Iris: Deep Reinforcement Learning Driven Shared Spectrum Access Architecture for Indoor Neutral-Host Small Cells

Xenofon Foukas*, Mahesh K. Marina* and Kimon Kontovasilis†

*The University of Edinburgh, United Kingdom

†NCSR "Demokritos", Greece

Abstract

We consider indoor mobile access, a vital use case for current and future mobile networks. For this key use case, we outline a vision that combines a neutral-host based shared small-cell infrastructure with a common pool of spectrum for dynamic sharing as a way forward to proliferate indoor small-cell deployments and open up the mobile operator ecosystem. Towards this vision, we focus on the challenges pertaining to managing access to shared spectrum (e.g., 3.5GHz US CBRS spectrum). We propose Iris, a practical shared spectrum access architecture for indoor neutral-host small-cells. At the core of Iris is a deep reinforcement learning based dynamic pricing mechanism that efficiently mediates access to shared spectrum for diverse operators in a way that provides incentives for operators and the neutral-host alike. We then present the Iris system architecture that embeds this dynamic pricing mechanism alongside cloud-RAN and RAN slicing design principles in a practical neutral-host design tailored for the indoor small-cell environment. Using a prototype implementation of the Iris system, we present extensive experimental evaluation results that not only offer insight into the Iris dynamic pricing process and its superiority over alternative approaches but also demonstrate its deployment feasibility.

Index Terms

Indoor mobile access, small cells, neutral host, RAN slicing, C-RAN, shared spectrum, dynamic pricing, deep reinforcement learning.

I. Introduction

A. Background and Motivation

Mobile data traffic growth over the past decade and forecasts have been driving research on scaling capacity of mobile networks. Much of this demand is from indoors, amounting to 80% as of 2014 according to a Gartner study and expected to rise to over 95% by the time 5G gets deployed [1]. Indoor cellular coverage, however, has traditionally been poor. Outdoor solutions for indoor coverage are expensive due to building penetration losses [2]. Even Distributed Antenna Systems (DAS) are found to be expensive except for a few large venues like stadiums [3], [4]. Indoor small cells are considered relatively promising to address the coverage issue and scale the infrastructure with user density/demand. Indeed, making cells smaller and denser has historically been the biggest contributor to capacity scaling of cellular networks [5]. Despite this potential, indoor small cell deployments have been hampered due to operator concerns over deployment costs (and return on that investment) and issues such as site access and backhaul.

For the cost-efficient and simplified deployment of indoor small-cell networks for all operators, there is an emerging consensus around the notion of a *"neutral-host"* [6]–[15]. The key idea is that the site owner (i.e. the neutral-host) offers indoor mobile access as a building amenity by taking the responsibility of deploying and managing the small-cell infrastructure and by allowing multiple operators to share it for a fee that covers the neutral-host's CapEx and OpEx (e.g. deployment, management and electricity cost), thus offering *small-cells as a service*[1]. The neutral-host becomes the only entity that needs to address issues such as power and backhaul, relieving the operators of deploying their own infrastructure and dealing with the associated challenges. Considering the ever-increasing significance of mobile access for users, the site owner is motivated to act as a neutral-host by the need to provide a high quality of experience for the building residents and visitors (thus increasing the value of the property), while the operators are motivated to pay a fee to gain indoor access rather than relying on their outdoor RAN infrastructure in order to improve their indoor coverage [16] (although providing a service with degraded quality to indoor users through their outdoor infrastructure is still a valid option to avoid paying a fee to the neutral-host).

As virtualization is a natural means for sharing the small-cell infrastructure, the neutral-host concept aligns well with the 5G vision of supporting a diverse array of services across

[1]The neutral-host is a more general concept that could also be applicable in other settings (e.g. outdoor, rural, etc.), however our focus here is on the indoor scenario.

different Mobile Virtual Network Operators (MVNOs) and verticals via network slicing. From this perspective, the neutral-host provides each operator (also referred to as a *tenant* henceforth) a virtual radio access network (vRAN) spanning the area of the indoor environment it covers; this vRAN becomes part of the operator's end-to-end network solution, including its existing core network or a cloud realization of the core (e.g., [17]). However, a vanilla realization of the neutral-host concept that serves just traditional mobile network operators (MNOs) bringing their own licensed spectrum offers limited incentives for the neutral-host and operators alike [6].

We envision that the potential of the neutral-host's infrastructure sharing capability would be significantly amplified through access to a pool of spectrum that is dynamically shared among operators. Firstly, traditional MNOs would be able to gain access to additional spectrum for increasing their capacity and for offloading purposes. Secondly, by removing the requirement to possess licensed spectrum (which typically only a handful of operators have), it allows new non-traditional operators into the fray, who may come with innovative revenue models differing from the traditional subscription-based model (e.g., free access that is monetized by advertising and analytics *a la* Internet services and free mobile apps). Lastly, the aforementioned increase in the network capacity offered by the additional spectrum and the potential cost reductions offered to users (or even free access) can greatly improve their quality of experience, which as already mentioned is the main incentive of the site owner to act as a neutral-host in the first place. In fact, there is more support for the neutral-host model following the 3GPP defined multi-operator core network (MOCN) form of network sharing, which requires use of common spectrum shared between operators [6].

The neutral-host's common (and dynamic) spectrum pool could in principle be made up of licensed spectrum pooled from different MNOs, unlicensed spectrum or shared access spectrum [18]. Regarding the latter, recent regulatory developments below 6 GHz allow sharing of lightly used spectrum held by legacy or public-sector incumbents (e.g., radars) via tiered spectrum access models [19]–[21], offering substantial amounts of spectrum at a lower acquisition cost compared to licensed spectrum and without the complex coexistence issues of unlicensed spectrum. The Citizen Broadband Radio Service (CBRS) in the US [22] is a case in point, allowing the shared use of the 3.5 GHz band via a three-tier access model; a management entity called Spectrum Access System (SAS) ensures that when higher tier users need to use the spectrum, they get interference protection from lower tier ones. Licensed shared access (LSA) model for spectrum sharing [23] that is promoted for some bands in Europe, especially in its dynamic form, is another such relevant development. *In view of the above, we consider the scenario where the*

neutral host is powered by shared access spectrum in the style of CBRS or LSA.

B. Paper Overview and Contributions

The focus of this work is on addressing the challenges that arise with respect to managing access to shared spectrum in an indoor neutral-host small-cell environment, which constitute **the requirements for the desired system**:

1) As the neutral-host needs to support multiple (traditional and non-traditional) operators all competing in offering broadly the same type of service to users, the system should facilitate for each tenant service differentiation over rival tenants and control over its share of resources without requiring direct/explicit interaction among tenants. These are key concerns from the operators perspective to incentivize their participation in neutral-host small cells [6]. This requirement means that tenants should not have to reveal any private information regarding their business model to the neutral-host; instead they should operate in isolation with respect to each other and the neutral-host, and should be able to dynamically change their private spectrum valuations.
2) The system should provide a control mechanism to enable the efficient and dynamic spectrum sharing among tenants by aiming to closely match the spectrum supply with the tenants' demand. This should be done in a way that tenants who value the spectrum most get it, especially during periods of congestion (e.g., due to insufficient spectrum availability). Moreover, relying on shared spectrum implies that strict service level agreements (SLAs) may be infeasible for the tenants depending on the spectrum availability, so tenants should be served respecting this constraint.
3) The neutral-host should be able to cover its expenses for offering the service, including the fixed costs (e.g. deployment, management and electricity) and also a time-varying spectrum acquisition cost [24], depending on the amount of the shared spectrum acquired to meet the overall demand. This last cost needs to be recouped from the tenants in a dynamic manner, since any pre-agreed static fee may either overcharge the tenants or put the neutral-host in losses. Crucially, as already explained, the primary goal of the neutral-host is to provide a high quality of experience for building residents and visitors, and doing so without incurring losses. So revenue maximization is not the main driver although a revenue target linked with the neutral-host's incentive to provide the service with some profit margin (adjusted depending on the deployment environment) could also exist. Note that the environment in which the neutral-host operates is not a monopoly (e.g., tenants could opt to use their own external RAN with degraded quality of service).

4) The solution approach meeting the above requirements should be realizable in the context of a shared spectrum based neutral-host small cell system architecture that is practical in terms of algorithmic complexity, signaling overhead, etc.

Our key insight in this paper is that pricing can be an effective control mechanism to meet the first two aforementioned requirements. Pricing has been effectively employed in other contexts [25]–[28] to regulate demand and enable efficient sharing of resources with service differentiation, while it also naturally allows meeting the third requirement of neutral-host cost recovery and achieving a revenue goal if it exists. Given that tenant behaviors and traffic demands as well as spectrum availability can vary over time, a single optimal fixed price may not exist and thus pricing has to be dynamic. On the complementary side, we view a cloud RAN (C-RAN) [29] architecture to be more suitable for the indoor neutral-host small cell environment, due to the better scaling it offers in terms of spectrum availability, number of tenants etc., while allowing a cheaper and denser small-cell radio infrastructure. The result is our proposed approach Iris, a novel dynamic pricing shared spectrum access architecture for indoor neutral-host small-cells. The key components of the proposed Iris approach and our contributions are outlined below:

- (§III) At the core of Iris lies a dynamic pricing mechanism that regulates the allocation of spectrum to tenants by determining the price at which tenants can obtain a share of spectrum at any given time instant, while also considering the cost/revenue requirements of the neutral-host. In view of the stochastic nature of the neutral-host's environment with several unknowns (tenant behaviors, future demands and spectrum availability), we model the pricing decision problem as a Markov Decision Process (MDP) and resolve it using reinforcement learning. As the large state space and continuous action space of the problem make common reinforcement learning techniques slow and inefficient (as we experimentally demonstrate), we leverage recent machine learning advances and employ deep reinforcement learning. Unlike the Iris dynamic pricing mechanism, existing spectrum sharing mechanisms relevant for the neutral-host context [30]–[37] fail to meet some of the necessary requirements listed above, as discussed in the next section.
- (§IV) We design a neutral-host system following C-RAN and RAN slicing design principles that embeds the above pricing mechanism. We also develop a prototype implementation of Iris, with the goal of demonstrating the feasibility of our proposed mechanism, thereby satisfying the fourth practicality requirement above. To our knowledge, relative to existing neutral-host designs [38]–[40], this is the first design accounting for the peculiarities of spectrum sharing and the indoor small cell environment, along with a concrete implementation.

- (§V) Using the above mentioned prototype implementation, we demonstrate the system's feasibility in practice and conduct extensive experimental evaluations — characterizing the learning behavior of Iris , its performance in different conditions, and highlighting its superiority with respect to static pricing and alternative approaches from the literature [37], [41].

II. Related Work

Dynamic pricing in other contexts. Fundamentally, the pricing problem we have bears similarity with the pricing work in the Internet congestion control context [25], [41]. In these works, pricing is used as a signal to regulate the rates of senders for efficiently sharing network resources (e.g., bandwidth of links). Referring to [41], for example, each link in the network sets a price depending on its aggregate demand from all senders and each sender adjusts its rate independently in a way that maximizes its net utility after accounting for the bandwidth cost. The key difference from our case is that these works do not have the equivalent of requirement (3) (§I-B), regarding the need of the neutral-host to reach a revenue target that will allow it to cover its expenses.

Dynamic pricing has also been successfully applied in various other contexts where regulation of demand is required [27], [28], [42]. In those cases the focus is on controlling the end-user demand by the operators, unlike our case of spectrum sharing among operators via the neutral-host.

Spectrum sharing in the RAN slicing context. Neutral-host spectrum sharing can be seen as a specific form of RAN slicing and as such, RAN slicing mechanisms are relevant. There exist several algorithmic works [30]–[36] focusing on either the base station level (e.g., [33], [35]) or the RAN level (e.g., [30], [31], [34]), allocating radio resources to slices based on their SLAs. As all these mechanisms centralize the resource allocation at the infrastructure provider (neutral-host in our setting), they fail to meet requirement (1) (§I-B). Also, with the exception of [32] where revenue maximization for the infrastructure provider is considered, others do not meet requirement (3) of recovering costs and reaching the revenue target of the neutral-host. With respect to requirement (2), the focus on strict SLAs in these works may also be limiting when dealing with shared access spectrum.

A recent work [37] explicitly targets the shared spectrum neutral-host setting but shares the same limitations as the above mentioned works. It presents several pre-determined spectrum allocation policies at the neutral-host, mostly SLA based with the exception of one that assumes all tenants have the same utilities and allocates spectrum proportional to their traffic loads. We consider the latter in our comparative evaluations to highlight the service differentiation benefit of Iris.

Spectrum sharing without infrastructure sharing. The allocation of shared spectrum has also been considered in settings where operators deploy independent infrastructures [43]–[49]. Some works assume that participating operators have predetermined agreements regarding their priority for accessing the spectrum in cases of congestion (e.g., [43]), while others focus on the architectural aspect of the system (e.g., [45]) or on the coordination among operators (e.g., [47], [48]).

