Joint Spectrum Pricing and Admission Control for Heterogeneous Secondary Users

Changkun Jiang*, Lingjie Duan**, and Jianwei Huang*
*Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong
**Engineering Systems and Design Pillar, Singapore University of Technology and Design, Singapore
Email: *{jc012, jwhuang}@ie.cuhk.edu.hk, **lingjie_duan@sutd.edu.sg

***Abstract*—This paper solves the problem of long-term revenue maximization of a spectrum database operator, through joint pricing of spectrum resources and admission control of secondary users. A unique feature that we consider is the stochastic and heterogeneous nature of secondary users' demands. We formulate the problem as a stochastic dynamic programming problem, and consider the optimal solutions under both static and dynamic prices. In the case of static pricing, we constrain the prices to be time-independent while allowing the admission control policies to be time dependent. We show that in most cases a stationary (time independent) admission policy is in fact optimal in this case. We further look at the general case of dynamic pricing, where both the prices and admission control policies can be time dependent. We show that the flexibility of dynamic pricing can significantly improve the operator's revenue (by more than 30%) when secondary users have high demand elasticities.**

I. Introduction

Database-assisted spectrum sharing is a promising approach to improve the utilization of limited spectrum resources [1], [2]. In such an approach, primary licensed users (PUs) report their spectrum usage patterns to a third-party spectrum database, which coordinates the spectrum access of secondary unlicensed users (SUs). Many government regulators, such as FCC in the US and Ofcom in the UK, strongly advocate such an approach due to its high reliability (comparing with other approaches such as spectrum sensing). The wireless industry also embraces such an approach, and companies like Spectrum Bridge [3], Microsoft [4], and Google [5] are pioneers in the growing list of spectrum database operators. Though extensive research efforts have made significant progress in addressing various technical issues in terms of database system management and resource allocation [6]–[8], the study of spectrum database economics remains an under-explored research area. Without a proper economic mechanism, the database operator may not have enough incentives to serve the SUs. This motivates us to consider a database operator's long-term revenue maximization problem in this paper.

There are two key challenges when considering such a revenue maximization problem. First, SUs' demands are often heterogeneous in terms of spectrum occupancy. For example, a large file download may require several time slots to finish, while sending a short text message or location information in location based services can be completed in a single time slot. Instead of using the same unit price per time slot, the database may improve its revenue by charging different prices to different types of demands. Second, SUs request resources from the database at random time instances. As different types of demands may bring different amount of revenue to the database, it may not be optimal to serve the SUs' demands according to their order of arrivals. The database needs to perform proper admission control to reserve spectrum resources to serve the most profitable future SU demands.

To address the above two issues, we propose a joint spectrum pricing and admission control mechanism for the database operator to maximize its long-term revenue. As an initial study of this challenging research problem, here we focus on a stylized model, where the database operator optimizes the admission control and prices of multiple time slots of a single channel. The database operator needs to determine the optimal prices for different types of SU demand in each time slot. These prices can vary over time and will affect SUs' demands. When SUs randomly arrive and inform the database of their demands, the operator also needs to determine the optimal admission control policy to maximize the revenue.

Let us briefly summarize the related literature and contrast them with our approach. We can categorize the recent studies on pricing limited wireless resources into static and dynamic pricing schemes. Under static pricing, the pricing decisions may be simply flat rate or depend on usage, reservation, and priority, but do not depend on time (e.g., [7]–[14]). In contrast, many dynamic pricing works focus on revenue maximization over a finite time horizon, by optimally setting dynamic prices for a perishable product (e.g., [15], [16] and those on airline seat management, hotel room booking, and transportation networks). Other studies on dynamic pricing are summarized and surveyed in detail in [17]. Several recent studies have looked at dynamic pricing of wireless resources (e.g., [18]–[20]). Song *et al.* in [18] studied the network revenue maximization problem by using dynamic pricing in a wireless multi-hop network. Ha *et al.* in [19] proposed a new system architecture of time-dependent (dynamic) pricing, which increases wireless operators' revenue and decreases customers' cost. Ma *et al.* in [20] proposed time and location based pricing for mobile data traffic. Our approach is novel comparing to the prior literature, in the sense that none of the previous results considered admission control with dynamic pricing, or considered both stochastic demands *and* different consecutive spectrum occupancy requirements.

This work is supported by the General Research Funds (Project Number CUHK 412713 and CUHK 412511) established under the University Grant Committee of the Hong Kong S.A.R., China. This work is also supported by SUTD-MIT International Design Center Grant (Project No. IDSF1200106OH) and SUTD Start-up Research Grant (Project No. SRG ESD 2012 042).

Our main results and key contributions are summarized as follows.

- *Joint spectrum pricing and admission control scheme*. To our best knowledge, this is the first work that jointly prices and allocates the spectrum resource to serve heterogeneous and stochastic SU demands. We elaborate the model and formulate the problem as a stochastic dynamic programming problem in Sec. II.
- *Optimal static pricing and stationary admission policies*. In Sec. III, we constrain ourself to both static pricing and stationary admission control, and derive the optimal policy. The optimal policy has simple threshold structures, and remains to be optimal most of the time even if we allow dynamic admission decisions. These easy-to-use stationary policies are threshold-based, depending on SUs' demand elasticities.
- *Optimal dynamic pricing and dynamic admission*. In Sec. IV, we further allow the prices to be dynamic, which change based on different SUs' stochastic demands over time. Although the optimal prices and admission decisions are coupled, we are able to compute the optimal decisions through proper decomposition in each time slot.
- *Revenue improvement evaluation*. By comparing the static and dynamic pricing both under dynamic admission, we show that dynamic pricing can significantly improve the database operator's revenue (by more than 30%) when SUs have high demand elasticities.

Most of the proof details can be found in our online appendix [21].

II. System Model and Problem Formulation

We consider a database operator who knows which channel is not used by PUs and can be used to serve SUs. The database operator wants to maximize the revenue through selling spectrum opportunities in the secondary spectrum market. We consider a period of $\mathcal{N} = \{1, \cdots, N\}$ time slots during where a single channel will not be used by the PUs. The value of N depends on the type of PU traffic, and is known in advance as the PUs need to register all traffic with the database (e.g., [1], [2], [6]).

SUs randomly arrive and request channel access at the beginning of each time slot. We classify SUs into two types according to their required number of consecutive time slots to access the channel. A light-traffic SU only needs to use the channel for one time slot. For simplicity, we assume that a heavy-traffic SU needs to occupy two consecutive time slots. In [21], we also extend our analysis to the case where a heavy-traffic SU occupies more than two time slots.

Once an SU is admitted, the database operator will serve and charge the SU either $r_l(n)$ or $r_h(n)$, depending on whether it is a light- or heavy-traffic SU. We assume that SUs are price-sensitive, and the arrival rates of SUs are decreasing in the prices. As we focus on a single channel scenario, the database operator will admit at most one SU (light- or heavy-traffic) at a time. Once an SU arrives and requests resource from the database, it will leave the system immediately if such request is not accommodated. This is reasonable when SUs have delay-intolerant applications such as VoIP and video conferencing.

