Strategic Resource Pricing and Allocation in a 5G Network Slicing Stackelberg Game
Mandar Datar, Eitan Altman, Hélène Le Cadre

To cite this version:
Mandar Datar, Eitan Altman, Hélène Le Cadre. Strategic Resource Pricing and Allocation in a 5G Network Slicing Stackelberg Game. IEEE Transactions on Network and Service Management, inPress, 10.1109/TNSM.2022.3216588 . hal-03824540

HAL Id: hal-03824540
https://inria.hal.science/hal-03824540
Submitted on 21 Oct 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Strategic Resource Pricing and Allocation in a 5G Network Slicing Stackelberg Game

Mandar Datar†‡, Eitan Altman†‡§, Hélène Le Cadre¶

*Inria Sophia Antipolis - Méditerranée †CERI/LIA, University of Avignon, Avignon, France
‡LINCS Lab, Paris §Univ. Lille, Inria, CNRS, Centrale Lille, UMR 9189 - CRISTAL, F-59000 Lille, France

Abstract—We consider a marketplace in the context of 5G network slicing, where Application Service Providers (ASP), i.e., slice tenants, providing heterogeneous services, are in competition for the access to the virtualized network resource owned by a Network Slice Provider (NSP), who relies on network slicing. We model the interactions between the end users (followers) and the ASPs (leaders) as a Stackelberg game. We prove that the competition between the ASPs results in a multi-resource Tullock rent-seeking game. To determine resource pricing and allocation, we devise two innovative market mechanisms. First, we assume that the ASPs are pre-assigned with fixed shares (budgets) of infrastructure, and rely on a trading post mechanism to allocate the resource. Under this mechanism, the ASPs can redistribute their budgets in bids and customise their allocations to maximize their profits. In case a single resource is considered, we prove that the ASPs’ coupled decision problems give rise to a unique Nash equilibrium. Second, when ASPs have no bound on their budget, we formulate the problem as a pricing game with coupling constraints capturing the shared resource finite capacities, and derive the market prices as the duals of the coupling constraints. In addition, we prove that the pricing game admits a unique variational equilibrium. We implement two online learning algorithms to compute solutions of the market mechanisms. A third fully distributed algorithm based on a proximal method is proposed to compute the Variational equilibrium solution of the pricing game. Finally, we run numerical simulations to analyse the market mechanism’s economic properties and the convergence rates of the algorithms.

Index Terms—Communication service market, Game theory, Trading post mechanism, Pricing, 5G network slicing, Resource allocation.

I. INTRODUCTION

Next-generation wireless network is expected to deliver support to emerging sectors like Virtual Reality (VR) live broadcast, automotive, healthcare, manufacturing, etc. Critical challenges in mobile network applicability to the sectors mentioned above are their heterogeneity and conflicting communications needs, the current monolithic network is insufficient to meet. Several new concepts have been proposed for the upcoming 5G network design to satisfy these critical needs. Probably one of the most important one is “network slicing”.

Network slicing is the concept of running multiple independent logical networks (slices) on top of a common shared physical infrastructure. Each independent logical network (slice) is then explicitly dedicated to meeting each slice tenant’s needs, contrary to the “one-size-fits-all” approach that was the mainstream allocation method in the previous mobile generations [2]. Network slicing brings a paradigm shift towards a multi-tenancy ecosystem where multiple tenants, owning individual slices, negotiate with multiple virtualized infrastructure network providers, i.e., NSPs, to request resources for service provision. In this competitive multi-agent setting, the Application Service Providers (ASPs), also called slice tenants, generally express a demand for a dedicated virtual network with full ownership of their Service Level Agreement (SLA). On the contrary, NSPs aim to maximize their return on investment by enabling the dynamic sharing of the infrastructure, as this lowers their operational and capital costs and allows them to monetize their infrastructure to its fullest potential. However, the infrastructure sharing may expose the tenants to the risk of violating their SLAs. Hence, one of the fundamental issues in network slicing is an efficient sharing of the network resources, which arbitrages between two conflicting interests, i.e., inter-slice isolation and efficient network resource utilization. In order to balance the inter-slice isolation and efficient resource utilization, Caballero et al. in [3] proposed the “Share-Constrained Proportional Allocation” (SCPA) scheme where each slice is pre-assigned with a fixed share (budget) of infrastructure. Slices are then allowed to redistribute their shares and customize their allocation according to the load dynamics. In turn, NSP allocates each resource to slices in proportion to their shares on that resource. This approach allows a dynamic sharing, where tenants can redistribute their network share based on the load dynamics. At the same time, it provides the slice tenants a degree of protection by keeping the pre-assigned share intact.

Game-theoretic models have been employed for strategic resource allocation in communication networks, power systems, and more generally, a large number of deregulated industries. When dealing with strategic resource allocation, each player’s utility function depends on its own decision variables, and on that of the other players. The players’ feasibility sets can also be coupled through some global and local coupling constraints, capturing the laws of physics or, simply, shared capacity constraints. Extending duality results from standard continuous optimization to non-cooperative games, the dual variables of the coupling constraints can be interpreted as market prices, also called shadow prices or locational marginal prices, capturing the state of the network, e.g., congestion. Applying a similar model to dynamic resource trading in a
5G network, we design a new communication service market where the NSP charges dynamically the ASPs, depending on how much they contribute to the infrastructure utilization. The market prices are locationally, being differentiated by cell and resource. From this literature, this approach seems promising regarding ASPs’ cost reduction, as analysed in [4] that considers a service market where dynamic market prices are adjusted according to supply and demand (im)balances.

Relying on network slicing, we consider a market design where a set of ASPs lease their respective networks from a NSP and employ a network slicing mechanism to request resources required for their service provision. Network Slice Providers (NSPs) can be either Mobile Network Operators (MNOs), owning their own spectrum, or Mobile Virtual Network Operators (MVNOs). They rely on network slicing to allocate resources based on priorities (Quality of Services). We assume that the ASPs offer dedicated services to their users, and the resource inventory available with ASPs characterizes their service performance. The users are free to choose their ASPs. Their decisions are made based on the service satisfaction attained from ASPs. Furthermore, the ASPs collect revenue by providing services to their customers. Assuming a dynamic resource sharing mechanism and that ASPs maximize their utilities, it is highly expected that selfish ASPs may exhibit strategic behaviour. For example, they might strategically distribute their shares on the resources conditioned on the trade-off between the Quality of Service (QoS) they want to offer and the congestion perceived by the users. In this work, we focus on: 1) building a game-theoretic model of the communication service market where ASPs negotiate with NSP to request resources and compete with one another to serve a pool of end users; 2) developing a dynamic resource allocation and pricing mechanism under a competitive environment.

A. Related Work

There is a large part of the literature dedicated to the design of communication service markets. Broadly, communication service markets have been studied as a two-stage non-cooperative game involving four categories of agents: Infrastructure network Provider (InP), Network Slice Providers (NSPs) Application Service Provider (ASP) and End Users (EU). We focus on ASPs who allow the EUs to access applications (identically called services in the rest of the paper) by connecting to slices which are built on top of the resources bought from an NSP [5]. At the upper level of the game, ASPs (buyers) lease the resources from the NSPs (sellers), negotiating for resource prices and quantities. At the lower level, ASPs (buyers) use the acquired resources from NSPs to offer a certain service to their EUs (buyers). At this level, ASPs decide on their service price and their resource schedule, while EUs make their subscription decisions. In [6], ASPs’ strategic service pricing has been analyzed as a Cournot game.

In [7], Korcak et al. considered that the Quality of Service (QoS) achieved by the ASP’s users depends on the number of subscribers of that ASP, and users’ choice behaviour can be analyzed relying on Evolutionary Game Theory (EGT). Li et al. in [8] integrated both the users’ choice evolution and the ASPs pricing scheme. They formulated the resulting problem as a Stackelberg game. The ASPs, interpreted as leaders, strategically decide the price to attract the users, and the users, seen as followers choose the ASPs to maximize their service satisfaction level. Also, the number of subscribers of the ASPs depends on the perceived QoS and, consequently, on their resources availability. Focusing on competitive aspects, the ASPs can act strategically when computing their resource demand, giving rise to a simultaneous non-cooperative game [9]. In [10], D’Oro et al. used a similar Stackelberg game formulation for resource allocation and scheduling in the network functions virtualization scenario. In [11], Azouzi et al. considered that the competition between ASPs takes place both in prices and in QoS. In practice, ASPs may not have complete information about the other ASPs’ resources. Dealing with such an incomplete information setting, Li et al. in [12] studied ASPs’ pricing strategies relying on a Bayesian game formulation, where ASPs compute their prices based on their beliefs about the resource availability. Li et al. also considered the possibility that the ASPs can coordinate and analysed the impact of ASPs coordination on the pricing scheme.

In [13], Xi et al. proposed a dynamic resource polling and trading mechanism for network virtualization relying on Stackelberg game, where the carrier is considered as the leader while the users are the followers. The techno-economic interaction of NSPs and ASPs is modelled as a multi-leader multi-follower Stackelberg game in [14] in their proposed model, NSPs act as leaders who decide the price per unit of cell capacity, while ASPs act as followers who select an NSP which maximizes its profit. In [15], Ho et al. proposed a two-stage Stackelberg game based dynamic pricing for resource allocation in wireless network virtualization, where the NSP aims to maximize its revenue by leasing the bandwidth to the Mobile Virtual Network Operator (MVNO); on the contrary, MVNOs want to minimize their costs while serving their users at the best performance. In [16], Hu et al. considered an Orthogonal Frequency Division Multiple Access (OFDMA) system consisting of an NSP, multiple MVNOs and downlink users. They formulated the resource allocation in a radio access network slicing as a tri-level Stackelberg game. In [17], Tran et al. discussed the resource allocation and pricing problem for network slicing as a multi-leader multi-follower Stackelberg game that captures the interactions among access and backhaul ASPs, and their end users.

In [18], Kazmi et al. modelled the service selection and resource purchasing problem as a two-stage combinatorial optimization problem and solved it using a hierarchical matching game-based scheme, where matching between user and ASPs is performed at the low level, while matching between ASPs and NSP takes place at the upper level. In [19], Raveendran et al. proposed a cyclic three-sided matching game-based wireless resource allocation mechanism for slicing. In their proposed mechanism, allocation is performed by matching between radio resources, physical infrastructure, and mobile users. In [20], Lotfi et al. studied the economics of competition.
and cooperation between NSPs and ASPs using a sequential game that they solve by computing a Subgame Perfect Nash Equilibrium (SPNE).

On the contrary, the literature devoted to multi-resource allocation with finite budget agents is rather limited, to the extent of our knowledge. In [21], Nguyen et al. modeled edge computing resource allocation problem for service as a Fisher market. In the same vein, Moro et al. in [22] formulated a resource allocation problem for 5G network slicing as Fisher market, wherein apart from edge resources like computation and memory, authors also included radio resource. Although [21], [22] deal with multi-resource service provisioning with budget constraints similarly to us, their utility functions do not describe economic quantities such as profits and costs.

