Online Pricing of Secondary Spectrum Access with Unknown Demand Function

Huseyin Mutlu, Murat Alanyali, David Starobinski and Aylin Turhan

***Abstract*—We consider a wireless provider who caters to two classes of customers, namely primary users (PUs) and secondary users (SUs). PUs have long term contracts while SUs are admitted and priced according to current availability of excess spectrum. The average rate at which SUs attempt to access the spectrum is a function on the currently advertised price, referred to as the demand function. We analyze the problem of maximizing the average profit gained by admissions of SUs, when the demand function is *unknown*. We introduce a new on-line algorithm, called Measurement-based Threshold Pricing (MTP), that requires the optimization of only two parameters, a price and a threshold, whereby SU calls are admitted and charged a fixed price when the channel occupancy is lower than the threshold and rejected otherwise. At each iteration, MTP measures the average arrival rate of SUs corresponding to a certain test price. We prove that these measurements of the secondary demand are sufficient for MTP to converge to a local optimal price and corresponding optimal threshold, within a number of measurements that is logarithmic in the total number of possible prices. We further provide an adaptive version of MTP that adjusts to time-varying demand and establish its convergence properties. We conduct numerical studies showing the convergence of MTP to near-optimal online profit and its superior performance over a traditional reinforcement learning approach.**

***Index Terms*—Management of electromagnetic spectrum, secondary markets, congestion pricing, real-time algorithms.**

I. INTRODUCTION

As a result of continuing efforts to deregulate wireless spectrum management, policy agencies are granting providers with the right to lease their spectrum [2]. This policy reform promises more efficient use of excess spectrum, which otherwise may be wasted. Implications of this reform can be seen in the novel services provided by spectrum brokerage companies, which match potential lessees and spectrum providers (licence holders). One such service is an on-line spectrum trading and leasing platform, called SpecEx.com, which was launched by Spectrum Bridge Inc. in 2008 [3].

The aforementioned spectrum reforms call for the design of efficient pricing strategies, since a spectrum provider strives to maximize its profit from leasing its excess spectrum. In this paper, we aim at developing a realistic pricing framework to achieve this goal. We consider a set-up consistent with the private commons model envisioned by FCC [2, 4], in which a wireless spectrum provider caters to two classes of customers, namely primary users (PUs) and secondary users (SUs). PUs have long term contracts and are not subjected to on-line pricing. On the other hand, SUs are admitted and priced according to the current availability of excess spectrum. The average rate at which SUs attempt to access the spectrum depends on the currently advertised price. The function describing this dependency is referred to as the *demand function*. The provider must ensure that admission of SUs does not significantly affect quality of service of PUs. This is because presence of SUs may increase *blocking* of PU calls, i.e., the rejection of PUs due to lack of channel availability, and hence lead to a punishment in the form of loss of business due to poor service.

The problem of pricing of shared resources has been widely studied in the literature [5–7]. Recent works introduced pricing strategies specifically tailored for secondary access of resources [8–10]. Yet, the overwhelming majority of papers in this area assume that the demand function of users is known and static (see Sec. II for exceptions). Precise knowledge of the demand function, which may vary over time, is, however, hard to acquire, and raises the question of how to apply this body of existing work in practice.

The contributions of this paper are the following. First, we propose a new on-line algorithm, called *Measurement-based Threshold Pricing* (MTP), for efficiently pricing secondary spectrum access and maximizing average profit when the demand function is unknown, but satisfies certain mild assumptions. MTP belongs to the class of *occupancy-based* pricing policies that depend only on the total number of ongoing (SU and PU) calls in the system, and whose performances are insensitive to the call length distribution, except through the mean [11]. While a general occupancy-based policy (including the optimal one) requires the optimization of a different price for each channel occupancy level, MTP requires the optimization of only two parameters, namely, a *threshold* and a *price*. Thus, SU calls are admitted and charged a fixed price (per unit of time or per call) if the number of occupied channels upon their arrivals is smaller than the specified threshold, and rejected otherwise.

MTP is an iterative algorithm that applies Fibonacci search to optimize an unknown profit function that depends on price only. At each iteration, MTP measures the average arrival rate of SUs corresponding to a certain *test* price and narrows down the search interval. We show that these price-based measurements are sufficient to derive both a locally optimal price and optimal threshold. Though the profit function may be multimodal, we analytically prove that MTP converges to

The authors are affiliated with the Department of Electrical and Computer Engineering at Boston University, Boston, MA 02215, USA (e-mail: hmutlu@gmail.com, alanyali@bu.edu, staro@bu.edu, aylinturhan@gmail.com). Corresponding author: David Starobinski (phone: +1-617-353-0202, FAX: +1-617-353-7337).