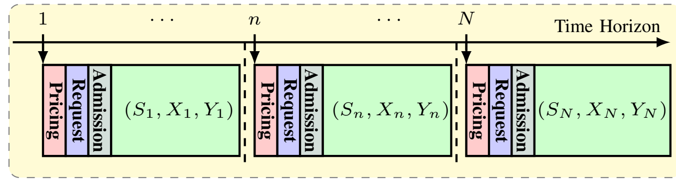

Fig. 1. Database operation: The blocks denote time slots available to serve SUs. At the beginning of each time slot, the database operator's pricing decision will affect the SUs' incoming requests and the revenue of the current time slot. After observing the requests, the database operator then makes admission decisions. (S_n, X_n, Y_n) denotes the resultant channel occupancy and two SU types' demand information (will be further explained in Sec. III).

Fig. 1 summarizes the database's operations in terms of the joint pricing and admission control. At the beginning of each time slot n, the database operator first announces two prices $r_l(n)$ and $r_h(n)$ for the light- and heavy-traffic SU types, then some SUs arrive and request to access the channel, and finally the database operator admits at most one SU and rejects the others. After these three phases, the admitted SU will transmit data over the channel during the rest of the time slot.[1]

To maximize its long-term average revenue, the database operator wants to optimize spectrum prices and admissions over all N time slots. In this optimization problem, the database operator's admission of a heavy-traffic SU in one time slot will prevent serving a light-traffic SU in the next time slot, hence the operation decisions over time are correlated. We will model the problem as a dynamic programming problem, and propose the optimal admission policies under static pricing in Sec. III and under dynamic pricing in Sec. IV.

III. Optimal Database Operation under Static Pricing and Dynamic Admission

Before studying the general case of optimal dynamic pricing problem, we will first consider the simplified case of static pricing, where prices do not change over time. Static pricing will reduce the signaling overhead, as we no longer need to update prices in every time slot as in Fig. 1. It will also serve as a benchmark and help us to quantify the performance benefit of dynamic pricing in Sec. IV. More precisely, with static pricing the database only needs to optimize and announce prices once in time slot 1, and keeps the prices fixed for the rest $N-1$ time slots, i.e., $r_l(n) = r_l$ and $r_h(n) = r_h$ for each time slot $n \in \{1, \cdots, N\}$.

We will formulate the revenue maximization problem as a stochastic dynamic programming problem. In Sec. III-A and Sec. III-B, we will solve the optimal admission control problem through backward induction from the last time slot, under any *fixed* prices. In Sec. III-C, we will optimize the static prices, considering the optimal admission policies derived in Sec. III-A and Sec. III-B.

[1] We assume that the signaling overhead is small in each time slot. This is especially true when we have a stationary admission policy as discussed in Sec. III.

A. Admission Control Optimization under Fixed Prices

In this subsection, we assume that the prices are fixed at r_l and r_h, and optimize the channel admission in each time slot. Such optimization needs to consider the channel availability and SU arrivals in the current time slot, as well as SU arrivals in future time slots. We will formulate and solve the problem as a stochastic dynamic programming problem.

We first define the system state as follows.

Definition 1 (System State): The system state in time slot n is (S_n, X_n, Y_n). Here S_n is the binary channel state, where $S_n = 0$ means that the (single) channel is available for admission in time slot n, and $S_n = 1$ otherwise. $X_n = 1$ means that at least one light-traffic SU arrives at the beginning of the time slot *and* is willing to use the channel and pay for the price r_l, and $X_n = 0$ otherwise. Y_n is defined similarly as X_n but for the heavy-traffic SU.

The system state evolves over time, depending on the channel admission decisions and SU arrivals. The feasible set of admission actions in each time slot depends on the current system state. Formally, we define the state-dependent admission action set as follows.

Definition 2 (Admission Action Set): The set of feasible admission actions in time slot n is a state-dependent set $\mathcal{A}_n(S_n, X_n, Y_n)$. When $S_n = 1$ such that the current time slot is not available for admission due to the admission of a heavy-traffic SU in the previous time slot, then $\mathcal{A}_n(1, X_n, Y_n) = \{0\}$ for all (X_n, Y_n). When $S_n = 0$, the action set depends on which type SU requests we have in the current time slot. If no SUs request in time slot n (i.e., $(X_n, Y_n) = (0, 0)$), the set of actions is $\mathcal{A}_n = \{0\}$. If both light- and heavy-traffic SUs demand, i.e., $(X_n, Y_n) = (1, 1)$, then we can either serve no SU (denoted by 0), or a light-traffic SU (denoted by 1), or a heavy-traffic SU (denoted by 2), and thus the set of actions is $\{0, 1, 2\}$. The sets of actions with the other requests are similarly determined:

$$\mathcal{A}_n(0, X_n, Y_n) = \begin{cases} \{0\}, & \text{if } (X_n, Y_n) = (0, 0), \\ \{0, 1\}, & \text{if } (X_n, Y_n) = (1, 0), \\ \{0, 2\}, & \text{if } (X_n, Y_n) = (0, 1), \\ \{0, 1, 2\}, & \text{if } (X_n, Y_n) = (1, 1). \end{cases}$$

We further define the specific admission action at $t = n$ as $a_n(S_n, X_n, Y_n) \in \mathcal{A}_n(S_n, X_n, Y_n)$. Here, $a_n = 0$ means that we will not admit any SU, and $a_n = 1$ and $a_n = 2$ mean that we admit a light-traffic SU and a heavy-traffic SU, respectively.

When $S_n = 1$, we will not admit any SU, hence in the next time slot $S_{n+1} = 0 = S_n - 1$. When $S_n = 0$, the channel availability of the next time slot depends on the action a_n. If we admit the light-traffic SU with $a_n = 1$, then the channel is available in the next, i.e., $S_{n+1} = 0 = a_n - 1$. If we admit the heavy traffic SU with $a_n = 2$, then the channel is not available in the next time slot, i.e., $S_{n+1} = 1 = a_n - 1$. To summarize, we have the following state dynamics.

Lemma 1 (State Dynamics): The dynamics of the system state component S_n for any $n = 1, \cdots, N-1$ satisfies the following equation:

$$S_{n+1} = (S_n + a_n(1 - S_n) - 1)^+, \tag{1}$$

where $(x)^+ := \max\{0, x\}$, and $S_n \in \{0, 1\}$, for all $n \in \mathcal{N}$.

The system state components (X_n, Y_n) only depend on SU arrivals (and prices) in the current time slot, but do not depend on the action a_n in previous time slots. The key notations we introduced so far are listed in Table I.