In our paper, we leveraged the Tullock Contest (TC) framework [29] to model the competition between slices. This framework has been extensively used in the communication network literature, to model the interactions between competitive agents. To mention a few, in [30], the competition between social media users for visibility over the timeline was modeled as a TC. Luo et al. in [31] proposed a TC based incentive mechanism for crowdsourcing. The TC framework has been applied to the multipath TCP network utility maximization problem [32]. In [33], Altman et al. studied the multi-cryptocurrency blockchain from a game-theoretic perspective, where the competition between the miners is framed as a TC. To the best of our knowledge, the theoretical results on the TC framework and its applications in literature only deal with a single resource case. We extend the TC framework to a multi-resource scenario, and thus our results also contribute to the theoretical literature on the TC framework.

We list below the key contributions of our article.

**B. Main Contributions**

1) We model the interactions between the ASPs and the EUs as a Stackelberg game, where the ASPs act as leaders and the EUs as followers. Besides, we rely on replicator dynamics to model the latter’s supplier choice and determine the ASP’s market shares in closed form expression relying on EGT.

2) We extend the study on Tullock-rent seeking games by showing that the non-cooperative game induced by the competition between ASPs admits a unique Nash equilibrium (NE).

3) We design an innovative communication service market relying on the solving of a generalized Nash equilibrium problem, where ASPs are charged depending on how much they contribute to the infrastructure utilization. The economic performance of this innovative market is compared to benchmark post trading mechanisms with finite budget constraints.

4) Finally, we provide two online learning algorithms and a fully distributed proximal based algorithm, new to our application field, to reach the NE and the Variational equilibrium solutions of the post trading mechanism and generalized Nash equilibrium problem respectively. Numerically, we observe that the proximal method significantly outperforms the online learning benchmark algorithms in terms of scalability and convergence rates.

The rest of the paper is organised as follows: Section II introduces the system model. Section III details the game-theoretic model of competition between the ASPs. In Section IV, we describe the Stackelberg game between the ASPs and the EUs. In Section V, we analyse the existence and uniqueness
We assume that the service offered by the ASP in a particular context, of the NE. Section VI introduces the resource pricing and market equilibrium. In Section VII, we provide two (semi-decentralized) online learning algorithms and a fully distributed algorithm to compute market equilibria. In Section VII, we report on numerical results. A concluding section ends the paper.

II. SYSTEM MODEL

We consider a market design, where at the upper level, a set of ASPs, $S$, leases some resources from a NSP to create one or more slices to provide different heterogeneous vertical applications (services). The NSPs lease the physical resources from Infrastructure network Providers (InPs) to form their own softwarized network, which can then be used to tailor slices to a host of application services (vertical services) demanded by the ASPs. For example, an ASP can lease some resources from NSP in the form of two slices, one for ultra-reliable low latency communications applications and another for enhanced mobile broadband applications. Each slice created by NSPs can be either managed by themselves or through third-party application service providers (ASP). If NSPs manage some of their own created slices and provide direct service to end users, they act as both NSPs and ASPs simultaneously. At the lower level, the ASPs (sellers) use the leased resources and compete to offer heterogeneous services to the end users.

A. User Model

We assume that the users might need different categories of services, e.g., Virtual Reality (VR) service, online gaming, autonomous driving, etc. To balance their demand, they have to subscribe to one of the ASPs. Let $K$ denote the set of all the services that are available to the users. $N^k_c$ represents the number of users who need service type $k$ in cell $c$; vector $\mathbf{n} = (n_1, \ldots, n_K)$ indicates the set of all users who need service type $k$ in cell $c$. Each base station $c$ in the network consists of a set of base stations or cells, $C$. Each base station at different locations accommodates multiple types of resources such as bandwidth, CPU, memory, etc. Users are spread across the network. Let $N^c$ be the number of users present in cell $c$. We assume that the service offered by the ASP in a particular cell can only be supported by the resources available within that cell. The system model is depicted in Fig. 1.

Notation: Let $\mathbb{R}^n$ indicate the set of $n$ dimension real vectors, and $\mathbb{R}^n_+$ its nonnegative orthant. $\|\cdot\|$ represents the Euclidean norm. Given a vector $x$, $x^T$ denotes its transpose. Let col$(x_1, \ldots, x_N) := [x_1^T, \ldots, x_N^T]$. For a closed set $\mathcal{F} \subseteq \mathbb{R}^n$, the mapping $\text{proj}_\mathcal{F} : \mathbb{R}^n \to \mathcal{F}$ denotes the projection onto $\mathcal{F}$, i.e., $\text{proj}_\mathcal{F}(x) := \arg\min_{y \in \mathcal{F}} \|y - x\|$. Depending on the context, $|\cdot|$ will denote the absolute value of a scalar or the cardinal of a set. Table II summarizes the main notation used in the paper.

![Fig. 1. The ASPs lease the resources from a NSP at different locations and compete to offer heterogeneous services to the end users.](image_url)
We suppose that the ASPs offer different types of services to the users and let \( \mathcal{K} \) denote the set of the services provided by each ASP \( s \) to the users, and for each service type, ASPs operate through a separate slice. The ASPs aim to maximize their number of subscriber \( n_s^c = [n_s^c_1 \ldots n_s^c_K] \), by attracting users with a better QoS and lower price. We assume that the QoS provided by each ASP depends on the resource inventory available at the slice and is defined according to the relationship \( q_{sk}^c = q_{sk}^c(d_{sk}) \). Let \( d_{sk} \) denote a bundle of resources available with ASPs \( s \), \( d_{km}^{c,m} \) captures the amount of resource type \( m \) acquired by ASP \( s \) for slice \( k \) at cell \( c \). We assume that for all \( c \in \mathcal{C}, k \in \mathcal{K} \) and \( s \in \mathcal{S} \), the function \( q_{sk}^c (d_{sk}) \) is concave non decreasing in \( d_{sk}^c \). This assumption is classical in economics, reflecting the principle of diminishing marginal returns.

Each ASP \( s \in \mathcal{S} \) collects revenue from the fees paid by its subscribers. The expected revenue of ASP \( s \) over the network is defined as

\[
R_s = \left( \sum_{c \in \mathcal{C}} \sum_{k \in \mathcal{K}} p_{sk} n_{sk}^c \right).
\] (2)

Each ASP needs to pay for the resources it leases from the NSP. Let \( \omega_m^c \) be the price per unit of resource of type \( m \) charged by the NSP, at base station \( c \). Let \( d_{s,m}^c = \sum_{k \in \mathcal{K}} d_{sk}^{c,m} \) denotes total amount of resource of type \( m \) allocated to ASPs \( s \) at cell \( c \)

The total cost each ASP \( s \) needs to pay to the NSP for resource activation is therefore \( \sum_{c \in \mathcal{C}} \sum_{m \in \mathcal{M}^r} \omega_m^c d_{s,m}^c \). We define the profit gained by ASPs as a quasi linear utility function

\[
U_s = R_s - \sum_{c \in \mathcal{C}} \sum_{m \in \mathcal{M}^r} \omega_m^c d_{s,m}^c.
\] (3)

Depending on the ASPs’ budgets, we consider two possible cases in the following.

In Case I, we assume that each ASP \( s \) has a finite budget \( B_s \), which captures the market (purchasing) power of the ASP. Another relevant interpretation in the context of network slicing is that it represents the ASP’s priority or a fixed share of the available resource pool, such that \( \sum_{s \in \mathcal{S}} B_s = 1 \). In this case, each ASP \( s \) must satisfy \( \sum_{c \in \mathcal{C}} \sum_{m \in \mathcal{M}^r} \omega_m^c d_{s,m}^c = B_s \). This budget constraint makes ASP \( s \)’s decision variables coupled over the cells \( c \in \mathcal{C} \). As a result, Case I can not be decomposed over each cell.

In Case II, we assume that ASP \( s \) has no bound over its budget. Its strategy set is defined as the set of vectors \( d_s \) such that \( d_{s,m}^c \geq 0, \forall m \in \mathcal{M}^r, \forall c \in \mathcal{C} \). A set \( \mathcal{I} \) of additional linear coupling constraints can be included in the form \( g_i(d) \leq 0, \forall i \in \mathcal{I} \). Note that contrary to Case I, Case II can be decomposed over each cell.

C. Resource Allocation and Pricing Problem

In this section, we discuss the central problem that we address in this paper. We observe from equation (2) that if an ASP wants to increase its market share, it needs to propose to EUs the best QoS at the lowest price. Indeed, at the lower level, EUs will choose to subscribe to the ASP that provides the best QoS and lowest price. The resource inventory available with the ASP characterizes their service performance: larger resources availability with the ASP guarantees a better QoS. However, the resources available with the NSP are limited, resulting in double-sided competitive interactions between the ASPs: one side of the interactions captures the competition for service provision, while the other side represents the competition for resource procurement. Undoubtedly, allocating and pricing resources in such a double-sided competitive environment is a significantly challenging task. In this work, our primary focus is on designing a resource pricing and allocation mechanism for the communication marketplace, which maximizes the network resources utilization and assigns slice tenants with their favourite bundle. In the next section, we model the interactions between users, ASPs and NSP as a generalized Stackelberg game involving coupling constraints at the lower level.
III. GAME THEORETIC MODEL

We assume that each user is utility-maximizer and selfish, and takes decisions to maximize its profit. From (1), we observe that the utility of each user depends on the total number of users of the ASP. On the one hand, as the number of users connected to the ASP increases, the utility of the user decreases. Therefore, the decision made by each user is also influenced by decisions taken by the other users. On the other hand, ASPs maximize their revenues by attracting the maximum number of users. Naturally, each ASP anticipates the users’ behaviour while computing their strategy. Therefore, it is highly expected that users and ASPs exhibit strategic behaviors. In our work, the ASPs take selfish decisions while anticipating the rational reactions of the users. We model the interactions between the users and the ASPs as a generalized Stackelberg game, where the ASPs act as leaders while users reactrationally as followers, computing their best responses to the signal sent by the ASPs. In the first stage, ASPs compete in terms of QoS to attract the maximum number of users. In the second stage, users optimally select their ASPs to maximize their utility given prices and QoS offered by ASPs.

In classical game theory, Nash equilibrium is the most popular solution concept to analyze non-cooperative game solutions. This concept is based on the assumptions that each player has an exact knowledge about all other players’ strategies at the equilibrium, and no player has an incentive to deviate from its own strategy at equilibrium. In many cases, knowing the exact information about all other players equilibrium strategies is a strong assumption, particularly when there are many users, and information about the strategy profile of all opponents is rarely perfectly known. In light of these limitations, we model the interactions between the users as a population game that extends the formulation of a non-cooperative game by incorporating the notion of population.