This work was supported in part by the US National Science Foundation under grant CCF-0964652.

A preliminary version of this paper appeared in the proceedings of IEEE INFOCOM 2010 [1].

a local optimum as fast as if the function were unimodal (a function is unimodal over a certain interval, if it has a single maximum over that interval). Specifically, we show that the number of iterations (measurements) required by MTP is logarithmic in the total number of possible prices and independent of other variables, such as the total number of channels.

Next, we evaluate through simulations the performance of MTP with finite measurement windows, which implies that the estimation of the SUs arrival rate at each iteration is noisy. Defining the mean call length to be one unit of time and setting the measurement window length to be one time unit as well, we show that, on average, MTP converges to a profit within 10% of the optimal occupancy-based policy (which knows the demand function *a-priori*) within only 5 time units, assuming a range of 10^4 different possible prices. Larger measurement windows of length 10 and 100 time units, bring the average profit of MTP within 5% and 2% of the optimal occupancy-based policy, respectively. We contrast the performance of MTP to a traditional Q-Learning algorithm, which is shown to only achieve about one third of the profit of MTP.

Last, we consider the case when the arrival rate of PUs and the demand function of SUs are time-varying. We introduce an extended version of MTP, called *Adaptive Measurement-based Threshold Policy* (AMTP), for this purpose. AMTP uses a hill climbing method to determine an interval that contains a local optimum of the profit function. We prove that AMTP converges to this local optimum by following similar steps to MTP. We simulate AMTP for scenarios where the arrival rate of PUs and the demand function of SUs change throughout the course of a day. These simulations show that compared to the optimal policy (for which the demand function of SUs and arrival rate of PUs are known), AMTP loses only a small portion of the total profit gained during the course of a day.

The rest of this paper is organized as follows. In Section II, we discuss related work. In Section III, we introduce the system model and problem formulation. In Section IV, the MTP algorithm for pricing with unknown demand function is introduced and analyzed. In Section V, we propose the AMTP algorithm for pricing spectrum with time-varying user demand. We provide numerical examples in Section VI and conclude the paper in Section VII.

II. Related Work

Our work is related to the problem of *congestion-based pricing*, that is, pricing that depends on the current level of resource usage in the system. Ref. [5] studies pricing of network resources when the arrival rate of all users can be regulated with price. It shows that static pricing (a single price is advertised regardless of occupancy level) achieves good performance and is optimal in some asymptotic regimes. This result was extended in [6] in the context of large network asymptotics.

Ref. [8] analyzes spectrum pricing with two different types of users: elastic and non-elastic. This paper shows that static pricing does not perform well with both elastic and non-elastic users, but threshold pricing performs close to optimal. In particular, the profit region of threshold pricing (i.e., the range of PU arrival rates for which positive profit can be achieved) is proven to be optimal.

Ref. [9] studies optimal and static pricing policies within the context of a generic rental management optimization problem with two types of customers, which are akin to our SUs and PUs. Ref. [12] provides a game theoretic analysis of revenue maximization problem for secondary spectrum access. Ref. [10] studies secondary spectrum access pricing strategies capturing the effects of network-wide interferences. All the previous work mentioned above assume a known demand function, in contrast to the model presented in this paper.

Next, we present related work on the less studied field of pricing with unknown demand function. Ref. [13] introduces an on-line algorithm for static pricing of calls. It considers a parametric demand function (meaning that the demand function depends on a fixed number of unknown parameters), while we consider a more general non-parametric demand function. Ref. [14] considers the problem of a seller holding an initial inventory of a single class of products which must be sold over a finite time period. The goal is to dynamically adjust prices in order to maximize the average profit. In contrast, our paper considers the temporal allocation of resources (i.e., channels) by a provider to primary and secondary users. The total amount of resources is fixed, and the goal of the provider is to maximize average profit over an infinite time horizon from the allocation of resources to the users. Ref. [15] establishes a critical price value above which secondary access is profitable for a provider if there exists any secondary demand. While [15] assumes the demand function to be unknown, its focus is on profitability (i.e., ensuring profit) rather than profit-maximization as considered in our paper.