TABLE I. KEY NOTATIONS AND PHYSICAL MEANING

Symbols	Physical Meaning
$\mathcal{N} = \{1, \cdots, N\}$	Time slot index of unused spectrum
(S_n, X_n, Y_n)	System state (occupancy, light, and heavy requests)
$a_n(S_n, X_n, Y_n)$	Admission action in time slot n
$\mathcal{A}_n(S_n, X_n, Y_n)$	Set of possible actions in time slot n
$R_n(S_n, X_n, Y_n)$	The total revenue from time slot n to N
$\pi^*(S_n, X_n, Y_n)$	Optimal admission policy (in time slot n)
$\mathbb{E}[R_n^*(S_n, X_n, Y_n)], \bar{R}_n^*(S_n)$	Optimal expected future revenue from n to N
$r_l(n), r_h(n)$	Prices for light- and heavy-traffic SUs
$p_l(r_l(n)), p_h(r_h(n))$	The probability of having at least one SU requesting for spectrum (in time slot n)
$V_n(r_l(n), r_h(n))$	Expected future revenue from n to N
k_l, k_h	Demand elasticities of SUs with prices change

We are now ready to introduce the revenue maximization problem. We define a policy $\boldsymbol{\pi} = \{a_n(S_n, X_n, Y_n), \forall n \in \mathcal{N}\}$ as the set of decision rules for states and time slots, and we let $\boldsymbol{\Pi} = \{\mathcal{A}_1(\cdot), \cdots, \mathcal{A}_N(\cdot)\}$ be the feasible set of $\boldsymbol{\pi}$. We define the total revenue from time slot n to N as $R_n(S_n, X_n, Y_n, a_n)$. Given $\boldsymbol{S} = \{S_1, \cdots, S_N\}$, $\boldsymbol{X} = \{X_1, \cdots, X_N\}$, $\boldsymbol{Y} = \{Y_1, \cdots, Y_N\}$, and the admission policies $\boldsymbol{\pi} \in \boldsymbol{\Pi}$, the database operator aims to find an optimal policy $\boldsymbol{\pi}^*$ that maximizes the expected total revenue from the entire N time slots:

P1: *Revenue Maximization by Dynamic Admission*

$$\begin{aligned} \text{Maximize} \quad & \mathbb{E}^{\boldsymbol{\pi}}_{\boldsymbol{X},\boldsymbol{Y}}[R_1(\boldsymbol{S}, \boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{\pi})] \\ \text{Subject to} \quad & a_n(S_n, X_n, Y_n) \in \mathcal{A}_n(S_n, X_n, Y_n), \forall n \in \mathcal{N}, \\ & S_{n+1} = (S_n + a_n(1 - S_n) - 1)^+, \\ & \qquad \forall n \in \{1, \cdots, N-1\}, \\ \text{Variables} \quad & \boldsymbol{\pi} = \{a_n(S_n, X_n, Y_n), \forall n \in \mathcal{N}\}, \end{aligned} \tag{2}$$

where $\mathbb{E}^{\boldsymbol{\pi}}_{\boldsymbol{X},\boldsymbol{Y}}[R_1(\cdot)]$ denotes the expected total revenue and the expectation taken over SUs' random requests $(\boldsymbol{X}, \boldsymbol{Y})$.

We proceed to analyze the problem **P1** by using backward induction [22]. At the beginning of n, the operator makes the admission action a_n to maximize the total revenue $R_n(S_n, X_n, Y_n, a_n)$. The total revenue has two parts, the *immediate revenue* $r(a_n)$ for the current admission action a_n, where $r(a_n) = 0$, r_l, or r_h if $a_n = 0$, 1, or 2, respectively; and the *expected future revenue* from $n+1$ to N, i.e., $\mathbb{E}[R_{n+1}(S_{n+1}, X_{n+1}, Y_{n+1})]$, where the expectation is taken over the SU requests in the next time slot $n+1$, i.e., (X_{n+1}, Y_{n+1}).[2] Then the problem for stage n in the backward induction is

$$R_n^*(S_n, X_n, Y_n) = \max_{a_n \in \mathcal{A}_n} R_n(S_n, X_n, Y_n, a_n), \tag{3}$$

where the revenue dynamic recursion is

$$R_n(S_n, X_n, Y_n, a_n) = r(a_n) + \mathbb{E}[R_{n+1}^*(S_{n+1}, X_{n+1}, Y_{n+1})]. \tag{4}$$

As a boundary condition, we set $R_{N+1}^*(\cdot) = 0$ for any $(S_{N+1}, X_{N+1}, Y_{N+1})$ and a_{N+1}. By using backward induction, we start with the final time slot N and derive the optimal

[2] In this paper, the expectation $\mathbb{E}[R_n(S_n, X_n, Y_n)]$ is always taken over SU requests $(X_n, Y_n), \forall n \in \mathcal{N}$, unless otherwise mentioned.

TABLE II. THREE STATIONARY ADMISSION POLICIES

Tables	(S_n, X_n, Y_n)			
	$(0,0,0)$	$(0,0,1)$	$(0,1,0)$	$(0,1,1)$
Tab.II–H: a_n^{H*}	0	2	1	2
Tab.II–M: a_n^{M*}	0	2	1	1
Tab.II–L: a_n^{L*}	0	0	1	1

decisions stage by stage. In time slot n, the admission decision is made by comparing the corresponding total revenues in terms of different admission actions from time slot n to N, i.e., $R_n(S_n, X_n, Y_n, a_n), a_n \in \mathcal{A}_n$, which can be computed iteratively based on different states and the prices by (3) and (4). We present the algorithm that computes the optimal admission policy in [21].

B. Stationary Admission Policies

The optimal admission control solution in general does not have a closed form characterization nor clear engineering insights. In the following, we will focus on a class of stationary admission policies, where the admission rules do not change over time. We will try to understand when these stationary policies will be optimal. Note that even with the stationary admission policy, the actual admission decision in different time slots may still be different as it depends on the system state.

Table II shows all three possible stationary policies. Recall that (S_n, X_n, Y_n) is the system state, and the three rows, Tab.II–H: a_n^{H*}, Tab.II–M: a_n^{M*}, and Tab.II–L: a_n^{L*}, correspond to the heavy-, mixed-, and light-traffic admission policies that we will introduce. We will derive the conditions of the static prices r_l and r_h, under which one-of-the-three stationary policies achieves the optimality of the problem **P1**. Notice that when $S_n = 1$, we have $a_n^* = 0$ for any values of X_n and Y_n. Thus Table II only focuses on the case of $S = 0$. We further write the optimal expected future revenue $\mathbb{E}[R_n^*(S_n, X_n, Y_n)]$ as $\tilde{R}_n^*(S_n)$, since the expectation is taken over (X_n, Y_n).