Auction-based dynamic spectrum sharing mechanisms. A rich body of literature on dynamic spectrum auction mechanisms is broadly related [50]–[59]. The most relevant from our context are [51], [52] but both have limitations from a practicality standpoint. The mechanism in [51] requires continual exchange of information between tenants and the neutral-host about each end-user device, and it allows only discrete number of traffic rates for tenant resource requests. [52] proposes a hierarchical auction-based mechanism that requires the involvement of end-users in the auction. More fundamentally, any auctioning mechanism for our setting has to handle a time-varying spectrum acquisition cost for the neutral-host along with its revenue target, which requires a *dynamic* reserve price. Iris's dynamic pricing mechanism essentially meets this requirement and thus can be seen as an enabler of auction-based spectrum sharing mechanisms for our setting.

Neutral-host system designs and specifications. A number of recent designs that consider multi-tenancy support in mobile RANs are applicable to the indoor neutral-host small cell setting. Perhaps the ones most relevant are: Orion [40], SESAME [38] and ESSENCE [39]. However, these works do not consider the use of shared spectrum and its implications, the main focus of this paper.

In terms of specifications targeting the neutral-host setting, nFAPI [60] is the most relevant one in which a functional split at the MAC layer is specified and each virtual operator is assigned a VNF implementing the higher-layer protocols. However, in contrast to our work, each tenant is assigned a static chunk of spectrum. Another closely related specification is MulteFire [61], which is a form of LTE deployment in unlicensed bands. In contrast to our work, the focus of MulteFire is on the ways to enable co-existence with other technologies operating over unlicensed spectrum (e.g. Wi-Fi).

III. Iris Dynamic Pricing Mechanism

This section describes the core component of Iris – its dynamic pricing mechanism.

A. System Model

Tenant resource requests. In our model, tenants express their resource requests in terms of radio resource blocks (RBs). We assume that tenants have a way to map their aggregate throughput

demands into the number of RBs required (e.g., by assuming an average spectral efficiency for every RB). Such a mapping is reasonable, considering that indoor small cell deployments are typically planned to provide near-optimal performance to users (UEs) within 20-30m [62].

Neutral-host and tenant interaction. The neutral-host follows a time slotted operation for the allocation of shared spectrum; we refer to each slot as an *epoch* henceforth. The duration of an epoch is expected to be short (e.g., 20-100ms), allowing the neutral-host to allocate radio resources in real-time. In each epoch t, the neutral-host determines a resource block price $p^t \in [p_{min}, p_{max}]$ with which the tenants can buy the available resources. The range of possible prices is assumed to be known to the tenants a priori (e.g., specified in their contract). Without loss of generality, we consider dynamic pricing for the allocation of downlink radio resources; the uplink can be treated similarly.

In each epoch t, all tenants see the price p^t announced by the neutral-host and decide how many resources to buy. The neutral-host is oblivious to the behavior of the tenants, not knowing the internal mechanism (possibly changing over time) that governs their decisions. Consequently, the high level goal of the neutral host would be to "predict" the demand of tenants at any point in time and dynamically decide on a price that would utilize the resources as efficiently as possible while recovering its cost and maximizing the tenant satisfaction, by allocating the (virtual) radio resources according to the expressed tenant demands. As discussed later, the model presented here is compatible with very general tenant behavior patterns, deterministic (e.g., driven by the optimization of utility functions) or not.

To model the temporal evolution of the tenants' demand, we divide a day into H periods, each e epochs long, so that He epochs make up 24h in the day. This construction makes a period correspond to an appropriate time interval within a day (e.g., an hour in a day) so that tenants' behavior is not expected to vary within a period but could across periods. Clearly, the shorter the period, the finer the granularity at which tenant behavioral changes can be captured. Without restricting generality, one may index periods within a day in the range $0 \leq h < H$ and may take the evolution of the system to start at epoch $t = 0$ coinciding with the beginning of a day. With this convention, the index of the current epoch t maps to the index of the current period of the day as: $h(t) = (t \operatorname{div} e) \bmod H$. It should be noted that the scheme imposes a natural synchronization, in which all tenants can always refer to the correct current epoch. In the rest of this section, we will use superscripts of the form $\cdot^t$ to denote the time dependency of any quantity including cases when it occurs indirectly through $h(t)$.

Shared spectrum acquisition cost and revenue target of the neutral-host. Let $n^t \in \mathbb{Z}$ be the number of RBs obtained by the neutral-host from an external/public spectrum repository in epoch t. To maintain flexibility, the pricing mechanism regards the interaction between the public repository and the neutral-host in abstract terms. Consequently, $\{ n^t, t \geq 0 \}$ is a stochastic process and the neutral-host, although informed about the current value n^t, is unaware of the process' future dynamics so dynamics of a very general form can be accommodated. The only assumption (to enable the MDP framework discussed later) is that n^{t+1}, conditioned on the value of n^t, follows a probability distribution (unknown to the neutral-host) that may depend on t and/or the current load of the tenants. This is a very mild assumption compatible with virtually all scenarios of practical interest.

To capture the neutral-host's incentive for participation, we introduce a target revenue level T. The value of T represents the per epoch revenue that the neutral-host expects to obtain through the dynamic pricing scheme for the particular small-cell in question. Generally, T can change dynamically as the neutral host seeks to offset its OpEx that encapsulates not just fixed costs like electricity and management of the infrastructure, but also dynamic costs like the the cost for the amount of RBs n^t obtained from the external spectrum repository in epoch t. In the following we will use the notation $T(n^t)$ to signify this functional dependence. This notion of a target revenue level is general enough to also capture other types of expenses (e.g. electricity) and could also be used to accommodate more general profit aims of the neutral-host (e.g., to dynamically adjust its profit margin according to the conditions of its environment).

System dynamics and neutral-host's small-cell resource allocation. Let I be the set of tenants served by the system. For each tenant $i \in I$, the expected load of a cell in epoch t is denoted by l_i^t, representing the total traffic that tenant i is expected to serve during epoch t. For example, this could be the bytes stored in the transmission buffers of all the UEs of the tenant in the cell and a forecast of any new traffic expected during epoch t. This can accommodate very general dynamics for the evolution of l_i^t, for all $i \in I$. The only assumption made (to enable the MDP formulation) is that l_i^{t+1}, conditioned on the value of l_i^t, follows a probability distribution that may depend on one or more of: the time t, the amount of radio resources n^t, and the price p^t.

The tenant i's behavior at epoch t is captured through its RB request $\nu_i^t \in \mathbb{Z}$. The dynamics of ν_i^t can be general, the only restriction being that ν_i^{t+1}, conditioned on the value of ν_i^t, follows a probability distribution (unknown to the neutral-host), whose form may depend on one or more of: the time t, the current tenant load l_i^t and the price p_t.

The collective request across all tenants may exceed the amount of available resources, i.e., it is possible for $\sum_{i\in I}\nu_i^t > n^t$. In such a case, the neutral-host would allocate the available resources proportionally to the tenants' requests. By using u_i^t to denote the resources actually allocated to tenant i in epoch t, this allocation rule translates into

$$u_i^t(\vec{\nu}^t) = \frac{\nu_i^t}{\sum_{j\in I}\nu_j^t}\min\{\, n^t, \sum_{j\in I}\nu_j^t \,\}, \qquad \forall i \in I, \tag{1}$$

where $\vec{\nu}^t$ stands for a vector containing the tenants' requests. Always, $\sum_{i\in I} u_i^t \leq n^t$. And $u_i^t = \nu_i^t$ for all $i \in I$, when $n^t \geq \sum_{i\in I}\nu_i^t$

B. Problem Formulation

Given the system model just described, we now formulate the neutral-host's dynamic pricing problem as a discrete-time, continuous state and action space MDP. Essentially, the neutral-host observes the state of its environment and makes a decision for an action (price p^t) based on this observation, getting a reward while transitioning the environment into a new state. We denote this action as $a^t \in A$ where $a^t = p^t$.

The state (x^t) of the neutral-host's environment in epoch t is made up of the vector $\vec{\nu}^{t-1}$ of virtual radio resources requested by the tenants in the previous epoch $t-1$, a vector $\vec{l^t}$ containing the current loads l_i^t of the tenants and the number of available radio resources at the neutral-host n^t. That is,

$$x^t := (\vec{\nu}^{t-1}, \vec{l^t}, n^t) \in X. \tag{2}$$

Note that x^t is known to the neutral-host as it either contains information maintained by itself or obtained from the tenants every time they request resources.

The reward function of the neutral-host is designed so that it can capture the first three requirements of the system as identified in Section I-B. Using the action notation a^t for the price decision p^t, the reward function is defined as

$$r(x^{t+1}, a^t \mid x^t) = f(\tfrac{n^t - \sum_{i\in I}\nu_i^t}{n^t}) g(\tfrac{a^t \sum_{i\in I} u_i^t(\vec{\nu_i^t})}{T(n^t)}). \tag{3}$$

Indeed, the first requirement of the system is met, as the tenant behaviors are hidden from the neutral-host, which only has tenant resource requests (ν_i^t) to glean that information (as part of the state given as input to the reward function).

The second requirement, i.e., avoiding a mismatch between resource supply and demand is handled through the first factor on the right hand side of (3). In the argument of the function f, this mismatch is expressed in a relative sense to make the reward function behave in the same way regardless of the amount of available spectrum. A zero value of this argument signifies a desirable perfect match between supply and demand. Positive values of the argument signify resource under-utilization ($n^t > \sum_{i\in I} \nu_i^t$). Avoiding it would leave room to allocate more resources to tenants and increase their satisfaction. Negative values of the argument signify excess demand ($n^t < \sum_{i\in I} \nu_i^t$). This should be avoided, as the proportional allocation in rule (1) gives some tenants fewer RBs than those requested at this price, leading to a decrease in their satisfaction. These are all met by defining the function f as

$$f(x) = e^{-x^2/\sigma^2}, \qquad \sigma > 0. \tag{4}$$

With this Gaussian form, the highest contribution to the reward is attained when $x = 0$ (perfect match) while positive or negative mismatches are penalized at an exponential rate. The parameter σ tunes the "sensitivity" of f – smaller values of σ penalize resource mismatches more aggressively.

The second factor of (3) corresponds to the third system requirement of avoiding a mismatch between the actual and target levels of revenue. The argument of the function g is the ratio of these two levels. The ideal case is when this argument is equal to 1, a perfect match between actual and target revenues. When the argument is smaller than 1, the actual revenue is below the target. This should be avoided as it signifies a reduced incentive for the neutral-host to provide the service (the neutral-host is experiencing losses). When the argument is greater than 1, the revenue exceeds the target level. This should also be avoided as it suggests that a lower price could also satisfy the goal of the neutral-host which could potentially improve the satisfaction of the tenants. All these features can be incorporated, by letting

$$g(x) = \Big(\frac{\min\{1, x\}}{\max\{1, x\}}\Big)^{\delta}, \qquad \delta \geq 0. \tag{5}$$

The highest reward contribution occurs when there is a perfect match ($x = 1$) while mismatches are penalized according to a power law. The value of δ tunes the sensitivity of g, higher values of δ penalizing revenue mismatches more aggressively. Moreover, the joint tuning of the parameters σ and δ can adjust the relative importance between the two factors f and g of the reward function. By making the values of any of these parameters smaller, the neutral-host tends to care more about the utilization of resources by the tenants and to disregard its own level of revenue. Increasing the values of the parameters has a reciprocal effect.

With the reward function (3), the behavior of the neutral-host is defined by a policy π, which maps the states to a probability distribution over the actions $\pi : X \rightarrow Pr(A)$. With the mild assumptions stated in §III-A and the neutral-host's state as in (2), the state transitions from x^t to x^{t+1} given the action a^t satisfy the Markov property and thus, applying a policy π to this MDP defines a Markov chain. We denote expectations over this chain by $\mathbb{E}_\pi$. We define the return from a state x^t as the sum of the discounted future rewards: $R^t = \sum_{\tau=t}^{\infty} \gamma^{(\tau-t)} r(x^{\tau+1}, a^\tau \mid x^\tau)$ for a discounting factor $\gamma \in [0, 1]$ [63]. The goal of the neutral-host then is to find a pricing policy that will maximize its expected returns from the start state $\mathbb{E}_\pi [R^0]$ over an infinite horizon. It should be noted that the choice for using a discounted rather than an average reward was mainly driven by the unpredictability of the environment, which can potentially change over time (e.g. tenants changing their spectrum acquisition policy).