A possible approach to deal with unknown demand functions is to apply one of the well-known reinforcement learning algorithms, such as Q-Learning [16, 17]. Yet, because these algorithms are generic, they have the disadvantage of not exploiting the specific structure of the problem at hand. In particular, these algorithms generally do not scale well with large state-space or action-space because they need to find the optimal action for each state. On the other hand, the new MTP and AMTP algorithms presented in this paper only need to learn two parameters (threshold and price) to achieve near-optimal performance.

III. Model and Problem Formulation

In this section, we introduce our model and objective. We consider a single-cell wireless network which provides access to C channels. Calls from PUs arrive according to a Poisson process with fixed rate $\lambda_p > 0$. A punishment in the amount of K monetary units is imposed on the provider if all the channels are busy and a PU call is blocked. SU call arrivals also form a Poisson process with rate $\lambda_{SU} > 0$ that is independent of the PUs. We note that the measurement study in [18] justifies the use of the Poisson process to model call arrival rates. When an SU call arrives, it accepts with probability $p(u)$ the price u advertised by the provider and attempts to join the network. Therefore, the rate at which SUs attempt to access the spectrum is $\lambda_s(u) = \lambda_{SU} p(u)$. We refer to $\lambda_s(u)$ as the *demand function*. This function is unknown *a-priori*.

Some of the results in this paper assume one or both of the following assumptions on the demand function. We specifically state whenever these assumptions are required.

Assumption 3.1: There exists a maximum price $u_{\max}$ for which $\lambda_s(u_{\max}) = 0$. Moreover, $\lambda_s(u)$ is a strictly decreasing, differentiable function in u over the interval $[0, u_{\max}]$.

The second assumption enables development of our efficient on-line optimization procedure presented in Section IV.

Assumption 3.2: Let $u(\lambda_s)$ be the inverse of $\lambda_s(u)$ on the interval $0 \leq u \leq u_{\max}$. Then $\lambda_s u(\lambda_s)$ is concave with respect to λ_s.

Assumption 3.2 implies that the marginal instantaneous profit is decreasing with respect to user demand, and ensures a "well-behaved" revenue function (that is, the function $\lambda_s u(\lambda_s)$ is either monotone or unimodal in λ_s [19]). This assumption is widely made in the literature [5, 14, 19] and is satisfied by variety of demand functions such as functions with exponential, linear and polynomial decay.

In the model under consideration, once an SU is admitted, it occupies the channel throughout the entire length of its call. A service level agreement for PUs can be achieved by choosing an appropriate punishment K for blocked PU calls (higher K means lower PU blocking probability).

We assume that PU and SU call lengths have a common *general* distribution with mean $1/\mu$. Therefore, once accepted, PU and SU calls are statistically indistinguishable. This assumption is valid for scenarios where PUs and SUs utilize similar applications. The call length distribution is unknown, except for its mean. Without loss of generality, we assume that $\mu = 1$, i.e., the mean call length time is one unit of time. Note that while many earlier papers in the literature assume (for analytical tractability) that call lengths are exponentially distributed, a recent study based on measurement of real traces in a cellular network shows that this assumption does not hold in practice [18].

In this paper, we restrict our attention to pricing policies that are based solely on the total number (PU and SU) of ongoing calls in the system. We refer to these policies as *occupancy-based* policies. Note that the total occupancy is not Markovian unless call lengths are exponentially distributed; hence an optimal policy would typically entail further information such as the amount of time each call has already been in the system. An occupancy-based pricing policy sets an advertised SU price u_n when there are $n < C$ ongoing calls in the system. This price can either be applied on a per call basis or it can be based on the length of a call. In this latter case, u_n corresponds to the price charged per unit of time. Since $\mu = 1$, u_n represents in that case the average revenue per call, where the average is taken over all of the calls joining the system in the presence of n ongoing calls upon their arrivals.

Therefore, a pricing policy can be defined as a vector $\mathbf{u} = (u_0, u_1, u_2, ..., u_{C-1})$. We are interested in finding the vector $\mathbf{u}$ which maximizes the average profit per unit of time gained from accepting SUs. Prices for $\mathbf{u}$ are selected from a discrete set $\mathbb{U}$, taking values between 0 and u_{max}. The price granularity is Δu, i.e., any two consecutive prices are Δu monetary units apart. Thus, from now and on, the interval notation $[u_a, u_b]$ represents the discrete set of prices $\{u_a, u_a + \Delta u, \ldots, u_b\}$.