We first analyze the performance of the heavy-traffic admission policy (in Tab.II–H: $a_n^{H*}, \forall n \in \mathcal{N}$). Under this policy, we will serve a heavy-traffic SU whenever possible, and only serve a light-traffic SU if there is no heavy-traffic SU. Such a stationary policy is optimal if the following two conditions hold for each slot $n \in \{1, \cdots, N-1\}$,

$$r_h + \tilde{R}_{n+2}^*(0) \geq 0 + \tilde{R}_{n+1}^*(0), \tag{5}$$

$$r_h + \tilde{R}_{n+2}^*(0) \geq r_l + \tilde{R}_{n+1}^*(0). \tag{6}$$

Inequality (6) means that serving a heavy-traffic SU is better than serving a light-traffic SU in the current time slot followed by serving another potential light-traffic SU in the next time slot (with certain positive probability computed based on the system parameters). Since (6) implies (5), we only need to consider (6).

Similarly, we can derive the condition under which the mixed-traffic admission policy (in Tab.II–M: a_n^{M*}) is optimal, i.e., $0 + \tilde{R}_{n+1}^*(0) < r_h + \tilde{R}_{n+2}^*(0) < r_l + \tilde{R}_{n+1}^*(0)$ for all n. Under this policy, when both types of SUs are available, we will always serve the light-traffic SU. Finally, we can derive the condition under which the light-traffic admission policy (in Tab.II–L: a_n^{L*}) is optimal, i.e., $r_h + \tilde{R}_{n+2}^*(0) \leq 0 + \tilde{R}_{n+1}^*(0)$ for all n. Under this policy, we will choose to serve a light-traffic SU whenever possible.

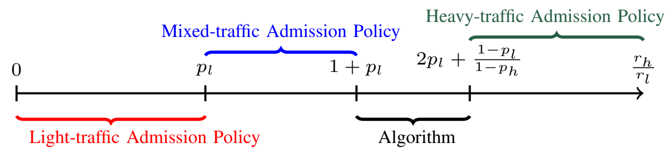

Fig. 2. Stationary admission policies for any price ratio $r_h/r_l \in [0, \infty)$. No stationary policy exists in the "Algorithm" segment, we need to resort to the dynamic admission algorithm (see [21]) for the general switch-over policy.

According to our analysis, it turns out that each of the three stationary policies is optimal (i.e., the corresponding condition holds) under a particular range of price ratio r_h/r_l. We summarize the result in the following theorem.

Theorem 1 (Optimality of Stationary Admission Policies): A stationary admission policy becomes the optimal policy to solve the problem **P1** if the following is true:

- The heavy-traffic admission policy a_n^{H*} in Tab.II–H for all $n \in \mathcal{N}$ is optimal if $\frac{r_h}{r_l} \geq 2p_l + \frac{1-p_l}{1-p_h}$.
- The mixed-traffic admission policy a_n^{M*} in Tab.II–M for all $n \in \mathcal{N}$ is optimal if $p_l \leq \frac{r_h}{r_l} \leq 1 + p_l$.
- The light-traffic admission policy a_n^{L*} in Tab.II–L for all $n \in \mathcal{N}$ is optimal if $\frac{r_h}{r_l} < p_l$.

Fig. 2 illustrates the results stated in Theorem 1. In this figure, we divide the feasible price choices r_h/r_l (i.e., the range between $[0, \infty)$) into four segments, among which three table segments corresponds to three stationary policies that we have discussed. The "Algorithm" segment corresponds to the case where we do not know whether a stationary policy is optimal or not, since the conditions in Theorem 1 are sufficient but not necessary. In that case, we need to use the algorithm (in our online appendix [21]) to compute the optimal dynamic admission policy for the problem **P1**.

Except the "Algorithm" segment, we are able to derive the closed-form optimal revenue (the maximum objective function) of the problem **P1** for all the other three stationary policy segments as functions of the prices (detailed analysis is presented in [21]). Next we will proceed to consider the static price optimization problem for the database operator.

C. Optimization of Static Pricing

We have developed the optimal admission choices in Theorem 1 given fixed prices r_l and r_h. Now we are ready to compute the optimal static pricing. As introduced in Sec. II, we assume that prices will affect the number of SUs requesting the resource. In particular, a higher price will drive down the SUs' demands. As an example, we will consider the linear demand function in economics [23], where the probability of having an SU of type i demanding spectrum in a time slot is $p_i(r_i) = 1 - k_i r_i, i \in \{l, h\}$.[3] The parameters k_l and

[3]Linear demand function is also a widely used model in the revenue management literature (e.g., [15], [16]). Changing to some common nonlinear functions are unlikely to change the key results.

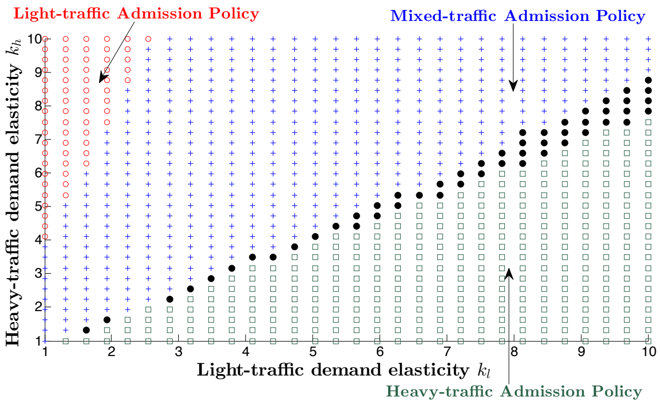

Fig. 3. Optimal admission policies with optimized prices, for different values of k_l and k_h. The red circle ○ regime is the light-traffic admission policy, the blue cross + regime is the mixed-traffic admission policy, and the dark green square □ regime is the heavy-traffic admission policy. These three regimes correspond to the three stationary policy segments in Fig. 2, respectively. The black dot ● regime corresponds to the "Algorithm" segment in Fig. 2.

k_h characterize the demand elasticity of light- and heavy-traffic SUs, respectively, and larger values indicate higher price sensitivities.

We proceed to optimize the prices that maximize the total revenue in the problem **P1**. Although we may derive the closed-form optimal revenue $\tilde{R}_n^*$ as a function of prices r_l and r_h (if one of the stationary admission policy is optimal, see Sec. III-B), the function $\tilde{R}_n^*(r_l, r_h)$ turns out to be non-convex in general, and the optimal prices cannot be solved in closed form.

However, notice that the key benefit of static pricing is that it does not need to be recomputed and updated frequently over time, thus we can compute the optimal static prices *offline* and afford a high computation complexity.

Recall that the three conditions in Theorem 1 are characterized by the price ratio r_h/r_l. It is natural to ask whether the optimal static pricing satisfies one of the conditions so that it is indeed optimal to choose a stationary policy. Fig. 3 illustrates the corresponding result, showing that when a stationary admission control policy is optimal under the optimal static prices for particular system parameters k_l and k_h. As we can see, except few black dots which correspond to the case of "Algorithm" segment in Fig. 2, the stationary policies are optimal in most cases (more than 94% in Fig. 3) once we optimize the static prices.