A. Multi-Population Game $\mathcal{E}$ Among Users

The population game provides an alternative to the classical equilibrium approach by involving a dynamic model. Unlike a single-play game or repeated games where all agents take their decisions simultaneously and repetitions occur at regular time periods, in a population game, each agent revises its decision sporadically, and the decision made by the revising agent only depends on the current system state and available payoff opportunities. Now for each cell $c$, and each service type $k$ we define the population game $\mathcal{E}_k^c$ as presented in the Fig. 2.

- Population: set of users requesting service type $k$, $N_k^c := \{1 \ldots N_k^c\}$ in cell $c$.
- Strategy: it is the choice of ASP $s \in S$ that each user in cell $c$ opts to join.
- Utility: the utility achieved by each user of ASP $s \in S$ with service type $k$ is equal to $\nu_{sk}$.

Remark 2. To simplify the notation and because the problem is decomposable over the set of services, we remove the subscript $k$, used for denoting service type $k$, from the mathematical analyses in the rest of the paper (unless explicitly specified).

In a population game, each agent revises its decision occasionally after some random duration of time. Whenever an agent reconsiders its decision, it depends on the system state and payoff opportunity available at that time. A general model of decision of the game is based on the concept of revision protocol. It is a mapping that translates the current population state (i.e., distribution of user) and available payoff (i.e., utilities (1)) into a switching rate which determines when users might update their choice of provider. Let $\tau_{s,s'}(n^c, U^c)$ be the revision protocol which defines the switching rate at which users switch their choice from ASP $s$ to ASP $s'$, given population state $n^c = [n_1^c \ldots n_c^c]$ and utility vector $\nu^c = [\nu_1^c \ldots \nu_S^c]$. Let $\mathcal{N}_c^c = \{n^c | \sum_{s \in S} n_s^c = N^c\}$ define the set of all possible population states. Population game $\mathcal{E}$ with revision protocol $\tau$ generates a continuous time evolutionary process on set $\mathcal{N}_c^c$ defined as

$$\dot{n}_s^c = \sum_{s'} n_{s'}^c \tau_{s,s'}^c - n_s^c \sum_{s'} \tau_{s,s'}^c. \tag{4}$$

The first-term in the right-hand side of equation (4) measures the rate at which users connect to ASP $s$. The second term measures the rate at which the portion of the population connected to ASP $s$ disconnects. A different choice of revision protocol results in a different dynamics. In this work, we assume that the users follow the pairwise proportional imitation behavior, e.g., after every random interval of time, each user interacts with its opponents (i.e., other users), and only if users meet an opponent with a higher utility than its own, it imitates the opponent with a probability proportional to the utility difference. The switching rate at which users in cell $c$ switch from ASP $s$ to ASP $s'$ takes the form

$$\tau_{s,s'}^c = \frac{n_{s'}^c}{N^c} [\nu_{s'}^c - \nu_{s}^c]^+. \tag{5}$$

After replacing $\tau_{s,s'}^c$ in (4) with (5) and after some analytical calculations detailed in Appendix C we get the replicator dynamics

$$\dot{n}_s^c = n_s^c \left[\nu_{s}^c - \frac{1}{N^c} \sum_{s'} n_{s'}^c \nu_{s'}^c \right]. \tag{6}$$

An Evolutionarily Stable Strategy (ESS) characterizes the equilibrium solution concept for population games. Once the evolutionary process reaches an ESS, the population state will not change. It is defined as the fixed point of the dynamical system defined through equation (4).

**Proposition 1.** For all $c \in C$ and for any bundle of resources available with ASP $s$, the replicator equation (6) admits a unique evolutionary equilibrium $\hat{n}_s$. Moreover, the number of users $\hat{n}_s^c$ in cell $c$ associated with ASP $s$ at the equilibrium point can be defined as

$$\hat{n}_s^c = \frac{N^c q_s^c e^{-p_s}}{\sum_{s' \in S} q_{s'}^c e^{-p_s}}. \tag{7}$$

**Proof.** The replicator equation (6) is nothing but a set of
ASP S

ASPS

\(A_{sk}(d^c, p) = \frac{f^c_s(d^c_{sk}, p^s_k)}{\sum_{s' \in S} f^c_{s'}(d^c_{s'k}, p^s_{s'k})}, \forall s \in S, \forall c \in C, \forall k \in K.\)  

(8)

**Remark 3.** Bernstein and Federgruen proposed a very well known general equilibrium model, named as attraction model, for industries with price and service competition in [32]. It is very similar to our slice association probability function.

In [8], the number of potential users in each cell as well as the slice association probability for each slice, might vary from cell to cell. The expected number of users choosing ASP \(s\) is defined as

\[
\sum_{c \in C} \sum_{k \in K} N_k^c A_{sk}(d^c, p) = \sum_{c \in C} \sum_{k \in K} N_k^c f^c_s(d^c_{sk}, p^s_k),
\]

(9)

Incorporating (7) and (2) in (3), we get

\[
U_s(d_s, d_{-s}) = \sum_{c \in C} \sum_{k \in K} p^s_k N_k^c f^c_{sk}(d^c_{sk}, p^s_k) - \sum_{m \in M^c} \omega_{mk} d^c_{sm}.
\]

(10)

In this work, we set

\[
f^c_{sk}(d^c_{sk}, p^s_k) = q^c_{sk}(d^c_{sk}) \epsilon^{-p^s_k}.
\]

(11)

We assume that \(f^c_{sk}(d^c_{sk}, p^s_k)\) is an increasing concave function in \(d^c_{sk}\) for a fix value of \(p^s_k\), while it is a convex decreasing function in \(p^s_k\) given fix value of \(d^c_{sk}\). The function translates the effort exerted by ASP in terms of resource and prices charged by them to the total utility experienced by its potential subscribers. We assume that ASPs are selfish, and that each ASP aims at maximizing its profit. They take into account the decisions of the other ASPs when computing their own decision. To theoretically analyze the outcome of these strategic interactions, we define the non-cooperative game \(\mathcal{G} \triangleq \langle \mathcal{S}, (\mathcal{F}_s)_{s \in S}, (U_s)_{s \in S} \rangle\) as follows:

- **Player set:** the set of application service providers \(\mathcal{S}\).
- **Strategy:** the vector of resource demand \(d_s = (d^1_s, \ldots, d^c_s)\) where \(d^c_s\) is the amount of resource to be requested by each base station \(c\). The strategy set for each ASP \(s\) is \(\mathcal{F}_s\).
- **Utility:** the utility of each ASP \(s\) is defined as \(U_s\).

We study the competition between ASPs in terms of QoS, i.e., how ASPs strategically spend their budget on the resources to attract the maximum number of users and, in turn, maximize their profits. The ASPs’ profit depends on both their individual decision and the decision taken by the other ASPs. Let \(d_s\) be the vector of strategy of ASP \(s\), \(d_{-s} \triangleq \text{col} \left( (d^c_{s'})_{s' \neq s} \right)\) is the stack vector which contains the vector of strategies of all the ASPs in \(\mathcal{S}\) except \(s\). The decision problem of each ASP \(s\) is
defined as
\[ Q_s \maximize_{d_s \in \mathcal{F}_s} U_s(d_s, d_{-s}). \]

To study the outcome of the non-cooperative game \( \mathcal{G} \), we recall the solution concept of Nash equilibrium (NE)

**Definition 1.** (37) A strategy profile \( d^* = (d_1^*, \ldots, d_S^*) \) is a Nash equilibrium of the game \( \mathcal{G} \) if
\[
\forall s \in \mathcal{S}, \quad U_s(d_s^*, d_{-s}) \geq U_s(d_s, d_{-s}^*), \forall d_s \in \mathcal{F}_s. \tag{12}
\]
Here, \( (d_s^*, d_{-s}^*) \) denotes the strategy profile with \( s^{th} \) element \( d_s^* \) and all other elements equal \( d_{-s}^* \) (for any \( s' \neq s \)).

In the next section, we analyze the existence and uniqueness of the Nash equilibrium (NE) solution of the non-cooperative game \( \mathcal{G} \).

**V. Existence and Uniqueness of the Nash Equilibrium**

In this section, we establish the existence and uniqueness of the NE of game \( \mathcal{G} \). To prove the uniqueness of the NE, we rely on the concept of diagonally strict concavity (DSC) introduced by Rosen [38]. Intuitively, DSC is a generalization of the idea of convexity to a non-cooperative game setting.

**Definition 2** (Diagonal Strict Concavity [38]). A game with profiles of strategies \( d \) and profiles of utility functions \( U \) is called diagonally strict concave (DSC) for a given vector \( r \) if for every distinct \( d \) and \( d' \),
\[
\left[ g(d, r) - g(d', r) \right] (d - d')' < 0, \tag{13}
\]
with \( g \) the concatenation of the weighted gradients of the players’ utility functions
\[
g(d, r) = \begin{bmatrix} r_1 \nabla U_1(d), r_2 \nabla U_2(d), \ldots, r_S \nabla U_S(d) \end{bmatrix}, \tag{14}
\]
where \( \nabla U_s(d) \) denotes the gradient of utility of player \( s \) with respect to his own strategy \( d_s \).

**Theorem 1.** The game \( \mathcal{G} \) admits a unique NE.

**Proof.** The utility of each ASP in game \( \mathcal{G} \) is continuous, increasing, and concave, while the strategy space for each ASP is convex and compact. Therefore, the existence of an equilibrium for the game follows from [38], Thm.1. To prove the NE uniqueness, we note that if the players’ utilities in the game \( \mathcal{G} \) satisfy the DSC property, then \( \mathcal{G} \) admits a unique NE (see [38], Thm.2).

Let \( G(d, r) \) be the Jacobian of \( g(d, r) \) with respect to \( d \), where \( d \) is any profile of strategies. In order to prove the strict DSC of \( g(d, r) \), from [38], Thm.6, we note that it is sufficient to prove that the symmetrized version of the pseudo-Jacobian, i.e., \( \bar{G}(d, r) \triangleq G(d, r) + G(d, r)' \), is negative definite over the domain of interest. To show that \( G(d, r) \) is negative definite, we must prove that the following three conditions hold simultaneously:

**C 1.** each \( U_s(d) \) is a regular strictly concave function of \( d_s \) (i.e., its Hessian is negative definite).

**C 2.** each \( U_s(d) \) is convex in \( d_{-s} \).

**C 3.** there is some \( r > 0 \) such that function \( \sigma(d, r) = \sum_s r_s U_s(d) \) is concave in \( d \).

The negative definiteness of \( [G(d, r) + G'(d, r)] \) follows from [39], Lem.1. We first consider the case of a single base station \( c \) with a single population demanding service \( k \) and show that \( \bar{G}_k(d, r) \) is negative definite for this case. We compute the Hessian \( \left( \bar{U}_s U^c_{sk} \right) \) of utility of any ASP \( s \) with respect to ASP \( s' \)’s own strategy
\[
H_s U^c_{sk} = -2 \left( \sum_{s' \in S, s' \neq s} f^c_{sk} \right) \sum_{s' \in S} f^c_{s'k} \tag{15}
\]
On the right hand side of (15), matrix \( (\nabla_s f^c_{sk})^T \nabla_s f^c_{sk} \) is positive semi-definite, where \( \nabla_s f^c_{sk} \) is the gradient row vector of \( f^c_{sk} \) with respect to ASP \( s' \)'s own strategy \( d^c_{sk} \). \( H_s(f^c_{sk}) \) is the Hessian of \( f^c_{sk} \) with respect to \( d^c_{sk} \) and it is negative definite as \( f^c_{sk} \) is concave. Thus, the Hessian of utility \( H_s U^c_{sk} \) is negative definite and satisfies the first condition (C 1).