Due to practical concerns, we limit the search for an optimal pricing policy to occupancy-based policies; the specific form of the call length distribution is often unavailable or cannot be properly formalized. Even in such cases where the distribution is known, it is hard to price optimally due to uncertainty in the future length of ongoing calls. On the other hand, occupancy-based policies are provably insensitive to the call length distribution, except through the mean, that is, they induce the same equilibrium occupancy distribution for all call length distributions with the same mean [1, Theorem 4.2][11, Theorem 3.2.2]. Extensive numerical studies conducted in [1, 11] for call lengths with various phase-type distributions (i.e., hyper-exponential, hypo-exponential, and Coxian) indicate that the optimal occupancy-based policy performs very close to the optimal general pricing policy (the maximum observed difference in the profit is about 1%). Note that if the call length distribution is exponential, then the optimal occupancy-based policy is the same as the optimal general pricing policy due to the memoryless property of the exponential distribution.

A. Optimal occupancy-based dynamic and threshold pricing policies

The results in this section require Assumption 3.1 on the demand function. Based on our model, the average profit function for a given dynamic occupancy-based pricing policy with price vector $\mathbf{u}$ is given by:

$$R = \sum_{n=0}^{C-1} \pi_n \lambda_s(u_n) u_n - \pi_C \lambda_p K + E(\lambda_p, C)\lambda_p K, \qquad (1)$$

where π_n is the steady-state probability of finding n users (PUs and SUs) in the system, with $0 \leq n \leq C$, and where $E(\lambda_p, C)$ is the blocking probability of PUs in the absence of SU arrivals. This quantity corresponds to the well-known *Erlang-B* formula

$$E(\lambda_p, C) = \frac{\frac{\lambda_p^C}{C!}}{\sum_{n=0}^{C} \frac{\lambda_p^n}{n!}}. \qquad (2)$$

The first term in Eq. (1) represents the sum of the average revenues collected from SUs in each state. Specifically, for each state $0 \leq n \leq C-1$, we multiply the average rate of SU calls joining the system $\lambda_s(u_n)$ with the (average) revenue gained per call u_n. Then, the resulting term is scaled with the probability that an SU call arrival finds n users in the system. Due to the PASTA (Poisson Arrival See Time Averages) property, this probability is the same as the steady-state probability π_n. The second term in Eq. (1) is the average revenue loss due to rejected PUs. This quantity corresponds to the probability π_C that a PU call arrival finds the system full multiplied with the average rate of PU call arrivals λ_p and the punishment K incurred for each blocked PU call. The last term in Eq. (1) acts as a "normalization" term to ensure that the profit is zero when all SUs are rejected. Thus, R represents the difference in the profit with respect to the case where no SU is admitted. Note that $E(\lambda_p, C)$ is nothing but π_C in the absence of SUs.

The steady-state probabilities π_n are unknown a-priori, as their computations require knowledge of the arrival rates of SUs [8]. These rates are estimated using the MTP algorithm described in the sequel.

The average profit function in Eq. (1) and the optimal occupancy-based policy that maximizes it do not depend on the call length distribution, except through the mean. Thus, the optimal occupancy-based policy can be as well calculated by assuming that the call lengths are exponentially distributed. Under this assumption, occupancy-based pricing can be modeled as an average reward dynamic programming problem with C states, and the optimal prices can be calculated using standard techniques such as value iteration, policy iteration, or linear programming [11].

In a *threshold pricing* policy, SU calls are admitted and charged a price u when the channel occupancy is smaller than some threshold T and rejected otherwise. This is equivalent to having the price vector

$$\mathbf{u} = (\underbrace{u, u, ..., u}_{T}, \underbrace{u_{max}, u_{max}, ..., u_{max}}_{C-T}).$$

Consequently, the total arrival rate until the occupancy level reaches T channels is $\lambda_p + \lambda_s(u)$ and λ_p afterwards.

Assuming the above price vector, the average profit of a threshold pricing policy is

$$R_T(u) = \sum_{n=0}^{T-1} \pi_n \lambda_s(u) u - \pi_C \lambda_p K + E(\lambda_p, C)\lambda_p K. \quad (3)$$

The optimization of the threshold pricing policy involves finding the optimal values for the price u and threshold T.

IV. Spectrum pricing with unknown demand function

Typically, the SU demand function $\lambda_s(u)$ is unknown. In this section, we introduce an algorithm, called Measurement-based Threshold Pricing (MTP), to calculate the threshold pricing policy under this condition.