IV. Optimal Database Operation under Dynamic Pricing and Dynamic Admission

A. Joint Dynamic Pricing and Dynamic Admission Formulation

Now we further study the general case of dynamic pricing, where the database operator has the flexibility of changing prices over time. The database operator's goal is to compute the optimal prices $\boldsymbol{r}_l^* = \{r_l^*(1), \cdots, r_l^*(N)\}$ and $\boldsymbol{r}_h^* = \{r_h^*(1), \cdots, r_h^*(N)\}$, and the optimal admission policy $\boldsymbol{\pi}^* = \{a_1^*(S_1, X_1, Y_1), \cdots, a_N^*(S_N, X_N, Y_N)\}$ from time slot 1 to time slot N to maximize its expected long-term revenue, i.e.,

P2: *Joint Dynamic Pricing and Dynamic Admission*

$$\begin{aligned}
\text{Maximize} \quad & \mathbb{E}^{\boldsymbol{\pi}}_{\boldsymbol{X},\boldsymbol{Y}}[R_1(\boldsymbol{S}, \boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{\pi}, \boldsymbol{r}_l, \boldsymbol{r}_h)] && (7)\\
\text{Subject to} \quad & a_n(S_n, X_n, Y_n) \in \mathcal{A}_n(S_n, X_n, Y_n), \forall n \in \mathcal{N},\\
& S_{n+1} = (S_n + a_n(1 - S_n) - 1)^+,\\
& \qquad\qquad \forall n \in \{1, \cdots, N-1\},\\
& 0 \le r_l(n) \le r_l^{\max}, \forall n \in \mathcal{N},\\
& 0 \le r_h(n) \le r_h^{\max}, \forall n \in \mathcal{N},\\
\text{Variables} \quad & \{\boldsymbol{\pi}, \boldsymbol{r}_l, \boldsymbol{r}_h\}.
\end{aligned}$$

We can again use backward induction to solve the problem **P2**, by starting from the last time slot and analyze till the first. The subproblem in each time slot is to jointly determine the prices and the admission decision to maximize the expected future revenue from the current time slot to time slot N. For the ease of exposition, at any time slot n, we denote the expected future revenue $\mathbb{E}_{X_n,Y_n}[R_n(\cdot)]$ as $V_n(\cdot)$, and its optimized value over $r_l(n)$, $r_h(n)$ and $a_n(S_n, X_n, Y_n)$ as V_n^*. The key notations have been listed in Table I. The subproblem in each time slot $n \in \mathcal{N}$ is

P3: *Pricing-Admission Subproblem in time slot* n

$$\begin{aligned}
\text{Maximize} \quad & V_n\big(r_l(n), r_h(n), a_n(S_n, X_n, Y_n)\big) && (8)\\
\text{Subject to} \quad & a_n(S_n, X_n, Y_n) \in \mathcal{A}_n(S_n, X_n, Y_n), && (8.1)\\
& 0 \le r_l(n) \le r_l^{\max}, && (8.2)\\
& 0 \le r_h(n) \le r_h^{\max}, && (8.3)\\
\text{Variables} \quad & \{a_n(S_n, X_n, Y_n), r_l(n), r_h(n)\}.
\end{aligned}$$

The key challenge of solving the problem **P3** is due to the coupling among the decisions variables. Next we will propose a decomposition scheme that helps us solve the problem **P3** in each time slot n.

B. Decomposition of Pricing and Admission in Each Time Slot

First we want to clarify the difference between an *admission strategy* and an *admission policy*. An admission strategy specifies admission actions for a specific time slot n, while an admission policy specifies the admission strategies for all time slots in $\mathcal{N}$. Next we consider all possible admission control strategies for a particular time slot n, as shown in Table III.[4] In this table, each strategy is accompanied by a condition of the total revenue from time slot n to time slot N, which consists of the immediate revenue for an admission action at time slot n and the expected future revenue after this admission action is made. The strategy is optimal to use in a time slot if the condition is satisfied. Let us take the heavy-traffic strategy as an example. It means that admitting a heavy-traffic SU is optimal when it is possible to do so, i.e., choose $a_n = 2$ if $Y_n = 1$, regardless of the value of X_n. When $Y_n = 0$, admitting a light-traffic SU is optimal if it is possible, i.e., $a_n = 1$ if $(X_n, Y_n) = (1, 0)$ and $a_n = 0$ if $(X_n, Y_n) = (0, 0)$. Finally, by combining all the analysis together, the decision under the heavy-traffic strategy can be summarized as $a_n = (2 - X_n)Y_n + X_n$. The corresponding condition for the heavy-traffic strategy implies that the total

[4]Note that the heavy-traffic policy in Sec. III requires that the heavy-traffic strategy holds for all time slots $n \in \mathcal{N}$.

TABLE III. ADMISSION STRATEGIES IN TIME SLOT n

Three Admission Strategies in time slot n	Conditions
Heavy-traffic Strategy $a_n = (2 - X_n)Y_n + X_n$	$r_h(n) + V^*_{n+2} \geq r_l(n) + V^*_{n+1}$
Mixed-traffic Strategy ($\mathbf{1}_{\{\cdot\}}$ is the indicator function) $a_n = X_n \cdot \mathbf{1}_{\{Y_n=0\}} + (2 - X_n) \cdot \mathbf{1}_{\{Y_n=1\}}$	$r_h(n) + V^*_{n+2} < r_l(n) + V^*_{n+1}$ $r_h(n) + V^*_{n+2} > 0 + V^*_{n+1}$
Light-traffic Strategy $a_n = X_n$	$r_h(n) + V^*_{n+2} \leq 0 + V^*_{n+1}$

revenue of admitting a heavy-traffic SU is no less than that of admitting a light-traffic SU, i.e., $r_h(n) + V^*_{n+2} \geq r_l(n) + V^*_{n+1}$. The conditions for the other two admission strategies can be understood similarly. By looking at the conditions in Table III, we can see that the three admission strategies cover all possible decisions in time slot n. This observation helps us decompose the problem **P3** into three subproblems.

More specifically, in the pricing phase of time slot n (see Fig. 1), the pricing-admission decomposition approach involves two steps:

- *Price optimization under a chosen admission strategy:* Assume that one of the three admission strategies in Table III will be used in time slot n, we optimize prices $r_l(n)$ and $r_h(n)$ to maximize the expected future revenue.
- *Admission strategy optimization of the one-out-of-three admission strategies:* Compare the maximized expected future revenues under the three admission strategies, and pick the best admission strategy (and the corresponding optimized prices) that leads to the largest revenue.

Next, we will derive the closed-form optimal pricing under any chosen admission strategy, respectively.