We still need to show that the utility of each ASP \( s \) is convex in the strategy of all other ASPs. For that purpose, consider the Hessian of utility of ASP \( s \) with respect to strategy of all other ASPs
\[
H_{-s} U_s = 2 \left( \sum_{s' \in S} f^c_{sk} \right) \left[ M^c_{sk} - \text{diag}_{-s} \{ H(f^c_{uk}) \} \right], \tag{16}
\]
where is \( M^c_{sk} \) block matrix and \( uv^{th} \) block is defined as
\[
M^c_{uku} = (\nabla_u f^c_{uk})^T \nabla_v f^c_{ku} \quad \text{where } u, v \neq s, u, v, s \in S. \tag{17}
\]
\( \nabla_u f^c_{uk} \) is the gradient row vector of \( f^c_{uk} \) with respect to ASP \( s' \)'s own strategy and \( \text{diag}_{-s} \{ H(f^c_{uk}) \} \) is the block diagonal matrix with block \( u \) where \( H(f^c_{uk}) \) is the Hessian of \( f^c_{uk} \) with respect strategy vector of \( u \) itself \( \forall u, u \neq s, u \in S \). In right hand side of equation (17) matrix \( M^c_{sk} \) is positive definite and the block diagonal matrix \( \text{diag}_{-s} \{ H(f^c_{uk}) \} \) is negative definite as the each diagonal matrix element. \( H(f^c_{uk}) \) is negative definite, thus \( H_{-s} U_s \) is positive definite, which satisfies the condition (C 2).

Finally, by choosing \( r_s = \frac{1}{p^c_{uk}} \forall s \in S \) we check that \( \sigma(d, r) \triangleq \sum_s r_s U_s(d) \) is concave in \( d \), therefore satisfying the condition (C 3).

We now want to extend the previous proof to the multi-service case. We have already shown that \( \bar{G}_k \) is negative definite for any single service \( k \). For \( K \) service types consider a \( \bar{G}^c \) symmetrized version of the pseudo Jacobian, after arranging columns and rows we get (see [40], Cor.2)
\[
(\bar{G}^c) = \text{diag} \left\{ \bar{G}_1^c, \ldots, \bar{G}_K^c \right\}.
\]
VI. RESOURCE PRICING AND EQUILIBRIUM

We have shown in the previous section that there exists a unique NE solution of the non-cooperative game $G$. We assume that the capacity of the resource released by the ASP in each cell is finite. Given the per-unit prices for resources decided by the ASP, the total resource requested by the ASPs at the NE of $G$ may violate the infrastructure capacity. Thus, the ASP’s primary concern is how to efficiently allocate the finite capacity constrained resources to competing ASPs. The desired allocation must satisfy all the ASPs’ constraints and simultaneously maintain high resource utilization. In this regard, we assume that the ASP optimizes the unit price of each resource such that at the NE of the game $G$, each ASP utilizes its entire budget and no resource remains leftover, i.e., the total demand of resources matches the available supply (resource capacity). In market economics, this pricing problem is formulated as a market clearing problem, e.g., a Fisher market, where the market prices are settled in such a way that the amount of resources requested by the buyers is equal to the amount of resources supplied by the sellers. We propose two approaches, introduced in Section II B, to deal with this challenge depending on whether the ASPs’ budget is binding.

One way to compute the market equilibrium is through Walras’ “équation” process, i.e., if the demand exceeds the resource capacity, the market operator increases the resource’s price. Conversely, the market operator decreases the resource’s price when the demand is smaller than the resource capacity. The process is repeated until demand equals supply (resource capacity). The disadvantage of this approach is that its outcome (known as a general equilibrium) relies on the strong assumption of perfect competition, which in practice does not hold. To overcome this limitation, we use the approach introduced by Shapley and Shubik in their pioneering work [41], also known as Trading post or Share constrained proportional allocation (SCPA) scheme [3]. Now we formally define the trading post mechanism.

A. Trading Post Mechanism

In the trading-post mechanism, each player (i.e., ASP) places a bid on each type of resource. Once all ASPs have placed their bids, each resource type’s price is determined by the total bids placed for that resource. Precisely, let ASP $s$ submit a bid $b^c_{sm}$ to resource $m$ at cell $c$. The price per unit of resource $m$ at cell $c$ is then set to $\frac{\sum_{s \in S} b^c_{sm}}{D^m_c}$. Accordingly, ASP $s$ receives a fraction of $d^c_{sm}$ in return to his spending of $b^c_{sm}$

$$d^c_{sm} = \begin{cases} \frac{b^c_{sm} \cdot D^m_c}{\sum_{s \in S} b^c_{sm}} & \text{if } b^c_{sm} > 0, \\ 0 & \text{otherwise.} \end{cases}$$

After replacing $d^c_{sm}$ in (3) and $(Q_s)$ in terms of bids, the decision problem of each ASP $s$ can be written as follows

$$\hat{Q}_s \max \sum_{b^c \in C} U_s (b^c_s, b^c_{-s}), \quad \text{subject to } \sum_{c \in C, m \in M^c} b^c_{s,m} \leq B^c_s, b^c_{s,m} \geq 0.$$

We may consider two possible behaviours for the ASPs. First, they are price takers, i.e., they accept the prices decided by the price setter (market operator), and they only act strategically in terms of QoS by optimizing their demand in the bundle resources. Second, ASPs are price makers, i.e., they anticipate the effect of their demand on the price of the bundle of resources. The trading post mechanism induces a new non-cooperative game $\hat{G}$ defined as follows:

- Player set: the set of ASPs $S$.
- Strategy: the vector of bids $b^c_s \in [b^c_{s,1}, \ldots, b^c_{s,C}]$ where $b^c_s$ is the bid to be submitted to cell $c$. ASP $s$ strategy set is $F_s \triangleq \left\{ b^c_s | b^c_{s,m} \geq 0, \forall m \in M^c, c \in C, \sum_{c \in C, m \in M^c} b^c_{s,m} = B^c_s \right\}$.
- Utility: The utility of each ASP $s$ is $U^c_s (b^c_s, b^c_{-s})$.

To study the outcome of the mechanism, we consider the standard notion of NE, applied to the trading post mechanism.

Definition 3. A multi-bid strategy $b^* = (b^c_1, \ldots, b^c_S)$ is called a NE of the game $\hat{G}$ if

$$\forall s \in S, U^c_s (b^c_s, b^c_{-s}) \geq U^c_s (b^c_s, b^c_{-s}), b^c_s \in F_s.$$ (19)

Here, $(b^c_s, b^c_{-s})$ denotes the strategy vector with $s^{th}$ element equals $b^c_s$ and all other elements equal $b^c_v$ (for any $v \neq s$).

For the proposed mechanism, a NE solution of game $\hat{G}$ constitutes a stable bidding policy where each ASP maximizes its utility and the NSF implements the resource allocation mechanism [18]. We investigate conditions for the existence and uniqueness of the NE solution of the game $\hat{G}$. This requires complex calculations. Thus, to keep the analysis tractable, we restrict the problem to a single resource (radio resource). We investigate conditions for the existence and uniqueness of the NE solution of the non-cooperative game $\hat{G}$ as follows:

Proposition 2. If for a single resource, the QoS provided by ASP $s$ in cell $c$ is defined by $q^c_s = (d^c_s)^{\rho^c_s}$ and $0 < \rho^c_s \leq 1$, then the game $\hat{G}$ admits unique NE.

Proof. If the QoS provided by ASP $s$ in cell $c$ is defined by $q^c_s = (d^c_s)^{\rho^c_s}$ and $0 < \rho^c_s \leq 1$, then the ASPs’ utility functions

$$A^c_s (d^c_s, p^c_s) = \frac{(d^c_s)^{\rho^c_s} e^{-p^c_s}}{\sum_{s \in S} (d^c_s)^{\rho^c_s} e^{-p^c_s}}.$$ (20)
satisfy the three conditions [2],[2] and [3]. The rest of the proof is the same as the proof of Theorem 1.

Moving ahead, we compare the profit gained by ASPs at the NE of the game with the baseline static proportional allocation scheme (SS), i.e., the allocation where each resource is allocated to an ASP \( s \) in proportion to its budget, e.g., \( \sum_{s' \in S} B_{s'} \).

**Proposition 3.** For two application service providers, the revenue gained under a dynamic resource sharing scheme is at least equal to the revenue gained under a proportional allocation scheme.

**Proof.** The proof is provided in Appendix B.

We have seen in the first part of this section, that when ASPs are constrained by budgets, the resource pricing can be implemented by a trading post mechanism. However, this mechanism requires a third-party player (market operator) to centralize the bids made by all the ASPs, and thus can only lead to semi-decentralized implementations. Furthermore, the network capacity constraints are only implicitly taken into account through the budget constraint.

In the next section, we design a pricing and resource allocation scheme for Case II introduced in Section II B, that explicitly takes into account network capacity constraints and can be implemented in a fully distributed way. Case II gives rise to a generalized Nash equilibrium problem (GNEP) involving global coupling constraints, which take into account the network finite capacities. To solve the GNEP, we rely on a variational reformulation of the non-cooperative game, which leads to a unique variational equilibrium (VE). Using that property, we implement two algorithms to compute the VE: the first one requires an extended game reformulation of the GNEP and is based on asymmetric projected gradient descent methods; the second one relies on an extension of the alternating direction method of multipliers (ADMM).

### B. Pricing Game

We consider a non-cooperative game where, similar to game \( \mathcal{G} \), each ASP aims at maximizing its profit by requesting resources under a set of local constraints that are not binded by a finite budget. However, we now assume that the ASPs take into account the infrastructure capacity while requesting resources, therefore giving rise to a global coupling constraint for each cell and each resource available within that cell

\[
\sum_{s \in S} d_{sm}^c \leq D_{m}, \forall c \in \mathcal{C}, m \in \mathcal{M}.
\]

Let \( \tilde{\mathcal{F}}_s \triangleq \{d_s | d_{sm}^c \geq 0, \forall m \in \mathcal{M}^c, c \in \mathcal{C}, K_s(d_s) \leq 0\} \). The decision problem faced by each ASP in this new non-cooperative game can be formulated as a parametrized optimization problem with local and global coupling constraints

\[
\begin{align*}
Q_s & \quad \text{maximize} \quad R_s(d_s, d_{-s}), \\
& \quad \text{subject to} \quad \sum_{s \in S} d_{sm}^c \leq D_{m}^c, \forall c \in \mathcal{C}, m \in \mathcal{M}, \quad (\lambda_{sm}^c)
\end{align*}
\]

where \( \lambda_{sm}^c \) at the right of (22) and between brackets, is the Lagrange multiplier (shadow price) of the coupling constraint (22).

