant

per service are investigated. The analysis is conducted considering two different time scales, i.e. short term and long term. In the short term, we assume that the infrastructure provider cannot react to the increase of the number of tenants $|M|$ (and thus users $|K|$), e.g., expanding the available capacity. In contrast, in the long term capacity can be scaled according to the demand.

Fig. 19(a) shows the result for the short term analysis, where the capacity is kept fixed while increasing $|M|$. On the other hand, Fig. 19(b) presents the result for the long term assumption, where the capacity is proportionally increased with $|M|$. Namely, we assume that the increase in capacity is achieved by the infrastructure provider increasing the total bandwidth. Results show that in the short term, see Fig. 19(a), increasing the number of tenants causes a resource scarcity and leads to a decrease of the average utility per tenant as expected. On the other hand, as shown in Fig. 19(a), the increase in $|M|$ also causes a decrease of the individual costs of tenants. In contrast, when considering a longer time scale (in the order of months), the infrastructure provider can react to the changes in $|M|$ and adjust the available capacity according to the needs. In this case, as depicted in Fig. 19(b), the average achieved utility and the average cost are not a function of $|M|$ (i.e. are almost constant when varying $|M|$).

Therefore, on one hand we can conclude that, in the long term, if the InP is able to expand the network capacity according to the tenants' needs, the proposed platform provides a sustainable resource sharing even when increasing $|M|$. On the other hand, in the short term, we cannot draw any conclusion only based on Fig. 19(a), since a decrease of the average utility could be compensated by a decrease in terms of cost (and hence price for the users). To evaluate the tradeoff between utility and cost (price), we use the concept of acceptance probability presented in [29]. In particular, the authors propose to model the acceptance probability as:

$$A_k(p, U_k) = 1 - \exp(-Cp^{-\epsilon}U_k^{\mu}), \tag{8}$$

which basically corresponds to the likelihood of user k to accept a service with price p and a corresponding utility U_k, where μ and ϵ are microeconomic parameters and C is a constant (that we set to the same values suggested in [29]).

To assess the sustainability of the sharing platform, we assume that each tenant aims to keep its profit constant, regardless of the number of tenants. This means that a variation of the costs directly affects the prices (that are computed as the sum of the costs and the profit). Therefore, increasing $|M|$ is accepted by the tenants, if the market share (i.e. the number of users) of each tenant is not decreasing, meaning that the acceptance probability ($A_k(p, U_k)$) should be a non-decreasing function of $|M|$.

By using (8), the condition above can be used, for two generic values $|M_1| \leq |M_2|$, as:

$$A_{k,M_1}(p_{M_1}, U_{k,M_1}) \leq A_{k,M_2}(p_{M_2}, U_{k,M_2}), \tag{9}$$

where

$$A_{k,M_1}(p_{M_1}, U_{k,M_1}) = 1 - \exp(-Cp_{M_1}^{-\epsilon}U_{k,M_1}^{\mu}),$$

$$A_{k,M_2}(p_{M_2}, U_{k,M_2}) = 1 - \exp(-Cp_{M_2}^{-\epsilon}U_{k,M_2}^{\mu}).$$

Assuming that the parameters μ, ϵ, and C are the same for both M_1 and M_2, (9) can be formulated as

$$\left(\frac{U_{k,M_1}}{U_{k,M_2}}\right)^{\mu} \leq \left(\frac{p_{M_1}}{p_{M_2}}\right)^{\epsilon}. \tag{10}$$

Satisfying (10) means that the variation in the average utility is accepted by the users since it is compensated by the decrease of the service price. In this case, the acceptance probability of $k \in K$ is a non-decreasing function of $|M|$.

Considering the same scenario of Fig. 19, Table V reports the numerical values for (10). As one can observe, inequality is always satisfied, which means that the users are paying less for their utility, and they are still willing to accept the service. Therefore, we can conclude that our proposed model provides a cost efficient and sustainable model even in short term.

TABLE V
VARIATION OF AVERAGE UTILITY AND TOTAL COSTS PER TENANT WITH THE NUMBER OF TENANTS IN SHORT TERM

$\lvert M_1\rvert \rightarrow \lvert M_2\rvert$	$\left(\frac{U_{k,M_1}}{U_{k,M_2}}\right)^{\mu}$	$\left(\frac{p_{M_1}}{p_{M_2}}\right)^{\epsilon}$
2 → 3	1,2834	3,7822
3 → 4	1,1744	2,6893
4 → 5	1,1372	2,2142

A further insight is given in Table VI, where Eq. (10) is evaluated for all the slice types (where 'yes' means that the Eq. (10) holds). In this case, we can see that, by increasing the number of tenants from $|M| = 4$ to $|M| = 5$, the acceptance probability of the elastic users decreases, whereas always increases for non-elastic services. This means that the tenants have a risk of losing some of the elastic traffic.