1) Optimal Pricing under the Heavy-traffic Strategy (HTS): Given HTS chosen in time slot n, we derive the expected future revenue $V^H_n\big(r^H_l(n), r^H_h(n)\big)$ by considering the nested form of revenues produced by actions a_n plus the optimal expected future revenue V^*_{n+1} or V^*_{n+2}, weighted by the probabilities of different SU arrivals (values of (X_n, Y_n)), i.e.,

$$
\begin{aligned}
V^H_n\big(r^H_l(n), r^H_h(n)\big) &= p_l(r^H_l(n))p_h(r^H_h(n)) \cdot \big(r^H_h(n) + V^*_{n+2}\big) \\
&+ p_h(r^H_h(n))\big(1 - p_l(r^H_l(n))\big) \cdot \big(r^H_h(n) + V^*_{n+2}\big) \\
&+ p_l(r^H_l(n))\big(1 - p_h(r^H_h(n))\big) \cdot \big(r^H_l(n) + V^*_{n+1}\big) \\
&+ \big(1 - p_l(r^H_l(n))\big)\big(1 - p_h(r^H_h(n))\big) \cdot \big(0 + V^*_{n+1}\big),
\end{aligned}
$$

where the superscript H indicates the HTS admission strategy for time slot n. Note that this does not imply that the database will always use HTS in future time slots. The database operator needs to solve the following problem.

P4: *Optimal Pricing for time slot n under HTS*

$$
\begin{aligned}
\text{Maximize}\quad & V^H_n(r^H_l(n), r^H_h(n)) && (9)\\
\text{Subject to}\quad & r^H_h(n) + V^*_{n+2} \geq r^H_l(n) + V^*_{n+1}, && (9.1)\\
& 0 \leq r^H_l(n) \leq r^{\max}_l, && (9.2)\\
& 0 \leq r^H_h(n) \leq r^{\max}_h, && (9.3)\\
\text{Variables}\quad & r^H_l(n) \text{ and } r^H_h(n).
\end{aligned}
$$

The first constraint guarantees that the heavy-traffic admission strategy is optimal to use in time slot n, where V^*_{n+2} and V^*_{n+1} are determined by the solutions of the subproblems in time slots $n+2$ and $n+1$ earlier in the problem **P4**. Since the optimization is a continuous function over a compact feasible set, the maximum is guaranteed to be attainable. It is easy to show that the problem **P4** is not a convex optimization problem. Thus any solution satisfying KKT conditions may be only a local optimum of the problem **P4**. Hence we need to find all local optimum solutions satisfying KKT conditions, and then compare these solutions to pick up the global optimum.

TABLE IV. OPTIMAL PRICING UNDER HEAVY-TRAFFIC STRATEGY

	$V^*_{n+1} - V^*_{n+2}$		
	$\leq \frac{4k_l - 3k_h}{4k_h k_l}$	$\left(\frac{4k_l - 3k_h}{4k_h k_l}, \frac{2 - \sqrt{1 + \frac{k_h}{k_l}}}{k_h}\right)$	$\geq \frac{2 - \sqrt{1 + \frac{k_h}{k_l}}}{k_h}$
$\frac{k_h}{k_l} < \frac{4}{3}$	$I0$	$E1$	$E2$
$\frac{4}{3} \leq \frac{k_h}{k_l} < 3$	N/A	$E1$	$E2$
$\frac{k_h}{k_l} \geq 3$	N/A	N/A	$E2$

To find out an optimal solution to the problem **P4**, we need to first examine the feasible region based on any possible parameter values $r^H_l(n)$ and $r^H_h(n)$. It turns out that the feasible region is a polyhedron on a two-dimensional plane. As such, we only need to check whether all the possible extreme points and the interior points satisfying KKT conditions are local optima. There are several possible extreme point solutions with either one or two active (binding) constraints. We can further show that most extreme points are not local optimal solutions. We skip the details due to space limit, and summarize the optimal pricing results in Proposition 1.

Proposition 1: Under the linear demand functions, the optimal dynamic pricing in time slot n under HTS is summarized in Table IV, which depends on the value of $V^*_{n+1} - V^*_{n+2}$.

In Table IV, the notations $I0$, $E1$, and $E2$ represent the interior point solution with no active constraint, the extreme point solution with one active constraint, and the extreme point solution with two active constraints, respectively. "N/A" represents the cases where the combinations of conditions are infeasible. For example, the first column of $V^*_{n+1} - V^*_{n+2}$ and the row $\frac{4}{3} \leq \frac{k_h}{k_l} < 3$ lead to the condition of $V^*_{n+1} - V^*_{n+2} \leq \frac{4k_l - 3k_h}{4k_h k_l} \leq 0$. This is not possible, since V^*_{n+1} includes one more time slot. Hence, the corresponding cell is labeled as "N/A".

2) Optimal Pricing under the Mixed-traffic Strategy (MTS): Given MTS chosen in time slot n, the expected future revenue $V^M_n\big(r^M_l(n), r^M_h(n)\big)$ can be similarly derived as the HTS case, where the superscript M indicates the MTS admission strategy. The database operator needs to solve the following problem.

P5: *Optimal Pricing for time slot n under MTS*

$$
\begin{aligned}
\text{Maximize}\quad & V^M_n(r^M_l(n), r^M_h(n)) && (10)\\
\text{Subject to}\quad & r^M_h(n) + V^*_{n+2} \leq r^M_l(n) + V^*_{n+1}, && (10.1)\\
& r^M_h(n) + V^*_{n+2} \geq V^*_{n+1}, && (10.2)\\
& 0 \leq r^M_l(n) \leq r^{\max}_l, && (10.3)\\
& 0 \leq r^M_h(n) \leq r^{\max}_h, && (10.4)\\
\text{Variables}\quad & r^M_l(n) \text{ and } r^M_h(n).
\end{aligned}
$$

The first two constraints guarantee that the mixed-traffic strategy is optimal to used in time slot n. Similarly to the analysis of the problem **P4**, we have the following proposition.

TABLE V. OPTIMAL PRICING UNDER MIXED-TRAFFIC STRATEGY

	$V^*_{n+1} - V^*_{n+2}$		
	$\geq \frac{2\sqrt{1-\frac{k_h}{k_l}}-1}{k_h}$	$\left(\frac{k_l-3k_h}{2k_hk_l}, \frac{2\sqrt{1-\frac{k_h}{k_l}}-1}{k_h}\right)$	$\leq \frac{k_l-3k_h}{2k_hk_l}$
$\frac{k_h}{k_l} < \frac{1}{3}$	$I0'$	$E1'$	$E2'$
$\frac{1}{3} \leq \frac{k_h}{k_l} < \frac{3}{4}$	$I0'$	$E1'$	N/A
$\frac{k_h}{k_l} \geq \frac{3}{4}$	$I0'$	N/A	N/A

Proposition 2: Under the linear demand functions, the optimal dynamic pricing solutions in time slot n under MTS is summarized in Table V, which depends on the determined difference $V^*_{n+1} - V^*_{n+2}$.