We define a new non-cooperative game \( \mathcal{G}_p \triangleq \left\langle \mathcal{S}, (\tilde{\mathcal{F}}_s)_{s \in S}, (R_s)_{s \in S} \right\rangle \), where the set of players and utility is the same as in game \( \mathcal{G} \). However, the strategy set of the players are coupled through the capacity constraint (22), giving rise to a GNEP. Consider the generalized Nash equilibrium (GNE) as the solution to this game.

**Definition 4.** A strategy profile \( d^* = (d_1^*, \ldots, d_S^*) \) is called a GNE of the game \( \mathcal{G}_p \) if

\[
\forall s \in \mathcal{S}, R_s(d_s^*, d_{-s}^*) \geq R_s(d_s, d_{-s}^*),
\]

\[
d_s \in \tilde{\mathcal{F}}_s, d_{sm}^c \geq 0, \forall m, c \text{ and } \sum_{s \in S} d_{sm}^c \leq D_{m}^c, \forall c \in \mathcal{C}, m \in \mathcal{M}.
\]

Due to coupling, solving directly \( \mathcal{G}_p \) requires coordination among possibly all ASPs, which might be hard to enforce in practice. To solve \( \mathcal{G}_p \), we will make use of the duality approach as a natural way to obtain a hierarchical decomposition of the GNEP. To that purpose, we start by characterizing the GNE solutions of game \( \mathcal{G}_p \) in terms of KKTs [42]: any strategy profile \( d \) is a GNE of the game \( \mathcal{G}_p \) if and only if it satisfies the KKT conditions, which are:

\[
\begin{cases}
\frac{\partial R_s}{\partial d_{sm}^c} (d) = \lambda_{sm}^c, \\
\lambda_{sm}^c \left( \sum_{s \in S} d_{sm}^c - D_{m}^c \right) = 0,
\end{cases}
\]

\[
\lambda_{sm}^c \geq 0, \forall m \in \mathcal{M}.
\]

In the above KKT conditions, we are primarily interested in \( \lambda_{sm}^c \), the Lagrange multipliers of (22), as these Lagrange multipliers can be interpreted as shadow prices for the resource allocation and can be used in the game \( \mathcal{G}_p \), as the evaluations by the ASPs of the prices charged by the NSP per resource unit. However, notice that if implemented without coordination, the Lagrange multipliers for each ASP are different, resulting in possibly discriminatory pricing. Moreover, there can be multiple possible GNEs. In fact, there are infinite GNEs solutions of \( \mathcal{G}_p \) in terms of KKTs [42]: any strategy profile \( d \) is a GNE of the game \( \mathcal{G}_p \) if and only if it satisfies the KKT conditions, which are:

**Definition 5.** An \( r \)-normalized Nash equilibrium point is such that there exists a \( \lambda_{sm}^c > 0 \) associated to each resource at each cell such that for all end users \( \lambda_{sm}^c = \lambda_{m}^c / r_s \), for a suitable vector of nonnegative coefficients vector \( r \).

It is very common in the literature, to relate normalized Nash equilibrium to the concept of variational equilibrium (VE) [42]. We will use both concepts without distinction in the following. The parameters \( \{r_1, \ldots, r_S\} \) intuitively show the proportion of a burden on ASP \( s \) for satisfying the coupling constraints among all other application service providers in the set. Notice, \( \lambda_{m}^c \) is
the same for all the ASP and thus can be treated as the base price. Next, we prove that such r-normalized Nash equilibrium (variational equilibrium), is unique for game $G_p$.

**Corollary 1.** The Pricing Game $G_p$ admits a unique normalized equilibrium for $r_s = \frac{1}{p_s}$.

**Proof.** In the proof of Theorem 1, we have shown that the Game $G$ has the DSC property $\forall c \in C$ and $\forall c \in M$ and any $\omega_m^c \geq 0$. The utilities of players in the Game $G_p$ (i.e. revenues) are the same as in $G$ with $\omega_m^c = 0$. Hence, the Pricing Game $G_p$ also satisfies the DSC property. The proof is a consequence of [38], Thm.4.

**Theorem 2.** Every r-normalized Nash equilibrium of the Pricing Game $G_p$ with shadow prices $\lambda_m^c = \frac{\lambda_m}{r_m^c}$ for all $s \in S, m \in M, c \in C$ is a NE for the corresponding Game $G$ with $\omega_m^c = \lambda_m^c, \forall m \in M, \forall c \in C$.

**Proof.** We reformulate $G_p$ using an augmented system-like utility function, that we call the Nash game (NG)-game utility function [43]. The NG utility function is defined as a two argument function

$$
\tilde{R}(d; x) \triangleq \sum_{s=1}^{S} r_s R_s(d_s, x_s), \forall x \in \bar{F} \triangleq \prod_s \bar{F}_s,
$$

where $x \triangleq (x_s)$, and $d_s$ defined as before. We note that Definition 5 can be equivalently formulated with respect to the NG utility function. A vector $d^* \in \bar{F}$ is called a NE solution of this game if its NG utility function $\tilde{R}$ satisfies

$$
\tilde{R}(d^*; d^*) \geq \tilde{R}(d^*; x) \ \forall x \in \bar{F}.
$$

The above condition can equivalently be written as follows, for given $d^*_s$:

$$
\sum_{s=1}^{S} r_s R_s(d^*_s; d^*_s) \geq \sum_{s=1}^{S} r_s R_s(d^*_s; x_s), \forall x \in \bar{F}.
$$

Note that the NG utility function $\tilde{R}$ is separable in the second argument $x$ for any given first argument $d^*$. The existence of NE is guaranteed by Theorem 2 [43]. Now to extend the NG utility function formulation to coupled constrained game, i.e., pricing game, we use the fact that the pricing game is related to a constrained maximization of NG utility function with respect to the second argument keeping the first argument as a fixed point solution. Consider that the ASPs maximize their revenue subject to coupled constraints $g_i(d) \leq 0, i \in I \triangleq C \times M$ where $g_i(d) \triangleq D_m - \sum_d d_{m}$. Then

$$
\tilde{R}(d^*; d^*) \geq \tilde{R}(d^*; x) \in F, \tilde{g}(d^*_s; x_s) \leq 0,
$$

where $\tilde{g}(d^*; x) = \sum_{s=1}^{S} g(d^*_s; x_s)$. We introduce the augmented Lagrangian function of the constrained NG utility maximization problem, with $\lambda$ a Lagrange multiplier vector

$$
\tilde{L}(d; x; \lambda) \triangleq \tilde{R}(d; x) + \lambda^T \tilde{g}(d; x).
$$

In our case $\forall s \in S, R_s$ is increasing concave and continuously differential, and $g$ is affine. Thus, all the constraints are active at the equilibrium. If $d^*$ is an equilibrium solution of the pricing game $G_p$, then by [43], Lem.2, there exists a unique $\lambda^* > 0$ such that $\nabla_d \tilde{L}(d; x, \lambda^*) = 0$ and $d^*$ maximizes the Lagrangian $\tilde{L}$, over $x \in \bar{F}$ as a fixed point.

Relying on the duality framework, we prove that we can decompose the coupled constrained game $G_p$ into the equivalent game with no coupled constraints, and indeed the equivalent game coincides with the non-cooperative game $G$ with $\omega_m^c = \lambda_m^c$. To that purpose, we consider the dual cost function $D(\lambda)$ defined as

$$
D(\lambda) \triangleq \tilde{L}(d^*; d^*; \lambda).
$$

Equivalently, relying on the notion of fixed point, the dual cost can be written as

$$
D(\lambda) \triangleq \left[ \max_{x \in \Omega} \tilde{L}(d; x; \lambda) \right]_{x=d}.
$$

The dual NG can then be defined as the minimization of the dual cost function

$$
D^* = \min_{\lambda \geq 0} D(\lambda).
$$

The Lagrangian function $\tilde{L}$ is separable over each ASP. Thus, the dual function can be separately written for each player as

$$
D(\lambda) \triangleq \sum_{s \in S} \left[ \max_{x_s \in \Omega_s} L_s(d_s - s; x_s; \lambda_s) \right]_{x_s=d_s} = \sum_{s \in S} L_s(u_{s}^*(\lambda), u_{s}^*(\lambda), \lambda),
$$

where

$$
L_s(d_s - s; x_s; \lambda) = r_s R_s(d_s - s; x_s) + \lambda^T g(d_s - s; x_s).
$$

From [43], Thm.3, we prove that $D(\lambda)$ can be obtained by solving the relaxed game with utility function $L_s$ and no coupled constrains. Indeed, that relaxed game is the same for all the ASP and thus can be treated as the base price. Next, we prove that such r-normalized Nash equilibrium (variational equilibrium), is unique for game $G_p$.

This approach enables us to reformulate the GNEP $G_p$, as a lower-level non-cooperative Nash game with utility function $L_s$ and no coupled constrains. Indeed, that relaxed game is the same for all the ASP and thus can be treated as the base price. Next, we prove that such r-normalized Nash equilibrium (variational equilibrium), is unique for game $G_p$. We consider a game with $S + 1$ players, where the first $S$ players are the ASPs and the...
We have proved in Section V that \( Q \) is also monotone on \( \tilde{G} \), for any \( Q \) is that in the former, there are no coupled constraints – complementary conditions are treated as the utility of an additional player (NSP).

**Proposition 4.** If \( \tilde{\alpha} \) is a \( \nu \)-normalized equilibrium of the pricing game, then there exists \( \tilde{\lambda} \geq 0 \) such that \( \langle \tilde{\alpha}, \tilde{\lambda} \rangle \) is an equilibrium of the extended pricing game.

**Proof.** We have already proved that the pricing game \( \tilde{G} \) is monotone on \( \tilde{F} \), which implies that the extended pricing game is also monotone on \( \tilde{F} \times \mathbb{R} \), the proof follows from [46], Prop.4.

**VII. ALGORITHMS TO COMPUTE MARKET EQUILIBRIA**

In this section, we introduce two semi-decentralized algorithms to compute the equilibria of the pricing game.