C. Deep Reinforcement Learning Solution

Reinforcement learning is a common way to solve MDP problems where an exact model describing the dynamics of the environment (e.g., tenant behavior and network traffic in our case) is unavailable. Q-learning [64] is a well-known algorithm for such problems. Q-learning employs an action-value function Q^π which describes the expected future return after taking the action a^t in some state x^t and following policy π from that point on, i.e.,

$$Q^\pi(x^t, a^t) = \mathbb{E}_\pi \left[R^t | x^t, a^t \right]. \tag{6}$$

This function can be expressed through a recursive relationship known as the Bellman equation:

$$Q^\pi(x^t, a^t) = \mathbb{E}_{r^t, x^{t+1} \sim X} \left[r(x^t, a^t) + \gamma Q^\pi(x^{t+1}, \pi(x^{t+1})) \right]. \tag{7}$$

The policy used for the estimation of the discounted future reward of Q-learning is the greedy policy $\pi(x) = \arg\max_a Q(x, a)$ whereas an exploration policy is employed for the state transitions (e.g., take random actions). This makes Q-learning an off-policy method in that the policy π used to estimate the discounted future reward is different from the policy used for the action of the learning agent in a state transition.

Though an obvious choice, Q-learning is not appropriate to our problem for several reasons. Firstly, it uses a table to store its Q-values. When the state space of the problem is continuous or very large (as in our problem due to the range of possible values for l_i, ν_i and n), calculating

Q^π using a table becomes challenging. To overcome this, we need to rely on function approximators [65] parametrized by θ^Q. These parameters can be optimized by minimizing the loss:

$$L(\theta^Q) = \mathbb{E}_{\pi'}\left[(Q(x^t, a^t|\theta^Q) - y^t)^2\right], \tag{8}$$

where

$$y^t = r(x^t, a^t) + \gamma Q(x^{t+1}, \pi(x^{t+1})|\theta^Q). \tag{9}$$

In addition to the large state space, we also have to deal with a continuous action space (the announced price) which needs to be discretized in order to use Q-learning. However, there is no obvious or straightforward way to discretize the prices since the price range and its interpretation can be environment dependent [65].

We find that policy gradient actor-critic algorithms (e.g., [66]) are more suitable for our purpose. Such algorithms maintain a parametrized actor function $\pi(x|\theta^\pi)$ that estimates an action policy and a parametrized critic function $Q(x, a|\theta^Q)$ that estimates the Q-values of action-state pairs through the Bellman equation, as in Q-learning. The actor policy is improved at each step by performing a gradient descent considering the estimated values of the critic. Recent works (e.g., [67]–[70]) show that using deep neural networks as the function approximators for the estimation of actors and critics can produce better results compared to using linear approximators, when the learning task presents similar complexity to the one we have in terms of its dimensionality, including higher rewards (avoiding local optima) and improved convergence speed in some cases.

In view of the above, we choose to use a state-of-the-art deep reinforcement learning actor-critic algorithm called DDPG [67], which has been shown to consistently provide good results for a wide range of problems and learning environments [69], [71]. The use of deep neural network approximators allows DDPG to scale to high-dimensional state spaces and operate over continuous action spaces, ideal for our problem. One of its key features is the use of replay buffers (a type of cache) to sample prior transitions $(x^t, a^t, x^{t+1}, a^{t+1})$ which are used to train the neural networks. It also uses a technique called batch normalization that improves the effectiveness of the learning process when using features with different units and ranges (e.g., RBs and time). Finally, it uses a technique that employs slow-changing copies of the actor and critic networks, called target networks, which are used for calculating y^t. This has been shown to greatly improve the stability of the learning method.

Algorithm 1 gives an outline of the dynamic pricing mechanism in Iris. A new price p^t is selected at each epoch t (`line 5`) using the policy of the actor $\pi(x|\theta^\pi)$. Some exploration noise ϵ is added to the price to allow the agent to explore other states. The price is announced

Algorithm 1 Iris Dynamic Pricing Mechanism Outline

1: **procedure** DYNAMICPRICE
2: $t \leftarrow 0$
3: Receive initial network state x^0 (`state in first epoch of the day`)
4: *loop*:
5: Choose a price p^t given the policy of the actor
6: $a^t \leftarrow p^t + \epsilon$, where ϵ is exploration noise
7: Execute action a^t (`announce price to tenants`)
8: Collect the radio resource requests of tenants $\vec{\nu^t}$ and distribute the allocated RBs $u_i^t(\vec{\nu^t}), \quad \forall i \in I$
9: Calculate the reward r^t and transition to the state x^{t+1}
10: Update the actor-critic parameters θ^Q and θ^π (`DDPG`)
11: $t \leftarrow t + 1$
12: **goto** *loop*.
13: **end procedure**

to tenants (`line 7`) and their radio resource requests are collected in return[2]. Based on these requests, Iris neutral-host allocates the radio resources following the rule in (1) (`line 8`); then calculates the reward r^t and transitions into a new state x^{t+1} (`line 9`). The parameters of the actor and the critic network are updated based on the DDPG algorithm (`line 10`), which is the training step, and a new epoch $t + 1$ begins. Note that the learning process of Iris never stops, allowing the pricing mechanism to re-train and adapt to new environments (e.g., as the tenants change their valuations for the radio resources over time). To achieve this Iris employs a constant learning rate for both the actor and the critic to update policies and, as already mentioned, uses a discounted reward to account for the unpredictability of the environment.

IV. IRIS SYSTEM ARCHITECTURE

Having described the dynamic pricing mechanism of Iris, we now present its overall system design and implementation.

A. Design

Our design builds on the observation that the small-cell infrastructure sharing capability offered by a neutral-host is a particular albeit compelling use case of the broader RAN slicing in the 5G

[2]An empty request is assumed if a tenant fails to respond at some epoch.

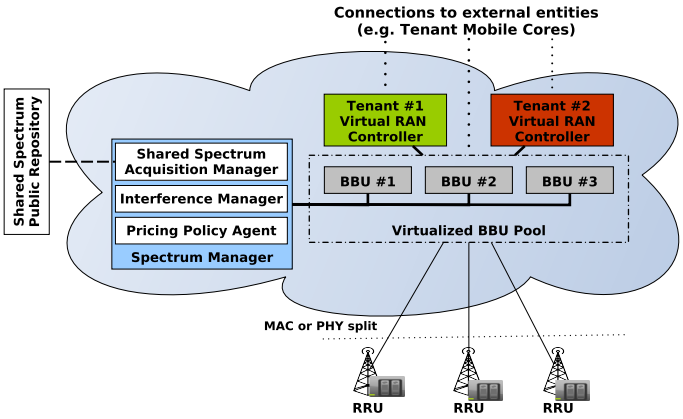


Fig. 1: Schematic of Iris neutral-host system architecture.

context. However, a vanilla RAN slicing system would be insufficient to address the specific needs of shared spectrum management and indoor small-cell environments. In the following, we highlight how we address these needs in our design (schematic shown in Fig. 1). It embraces the cloud RAN (C-RAN) design paradigm, with baseband processing units (BBUs) centralized in a virtualized BBU pool located in an edge cloud (e.g., in the basement of the indoor space) and remote radio units (RRUs) deployed throughout the building in a planned manner. The RRUs are connected to the BBUs over high speed channels (e.g., 10-Gigabit Ethernet or Fiber). This approach places most of the RAN processing on the edge cloud which allows the system to scale better as BBU resources can be adaptively provisioned depending on the number of tenants and the spectrum availability. It also lowers the form factor of the RRUs, making their deployment easier and discreet from a building aesthetics viewpoint.

Each tenant is allocated a *Virtual RAN Controller*, deployed as a Virtual Network Function (VNF) over the edge cloud. The controllers interface with the BBUs using message-based communication and provide tenant-specific functions such as schedulers and mobility managers, as well as an agent for the allocation of shared spectrum (discussed shortly).

At the heart of Iris lies the *spectrum manager*, a centralized controller managed by the neutral-host. This controller informs the BBUs about the amount and type of available spectrum (shared or privately owned) and about its valid allocations, depending on the access rights of tenants, distinguishing in particular between tenants operating exclusively over shared spectrum and tenants that can also use their own private licensed spectrum. A shared spectrum acquisition manager acquires the shared spectrum in a demand driven manner through a public repository

(e.g., SAS in the CBRS context). Moreover, this controller manages interference among small cells. Due to the system's C-RAN based design, the VNF of the spectrum manager co-exists with the virtualized BBU pool over the same edge cloud, simplifying its coordination with the BBUs through low-latency and high bandwidth channels and enabling the use of advanced interference management techniques like Coordinated Multipoint (CoMP) [72].

Shared spectrum allocation process in Iris. Crucially, the spectrum manager hosts a pricing policy agent that is responsible for dynamically deciding the price for the tenants to use shared spectrum resources. The functionality of the Iris dynamic pricing mechanism (§III) is distributed among three distinct agents as illustrated in Fig. 2.

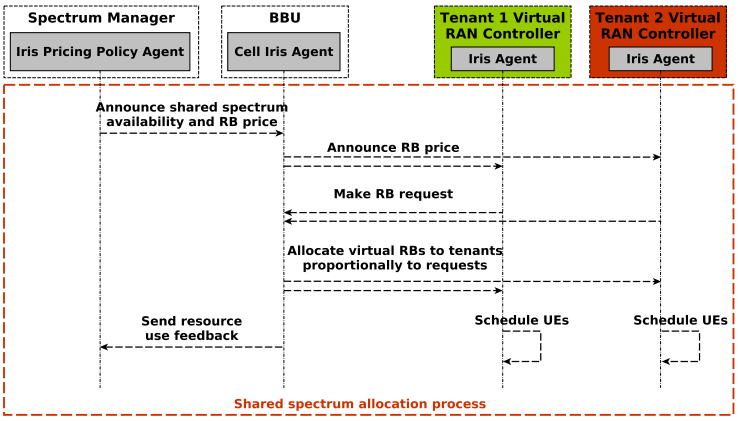


Fig. 2: Iris agents involved in dynamic pricing mechanism.

The *pricing policy agent* initiates the shared spectrum allocation process in each epoch, deciding on the price for each cell using the deep reinforcement learning algorithm described earlier in §III-C. The pricing policy agent announces the current epoch, the spectrum availability and the cell specific prices to the respective *cell agents* residing in the BBUs, which in turn convey the price to the *tenant agents* residing in the tenants' virtual RAN controllers. Each tenant considers the announced price along with its traffic load at the cell in question to decide on quantity of resource to be requested as dictated by its internal *private* policy. The tenant requests are aggregated at the cell agent, which distributes the available shared spectrum proportionally to the tenants' requests following rule (1) and notifies the pricing policy agent about the allocated resources, the load of the tenants etc. The schedulers running in the virtual RAN controllers of the tenants use the allocated resources to serve the traffic of their UEs as per the tenants' internal policies. Once the allocation process is complete, the pricing policy agent is further *trained* using the feedback from the cells and decides on a new price for the upcoming epoch.

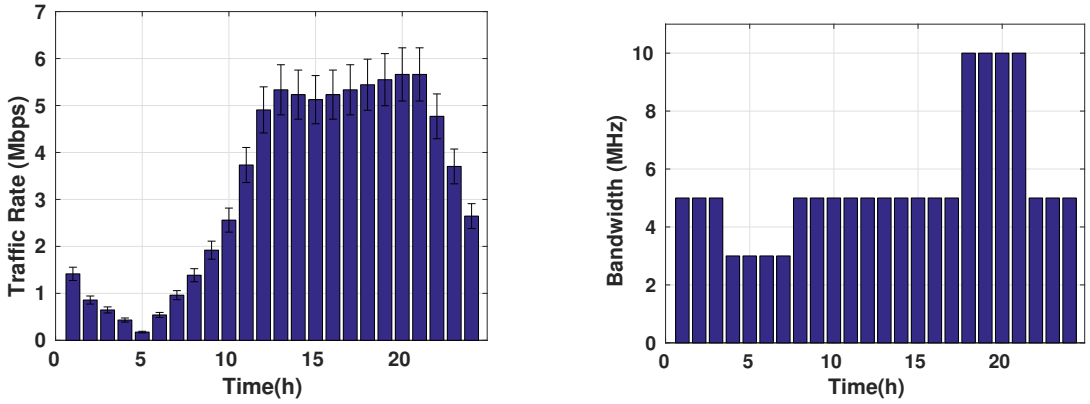
B. Implementation

Following the design described above and in order to be able to assess its practicality for a real deployment (as explored in §V-B), we developed a prototype implementation of Iris, considering LTE as the radio access technology (RAT). To realize RAN slicing, we leveraged the Orion RAN slicing system [40], which provides functionally isolated virtual control planes (RAN controllers) for tenants and virtualized radio resources revealed to them through a Hypervisor. Orion is in turn built on top of the OpenAirInterface (OAI) LTE platform [73]. OAI has built-in C-RAN support offering three functional splits: lower-PHY, higher-PHY and nFAPI [60]. Although in principle any of these functional splits could be used in Iris, the Orion implementation is only compatible with the first two. Between them, considering their differences in fronthaul bandwidth requirements (1Gbps with lower-PHY versus 280Mbps for higher-PHY for a 20MHz carrier) [74], [75], we opt for the higher-PHY functional split.