TABLE VI
EVALUATION OF THE USERS' ACCEPTANCE PROBABILITY FOR ALL SLICE TYPES (WE USE 'YES' TO INDICATE THAT EQ. (10) HOLDS, 'NO' OTHERWISE)

$\lvert M_1\rvert \rightarrow \lvert M_2\rvert$	Elastic	Inelastic	M2M	Background
2 → 3	Yes	Yes	Yes	Yes
3 → 4	Yes	Yes	Yes	Yes
4 → 5	No	Yes	Yes	Yes

The decrease in the acceptance probability of elastic services can be handled by an accurate and timely capacity expansion. The proposed pressure cost allows the infrastructure provider to accurately estimate the capacity needs and the expansion time. Even though increasing $|M|$ leads to lower utilities, since the collected pressure cost proportionally increases with the utility decrease, higher $|M|$ also implies faster capacity expansions.

V. Conclusion

We have shown that dynamic network slicing offers an efficient way of exploiting variable traffic and channel conditions to share resources among tenants, following different strategies, bearing different characteristics. Our proposed scheme defines a new platform where tenants can acquire resources within a short time frame, negotiating through a set of network and economic parameters. Our numerical results demonstrate that the proposed approach provides fairness among both tenants and services and can improve the efficiency of resource allocation up to 40% by exploiting simple prediction mechanisms. Despite the tenants share a common infrastructure, results have also demonstrated that it is possible for them to differentiate their services by tuning model parameters. We have also shown that the pricing model can allocate economic resources for capacity expansion and that this is crucial to keep infrastructure sharing convenient for the tenants.

Acknowledgment

This work is funded by the European Unions Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No. 643002.

The authors would like to thank Ms. Noemi Wagner for copy editing the publication material.

References

[1] Cisco, "The zettabyte era: trends and analysis," 2017. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/vni-hyperconnectivity-wp.pdf

[2] W. Lemstra, "Leadership with 5G in Europe: Two contrasting images of the future, with policy and regulatory implications," *Telecommunications Policy*, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0308596118300491

[3] China Mobile Communications Corporation, Huawei Technologies, Deutsche Telekom, and Volkswagen, "5G service-guaranteed network slicing white paper," 2017.

[4] OECD, "Wireless market structures and network sharing," 2014. [Online]. Available: http://dx.doi.org/10.1787/5jxt46dzl9r2-en

[5] N. C. Luong, P. Wang, D. Niyato, Y. Wen, and Z. Han, "Resource management in cloud networking using economic analysis and pricing models: A survey," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 2, pp. 954 – 1001, 2017.

[6] D. Zhang, Z. Chang, and T. Hamalainen, "Reverse combinatorial auction based resource allocation in heterogeneous software defined network with infrastructure sharing," in *IEEE Vehicular Technology Conference (VTC Spring)*, 2016.

[7] P. Cramton and L. Doyle, "An open access wireless market supporting competition, public safety, and universal service," 2016. [Online]. Available: http://www.cramton.umd.edu/papers2015-2019/cramton-doyle-open-access-wireless-market.pdf

[8] M. R. Hassan, G. C. Karmakar, J. Kamruzzaman, and B. Srinivasan, "Exclusive use spectrum access trading models in cognitive radio networks: A survey," *IEEE Communications Surveys Tutorials*, vol. 19, no. 4, pp. 2192–2231, Fourthquarter 2017.

[9] O. U. Akgül, I. Malanchini, V. Suryaprakash, and A. Capone, "Dynamic resource allocation and pricing for shared radio access infrastructrue," in *IEEE International Conference on Communications (ICC)*, 2017.

[10] G. S. Kasbekar, S. Sarkar, K. Kar, P. K. Muthuswamy, and A. Gupta, "Dynamic contract trading in spectrum markets," *IEEE Transactions on Automatic Control*, vol. 59, no. 10, pp. 2856 – 2862, 2014.

[11] M. Richart, J. Baliosian, J. Serrat, and J.-L. Gorricho, "Resource slicing in virtual wireless networks: A survey," *IEEE Transactions on Network and Service Management*, vol. 13, no. 3, pp. 462 – 476, 2016.

[12] X. Ting, P. Zhiwen, L. Nan, and Y. Xiaohu, "Inter-operator resource sharing based on network virtualization," in *International conference on Wireless Communication Signal Processing (WCSP)*, Oct 2015, pp. 1–6.

[13] M. I. Kamel, L. B. Le, and A. Girard, "LTE wireless network virtualization: Dynamic slicing via flexible scheduling," in *IEEE Vehicular Technology Conference (VTC Fall)*, Sept 2014, pp. 1–5.

[14] M. Jiang, M. Condoluci, and T. Mahmoodi, "Network slicing management & prioritization in 5G mobile systems," in *22 European Wireless 2016; 22th European Wireless Conference*. IEEE, May 2016, pp. 1–6.

[15] D. Zhang, Z. Chang, T. Hmlinen, and F. R. Yu, "Double auction based multi-flow transmission in software-defined and virtualized wireless networks," *IEEE Transactions on Wireless Communications*, vol. 16, no. 12, pp. 8390–8404, Dec 2017.