**A. Semi-Decentralized Learning Algorithms**

We have proved in Section [46, 47] that \( \tilde{G} \) admits a unique equilibrium for any price vector decided by the NSP. A similar result also holds for [46, 47] when dealing with a single resource. However, we still need to check whether ASPs can reach this equilibrium in a decentralized fashion. In this regard, we propose the use of the dual averaging or mirror–descent method suggested for continuous action convex games [47]. We proceed by describing the dual averaging method. In the dual averaging method, each player, e.g., ASP \( s \) estimates its marginal utility or utility gradient with respect to its own strategy. To increase their utilities, the players need to take action along the direction of their utility gradient while maintaining their action in the feasible action space. In order to achieve this, each player \( s \) at each time step \( n \) accumulates its discounted utility gradient in some auxiliary variable \( y_s \)

\[
y_s(n+1) = [y_s(n) + \alpha_n \nabla_{b_s} J_s(b_s(n), b_{-s}(n))].
\]

In the above equation \( \alpha_n \) denotes the discount factor or step size. Once the discounted gradient has been accumulated, every ASP \( s \) uses its own updated value of the auxiliary variable, \( y_s \), to take the next feasible action

\[
b_s(n + 1) = Q_s(y_s).
\]

In turn, each ASP \( s \) maps the recent value of auxiliary variable \( y_s \) to its decision space \( F \) using the mapping \( Q_s(y_s) \), e.g., \( Q_s \) can be interpreted as a projection map. The map \( Q_s(y_s) \) is defined more generally as

\[
Q_s(y_s) = \arg\max_{b_s \in F_s} \{\langle y_s(n), b_s \rangle - h_s(b_s)\},
\]

where \( h_s(b) \) is a regularization function, also called penalty function, over the feasible action set \( F \). The penalty \( h_s(b) \) aims to force the algorithm to converge within the interior of the feasible domain set. Different definitions of the regularization functions induce different maps. For instance, the use of \( l_2 \) norm \( h_s(\cdot) = ||\cdot|| \) as a regularizer, results in the well-known Euclidean projection map.

For the game \( \tilde{G} \), where application service providers actions are bounded by the users’ budgets, we use the Gibbs entropy function as a regularization function

\[
h_s(b_s) \triangleq \sum_{c \in \mathcal{C}} \sum_{m \in \mathcal{M}} b_{sm}^c \log(b_{sm}^c).
\]

We replace \( h_s(b_s) \) in equation [42] by the entropic regularization function and after some calculation we get the exponential mapping

\[
b_{sm}^c = \frac{B_c \exp(y_{sm}^c)}{\sum_{c \in \mathcal{C}} \sum_{k \in \mathcal{M}} \exp(y_{sk}^c)}.
\]

The induced map \( Q_s(y_s) \) is similar to the well-know Logit map, where each player distributes its budget (weights) to different resources depending on exponential of accumulated discounted gradients.

**Algorithm 1** Online Learning Algorithm for \( \tilde{G} \)

**Require:** \( \sum_{n=0}^{\infty} \alpha_n = +\infty, \alpha_n \to 0 \) as \( n \to +\infty 

1: \textbf{repeat} \ n = 1, 2, \ldots.
2: \textbf{for each} ASP \( s \in \mathcal{S} \)
3: \ \ \ \textbf{observe gradient of utility and update} \n4: \ \ \ \ y_s = [y_s + \alpha_n \nabla_{b_s} J_s(b_s, b_{-s})]
5: \textbf{end for}
6: \textbf{for each} ASP \( s \in \mathcal{S} \)
7: \ \ \ \textbf{for each} cell \( x \in \mathcal{C} \) and resource \( m \in \mathcal{M}^c \)
8: \ \ \ \ \text{Play} \ b_{sm}^c \leftarrow \frac{B_c \exp(y_{sm}^c)}{\sum_{c \in \mathcal{C}} \sum_{k \in \mathcal{M}} \exp(y_{sk}^c)}.
9: \textbf{end for}
10: \textbf{end for}
11: \textbf{until} \ ||(b(n) - b(n-1))|| \leq \epsilon

**Theorem 3.** If Algorithm 1 satisfies the required conditions for step size sequence, e.g., \( \sum_{n=0}^{\infty} \alpha_n = +\infty, \alpha_n \to 0 \) as \( n \to +\infty \), then it converges to the unique NE of the Game \( \tilde{G} \).

**Proof.** The proposed exponential algorithm is the special case of the dual averaging algorithm. If the NE of any continuous
action convex game is strictly $r$-variationally stable, then the converges of the dual averaging algorithm to a unique NE of the game is guaranteed by \cite{27}, Thm. 4.6. Hence to prove the convergence of the proposed algorithm, it is sufficient to show that the unique NE of game $\mathcal{G}$ is strictly $r$-variationally stable. The unique NE $\hat{b}$ of any convex game is strictly $r$-variationally stable if $\forall \hat{b}_s \in \mathcal{F}_s$

$$\sum_{s \in \mathcal{S}} r_s \nabla_s U_s (\hat{b}) (b_s - \hat{b}_s) < 0.$$  

(40)

As we have already shown in section VI the ASPs' utility functions in game $\mathcal{G}$ satisfy the DSC for $r_s = \frac{1}{p_s}, \forall s \in \mathcal{S}$

$$\sum_{s \in \mathcal{S}} r_s \left[ \nabla_s U_s (b) - \nabla_s U_s (\hat{b}) \right] (b_s - \hat{b}_s) < 0.$$  

(41)

We know that for any continuous action convex game, a feasible point $\hat{b}$ is a NE of the game if and only if

$$\sum_{s \in \mathcal{S}} r_s \nabla_s U_s (\hat{b}) (b_s - \hat{b}_s) \leq 0.$$  

(42)

Inequalities (42) and (41) imply (40), which proves that the unique NE of game $\mathcal{G}$ is strictly $r$-variationally stable and then by \cite{27}, Thm. 4.6, Algorithm 1 converges to the unique NE of game $\mathcal{G}$.

For Case II when the ASPs have no bound on their budgets, we have proved in Section VI B that the resource pricing scheme can be set up by solving the GNEP $\mathcal{G}_p$. Furthermore, we have also shown that the VE solution to $\mathcal{G}_p$ can be computed as the solution of an extended pricing game $\mathcal{Q}_+$. Now, we provide an online semi-decentralized learning algorithm that enables the ASPs and the NSP to reach the VE of $\mathcal{G}_p$. In the proposed semi-decentralized algorithm, we leverage on the framework from \cite{48}: the first $S$ players, i.e., the ASPs, follow similar steps as in Algorithm 1. However, an $(S+1)$th player, i.e., the NSP, accumulates the augmented discounted gradients of its utility in the auxiliary variable $y_{S+1}$

$$y_{S+1} = \lambda^c_m + \alpha_n \left[ \left( \sum_{s \in \mathcal{S}} d^c_{sm} - C^c_m \right) - \theta_n \lambda_n \right].$$  

(43)

Here rationale behind adding an extra term is that the original game is strictly monotone, and thus convergence is guaranteed in that case. However, the extended pricing game is just monotone and therefore, to make the algorithm converge to an equilibrium point, an additional term must be included \cite{48}. NSP updates the market price by projecting the stored auxiliary variable on the positive orthant

$$\lambda^c_m \leftarrow \text{proj}_{\mathbb{R}^+} \left( \lambda^c_m + \alpha_n \left[ \left( \sum_{s \in \mathcal{S}} d^c_{sm} - C^c_m \right) - \theta_n \lambda_n \right] \right).$$  

(44)

**Theorem 4.** \cite{48} If Algorithm 2 satisfies the required conditions for step size sequence, e.g., $\sum_{n=0}^{+\infty} \alpha_n = +\infty, \alpha_n \to 0$ as $n \to +\infty$ and for an augmented sequence $\theta_n, \frac{\theta_n}{\sum_{n=1}^{+\infty} \alpha_n} \to 0, N \to 0$, then it converges to the unique equilibrium of the extended pricing game.

**Algorithm 2 Online Learning Algorithm for $\mathcal{Q}_+$**

**Require:** $\sum_{n=0}^{+\infty} \alpha_n = +\infty, \alpha_n \to 0$ as $n \to +\infty$

1: repeat $n = 1, 2, \ldots$

2: for each ASP $s \in \mathcal{S}$

3: Observe gradient of utility and update

4: $y_s \leftarrow [y_s + \alpha_n, \forall d_s \nabla_s U_s (d_s, d_{-s}, \omega_s)]$

end for

5: for each ASP $s \in \mathcal{S}$

6: $d_s \leftarrow \text{proj}_{\mathcal{D}_s} [y_s]$

end for

7: NSP update the resource prices

8: for each Cell $c \in \mathcal{C}$

9: for each Resource $m \in \mathcal{M}$ update the base price

10: $\lambda^c_m \leftarrow \max [0, \lambda^c_m + \alpha_n \left( \sum_{s \in \mathcal{S}} d^c_{sm} - C^c_m \right)]$

end for

end for

11: until $\|d(n), (\omega(n)) - (d(n-1), \omega(n-1))\| < \epsilon$

**B. A Distributed Proximal Algorithm**

We assume a fully connected communication graph between the ASPs, e.g., $\Gamma_s \triangleq \mathcal{S} \setminus \{s\}, \forall s \in \mathcal{S}$. We want to compute the $r$-normalized Nash equilibrium solution of $\mathcal{G}_p$ relying on a fully distributed algorithm. To that purpose, we set $x_s \triangleq d_s$ as ASP $s$ own action, $x_s' \triangleq$ ASP $s$'s estimate of the other ASPs' actions, and $x^\ast \triangleq \text{col}(x^\ast_s, x^\ast_{-s})$ as the concatenation of ASP $s$ own action and estimate of the others’ actions. Let $\mathcal{F}_s \triangleq \{x_s^\ast | x_s^\ast > 0, \omega^T x_s^\ast = B_s \}$ be the strategy set of ASP $s$. Following \cite{49, 50}, we decompose the pricing game $\mathcal{G}_p$ per agent. Some slack variables $(v_{s,s'})_{s,s'}$ and $(w_{s,s'})_{s,s'}$ are introduced to guarantee the coincidence of the local copies. Let $M_{S-1}$ be the matrix made of $S-1$ blocks, each one of them containing the Identity matrix of size $\sum_s |\mathcal{M}_s| \times \sum_s |\mathcal{M}_s|$. Each ASP $s$ solves the local optimization problem

$$\min_{\lambda^c_s \geq 0, (w_{s,s'}, v_{s,s'})_{s,s'} \in \mathcal{F}_s \times \mathcal{F}_s} \max_{x_s \in \mathcal{F}_s} \left[ R_s (x_s^\ast, x_{-s}^\ast) - \lambda^c_s (x_s^\ast + M_{S-1} x_{-s}^\ast) - D \right],$$  

(45a)

s.t. $x_s' = v_{s,s'} \forall s, s' \in \mathcal{S}, \forall s', \in \mathcal{S}_s \in \Gamma_s', (\alpha^s_{s'}) \quad (45b)$

$x_s' = v_{s,s'} \forall s, s' \in \mathcal{S}, \forall s', \in \mathcal{S}_s \in \Gamma_s', (\gamma^s_{s'}) \quad (45c)$

$\lambda^c_s = r_s w_{s,s'} \forall s, s' \in \mathcal{S}, \forall s', \in \mathcal{S}_s \in \Gamma_s', (\delta^s_{s'}) \quad (45d)$