Edge Cloud Deployment. To realize the Iris system design, we leveraged the OpenStack edge cloud deployment of the University of Edinburgh presented in [76], which is composed of 5 compute nodes (24-core Xeon CPUs @2.1GHz and 32GB RAM each), optimized for real-time operation (disabled CPU C-states, low-latency Linux kernel, no CPU frequency scaling, VNF CPU pinning). For the RRUs, we employed USRP B210 Software-Defined Radios (SDRs), each interfaced to a small form factor PC (UP board with 4GB of RAM and Intel Atom x5 Z8350 CPUs @1.92GHz), the latter acting as a compute node for running the lower part of the PHY operations and for communication with the BBUs (over Gigabit Ethernet).

Spectrum Manager. We implemented a prototype Python-based spectrum manager to host the Iris pricing policy agent, employing an existing implementation of DDPG [77] that uses Tensorflow for the training of the deep neural networks. Given our hardware constraints, we used a Tensorflow flavor that supports execution only on CPU (no support for GPU). Regarding the parameters of DDPG, we retained the default values provided in the aforementioned implementation, with the actor and the critic neural networks both having two hidden layers with 400 and 300 units, respectively. For the representation of the state and action space, we employed the OpenAI Gym [78] toolkit.

In a full implementation, shared spectrum support would make use of carrier aggregation. However, given that this functionality is not currently supported by OAI, we used a contiguous band of spectrum to simulate scaling the available shared spectrum up/down (by the spectrum manager through signaling messages to the cell agents).



(a) Traffic load

(b) Spectrum availability

Fig. 3: Daily traffic profile of tenants and spectrum availability profile over a day at the neutral-host.

Cell Agents. Each cell agent in Iris is realized via modified Orion Hypervisor in the BBU. The radio resource allocation scheme of Iris following rule (1) was introduced into the radio resource manager of the Hypervisor. The Hypervisors were interfaced with the spectrum manager using Google Protocol Buffers [79] and ZeroMQ [80]. Finally, the protocol used for the communication of the Hypervisors with the virtual control planes of tenants was extended to support the messages required for the shared spectrum price announcements and the radio resource requests of the Iris tenant agents.

Tenant Agents. On the tenant side, we leveraged the Orion virtual control plane implementation, which we extended with the Iris tenant agents. Note that our design (specifically the dynamic pricing mechanism) is agnostic to tenant behaviors. For the sake of evaluations, our implementation supports a rich set of tenant behaviors enacted in the form of utility functions (described in §V-A). However, our implementation could also support other ways of expressing the tenant behaviors.

V. Experimental Evaluation

A. Evaluation Setup

For our evaluations, we employ the prototype implementation of Iris (§IV-B). The default experimental setup corresponds to 4 tenants per cell. For experiments with a single small cell, real UEs (LTE smartphones and dongles), one per tenant, and the D-ITG traffic generator [81] were used to generate a simulated aggregate UDP traffic for the tenant. UEs were simulated for scenarios with multiple small-cells due to the complexity of managing the experimental setup. It should be noted that even though certain aspects of the system like the generated traffic were simulated (and therefore a simpler simulation setup could also be used), employing the prototype

implementation is still crucial for the evaluation, since it provides critical insights regarding the applicability and the overhead of the proposed mechanism in real settings, as discussed in §V-B.

Tenant traffic loads and spectrum availability. To model the traffic loads of tenants, we employed the daily aggregate traffic pattern presented in [82] for an entertainment area, a representative indoor environment. We assume that the aggregate incoming hourly traffic of each tenant follows a normal distribution with mean and variance depending on the particular hour in the day considered as shown in Fig. 3a. With no real-world data to rely on, we consider a reasonable spectrum availability profile in Fig. 3b. The idea behind this profile is that the available shared spectrum is re-adjusted (with some delay) by the spectrum manager to approximately match the traffic load. While some of the experiments span the whole day and use the full profiles in Fig. 3, others focus on a particular hour and so use the traffic and spectrum values for that hour. The default evaluation configuration is for the hour starting at 3pm and a cell with 5MHz of available shared spectrum, emulating a CBRS-like service using LTE band 7. Small cells use a SISO transmission mode, which for 5MHz spectrum corresponds to a max throughput of 16Mbps.

Dynamic pricing mechanism settings. The epoch duration is set to 30ms (based on results in §V-B), while the presented results correspond to the parameter $\sigma = 1$ for (4) and $\delta = 1$ for (5). The price range is set to $[0, p_{max}]$ with $p_{max} = 2500$. In setting the target revenue level T, we consider the case where the neutral-host is concerned only with recovering the cost associated with shared spectrum acquisition. Accordingly, we set $T(n) = p_c n$, where p_c is the cost incurred to the small cell for the acquisition of a single RB. We use the value $p_c = 850$, unless explicitly stated otherwise.

Modeling different tenant behaviors. For this purpose, we came up with a generic parameterizable dis-utility function of the form in (10) based on the detailed analysis that can be found in the appendix.

$$\bar{U}(b; d, p) = \Big(\alpha(\max(0, d - b))^{\gamma_d} + (pb)^{\gamma_p}\Big)^{1/\gamma_p} \tag{10}$$

where, p denotes the price announced by the pricing policy agent while d and b, respectively, represent a tenant's traffic load and requested resources in terms of RBs. All arguments here refer to the same epoch. The parameters α, γ_d and γ_p characterize the individual tenant behavior; the settings of these parameters determine the sensitivity of the dis-utility function to the current load or price and can therefore allow modeling different tenant behaviors and reactions to price changes. These parameters can be modified on-the-fly, allowing the tenants to dynamically change their shared spectrum allocation policy. Raising the sum in (10) to the power $1/\gamma_p$ expresses the

TABLE I: Tenant profiles with different parameterizations of the generic disutility function and the resulting behaviors.

Profile Type	α	γ_p	γ_d	Effect
1. "Best effort"	3.5×10^8	2	1	The main focus of the tenant is to maintain a network presence, providing best effort services with a small amount of radio resources regardless of the price. The tenant is only willing to cover its load for low prices.
2. Price-driven	2×10^9	2	1	The tenant fully covers its load when the cost is low (e.g. off peak hours with no congestion). For high prices, it only covers part of its load.
3. Demand-driven	0.203	1	2	The tenant focuses on providing data-demanding services to its users (e.g. video streaming, IPTV). In times of high load the tenant is willing to buy a large amount of resources, regardless of the price. In other times, the tenant will queue its traffic until the load increases enough to buy in bulk.
4. "Medium" QoS level	1.1×10^5	2	2	Tenant tries to provide a medium level of service, asking a price-dependent fraction of its load.

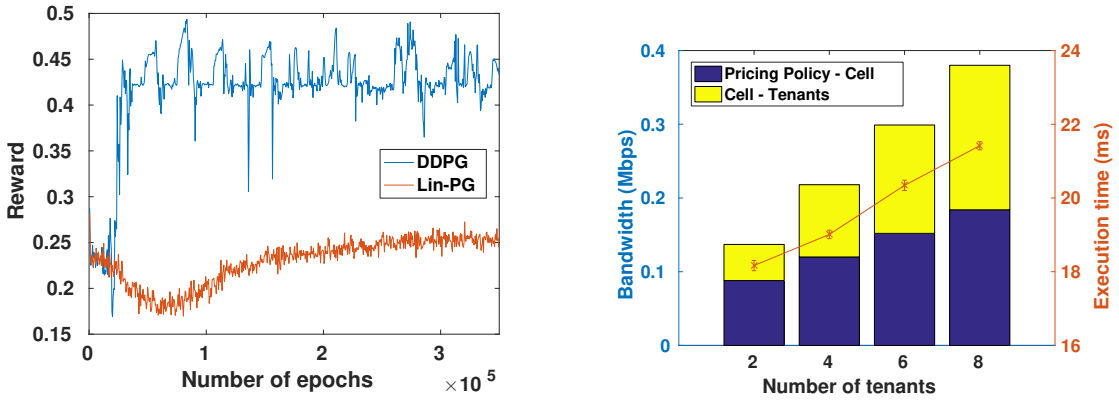
value of the dis-utility in units of "cost", bearing the same interpretation for all tenants. This allows introducing the notion of "total dis-utility" calculated as the sum of dis-utilities over all tenants. Through (10) and given the price p^t and the level of traffic load l_i^t, corresponding to $d(l_i^t)$ RBs, the agent of each tenant i requests from the Iris cell agent the number of RBs that minimizes its dis-utility i.e., $\arg\min_b \bar{U}_i(b; d(l_i^t), p)$.

Based on the above, we created 4 tenant profiles for our evaluations (Table I), using different parameterizations of (10) to model different tenant behaviors. The choice of parameters for these profiles was made based on the analysis in the appendix, with the goal of capturing a wide range of sensible and diverse tenant behaviors that would allow a more accurate and realistic evaluation of our mechanism. Unless explicitly stated otherwise, tenants were assigned these profiles in a cyclic manner, i.e., tenant 1 to profile 1, tenant 2 to profile 2, tenant 5 to profile 1 etc.

B. Deep Learning Benefits and Feasibility

We begin by examining the choice of employing deep reinforcement learning against simpler reinforcement learning algorithms for solving the problem formulated in §III-B. We compare the performance of Iris when using DDPG against the stochastic policy gradient algorithm of [83], which employs linear function approximators for the actor and the critic (Lin-PG). As illustrated in Fig. 4a, DDPG converges faster than Lin-PG and attains a much higher overall reward, both of which are critical characteristics for the success of the proposed mechanism in a real deployment.

Another very important aspect in terms of the mechanism's practicality is that the benefits of deep reinforcement learning and the requirement for the real-time communication of tenants



(a) Comparison of DDPG and policy gradient algorithm using linear function approximators.

(b) Iris single training step time and bandwidth requirements for Iris agents message exchanges

Fig. 4: Benefits/feasibility of deep reinforcement learning.

with the neutral-host should not come at the expense of the system's deployment and operation feasibility. For this reason, we use our prototype implementation to evaluate the performance of the system. As illustrated in Fig. 4b, the time required to calculate the new parameters of the actor and the critic functions by DDPG in a single training step increases linearly with the number of tenants, but remains below 22ms even for 8 tenants. This linear effect correlates with the computational complexity introduced by the linear increase in the number of input layer units in the neural networks of the actor and critic as the number of tenants grow – each tenant adds one load l_i and one request ν_i feature to the input layer. When setting the epoch duration, the additional overhead introduced by the message exchanges between the various Iris agents should also be taken into account. Therefore, setting the epoch duration to 30ms is a reasonable choice (also used in all our consecutive experiments), that provides a very fine granularity in terms of the neutral-host agent's training speed. Offloading the training computations to GPUs (rather than using the CPU as in our current implementation) can potentially lead to significant reductions in the execution time, which in turn allows a lower epoch duration and enables Iris to learn even faster. The bandwidth requirements of Iris for the message exchanges between pricing policy and cell agents as well as between cell and tenant agents are also illustrated in Fig. 4b. These requirements are minimal (less than 0.4Mbps) for all practical deployment scenarios of up to 8 tenants and posing a negligible overhead to the edge cloud deployment.

It should also be noted that the results illustrated in Fig. 4b only depend on the number of tenants sharing the infrastructure and are independent of the amount of traffic and the way it

was generated (simulated or real traffic) or of the actual number of UEs associated with the tenants of the system. Therefore, our prototype implementation provides us with a very accurate depiction of Iris's overhead, demonstrating the feasibility of the system's deployment.

C. Characterizing Iris Spectrum Management

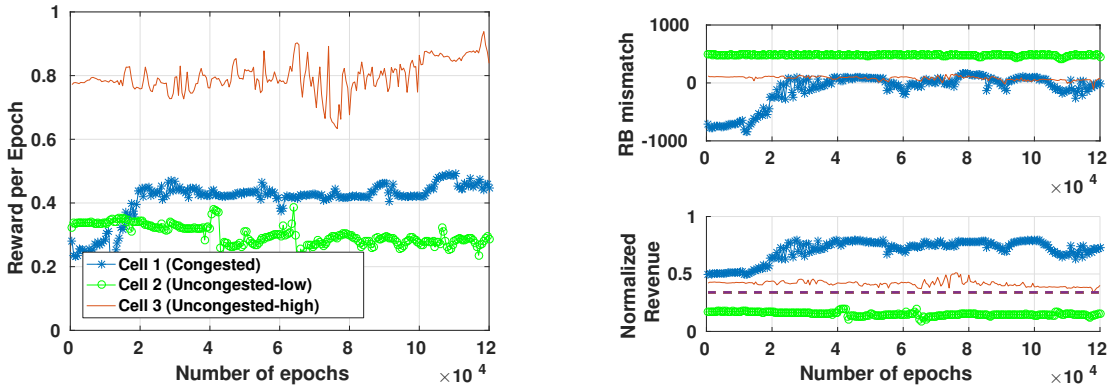
Learning behavior for various traffic loads. We evaluate the learning behavior of the pricing policy agent for four tenants under three different scenarios, each considering a cell with a different aggregate traffic load: (i) a congested cell (Cell 1), corresponding to the conditions at 3pm from the daily traffic and spectrum availability profiles of Fig. 3; (ii) an uncongested cell (Cell 2) with low traffic load, corresponding to 8am; and (iii) a cell with high traffic loads (Cell 3) but not in congested state, corresponding to 11am.