[16] K. Zhu, Z. Cheng, B. Chen, and R. Wang, "Wireless virtualization as a hierarchical combinatorial auction: An illustrative example," in *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, March 2017, pp. 1–6.

[17] D. Zhang, Z. Chang, T. Hmlinen, and W. Gao, "A contract-based resource allocation mechanism in wireless virtualized network," in *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, April 2018, pp. 474–479.

[18] A. Gran, S.-C. Lin, and I. F. Akyildiz, "Towards wireless infrastructure-as-a-service (WIaaS) for 5G software-defined cellular systems," in *2017 IEEE International Conference on Communications (ICC)*, 2017.

[19] O. U. Akgül, I. Malanchini, V. Suryaprakash, and A. Capone, "Service-aware network slice trading in a shared multi-tenant infrastructure," in *IEEE Global Communications Conference (GLOBECOM)*, 2017.

[20] J. Pérez-Romero, O. Sallent, S. Ferrús, and R. Agustí, "Admission control for multi-tenant radio access networks," in *IEEE International Conference on Communications Workshops (ICC Workshops)*, 2017.

[21] R. Berry, M. Honig, T. Nguyen, V. Subramanian, H. Zhou, and R. Vohra, "On the nature of revenue-sharing contracts to incentivize spectrum-sharing," in *IEEE International Conference on Computer Communications (INFOCOM)*, 2013.

[22] L. Cano, A. Capone, G. Carello, M. Cesana, and M. Passacantando, "Cooperative infrastructure and spectrum sharing in heterogeneous mobile networks," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 10, pp. 2617 – 2629, 2016.

[23] L. Zheng, J. Chen, C. Joe-Wong, C. W. Tan, and M. Chiang, "An economic analysis of wireless network infrastructure sharing," in *15th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, 2017.

[24] I. Malanchini and M. Gruber, "How operators can differentiate through policies when sharing small cells," in *IEEE 81st Vehicular Technology Conference (VTC Spring)*, May 2015.

[25] A. P. Avramova and V. B. Iversen, "Radio access sharing strategies for multiple operators in cellular networks," in *IEEE International Conference on Communication Workshop*, June 2015, pp. 1113–1118.

[26] J. S. Panchal, R. Yates, and M. M. Buddhikot, "Mobile network resource sharing options: Performance comparisons," *IEEE Transactions on Wireless Communications,*, vol. 12, no. 9, pp. 4470–4482, 2013.

[27] I. Malanchini, S. Valentin, and O. Aydin, "Wireless resource sharing for multiple operators: Generalization, fairness, and the value of prediction," *Computer Networks*, vol. 100, pp. 110 – 123, 2016.

[28] Gurobi Optimization Inc., "Gurobi optimizer reference manual," 2015. [Online]. Available: http://www.gurobi.com

[29] L. Badia, M. Lindstrom, J. Zander, and M. Zorzi, "Demand and pricing effects on the radio resource allocation of multimedia communication systems," in *IEEE Global Telecommunications Conference, GLOBECOM '03*, vol. 7, Dec 2003, pp. 4116–4121.

Özgür Umut Akgül holds an M.Sc. in Computer Engineering from Istanbul Technical University in Turkey. He is currently pursuing the Ph.D. degree at the Department of Electronics and Information, Politecnico di Milano in Italy. He is also involved in the EU H2020 ACT5G project. His main research interests are mostly related to techno-economic modeling and the analysis of network slicing in multi-tenant networks based on game theoretical models.

Ilaria Malanchini is a Senior Research Engineer and has been with Bell Labs Stuttgart since 2012. She received B.S. and M.S. degrees in telecommunications engineering from Politecnico di Milano, Italy, in 2005 and 2007, respectively, and a Ph.D. in electrical engineering from Drexel University, Philadelphia, and Politecnico di Milano in 2011. Ilaria was awarded the Meucci-Marconi Award and the Chorafas Foundation Prize for her Master and PhD thesis, respectively. She published more than 25 journals and conference papers and has more than 10 granted or filed patents. Her research interests focus on optimization models, mathematical programming, game theory, and machine learning, with the application of these techniques to wireless network problems such as wireless resource allocation, anticipatory network optimization, infrastructure and resource sharing, and network slicing.

Antonio Capone is currently a Full Professor at the Politecnico di Milano (Technical University of Milan), where he is also the Director of the ANTLab. His expertise is on networking and his main research activities include radio resource management in wireless networks, traffic management in software defined networks, network planning, and optimization. He has over 250 publications on these topics. He is an Editor of the IEEE TRANSACTIONS ON MOBILE COMPUTING, *Computer Networks*, and *Computer Communications*. He contributes to major conferences on networking as a Technical Program Comittee member. He was an Editor of the ACM/IEEE TRANSACTIONS ON NETWORKING from 2010 to 2014 as well.