Table V is similarly derived as the HTS case. In this table, the notations $I0'$, $E1'$, and $E2'$ represent the interior point solution with no active constraint, the extreme point solution with one active constraint, and the extreme point solution with two active constraints in the MTS case, respectively. Notice that they have different expressions from those in the HTS case (in Table IV). Likewise, we also use "N/A" to represent the three cases where the conditions lead to the impossible relationship of $V^*_{n+1} - V^*_{n+2} \leq 0$.

Table IV and Table V show the optimal dynamic pricing in each time slot n. Given the demand elasticities k_l and k_h, the solution will be uniquely given by one of the three cases of the difference $V^*_{n+1} - V^*_{n+2}, \forall n$. In Sec. IV-C, we will propose an algorithm to compute the difference of $V^*_{n+1} - V^*_{n+2}$.

3) Optimal Pricing under the Light-traffic Strategy (LTS): Given LTS chosen in time slot n, the expected future revenue $V^L_n(r^L_l(n), r^L_h(n))$ can also be similarly derived as the HTS case, where the superscript L indicates the LTS admission strategy. The database operator needs to solve the following problem.

P6: *Optimal Pricing for time slot n under LTS*

$$\begin{aligned} \text{Maximize} \quad & V^L_n(r^L_l(n), r^L_h(n)) && (11)\\ \text{Subject to} \quad & r^L_h(n) + V^*_{n+2} \leq 0 + V^*_{n+1}, && (11.1)\\ & 0 \leq r^L_l(n) \leq r^{\max}_l, && (11.2)\\ & 0 \leq r^L_h(n) \leq r^{\max}_h, && (11.3)\\ \text{Variables} \quad & r^L_l(n) \text{ and } r^L_h(n). \end{aligned}$$

Unlike the HTS and the MTS cases, we can derive the optimal prices under LTS in closed-form.

Proposition 3: Under the linear demand functions, the optimal prices in time slot n under LTS are

$$r^{LI}_l(n) = \frac{1}{2k_l}, r^{LI}_h(n) = \min\{V^*_{n+1} - V^*_{n+2}, r^{\max}_h\}. \quad (12)$$

C. Optimal Dynamic Pricing and Admission Policies

After deriving the optimal prices under each admission strategy, we can now compare the corresponding revenues and choose the best admission strategy for time slot n. We need to do this for each of the N time slots to derive the optimal admission strategies and prices. We show this process in the *policy development* phase in Algorithm 1, which involves the previous three propositions (Tables IV, V, and Equation (12)). More specifically, the algorithm iteratively computes the prices and revenues under the three admission strategies, respectively, and then selects the optimal prices which lead to the largest revenue (lines 4 to 16). The complexity of Algorithm 1 is low and linear in the number of time slots $\mathcal{O}(N)$, as it only needs to check Table IV, Table V, and Equation (12) that we derived. We summarize the optimality result as follows.

Algorithm 1: Policy Development and Implementation

1: *Policy Development Phase:*
2: Set $n \leftarrow N+1$, $V^*_{N+1} \leftarrow 0$
3: Set $r^*_l(N), r^*_h(N)$ by (12) and V^*_N by the objective function in the problem P6 $V^L_N(r^*_l(N), r^*_h(N))$.
4: **for** $n = N-1..., 2, 1$ **do**
5: Compute $r^{H*}_l(n), r^{H*}_h(n)$ and V^{H*}_n by Table IV.
6: Compute $r^{M*}_l(n), r^{M*}_h(n)$ and V^{M*}_n by Table V.
7: Compute $r^{L*}_l(n), r^{L*}_h(n)$ and V^{L*}_n by (12).
8: Then $r^*_l(n), r^*_h(n) \leftarrow \arg\max\{V^{H*}_n, V^{M*}_n, V^{L*}_n\}$.
9: **if** $r^*_l(n), r^*_h(n) = r^{H*}_l(n), r^{H*}_h(n)$ **then**
10: The heavy-traffic strategy in Table III is optimal.
11: **else if** $r^*_l(n), r^*_h(n) = r^{M*}_l(n), r^{M*}_h(n)$ **then**
12: The mixed-traffic strategy in Table III is optimal.
13: **else**
14: The light-traffic strategy in Table III is optimal.
15: **end if**
16: **end for**
17: **return** Dynamic pricing-admission policy $\boldsymbol{r}^*$ and $\boldsymbol{\pi}^*$.
18: *Policy Implementation Phase:*
19: Set $n \leftarrow 1$, $S_1 \leftarrow 0$
20: **while** $n \leq N$ **do**
21: Announce the prices $\boldsymbol{r}^*(n)$ and determine the SU requests (X_n, Y_n).
22: **if** $S_n = 1$ **then**
23: The admission action $a^*_n \leftarrow 0$.
24: **else**
25: Check $\boldsymbol{\pi}^*(a_n(0, X_n, Y_n))$ for the action a^*_n.
26: **end if**
27: Set $S_{n+1} \leftarrow (S_n + a_n(1-S_n) - 1)^+, n \leftarrow n+1$
28: **end while**

Theorem 2: The dynamic prices $\boldsymbol{r}^* = \{\boldsymbol{r}^*(n), \forall n \in \mathcal{N}\}$ and the admission policy $\boldsymbol{\pi}^* = \{a^*_n(S_n, X_n, Y_n), \forall n \in \mathcal{N}\}$ derived in the policy development phase in Algorithm 1 correspond to the optimal solution to the problem **P2**.

Theorem 2 can be proved by using the principle of optimality in dynamic programming [22]. Note that the optimal prices and admission policy form a *contingency* plan that contains information about the optimal prices and admission decision at all the possible system states $(\boldsymbol{S}, \boldsymbol{X}, \boldsymbol{Y})$ in any time slots $n \in \mathcal{N}$. To implement the optimal policy, the database operator needs to decide actual admission actions according to the realizations of random demands and the transition of system states (see the *policy implementation phase* in Algorithm 1). More specifically, at the beginning of each time slot n, the operator first announces prices $\boldsymbol{r}^*(n)$ and checks the actual arrivals (X_n, Y_n) (line 21). Then, the admission decisions are carried out based on the optimal policy $\boldsymbol{\pi}^*$ through checking a table (lines 22 to 26) and the state component S_n is updated accordingly (line 27).