Fig. 5 shows the results for reward obtained by the pricing policy agent for each of the three scenarios considered. It also shows the average mismatch between the amount n^t of RBs available during an epoch and the amount $\sum_{i \in I} \nu_i^t$ collectively requested by all tenants, as well as the actual (target) revenue received (set) by the neutral-host, normalized by the maximum possible revenue for an epoch (equal to $p_{max} n^t$). In view of this normalization, the value of the target revenue T (dashed line) maintains the same value (equal to p_c / p_{max}) for all three scenarios.

In the congested (cell 1) case, the agent begins with a very high RB mismatch, which gets close to 0 after the first 20000 epochs, indicating that the pricing policy agent is effectively and quickly learning how to control the requests of the tenants. The neutral-host achieves this by increasing the price of the RBs as evident from its revenue increase. It should also be noted that the big difference between achieved and target revenue levels has an effect to the overall reward of the neutral-host (through function g), which converges to a value that is less than half of the max reward 1.

For cell 2, the mismatch is always positive (underutilization) and close to 500, since the load of the tenants is very low and the demand can never match the supply. Due to the very low load, it is infeasible for the neutral-host to fully recover its costs for acquiring the shared spectrum, regardless of the pricing policy it follows, something reflected in its reward that is scaled down by (5).

Finally, in the case of cell 3 the agent presents a stable behavior, with its RB mismatch and revenue remaining relatively static throughout the experiment. The aggregate traffic of the tenants requires an amount of RBs that is roughly equal to the RBs that are available in the system, while the revenue that is achieved by the neutral-host is very close to the target revenue, leading to an overall reward for the neutral-host agent that is much closer to the max compared to the other two cases.



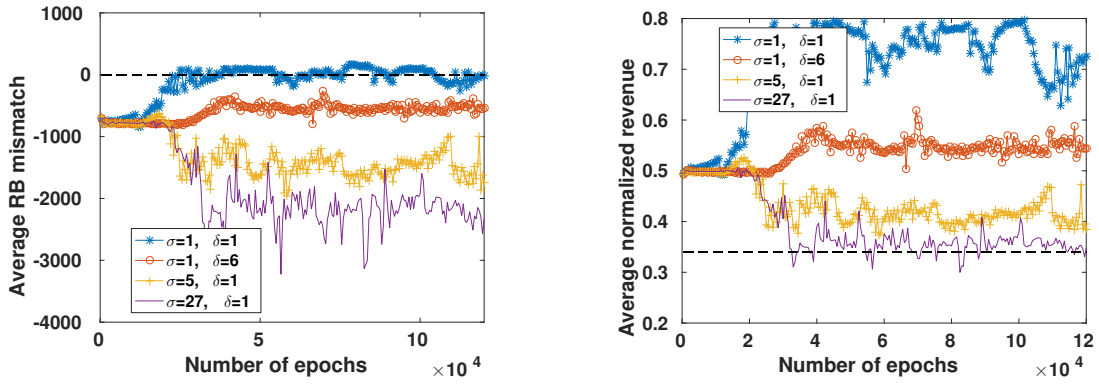
(a) Reward received by neutral-host per pricing epoch.

(b) RB mismatch and achieved revenue vs target revenue (dashed line).

Fig. 5: Learning behavior of pricing policy agent for cells with different levels of congestion/loads.

Effect of reward function. We explore how the configuration of the reward function affects the neutral-host's learning process, considering four variants of the reward function with different combinations for the parameters σ and δ, as shown in Fig. 6. In accordance with the discussion in §III-B, we can observe that as the value of σ increases, the RB mismatch becomes more unstable and/or higher from one round to the next (Fig. 6a), but at the same time the achieved revenue gets closer to the target revenue T (dashed line in Fig. 6b). The reason for this behavior is that higher values of σ can tolerate higher RB mismatches (since the bell curve of (4) widens). Therefore, even high mismatches yield relatively significant reward contributions from (4), something that simplifies the pricing decision, by making the pricing policy to be mainly driven by the other factor (5) of the reward function. The results in Fig. 6 also indicate (as per discussion in §III-B) that increasing the value of δ also promotes a closer match between actual and target revenue levels (driven by a more significant effect of (5)). Naturally, this has also an effect in the achieved RB mismatch, which increases. This is because the effort to match actual and target revenue levels triggers a lower price per RB, subsequently leading to an increased demand by the tenants.

Different number of tenants. We evaluate the learning behavior of the dynamic pricing mechanism as the number and behavior of active tenants vary (recall that the behavior of each tenant depends on its index, as explained in §V-A). We consider three cases with two, four and six tenants and a congested cell with the aggregate traffic of 16Mbps across all tenants with equal levels of traffic. Fig. 7a shows the average RB mismatch and the pricing choices of the neutral-host. For four and six tenants the system begins with a negative mismatch, while for two tenants with a positive



(a) RB mismatch.

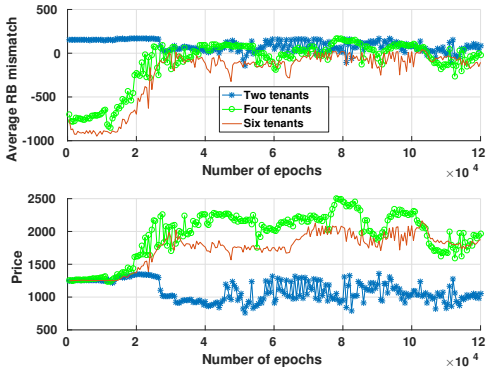
(b) Neutral-host revenue.

Fig. 6: Effect of reward function parameters on dynamic pricing behavior.

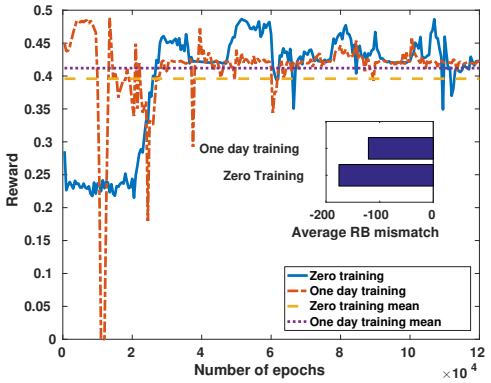
mismatch. This is related to the effect of neutral-host's pricing choices to the tenants, given their loads and shared spectrum allocation profiles. For two tenants, the initial prices are considered high, leading to the underutilization of the resources, despite the cell congestion. On the other hand, for four and six tenants and given the increased competition, the price is low, leading to excess demand. In all cases, the agent adapts and discovers an appropriate pricing policy to minimize the mismatch.

Effect of prior training. We evaluate the effect of prior training to the results achieved by the neutral-host. We perform an experiment for two consecutive days, starting from a state of no training and focus on the results obtained during the same hour of the day (3pm from the daily profile). As it can be seen in Fig. 7b, the second day yields improved results compared to the first day (zero training), which is evident both from the better mean reward (the brief drops of the instantaneous reward, lasting for less than a minute each, are inconsequential), as well as from the reduced average RB mismatch during the second day. However, the differences between the two days are small, demonstrating the effectiveness of the mechanism even without any substantial prior training.

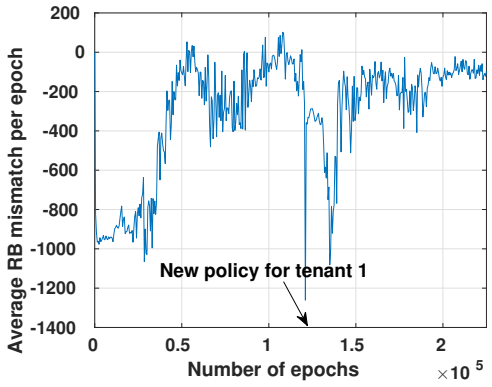
Effect of dynamic policy changes. This experiment demonstrates the adaptiveness of the dynamic pricing mechanism when tenants make policy changes dynamically. In this scenario, the experiment runs for 120000 epochs using the default tenant profiles. When this period elapses, and once the agent has identified an appropriate pricing policy for the given load, the first tenant's policy changes to profile #2 (Table I). This leads to a temporary failure of the agent to appropriately price the available radio resources, which is translated into a major RB mismatch (Fig. 7c). However, after about 30000 epochs (150000 epochs in the experiment), the neutral-host agent manages to re-adapt to the new behavior of tenant 1.



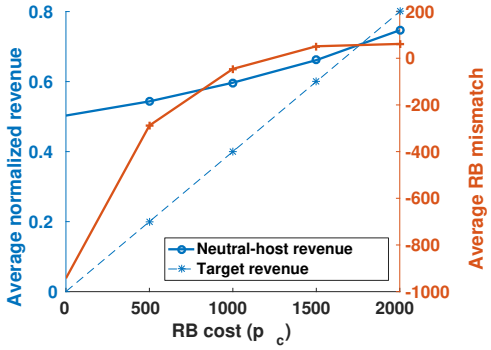
(a) Learning behavior for different number of tenants.



(b) Prior training effect to pricing mechanism.



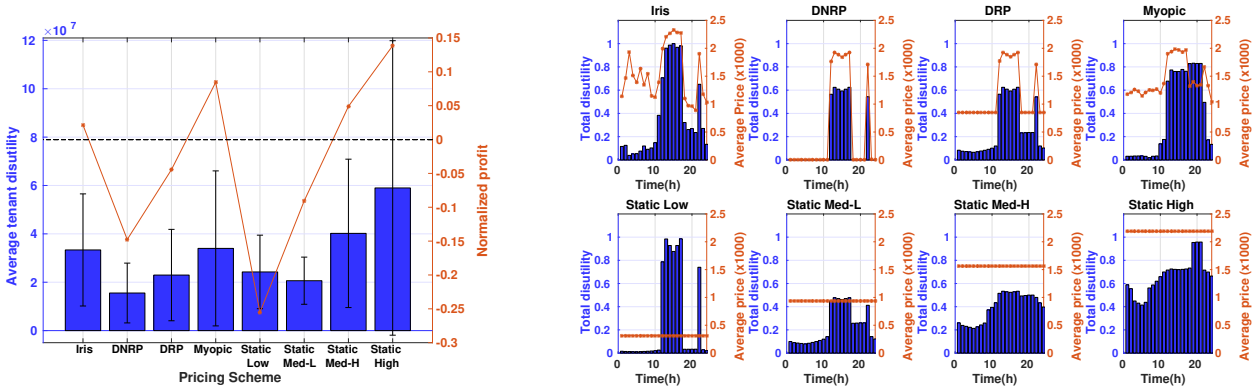
(c) Adaptation to dynamic policy changes.



(d) Impact of target revenue to neutral-host behavior.

Fig. 7: Behavior of Iris pricing mechanism under different conditions: varying number of tenants; prior training; dynamic tenant policy changes; and different costs for acquiring shared spectrum.

Effect of target revenue level/shared spectrum acquisition cost. We evaluate the effect of the target revenue level to the pricing decisions. Since we employ target revenue levels of the form $T(n) = p_c n^t$, changes to the target revenue level correspond to changes in the shared spectrum acquisition price p_c. We proceed by applying different values of p_c and comparing the actual (target) revenues received (set) by the neutral-host, again both normalized by the maximum possible revenue for an epoch. As it can be seen in Fig. 7d, both revenue levels increase along with p_c. For the lower acquisition prices, and due to the high load (traffic at 3pm) and the network congestion, tenants are willing to buy the RBs in prices much higher than p_c. Therefore, the neutral-host finds a balance on the goals of (3) by announcing lower prices (creating excess demand) in order to bring the actual and target revenue levels as close as possible (avoiding the overcharging of tenants for the resources). On the other hand, as the acquisition price increases, the gap between the target and



(a) Total dis-utility of tenants and profit of neutral-host.

(b) Hourly breakdown of tenants' total disutility and average price selected by neutral-host.

Fig. 8: Comparison of Iris with alternative approaches.

actual revenue levels decreases, with the target surpassing the actual revenue for $p_c = 2000$. At the same time, the RB mismatch becomes smaller and turns from excess demand to excess supply (positive mismatch) for $p_c = 1500$ and $p_c = 2000$. This is because the neutral-host, driven by its reward function, learns pricing policies that make the tenants buy less RBs on average, but at higher prices.

D. Comparison with Alternative Approaches

We compare the performance of the Iris dynamic pricing mechanism with alternative approaches in terms of the benefits provided to tenants. We consider the traffic generated for a whole day (full profile presented in Fig. 3) for four tenants with their behaviors defined in Table I. The agent is evaluated against other schemes without any prior training. This worst-case scenario is important to benchmark the effectiveness of the neutral-host agent's operation in volatile environments.