$\lambda^c_s = r_s w_{s,s'} \forall s, s' \in \mathcal{S}, \forall s', \in \mathcal{S}_s \in \Gamma_s', (\gamma^s_{s'}) \quad (45e)$

where $r_s' = \frac{1}{\beta_{s'}}$ and $r_s'' = \frac{1}{\beta_{s''}}$. Note that we use the convention to have superscript indices for primal variables, and lower script indices for duals of $\mathcal{G}_p$. A solution of $\mathcal{G}_p$ is obtained by assuming that each ASP $s$ solves the partial dual optimization problem \cite{43} and by identifying $x_s^\ast = d_s$ and $\lambda^c_s = \frac{1}{\beta_s \gamma_s}$. Let $\zeta > 0$ be a scalar coefficient. We follow the alternating direction method of multipliers (ADMM). To that purpose, we explicit the Lagrangian function associated
with (45)
\[
L_s(x_s^*, \{v, \alpha, \beta \}, \lambda_s, \{w, \gamma, \delta \}) := R_s(x_s^*) - \lambda_s^T (x_s^* + M_{s-1} x_{s-1}^* - D) - \sum_{s' \in \Gamma_s^e} \sum_{s'' \in \Gamma_s^f} \left[ (\alpha^{s''} x_s^*)^T (x_s^* - v^{s''}) + (\beta^{s''} x_s^*)^T (x_s^* - v^{s''}) \right] + \sum_{s' \in \Gamma_s^e} [\gamma s' (\lambda_s - r_{s'} w_{s'}^*) + \delta s' (\lambda_s - r_{s'} w_{s'}^*)],
\]
and associated KKTs, which give rise to the following relationships: \(\alpha^{s^*} + \beta^{s^*} = 0 \) and \(\gamma_{s^*} + \delta_{s^*} = 0\), \(\forall s^* \in \Gamma_s^e \).

To update the ASPs' strategies, we rely on the augmented Lagrangian associated with (45):
\[
L_s(x_s^*, \{v, \alpha, \beta \}, \lambda_s, \{w, \gamma, \delta \}) \triangleq L_s(x_s^*, \{v, \alpha, \beta \}, \lambda_s, \{w, \gamma, \delta \}) - \frac{\lambda_s^T}{2} \left( \sum_{s'' \in \Gamma_s^e} \left( ||x_s^* - v^{s''}||^2 + ||x_s^* - v^{s''}||^2 \right) + \frac{\xi}{2} \left( \sum_{s'' \in \Gamma_s^e} (\lambda_s - r_{s'} w_{s'}^*)^2 \right) \right).
\]

Following vanilla ADMM, the duals in (45) are updated according to the rules
\[
\alpha^{s''}(t) = \alpha^{s''}(t-1) + \frac{\xi}{2} \left( x_s^*(t-1) - x_s^{s''}(t-1) \right), \quad (46a)
\]
\[
\beta^{s''}(t) = \beta^{s''}(t-1) + \frac{\xi}{2} \left( v^{s''}(t-1) - x_s^{s''}(t-1) \right), \quad (46b)
\]
\[
\gamma_{s''}(t) = \gamma_{s''}(t-1) + \frac{\xi}{2} \left( \lambda_s - r_{s'} w_{s'}^* \right), \quad (46c)
\]
\[
\delta_{s''}(t) = \delta_{s''}(t-1) + \frac{\xi}{2} \left( \lambda_s - r_{s'} w_{s'}^* \right), \quad (46d)
\]

We update the slacks \(v, w\) by solving the following optimization problems
\[
\begin{align*}
v^{s''}(t) &= \arg \max_{v^{s''} \in S_s} \tilde{L}_s \left( x_s^*(t-1), \{v, \alpha(t), \beta(t)\}, \lambda_s(t-1), \{w(t-1), \gamma(t), \delta(t)\} \right), \quad (47a) \\
w_{s''}(t) &= \arg \min_{w_{s''}} \tilde{L}_s \left( x_s^*(t-1), \{v(t), \alpha(t), \beta(t)\}, \lambda_s(t-1), \{w, \gamma(t), \delta(t)\} \right). \quad (47b)
\end{align*}
\]

Assuming that \(\alpha^{s''}(0) = \beta^{s''}(0) = 0\) and \(\gamma_{s''}(0) = \delta_{s''}(0) = 0\) and relying on (46a)-(46d), the slack update rules (47a)-(47b) give rise to the following closed form expressions
\[
\begin{align*}
v^{s''}(t) &= \frac{1}{2} \left( x_s^*(t-1) + x_s^{s''}(t-1) \right), \quad (48a) \\
w_{s''}(t) &= \frac{1}{2} \left( \lambda_s(t-1) + \lambda_s(t-1) \right). \quad (48b)
\end{align*}
\]

Set \(\Phi^* \triangleq \sum_{s' \in \Gamma_s^e} (\alpha^{s''} + \beta^{s''})\) and \(\Psi_s \triangleq \sum_{s' \in \Gamma_s^e} (\gamma_{s''} + \delta_{s''})\). From (46a)-(46b) and (46c)-(46d), we get that \(\Phi^*\) and \(\Psi_s\) are updated according to the rules
\[
\Phi^*(t) = \Phi^*(t-1) + \zeta \sum_{s' \in \Gamma_s^e} \left( x_s^*(t-1) - x_s^{s''}(t-1) \right), \quad (49a)
\]
\[
\Psi_s(t) = \Psi_s(t-1) + \zeta \sum_{s' \in \Gamma_s^e} \left( \lambda_s(t-1) - \lambda_s(t-1) \right). \quad (49b)
\]

Let \(\tilde{\beta}_s > 0\) be a penalty factor for the proximal first-order approximation for \(s \in S\).

Following [50], from (48a)-(48b), the primal update rule for

**Algorithm 3 Distributed Proximal Algorithm for \(G_p\)**

**Require:** \(\zeta > 0, \tilde{\beta}_s > 0, \forall s \in S, \epsilon_{\text{stop}}, \epsilon_{\text{stop}}, t_{\text{max}}\)

1. **Initialization Step**
2. Each ASP \(s\) builds initial estimate \(x^0(0) \in \tilde{F}\) and \(\lambda_s(0) \geq 0\)
3. Set \(\alpha^{s^*} = \beta^{s^*} = 0\) and \(\gamma_{s^*} = \delta_{s^*} = 0\), \(\forall s^* \in S, \forall s^* \in \Gamma_s^e\)
4. while \(\epsilon_{\text{prim}}(t) \geq \epsilon_{\text{stop}}(t) \land \epsilon_{\text{dual}}(t) \geq \epsilon_{\text{stop}}(t) \land t \leq t_{\text{max}}\)
5. **Communication Step**
6. Each ASP \(s\) exchanges his previous estimate \(x^s(t-1)\) and his dual Lagrange multiplier \(\lambda_s(t-1)\) with his neighbors \(s' \in \Gamma_s^e\)
7. **Action Step Update**
8. for each ASP \(s \in S\)
10. \(x^s(t)\) is updated according to (49a)
11. \(\Psi_s(t)\) is updated according to (49b)
12. \(\lambda_s(t)\) is updated according to (51)
13. \(x^s(t)\) is updated according to (52)
14. end for
15. end while

ASP \(s\) is obtained by solving a local optimization problem
\[
x^s(t) = \arg \max_{x^s \in S_s} \left\{ \nabla_{x^s} R_s(x^s(t-1))^T (x^s - x^s(t-1)) - \frac{1}{2} \sum_{s' \in \Gamma_s^e} \lambda_{s'}(t-1) - \lambda_{s'}(t-1) - \frac{\lambda_{s'}(t-1)}{r_{s'}} \right\} + D x^s - D x^s(t-1)
\]

Dual update rule takes the form
\[
\lambda_s(t) = \text{proj}_{\sum_{s' \in \Gamma_s^e} [\lambda_{s'}(t-1) - \lambda_{s'}(t-1) + \frac{\lambda_{s'}(t-1)}{r_{s'}}]} \left( \frac{1}{2} \sum_{s' \in \Gamma_s^e} \lambda_{s'}(t-1) \right) - D
\]

**Theorem 5.** If \(f^*_s(\cdot)\) is \(K_s\) Lipschitz continuous for all \(s \in S, c \in C\), Algorithm 3 converges to the \(r\)-normalized Nash equilibrium solution to \(G_p\).
TABLE III

<table>
<thead>
<tr>
<th>Description of the parameters ($v_{a,s}$) in the ASPs’ QoS function.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td>Bandwidth ($v_{BW}$)</td>
</tr>
<tr>
<td>VCPU ($v_{CPU}$)</td>
</tr>
<tr>
<td>Memory ($v_{MEM}$)</td>
</tr>
</tbody>
</table>

**Proof.** See Appendix [D]

VIII. NUMERICAL EXPERIMENTS

In this section, we consider two numerical experiments to validate the proposed pricing and the multi-resource allocation schemes. In the first experiment, described in Section VIII.A, we consider the first case when application service providers own the finite budgets to procure the resources. In the second experiment, illustrated in Section VIII.B, we consider the second case where ASPs are not limited by their budgets.

A. Numerical Experiment I

In this experiment we consider a more general setting where ASPs can support multiple services; one service being defined per slice. For each type of service provisioning, ASPs require various kinds of resources. In this scenario, particularly to define the QoS as a function of multi resources, we consider a general class of utility function known as CES (constant elasticity of substitution), mathematically defined as

$$q_s(d_s) = \left( \sum_r v_{skr} d_{skr} \right)^{1/\rho},$$

where $\rho \in (-\infty, 0) \cup (0, 1]$ can be used to parametrize the whole family of utility functions. For example $\rho = 1$ corresponds to linear (additive) valuations $q_s(d_s) = \sum_r v_{sr} d_{sr}$, $\rho \to 0$ corresponds to Cobb Douglas function $q_s(d_s) = \Pi_r (d_{sr})^{\rho_{sr}}$, $\rho \to -\infty$ correspond to Leontief utility functions $q_s(d_s) = \min \{ d_{skr} \}$, and $D_s = (v_{s1} \ldots v_{srr})$, where $v_{srr}$ is the amount of resource type $r$ needed by ASP $s$ to support one unit of QoS. Linear valuation signifies the perfect substitutes, representing a scenario where the resources can replace each other in utilization. On the contrary, Leontief utility functions represent the perfect complement scenario where one resource may have no value without the other. For instance, the CPU and computer memory are both essential for completing a computing task. CSE utility function interpolate between perfect substitutes and the perfect complement through the parameter $\rho$. We consider that application service providers ASP 1 and ASP 2 support two types of services. ASP 1 and ASP 2 provide the service type 1 through slice $ASP_{11}$ and slice $ASP_{21}$, while they provide service type 2 through slice $ASP_{12}$ and $ASP_{22}$ respectively. We consider the number of end-users demanding service type 1 and service type 2 at cell $C_1$ are 120 and 180, while their numbers at cell $C_2$ are 130 and 170, respectively.