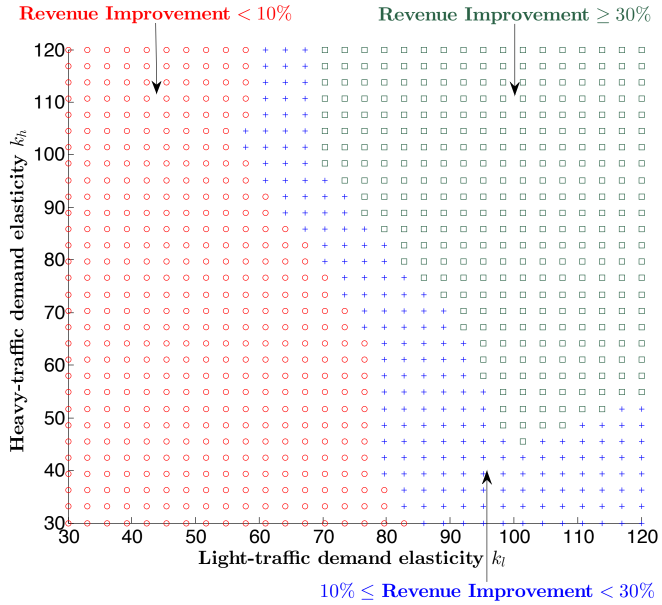

Fig. 4. Revenue improvement evaluation: $N = 100$. The revenue improvements are given in terms of k_l and k_h, where the red circle $\circ$ regime is less than 10%, the blue cross $+$ regime is between 10% and 30%, and the dark green square $\square$ regime is more than 30%.

D. Comparison between Static and Dynamic Pricing

Now we compare the total operator revenue under optimized dynamic pricing and optimized static pricing. Fig. 4 shows the revenue improvement of dynamic pricing over the static pricing under different demand elasticities of k_l and k_h. As shown in Fig. 4, dynamic pricing outperforms static pricing by more than 30% when both types of SUs are sensitive to prices and have high demand elasticities parameters k_l and k_h. When both types of SUs are not very price-sensitive, dynamic pricing leads to limited revenue improvement (less than 10%) than the static pricing, and it is better to implement static pricing due to its low complexity.

V. Conclusion

In this paper, we consider a spectrum database operator's revenue maximization problem through joint spectrum pricing and admission control. We incorporate the heterogeneity of SUs' spectrum occupancy and demand uncertainty into the model, and consider both the static and dynamic pricing schemes. In static pricing, we show that stationary admission policies can achieve optimality in most cases. In dynamic pricing, we compute optimal pricing through a proper pricing-admission decomposition in each time slot. Finally, we show that dynamic pricing significantly improves revenue over the static pricing when SUs are sensitive to prices.

In the future work, we will consider the pricing and admission control of multiple parallel channels. In this case, SUs may flexibly request different spectrum-time chunks in a two-dimensional time and frequency plane. One challenge is how to solve this Markov Decision Process (MDP), where the system state and state dynamics are much more complicated. We may further consider delay tolerant SUs who are willing to wait in queues if not admitted immediately. We may use the Lyapunov method to analyze the system stability and performance under such a scenario [24]. We may further use the queueing based MDP to analyze the pricing and admission decision for such a scenario.

References

[1] FCC 10-174, "Second Memorandum Opinion and Order," 2010.

[2] Ofcom, "Implementing geolocation," Nov. 2010.

[3] Spectrum Bridge, www.spectrumbridge.com.

[4] Microsoft Spectrum Observatory, spectrum-observatory.cloudapp.net.

[5] Google Spectrum Database, www.google.org/spectrum/whitespace/.

[6] X. Feng, J. Zhang, and Q. Zhang, "Database-Assisted Multi-AP Network on TV White Spaces: Architecture, Spectrum Allocation and AP Discovery," *Proc. IEEE DySPAN*, 2011.

[7] X. Feng, Q. Zhang, and J. Zhang, "Hybrid Pricing for TV White Space Database," *Proc. IEEE INFOCOM*, 2013.

[8] Y. Luo, L. Gao, and J. Huang, "White Space Ecosystem: A Secondary Network Operator's Perspective," *Proc. IEEE GLOBECOM*, 2013.

[9] P. Hande1, M. Chiang, R. Calderbank, and J. Zhang, "Pricing under Constraints in Access Networks: Revenue Maximization and Congestion Management," *IEEE INFOCOM*, 2010.

[10] H. Yu, L. Gao, Z. Li. X. Wang, E. Hossain, "Pricing for Uplink Power Control in Cognitive Radio Networks," *IEEE Transactions on Vehicular Technology*, vol 59, no. 4, pp. 1769-1778, 2010.

[11] X. Wang, Z. Li, P. Xu, Y. Xu, X. Gao, and H. Chen, "Spectrum Sharing in Cognitive Radio Networks-An Auction based Approach," *IEEE Transactions on System, Man and Cybernetics-Part B: Cybernetics*, vol 40, no. 3, pp. 587-596, 2010.

[12] J. Huang and L. Gao, "Wireless Network Pricing," *Synthesis Lectures on Communication Networks*, Morgan & Claypool, July 2013.

[13] C. Courcoubetis and R. Weber, "Pricing Communication Networks: Economics, Technology and Modelling," Wiley, England, 2003.

[14] C. Gizelis and D. Vergados, "A Survey of Pricing Schemes in Wireless Networks," *IEEE Commu. Surveys & Tutorials*, vol. 30, no. 7, 2012.

[15] G. Gallego and G. Van Ryzin, "Optimal Dynamic Pricing of Inventories with Stochastic Demand over Finite Horizons," *Management Science*, vol. 40, no. 8, pp. 999-1020, 1994.

[16] D. Zhang and Z. Lu, "Assessing the Value of Dynamic Pricing in Network Revenue Management," *INFORMS Journal on Computing*, vol. 25, no. 1, pp. 102-115, 2013.

[17] C. Joe-Wong, S. Ha, S. Sen, and M. Chiang, "Optimized Dynamic Pricing for Broadband Access," Princeton University, Tech. Rep., 2011. [Online]. Available: http://scenic.princeton.edu/paper/TR.pdf

[18] Y. Song, C. Zhang, Y. Fang, and P. Lin, "Revenue Maximization in Time-Varying Multi-Hop Wireless Networks: A Dynamic Pricing Approach," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 7, pp. 1237-1245, 2012.

[19] S. Ha, S. Sen, C. Joe-Wong, Y. Im, and M. Chiang, "TUBE: Time-Dependent Pricing for Mobile Data," *Proc. ACM SIGCOMM*, 2012.

[20] Q. Ma, Y. Liu, and J. Huang, "Time and Location Aware Mobile Data Pricing," *Proc. IEEE ICC*, 2014.

[21] C. Jiang, L. Duan, and J. Huang, "Appendix of Joint Spectrum Pricing and Admission Control for Heterogeneous SUs." [Online]. Available: http://jianwei.ie.cuhk.edu.hk/publication/AppendixDynamicPricing.pdf

[22] D.P. Bertsekas, "Dynamic Programming and Optimal Control," 3rd ed., Athena Scientific, Belmont, Massachusetts, 2005.

[23] A. Mas-Colell, M. D. Whinston, and J. R. Green, "Microeconomic theory," Oxford University Press, New York, 1995.

[24] S. Li, J. Huang, and S.Y. Li, "Dynamic Profit Maximization of Cognitive Mobile Virtual Network Operator," *IEEE Transactions on Mobile Computing*, vol. 13, no. 3, pp. 526-540, March 2014.