We compare the dynamic pricing mechanism of Iris against the distributed optimization algorithm proposed in [41]. Based on that, the neutral-host iteratively adjusts the price of the available RBs in order to control the behavior of tenants that are driven by the goal of minimizing their dis-utilities. Assuming a static environment, the algorithm in [41] has been shown to converge to an optimal solution in terms of the utilization of the available resources, but does not inherently capture the requirement of Iris regarding the revenue target of the neutral-host. For this, we consider two variants of [41]: (i) the vanilla version in which the neutral-host does not set a reserve price for the resources it distributes to the tenants (Distributed No Reserve Price—DNRP) and; (ii) a modified version, in which the neutral-host sets a reserve price equal to the cost of a resource block ($p_c = 850$), in an attempt to avoid experiencing losses (Distributed Reserve Price—DRP).

Another alternative we compare against and which could be viewed as a variant of the optimal solution is an unrealistic myopic pricing scheme in which the neutral-host knows the dis-utility functions of the tenants. Using this knowledge, it determines at each epoch, the price to charge the tenants by minimizing the sum of tenant dis-utilities, subject to the resource availability constraint and the requirement that the neutral-host matching or exceeding the target revenue level, i.e.,

$$\min_{p,\vec{\nu}} \sum_{i \in I} \bar{U}_i(\nu_i; d, p)$$

$$\text{s.t.} \quad p \sum \nu_i \geq T(n), \quad \sum \nu_i \leq n, \quad \nu_i \geq 0, \quad \forall i \in I$$

The neutral-host allocates the resources myopically during each epoch (in the sense that it views each epoch in isolation) so that it does not incur losses even in the short-term. A side-effect of this is that under very low traffic load, the neutral-host forces the tenants to buy more resources than they actually need, to recover the acquisition cost for the spectrum. It is noted that the comparative evaluation does not consider as a baseline the unrealistic but "optimal" solution which optimizes the allocation of resources considering the network dynamics (traffic, spectrum cost, tenant behaviors) throughout the whole day. The complex modeling requirements accounting for the dependency across epochs, the high computational complexity of obtaining the optimal solution and the fact that this needs to be performed over and over again in time makes it impractical even as a baseline.

In addition to the myopic scheme outlined above, we also consider four static pricing schemes, where the price announced by the agent during each epoch t is fixed to $p_{max}/8 \approx 312$ (Static Low), $3p_{max}/8 \approx 937$ (Static Med-L), $5p_{max}/8 \approx 1562$ (Static Med-H) and $7p_{max}/8 \approx 2187$ (Static High) correspondingly, to capture the whole range of possible prices.

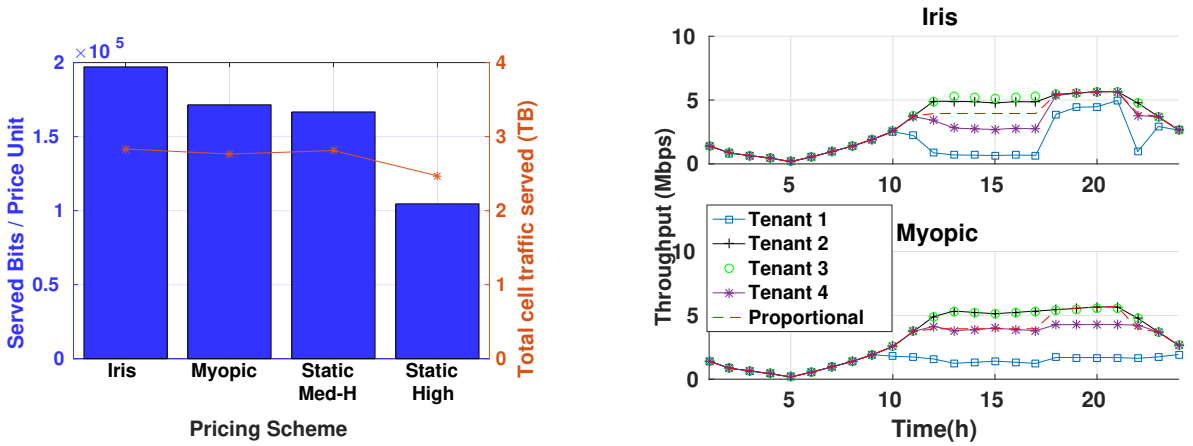
We begin by looking at the average dis-utility of the tenants for each pricing scheme and the total normalized revenue above the target level made by the neutral-host (Fig. 8a). The revenue is normalized by the maximum possible revenue of the neutral-host, i.e., selling all the available resources at the max price of p_{max}. In terms of the dis-utility, we can observe that Iris performs worse than DNRP and DRP as well as two of the lowest static pricing schemes (Static Low and Static Med-L). However, through the revenue results, we observe that for those four schemes the neutral-host experiences losses (negative profit), disincentivizing the neutral-host to provide its service in the first place. This could have been avoided if the pricing policy dynamically adapted not only based on the utilization of the resources, but also based on the revenue target set by the neutral-host.

The results are opposite for the myopic and the higher static pricing schemes (Med-H and High)

in that with these schemes the neutral-host obtains a revenue that is higher than the set target at the expense of a higher tenant dis-utility compared to Iris. For the static pricing schemes, this is due to the inability of the pricing mechanism to adapt to the traffic loads, charging high prices even at times of no congestion (e.g., 1am-10am when the traffic load is low or 6pm-9pm when there is abundance of spectrum). For the myopic scheme, however, this is due to the neutral-host agent forcing tenants to buy resources not needed to myopically recover its spectrum acquisition cost within each epoch. These behaviors are better seen in the hourly breakdown of the tenants' dis-utilities and corresponding prices decided by the neutral-host as shown in Fig. 8b. The Iris dynamic pricing mechanism manages to draw a balance between the needs of the tenants and the neutral-host more effectively, learning the right pricing policy that keeps the tenants as satisfied as possible, but without incurring a low revenue that would disincentivize the neutral-host.

In terms of the offered service, we measure the total traffic served by a cell throughout the day and calculate the average bits per price unit that the tenants bought for each pricing scheme. The results are in Fig. 9a. We omit DNRP, DRP and the static Low and Med-L schemes, given the losses they incur to the neutral-host. As we can observe, Iris offers the cheapest service, benefiting from the adaptiveness of its pricing scheme. Note that, although the same adaptiveness is also offered by the myopic scheme, the fact that tenants might be forced to buy unwanted resources raises the overall service cost. Another interesting observation is that, the total traffic served in the High static pricing scheme is significantly lower than that of Iris. This is because, due to the high prices, the tenants avoid buying radio resources despite the availability (evident from the fact that the other pricing schemes served more traffic with the same overall amount of resources).

Finally, we compare the service differentiation offered by Iris against the myopic scheme and the spectrum allocation policy proposed in [37]. The latter allocates RBs to the tenants proportionally to their load, so it can be viewed as a purely load dependent but pricing agnostic scheme. For this result, the myopic scheme can act as a baseline, since the neutral-host is aware of the dis-utility functions of the tenants and thus optimally distributes the resources among them. The results appear in Fig. 9b. As we can observe, Iris provides service differentiation among tenants, with results that are close to that of the myopic scheme. For the proportional scheme, no differentiation can be achieved (since every tenant generates the same traffic load). This can have a negative impact to the tenants' satisfaction, since the tenants that value the available spectrum the most end up getting less resources than they would like during hours of congestion (e.g., 12-6pm).



(a) Total traffic served and bits per price unit.

(b) Iris and myopic scheme service differentiation.

Fig. 9: Comparison of Iris with alternative approaches.

VI. Discussion and Future Directions

We believe that our work opens up a number of interesting research opportunities, which we discuss here.

Strategic tenants. In the current work it is assumed that tenants present a behavior that is invariant to the choices of other tenants and to their capability of affecting the price announced by the neutral-host through their actions. However, it is natural to expect that in many cases tenants could also develop strategic behavior, e.g. use their own learning agents, making resource requests that optimize their long-term benefits given the prices announced by the neutral-host. In such scenarios, we no longer have time-invariant transition rates from the point of view of any one agent (neutral-host or tenant), which can make the problem of solving the model much harder. One way to overcome this challenge could be to consider the problem in the context of a multi-agent reinforcement learning framework like [84]. Another approach could be to restrict the way that tenants behave and request resources, by enforcing the use of a mechanism that prohibits strategic behavior. Such a mechanism could for example restrict the frequency with which tenants can change their policy or to employ a domain specific language through which the Iris tenant agents could express their business models and demands in a constrained way.

Tenant tradeoffs driving use of the neutral-host deployment. As mentioned at the outset, traditional operators have the option to serve users in indoor spaces either by using their own outdoor RAN infrastructure or by using the indoor neutral-host deployment after paying some fee. Due to these options that operators have, a complementary problem to the one considered here

is how operators should decide whether it is preferable to use the neutral-host's infrastructure or to rely on their own. This decision could involve aspects like the level of the fee charged by the neutral-host, the number of users that would benefit from the presence of the operator in the indoor space, as well as the performance improvement that the users would experience through that.

Co-existence of multiple neutral-hosts in other settings. As already explained, in the setting of this work only a single neutral-host is expected to exist (e.g. due to indoor space constraints and regulations), with its main incentive for providing its service being the improvement of the quality of experience of residents and visitors. However, when considering settings where multiple neutral-hosts could be co-located (e.g. outdoor settings) the goals of the system could change. In such settings, attracting more users and maximizing profit would also be equally significant for the neutral-host in addition to the ones like the regulation of spectrum among tenants considered in this paper. In such scenarios an alternative framework would be required (e.g. a game-theoretic framework) to drive the behavior of each individual neutral-host, considering the actions of the other neutral-hosts.

Spectrum management related issues. One obvious extension of Iris is to expand its scope to also support pooled licensed as well as unlicensed spectrum. While our system design and dynamic pricing mechanism would still form the core solution in both of these cases, modifications would also be required due to the idiosyncrasies that these scenarios present. For pooled licensed spectrum, pricing needs to additionally account for revenue sharing with MNOs contributing licensed spectrum to the pool. On the other hand, in the case of unlicensed spectrum, coexistence issues with other technologies like Wi-Fi need to be addressed (e.g. using a technology like MulteFire [61]).

The dynamics of the interaction between the spectrum manager of Iris and the external repositories for shared spectrum acquisition is another relevant topic. Deciding on the amount of spectrum to request from an external repository can be a challenging problem for the neutral-host, due to the different loads and demands presented by different small-cells, which create a requirement to draw a balance between the spectrum acquisition cost and the satisfaction of the tenants' demands.

Multi-RAT support. Another interesting research topic is providing support for Iris in multi-RAT settings. Accommodating multiple disparate radio access technologies (e.g., 5G New Radio, LTE and Wi-Fi) as part of the same neutral-host system architecture is an approach in line with the 5G vision of native multi-access with an access agnostic core network architecture. However multi-RAT support presents its own set of challenges, with the main problem being on how tenants should decide which of the available technologies to use to accommodate the needs of their users, considering that each technology presents its own pros and cons in terms of performance, cost, capacity etc.

VII. Conclusions

We have presented Iris, a system architecture for neutral-host indoor small-cells based on shared spectrum. The design of Iris follows a C-RAN approach that allows scalable and efficient use of resources in the edge cloud while enable denser and cheaper small-cell radio infrastructure indoors. At the core of Iris lies a novel dynamic pricing radio resource allocation mechanism for shared spectrum. This mechanism employs deep reinforcement learning to discover pricing policies that allow tenants to request shared spectrum resources on demand, ensuring the differentiation of their services based on their valuation of the spectrum, while meeting the revenue target of the neutral-host that includes recouping the costs for shared spectrum acquistion. Using our prototype implementation of Iris developed for LTE, we have conducted extensive experimental evaluations to characterize the dynamic pricing mechanism of Iris under different conditions, show the benefits of the Iris approach compared to alternative approaches and examine its deployment feasibility.

Acknowledgment

The authors would like to thank Prof. George D. Stamoulis and Dr. Stefano V. Albrecht for their very helpful suggestions on improving this work.

References

[1] Cisco, "Vision 5G: Thriving Indoors," Feb 2017.

[2] Analysys Mason, "Small Cell Indoor Coverage Solutions," March 2016.

[3] Viavi Solutions, "DAS Deployment Overview: Streamlined approaches for DAS deployments," March 2017.

[4] Wireless 20—20, "Multi-Carrier Small Cell Solutions for In-Building Wireless," February 2017.

[5] J. Zander and P. Mähönen, "Riding the data tsunami in the cloud: myths and challenges in future wireless access," *IEEE Communications Magazine*, vol. 51, no. 3, pp. 145–151, 2013.

[6] 5G Americas, "Multi-operator and neutral host small cells: Drivers, architecture, planning and regulation," Dec 2016.