For numerical experiments, we consider that each service needs three types of resources, namely, Bandwidth (Gbps), vCPU, Memory (GB). We consider that the total available capacity of bandwidth and vCPU memory is fixed at 40 Mhz, 60 units and 400 GB respectively The values of parameters $v_{sr}$ for the application services providers and their respective services (slices) are as described in the table.

We apply our online learning algorithm to compute a NE. To evaluate the performance of the proposed allocation mechanism, we compare the revenue gained through the proposed resource allocation with three different baseline allocation schemes; first, with the static proportional resource allocation scheme (SS), where each resource required by the application service providers are allocated in proportion to their budgets; second, with the Fisher market equilibrium based resource allocation scheme (FM) proposed in [21] and [22]. We calculate the market equilibrium as a solution to the Eisenberg-Gale optimization problem, where the objective is to maximize the weighted proportional quality of service provided by ASPs

$$\text{maximize}_{d_e \in F_s} \sum_s \sum_c B_s N_c k \log \left( q_s^a (d_{sk}) \right),$$

subject to \( \sum_s \sum_c d_{ckm} \leq D_m \forall c \in C, m \in M \).

Third, we compare our proposed allocation scheme with the well-known socially optimal resource allocation scheme (SO), where resources are allocated to ASPs such that the weighted sum of the QoS provided by the ASPs is maximized [51]

$$\text{maximize}_{d_e \in F_s} \sum_s \sum_c B_s N_c k q_s^a (d_{sk}),$$

subject to \( \sum_s \sum_c d_{ckm} \leq D_m \forall c \in C, m \in M \).

For numerical simulations, we consider different values of parameter $\rho = 0.5, -5$ and budget $B_1 = 0.5$ and $B_1 = 0.7$. The bar graphs in Fig. [3] describe the respective revenue obtained by the ASPs under NE, SS, FM, and SO based resource allocations. We observe from the bar graphs that the revenues gained by ASPs for different values of parameters under NE and SS are almost equal and are in proportion to the budgets owned by the ASPs. We can observe that ASP with high marginal quality of service and high budget gains the maximum revenue under SO allocation. Thus, even if SO-based distribution maximizes the total system performance, it is at the cost of poor fairness in allocation. For the FM-based scheme, in Fig. [3(c)], we observe that even though ASP 1 has a higher budget than ASP 2, ASP 1 gains less revenue than ASP 2; again, this is also not fair from a business point of view. The bar graphs in Fig. [3(e)] describe the distribution of revenue gained by the ASPs from their different slices.

B. Numerical Experiment 2

We consider that application service providers ASP 1 and ASP 2 offer a service type 1 from the previous experiment.
For the bandwidth and vCPU, we keep their total availability fixed. At the same time, for the memory, we vary the available capacity from 100 GB to 400 GB, and we examine the its effects on the resources’ prices. First, we consider the case where the QoS provided by ASPs follows a substitutable relationship between resources, e.g., we rely on the CSE function with $\rho = 0.1$. Fig. 4(a) illustrates the effect of available capacity of resources on the resources price. As the total availability of memory increases. In this case, the cost of memory decreases. However, as the relationship between the resources is substitutable, we observe from the figure that a change in memory availability does not affect the price of the other resources.

Next, we consider the scenario where the ASPs’ QoS is defined by the CSE function with $\rho = -1.5$. In this case, the relationship between the resources is more complementary than the previous one. Fig. 4(b) illustrates the effect of available capacity of resources on the resources price. As the availability of memory increases, the cost of memory decreases. However, in this case, we observe that a change in memory availability also affects the price of the other resources. The cost of bandwidth and VCPU also increases with a rise in memory’s availability. An increase in the capacity of memory gives ASPs room to improve their QoS, but at the cost of increasing the other related resources, Bandwidth and VCPU. Thus, it causes congestion at Bandwidth and VCPU hence resulting in a rise in their prices. Similarly, Fig. 4(d) presents a change in the prices of the resources with respect to the capacity of the memory where QoS is considered a CSE function with $\rho = -2.5$. In Fig. 4(d) we demonstrate the fast convergence of Algorithm 3. For simulation purposes, we consider the availability of Bandwidth, VCPU and Memory as 40 Gbps 60 units and 100 GB, respectively. The plot in the figure shows the exact convergence of total demand for all three resources to their available capacity. Fig. 4(e) shows the convergence of error in decision variable (resources’ demand) by the ASPs and the convergence of error in price $\lambda$.

IX. Conclusion

In this work, we have considered a setting where application service providers lease resources from an network slice provider through a network slicing mechanism and compete with one another to serve a large pool of end-users. We have shown that the interactions between the end-users and application service providers can be modelled as a Stackelberg game, where the ASPs act as leaders and the end-users as followers. In addition, we have proved that the competition between the ASPs results in a multi-resource Tullock rent-seeking game, which admits a unique Nash equilibrium. The market price is computed by the NSP for each resource, taking into account the finite capacity of the network. To compute the market price and resource allocation, we have proposed two innovative market mechanisms. First, we have implemented a trading post mechanism taking into account the fact that the ASPs have bounds on their budgets. We have proved that the non-cooperative game induced by the trading post mechanism
admits a unique Nash equilibrium in case a single resource is considered. We have implemented a semi-decentralized exponential learning algorithm to compute the unique Nash equilibrium of this game. However, this mechanism does not enable an explicit incorporation of the network finite capacity constraints. To overcome that limitation, in a second design, when ASPs have no bound on their budgets but take into account the network finite capacity as a global coupling constraint, we have shown that the market equilibrium can be obtained by solving a generalized Nash equilibrium problem. We have provided a dual averaging-based semi-decentralized algorithm to compute solution of the extended game reformulation of the pricing game, and a proximal inexact-ADMM based distributed algorithm that provably converges to the Variational equilibrium of the pricing game. Finally, we have provided numerical results to analyse the economic properties of the two market designs, and confirm the fast convergence rate of the inexact-ADMM highlighting its practical applicability.

Appendix

A. Proof of Proposition 3

To find the equilibrium of the replicator dynamics defined in (6), consider \( \log \left( \frac{q^c_s}{n^c_s} \right) - p_s = \log \left( \frac{q^s_{s'}}{n^s_{s'}} \right) - p_{s'} \). Taking the exponential of both sides, we obtain \( \frac{q^c_s}{n^c_s} n^{c'}_{s'} = e^{p_{s'} - p_s} \). Summing over all \( s' \in S \) gives us \( \sum_{s'} \frac{q^c_s}{n^c_s} n^{c'}_{s'} = \sum_{s'} q^s_{s'} e^{p_{s'} - p_s} \), which can be rewritten as

\[ n^c_s = \frac{N^c q^c_s e^{-p_s}}{\sum_{s'} q^s_{s'} e^{-p_{s'}}}, \quad \text{i.e.,} \quad n^c_s = \frac{N^c f^c_s(d^c_s)}{\sum_{s'} f^s_{s'}(d^s_x)} e^{-p_s}. \]

B. Proof of Proposition 2

Consider that for any bid \( b^2_s > 0 \) submitted by ASP 2 at cell \( c \), ASP 1 places a bid of \( b^1_s = B_1 \frac{b^2_s}{b_2} \) at cell \( c \). Then, the quantity of resource received by ASP 1 at cell \( c \) is \( d^1_c = \frac{B_1 b^2_s}{b_1 + b_2} = B_1 \frac{b^2_s}{b_1 + b_2} \). This proves that for any strategy played by ASP there exists a strategy for the other ASP such that he receives the resource in proportion to his budget.

C. Revision Protocol

Let recall the revision protocol, which defines the switching rate at which users switch their choice from ASP \( s \) to ASP \( s' \) given population state \( n \tau_{s,s'} = n_{s'} [\nu_s - \nu_s]_+ \). Note that for the sake of simplicity, we omit the cell dependence (c).

Relying on the evolutionary process (4) and by substitution of the revision protocol, we get

\[ \dot{n}_s = \sum_{s'} n_{s's} \tau_{s,s'} - n_s \sum_{s'} \tau_{s,s'}, \]

\[ \dot{n}_s = \sum_{s'} n_{s's} [\nu_s - \nu_{s'}]_+ - n_s \sum_{s'} n_{s's} [\nu_{s'} - \nu_s]_+, \]

\[ \dot{n}_s = n_s \sum_{s'} n_{s's} [\nu_s - \nu_{s'}], \]

\[ \dot{n}_s = n_s \left[ \nu_s - \sum_{s'} n_{s's} \nu_{s'} \right]. \]
we obtain that linear in the ASPs' decision variables. Though we introduce $G$ then for any $L$

\[
\|\mathbf{x}\|_2 = \sqrt{\sum_{c} p_c N^c \left( \sum_{s' \neq s} f_{c}^{s'}(d_{c}^{s'}) e^{-p_{s'}} \right)^2} + \sum_{s' \neq s} f_{c}^{s'}(d_{c}^{s'}) e^{-p_{s'}}
\]

For any $d, \tilde{d} \in F$, $\|\nabla_{\mathbf{x}} \mathcal{R}_a(d) - \nabla_{\mathbf{x}} \mathcal{R}_a(\tilde{d})\|_2 \leq \sum_{c} p_c N^c \right\}
\leq \sum_{c} p_c N^c \max \left\{ \begin{array}{l}
\|\nabla_{\mathbf{x}} f_{c}^{s}(d_{c}^{s}) e^{-p_{s'}} \right\}_2 \| f_{c}^{s}(d_{c}^{s}) e^{-p_{s'}} \| - \sum_{s' \neq s} f_{c}^{s'}(d_{c}^{s'}) e^{-p_{s'}} \| \right\}_2 \|
\leq \sum_{c} p_c N^c \max \left\{ \begin{array}{l}
\|\nabla_{\mathbf{x}} f_{c}^{s}(d_{c}^{s}) e^{-p_{s'}} \right\}_2 \| f_{c}^{s}(d_{c}^{s}) e^{-p_{s'}} \| - \sum_{s' \neq s} f_{c}^{s'}(d_{c}^{s'}) e^{-p_{s'}} \| \right\}_2 \|
\leq L_s \left\| d - \tilde{d} \right\|_2 \leq L_s \sqrt{N \sum_{c} \|\mathbf{M}^c\|_F \| d - \tilde{d} \|_2}.
\]

This proves that $\nabla_{\mathbf{x}} \mathcal{R}_a(\mathbf{s})$ is Lipschitz continuous. In addition, the coupling constraints in the pricing game $G_p$ are linear in the ASPs' decision variables. Though we introduce projection operators in the norm of $\mathbf{M}^c$. Cauchy-Schwarz inequality implies that the norm of the projection matrix can be upper bounded by 1. This enables us to derive the same upper bound and statement as in [50]. Thm.1.

**REFERENCES**