[7] "Network sharing makes sense in-building, not outdoors, says AT&T," https://enterpriseiotinsights.com/20180523/channels/news/network-sharing-in-building-tag40, accessed: 2018-08-02.

[8] "Small Cell "Neutral Hosting" is it the future?" https://tinyurl.com/ya6g2fpy, accessed: 2018-08-02.

[9] "Dense air," http://denseair.net/, accessed: 2018-08-02.

[10] "ip.access Viper," https://tinyurl.com/ydabffvc, accessed: 2018-08-02.

[11] "Baicells NeutralCell," http://www.baicells.com/neutralcell.html, accessed: 2018-08-02.

[12] "Crown Castle Indoor Small Cells," https://tinyurl.com/y93vl78s, accessed: 2018-08-02.

[13] I. Giannoulakis *et al.*, "The emergence of operator-neutral small cells as a strong case for cloud computing at the mobile edge," *Transactions on Emerging Telecommunications Technologies*, vol. 27, no. 9, pp. 1152–1159, 2016.

[14] M. Matinmikko *et al.*, "Micro Operators to Boost Local Service Delivery in 5G," *Wireless Personal Communications*, vol. 95, no. 1, pp. 69–82, 2017.

[15] P. Ahokangas *et al.*, "Future micro operators business models in 5G," *The Business & Management Review*, vol. 7, no. 5, p. 143, 2016.

[16] ——, "Business Models for Local 5G Micro Operators," in *IEEE International Symposium on Dynamic Spectrum Access Networks (DYSPAN)*. IEEE, 2018.

[17] B. Nguyen *et al.*, "ECHO: A reliable distributed cellular core network for hyper-scale public clouds ," in *Proceedings of the 24th ACM MobiCom*. ACM, 2018.

[18] ATIS, "Neutral Host Solutions for Multi-Operator Wireless Coverage in Managed Spaces," September 2016.

[19] "Bringing the Sharing Economy to the Airwaves Will Boost Your Bandwidth," https://tinyurl.com/y9vsvsue, accessed: 2018-08-02.

[20] FCC, "FCC Rule Making on 3.5 GHz Band / Citizens Broadband Radio Service," April 2015.

[21] Ofcom, "3.8 GHz to 4.2 GHz Band: Opportunities for Innovation," April 2016.

[22] Mobile Experts, "CBRS: New Shared Spectrum Enables Flexible Indoor and Outdoor Mobile Solutions and New Business Models," March 2017.

[23] M. Matinmikko *et al.*, "Spectrum sharing using licensed shared access: the concept and its workflow for LTE-advanced networks," *IEEE Wireless Communications*, vol. 21, no. 2, pp. 72–79, 2014.

[24] M. Matinmikko-Blue *et al.*, "Analysis of Spectrum Valuation Approaches: The Viewpoint of Local 5G Networks in Shared Spectrum Bands," in *IEEE International Symposium on Dynamic Spectrum Access Networks (DYSPAN)*. IEEE, 2018.

[25] F. P. Kelly *et al.*, "Rate control for communication networks: shadow prices, proportional fairness and stability," *Journal of the Operational Research society*, vol. 49, no. 3, pp. 237–252, 1998.

[26] S. Sen *et al.*, "Smart data pricing: using economics to manage network congestion," *Communications of the ACM*, vol. 58, no. 12, pp. 86–93, 2015.

[27] N. Li *et al.*, "Optimal demand response based on utility maximization in power networks," in *Power and Energy Society General Meeting, 2011 IEEE*. IEEE, 2011, pp. 1–8.

[28] Z. Liu *et al.*, "Pricing data center demand response," *ACM SIGMETRICS Performance Evaluation Review*, vol. 42, no. 1, pp. 111–123, 2014.

[29] I. Chih-Lin *et al.*, "Recent progress on C-RAN centralization and cloudification," *IEEE Access*, vol. 2, pp. 1030–1039, 2014.

[30] V. Sciancalepore *et al.*, "Mobile Traffic Forecasting for Maximizing 5G Network Slicing Resource Utilization," *IEEE INFOCOM*, 2017.

[31] P. Caballero *et al.*, "Multi-Tenant Radio Access Network Slicing: Statistical Multiplexing of Spatial Loads," *IEEE/ACM Transactions on Networking*, vol. 25, no. 5, pp. 3044–3058, 2017.

[32] D. Bega *et al.*, "Optimising 5G infrastructure markets: The business of network slicing," in *INFOCOM 2017*. IEEE, 2017, pp. 1–9.

[33] R. Kokku *et al.*, "NVS: A Substrate for Virtualizing Wireless Resources in Cellular Networks," *IEEE/ACM Transactions on Networking (TON)*, vol. 20, no. 5, pp. 1333–1346, 2012.

[34] ——, "CellSlice: Cellular Wireless Resource Slicing for Active RAN Sharing," in *2013 Fifth International Conference on Communication Systems and Networks (COMSNETS)*. IEEE, 2013, pp. 1–10.

[35] M. Jiang *et al.*, "Network slicing management & prioritization in 5G mobile systems," in *Proceedings of 22th European Wireless Conference*. VDE, 2016, pp. 1–6.

[36] M. R. Crippa *et al.*, "Resource Sharing for a 5G Multi-tenant and Multi-service Architecture," in *Proceedings of 23th European Wireless Conference*. VDE, 2017, pp. 1–6.

[37] M. G. Kibria *et al.*, "Shared spectrum access communications: A neutral host micro operator approach," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 8, pp. 1741–1753, 2017.

[38] J. O. Fajardo *et al.*, "Introducing mobile edge computing capabilities through distributed 5G cloud enabled small cells," *Mobile networks and applications*, vol. 21, no. 4, pp. 564–574, 2016.

[39] I. P. Chochliouros *et al.*, "A Novel Architectural Concept for Enhanced 5G Network Facilities," in *MATEC Web of Conferences*, vol. 125. EDP Sciences, 2017, p. 03012.

[40] X. Foukas *et al.*, "Orion: RAN Slicing for a Flexible and Cost-Effective Multi-Service Mobile Network Architecture," in *Proceedings of the 23rd ACM MobiCom*. ACM, 2017, pp. 127–140.

[41] S. H. Low and D. E. Lapsley, "Optimization flow control. I. Basic algorithm and convergence," *IEEE/ACM Transactions on networking*, vol. 7, no. 6, pp. 861–874, 1999.

[42] S. Ha *et al.*, "Tube: Time-dependent pricing for mobile data," *ACM SIGCOMM Computer Communication Review*, vol. 42, no. 4, pp. 247–258, 2012.

[43] M. G. Kibria *et al.*, "Resource allocation in shared spectrum access communications for operators with diverse service requirements," *EURASIP Journal on Advances in Signal Processing*, vol. 2016, no. 1, p. 83, 2016.

[44] ——, "Heterogeneous networks in shared spectrum access communications," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 1, pp. 145–158, 2017.

[45] J. Luo *et al.*, "Multi-carrier waveform based flexible inter-operator spectrum sharing for 5G systems," in *IEEE International Symposium on Dynamic Spectrum Access Networks (DYSPAN)*. IEEE, 2014, pp. 449–457.

[46] T. Sanguanpuak *et al.*, "On spectrum sharing among micro-operators in 5g," in *2017 European Conference on Networks and Communications (EuCNC)*. IEEE, 2017, pp. 1–6.

[47] B. Singh *et al.*, "Co-primary inter-operator spectrum sharing over a limited spectrum pool using repeated games," in *IEEE International Conference on Communications (ICC)*. IEEE, 2015, pp. 1494–1499.

[48] ——, "Repeated spectrum sharing games in multi-operator heterogeneous networks," in *2015 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*. IEEE, 2015, pp. 221–228.

[49] C. Hasan and M. K. Marina, "Communication-free inter-operator interference management in shared spectrum small cell networks," in *IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN), 2018.* IEEE, 2018.

[50] X. Zhou *et al.*, "eBay in the sky: Strategy-proof wireless spectrum auctions," in *Proceedings of the 14th ACM MobiCom*. ACM, 2008, pp. 2–13.

[51] F. Fu and U. C. Kozat, "Stochastic game for wireless network virtualization," *IEEE/ACM Transactions on Networking (ToN)*, vol. 21, no. 1, pp. 84–97, 2013.

[52] K. Zhu and E. Hossain, "Virtualization of 5G cellular networks as a hierarchical combinatorial auction," *IEEE Transactions on Mobile Computing*, vol. 15, no. 10, pp. 2640–2654, 2016.

[53] X. Feng *et al.*, "FlexAuc: Serving dynamic demands in a spectrum trading market with flexible auction," *IEEE Transactions on Wireless Communications*, vol. 14, no. 2, pp. 821–830, 2015.

[54] T. Le *et al.*, "On a new incentive and market based framework for multi-tier shared spectrum access systems," in *2014 IEEE International Symposium on Dynamic Spectrum Access Networks (DYSPAN)*. IEEE, 2014, pp. 477–488.

[55] S. Sengupta and M. Chatterjee, "Designing auction mechanisms for dynamic spectrum access," *Mobile Networks and Applications*, vol. 13, no. 5, pp. 498–515, 2008.

[56] S. Gandhi *et al.*, "Towards real-time dynamic spectrum auctions," *Computer Networks*, vol. 52, no. 4, pp. 879–897, 2008.

[57] J. Jia *et al.*, "Revenue generation for truthful spectrum auction in dynamic spectrum access," in *Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing*. ACM, 2009, pp. 3–12.

[58] L. Gao *et al.*, "An integrated contract and auction design for secondary spectrum trading," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 3, pp. 581–592, 2013.

[59] W. Dong *et al.*, "Double auctions for dynamic spectrum allocation," *IEEE/ACM Transactions on Networking*, vol. 24, no. 4, pp. 2485–2497, 2016.

[60] Small Cell Forum, "nFAPI and FAPI specifications," May 2017.

[61] MulteFire Alliance, "MulteFire release 1.0 technical paper: A new way to wireless," 2017.

[62] G. Salami *et al.*, "LTE indoor small cell capacity and coverage comparison," in *IEEE 24th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC Workshops)*. IEEE, 2013, pp. 66–70.

[63] M. L. Puterman, *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

[64] C. J. Watkins and P. Dayan, "Q-learning," *Machine learning*, vol. 8, no. 3-4, pp. 279–292, 1992.

[65] L. Busoniu *et al.*, *Reinforcement learning and dynamic programming using function approximators*. CRC press, 2010, vol. 39.

[66] D. Silver *et al.*, "Deterministic policy gradient algorithms," in *ICML*, 2014.

[67] T. P. Lillicrap *et al.*, "Continuous control with deep reinforcement learning," in *ICLR 2016*, 2016.

[68] J. Schulman *et al.*, "High-dimensional continuous control using generalized advantage estimation," *ICLR 2016*, 2016.

[69] Y. Duan *et al.*, "Benchmarking deep reinforcement learning for continuous control," in *International Conference on Machine Learning*, 2016, pp. 1329–1338.

[70] S. Gu *et al.*, "Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 3389–3396.

[71] Y. Tassa *et al.*, "Deepmind control suite," *arXiv preprint arXiv:1801.00690*, 2018.

[72] R. Irmer *et al.*, "Coordinated multipoint: Concepts, performance, and field trial results," *IEEE Communications Magazine*, vol. 49, no. 2, pp. 102–111, 2011.

[73] N. Nikaein *et al.*, "OpenAirInterface: A flexible platform for 5G research," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 5, pp. 33–38, 2014.

[74] N. Makris *et al.*, "Experimental evaluation of functional splits for 5G Cloud-RANs," in *2017 IEEE International Conference on Communications (ICC)*. IEEE, 2017, pp. 1–6.

[75] C.-Y. Chang *et al.*, "FlexCRAN: A flexible functional split framework over ethernet fronthaul in Cloud-RAN," in *2017 IEEE International Conference on Communications (ICC)*. IEEE, 2017, pp. 1–7.

[76] X. Foukas *et al.*, "Experience Building a Prototype 5G Testbed," in *Proceedings of the 1st International Workshop on Experimentation and Measurements in 5G (EM-5G)*. ACM, 2018.

[77] "DDPG implementation," https://github.com/stevenpjg/ddpg-aigym, 2018.

[78] G. Brockman *et al.*, "OpenAI Gym," 2016.

[79] https://developers.google.com/protocol-buffers/.

[80] http://zeromq.org/.

[81] A. Botta *et al.*, "A tool for the generation of realistic network workload for emerging networking scenarios," *Computer Networks*, vol. 56, no. 15, pp. 3531–3547, 2012.

[82] H. Wang *et al.*, "Understanding mobile traffic patterns of large scale cellular towers in urban environment," in *Proceedings of the 2015 Internet Measurement Conference*. ACM, 2015, pp. 225–238.

[83] T. Degris *et al.*, "Model-free reinforcement learning with continuous action in practice," in *American Control Conference (ACC), 2012*. IEEE, 2012, pp. 2177–2182.

[84] R. Lowe *et al.*, "Multi-agent actor-critic for mixed cooperative-competitive environments," in *Advances in Neural Information Processing Systems*, 2017, pp. 6382–6393.

APPENDIX

MODELLING THE EFFECT OF TRAFFIC LOAD AND UNIT PRICE ON THE BEHAVIOR OF THE NEUTRAL-HOST TENANTS

A. Preliminaries

Let $d \geq 0$ be the traffic load of the tenant and let $p \geq 0$ be the price charged by the neutral-host per resource block. Suppose that the tenant requests $b \geq 0$ resource blocks from the neutral-host. This decision has a cost equal to pb and creates a backlog of traffic equal to $[d-b]^+$ (using the standard notation $[x]^+ \triangleq \max\{x,\ 0\}$). Obviously, higher values of b reduce (or eliminate) the backlog, but also increase the cost.

To express this trade-off, we consider dis-utility functions $\bar{U} : \mathbb{R}_0^+ \to \mathbb{R}_0^+$ of the form

$$\bar{U}(b; d, p) = \Big(a\big([d-b]^+\big)^{\gamma_d} + \big(pb\big)^{\gamma_p}\Big)^{1/\gamma_p}, \qquad a > 0, \quad \gamma_d, \gamma_p \geq 1. \tag{11}$$

The exponents γ_d and γ_p in (11) tune the sharpness of the dissatisfaction associated with the backlog of traffic and with the cost of the resources respectively, while the factor a expresses the relative importance of these two sources of dissatisfaction in the overall dis-utility. In the following, we will investigate the use of the dis-utility function for determining the optimal resource allocation request $b^*(d,p) \triangleq \arg\min_{b \geq 0} \bar{U}(b; d, p)$, under a given price and traffic load.

With respect to the units of the various quantities involved, and apart from the two exponents γ_d and γ_p, which are dimensionless, d and b are expressed in units of $[\text{resource block}]$. The price p is in units of $[\text{cost}]/[\text{resource block}]$, while the factor a is in units of $[\text{cost}]^{\gamma_p}/[\text{resource block}]^{\gamma_d}$. Finally, since the parenthesized sum in (11) is raised to the power of $1/\gamma_p$, the values of the dis-utility function $\bar{U}$ are expressed in units of $[\text{cost}]$.

This last feature is worthwhile because, apart from the use of the dis-utility function for determining b^*, the function is useful also for quantifying the dis-utility experienced by multiple tenants, whose dis-utility functions may employ different values for the parameters γ_d, γ_p and a. Regardless of such differences, all dis-utilities will be expressible in the same unit of $[\text{cost}]$, bearing the same interpretation for all tenants and being directly comparable. Additionally, it is possible to introduce a notion of 'overall dis-utility', calculated as the sum of dis-utilities over all tenants.

B. Structural properties

Since $(\cdot)^{\gamma_p}$ is strictly increasing, the minima of $\bar{U}(\cdot\,; d, p)$ and of $U(\cdot\,; d, p) \triangleq \bar{U}(\cdot; d, p)^{\gamma_p}$, coincide. Thus, we may work with the simpler function U, equal to the parenthesized sum in (11).

By construction, U is continuous and convex (because $\gamma_d, \gamma_p \geq 1$) throughout its domain. In fact, U is strictly convex (and its derivative strictly increasing), except when $\gamma_d = \gamma_p = 1$, in which case it is piecewise linear.

In view of (11), U is continuously differentiable in $[0, d) \cup (d, +\infty)$, with

$$U'(b; d, p) = \begin{cases} -a\gamma_d (d-b)^{\gamma_d - 1} + \gamma_p p^{\gamma_p} b^{\gamma_p - 1}, & 0 \leq b < d, \\ \gamma_p p^{\gamma_p} b^{\gamma_p - 1}, & b > d. \end{cases} \tag{12}$$

The derivative in (12) is continuous also at $b = d$ and $U'(d)$ exists unless $\gamma_d = 1$, in which case $U'(d^-) < U'(d^+)$.

By the second branch in (12), U is increasing for $b > d$, so the infimum of the function occurs within the closed and bounded interval $[0, d]$ and, by continuity, there exists a minimal point $b^* \in [0, d]$. Furthermore, by convexity, this minimal point is unique (except perhaps for the non-strictly convex case $\gamma_d = \gamma_p = 1$ when the minimum may be attained for all points of an interval within $[0, d]$).

C. The optimal resource allocation request $b^*(d, p)$

We now express b^* as a function of the given traffic load and price. As we will see, the shape of this function depends on the values of the exponents γ_d and γ_p.

1) The case $\gamma_d = \gamma_p = 1$ – extreme behavior: By (12), U is decreasing throughout $[0, d]$ when $p < a$, increasing when $p > a$, and constant when $p = a$. Thus,

$$b^*(d, p) = \begin{cases} d, & p < a, \\ \text{any } b \in [0, d], & p = a, \\ 0, & p > a. \end{cases} \tag{13}$$

The form (13) signifies "extreme" behavior. The factor a fixes a price threshold and the tenant either makes a resource request equal to the traffic load when the price is below the threshold, or backlogs all its traffic when the price exceeds the threshold.

2) The case $\gamma_d = 1$, $\gamma_p > 1$ – cost saving tendency; limiting allocations below a price-dependent threshold: By (12), if $U'(d^-) \leq 0$, i.e., if $a \geq \gamma_p p^{\gamma_p} d^{\gamma_p - 1}$ then U is decreasing throughout $[0, d]$ and $b^* = d$. Otherwise, the minimization occurs at the unique solution of $U'(b) = 0$. Putting these facts together,

$$b^*(d, p) = \min \left\{ \left(\frac{a}{\gamma_p p^{\gamma_p}} \right)^{\frac{1}{\gamma_p - 1}}, \; d \right\}. \tag{14}$$

This result may be interpreted as follows: Fixing a price determines a traffic load threshold d_0, equal to the first argument of the $\min$-operator within (14). For traffic loads lower than this threshold the tenant requests resource blocks equal to the load; for higher ones, resource blocks for a load of d_0 are requested and the remaining part of the traffic is backlogged.

An alternative interpretation may also be given, by first fixing the traffic load d and then varying p. Along this interpretation, as prices increase, starting from an initially low level close to 0, the traffic is fully covered. This occurs up to a load-dependent price threshold (obtained by equating the two arguments of the $\min$-operator in (14) and then solving for p), while higher prices introduce backlog. In this regime, the processed part of the traffic gradually diminishes as $p \to \infty$.

It may be seen that as $\gamma_p \downarrow 1$, the case considered in this section tends to the extreme case in Section C1 and (14) collapses to (13).

3) The case $\gamma_d > 1$, $\gamma_p = 1$ – limiting backlogs below a price-dependent threshold: In view of (12), when $U'(0) \geq 0$, i.e., when $p \geq a\gamma_d d^{\gamma_d - 1}$, the function U is increasing throughout $[0, d]$ and $b^* = 0$. Otherwise, the minimization occurs at the unique solution of $U'(b) = 0$. Overall,

$$b^*(d, p) = \left[d - \left(\frac{p}{a\gamma_d} \right)^{\frac{1}{\gamma_d - 1}} \right]^+ . \tag{15}$$

According to this result, fixing a price again determines a traffic load threshold d_0, now equal to the term subtracted from d in (15). For traffic loads higher than the threshold the tenant requests resource blocks that will only partially cover its load, creating a backlog of remaining traffic equal to d_0. Loads lower than the threshold are entirely backlogged. This behavior tends to favor higher traffic loads over lower ones.

For the alternative interpretation related to fixed traffic loads and varying prices, it may be seen that as prices increase, again starting from an initially low level close to 0, the unprocessed part of the traffic gradually increases. Past a load-dependent price (obtained by equating the two terms subtracted in (15) and solving for p), the entire traffic load is backlogged.

Again, it may be seen that as $\gamma_d \downarrow 1$, the case considered here tends to the extreme case in Section C1 and (15) collapses to (13).

4) The case $\gamma_d > 1$, $\gamma_p > 1$ and the balanced sub-case $\gamma_d = \gamma_p = \gamma > 1$: Now, $U'(0) < 0$, and $U'(d) > 0$, so b^* lies in the interior of $[0, d]$ and is determined as the unique solution of $U'(b) = 0$, equivalently the unique solution of the non-linear equation (in b)

$$\left(\frac{\gamma_p p^{\gamma_p}}{\gamma_d a} \right)^{\frac{1}{\gamma_d - 1}} b^{\frac{\gamma_p - 1}{\gamma_d - 1}} + b = d. \tag{16}$$

Generally, this equation must be solved numerically, but for $\gamma_d - 1 = 2(\gamma_p - 1)$ (closer in spirit to the case of Section C3) and for $\gamma_d - 1 = (\gamma_p - 1)/2$ (closer in spirit to the case of Section C2) the equation reduces to a quadratic leading to a closed form solution.

Again, it may be seen that by keeping γ_p fixed and letting $\gamma_d \downarrow 1$ the case in this section tends to the case in Section C2. Similarly, by keeping γ_d fixed and letting $\gamma_p \downarrow 1$, this case reduces to the case of Section C3. Finally, letting both exponents tend to unity leads to the case in Section C1.

When the exponents are equal and greater than unity, (16) collapses to a first order linear equation with solution

$$b^*(d,p) = \frac{d}{1 + (p^\gamma/a)^{\frac{1}{\gamma-1}}}. \tag{17}$$

It is seen that all levels of traffic loads are treated uniformly, with the tenant requesting resource blocks that will neither fully cover nor entirely backlog the existing traffic load. Instead, the price determines the fraction of the traffic to be processed. As $\gamma \downarrow 1$ this solution adopts the sharp characteristics of the case in Section C1.

D. Parametrizing the dis-utility function

Here we consider values for the parameters of the dis-utility function $\bar{U}$, to tailor the function to a particular tenant's behavior and create the tenant profiles presented in Table I of Section V-A. We address the cases in Sections C2, C3 and C4. In each of these, one needs to determine values for an exponent and for the factor a. The value of the exponent is chosen first, to determine the "sharpness" of the function's response. Then, to determine the factor a one proceeds as follows:

- For the "cost saving" case in Section C2, one specifies a traffic load threshold d_0 and a price threshold p_0. These determine the value of a, so that for prices and loads not greater than the thresholds the traffic load of a tenant is fully processed. By equating the two arguments of the $\min$-operator in (14), with the price and load therein set equal to the thresholds, the appropriate value of the factor is seen to be

$$a = \gamma_p p_0^{\gamma_p} d_0^{\gamma_p - 1}. \tag{18}$$

  Profiles 1 and 2 of Table I were based on this "cost saving" case, with $\gamma_p = 2$ (and $\gamma_d = 1$). For the first profile, a traffic load threshold $d_0 = 1750$ and a price threshold $p_0 = 100$ were used, leading to a value of $a = 3.5 \times 10^8$ and a "best effort" profile type, in which the tenant is willing to fully cover its load for very low prices (recall that $p_{\max} = 2500 \gg p_0$) and only a

small part of the load otherwise (just to maintain network presence). For the second profile, a traffic load threshold $d_0 = 1000$ and a price threshold $p_0 = 1000$ were used ($a = 2 \times 10^9$). This leads to a "price-driven" profile that is similar to the first one, with the difference that the tenant is willing to buy more resources for low to medium prices compared to the best-effort case (not just focusing on maintaining network presence).

- For the "bounded backlog" case in Section C3, one again specifies a traffic load threshold d_0 and a price threshold p_0. These determine the value of a, so that for prices greater than p_0 and loads lower than d_0 the traffic is backlogged entirely. By equating the two terms subtracted in (15), with the price and load therein set equal to the thresholds, the appropriate value of the factor is seen to be

$$a = \frac{p_0}{\gamma_d d_0^{\gamma_d - 1}}. \tag{19}$$

  The third profile of Table I was based on this analysis, with $\gamma_d = 2$ (and $\gamma_p = 1$). The traffic load threshold was set to $d_0 = 6000$ and the price threshold to $p_0 = 2436$ (i.e., close to $p_{\max}$), leading to a value of $a = 0.203$. This profile corresponds to a "demand-driven" tenant, who is willing to buy large amounts of resources regardless of the price when the traffic load is high and to queue its traffic until the load increases enough to buy in bulk in other times.

- Finally, for the balanced sub-case in Section C4, one must specify a price threshold p_0 and the corresponding fraction $\omega_0 \in (0, 1)$ of the traffic load that will be processed. Then, in view of (17), the appropriate value of the factor is seen to be

$$a = p_0^{\gamma} \Big(\frac{\omega_0}{1 - \omega_0} \Big)^{\gamma - 1}. \tag{20}$$

  Correspondingly, the last profile of Table I was set to represent a "medium" QoS level type of tenant with $\gamma = 2$, $p_0 = 600$ and $\omega_0 \approx 0.25$, yielding $a = 1.1 \times 10^5$.