When $\lambda_s(u)$ is unknown, a formula for the threshold pricing profit function $R_T(u)$ is unavailable. However, we can measure the arrival rate of SUs for a specific price u and threshold T and calculate the average profit $R_T(u)$ for that price and threshold. Measurements are conducted by observing the rate of SUs who accept the advertised price for a sufficiently long period of time. In this section, we assume that measurements are exact. In Section VI, we numerically study the robustness of MTP to noise due to finite measurement windows. The threshold T used during the measurement is irrelevant due to the following property of $R_T(u)$.

Lemma 4.1: For a given price u, $R_T(u)$ can be calculated for any threshold $1 \leq T \leq C$ using a single measurement window.

This lemma is a direct consequence of Eq. (3), which can be calculated for any threshold and a given price once the corresponding $\lambda_s(u)$, which is independent of the current threshold, is acquired as a result of measurements.

In practice, measurements have to be performed while the system is in operation. These measurements are often done with non-optimal parameters which causes profit loss.

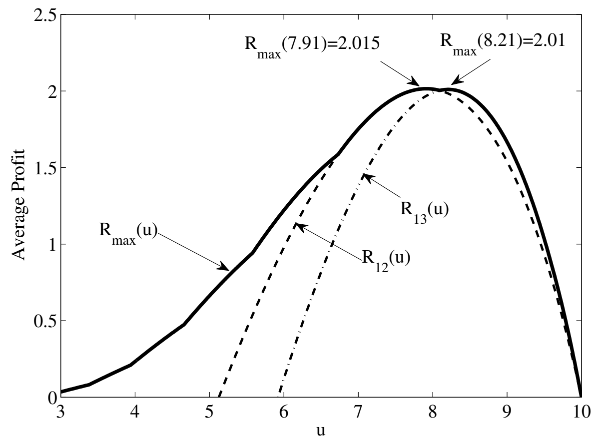


Fig. 1. Multimodal $R_{max}(u)$ and $R_T(u)$ for $T = 12$ and $T = 13$ on which two maxima occur. System parameters are $C = 20, \lambda_s(u) = (10 - u)_+, \lambda_p = 12.5$ and $K = 120$.

Therefore, our main goal is to calculate the optimal threshold pricing policy with as few measurements as possible.

A. Properties of the threshold pricing profit function

The properties we introduce in this section require both Assumption 3.1 and Assumption 3.2 on the demand function. These assumptions ensure that, for a fixed threshold, the profit function $R_T(u)$ is unimodal with respect to price in $[0, u_{max}]$ [8, Theorem 5.5]. This property enables efficient calculation of the optimal price for a given threshold. However, finding the optimal threshold requires a search over all possible threshold values. We circumvent this problem by introducing an auxiliary profit function which depends on price only:

$$R_{max}(u) = \max_{1 \leq T \leq C}(R_T(u)) \quad (4)$$

For a given price u, this function can be calculated with a single measurement window thanks to Lemma 4.1.

During our numerical studies, we observed that, for certain range of system parameters, $R_{max}(u)$ possesses the following property.

Claim 4.2: $R_{max}(u)$ can be multimodal in u for certain system parameters and demand functions.

Proof: Consider a 20 channel system with linear demand function $\lambda_s(u) = (10 - u)_+$, where $(\ldots)_+ \triangleq \max(\ldots, 0)$. The PU arrival rate is $\lambda_p = 12.5$ and the penalty for blocking a PU user is $K = 120$. The function $R_{max}(u)$, for this set up, is plotted in Figure 1. This specific function has two maximum points at $u = 7.91$ and at $u = 8.21$. The corresponding maximizing thresholds are $T = 12$ and $T = 13$, respectively. ■

Claim 4.2 is a rather undesirable property from an optimization point of view. Nevertheless, in the next section, we show that a local optimal price and threshold can be calculated as efficiently as if $R_{max}(u)$ were unimodal.

B. Measurement-based Threshold Pricing (MTP)

In this section, we describe the MTP algorithm and prove that it converges to a local maximum of $R_{max}(u)$. As expected, when the function is unimodal it converges to the global maximum. During our numerical studies, we observed that when $R_{max}(u)$ is multimodal, the average profits of local maximums are very close to each other, as observed in Fig. 1. Therefore, we do not expect significant profit loss when the algorithm converges to a local maximum rather than the global one.

While one can calculate the value of $R_{max}(u)$ for a given price with a single measurement window, the same is not true for its derivative which can be undefined at certain points (transition points from one $R_T(u)$ to another). Therefore, we base the `MTP` algorithm on the derivative-free *Fibonacci search* which was first introduced by Kiefer [20]. Fibonacci search is a sequential line search algorithm which maximizes a unimodal function. In every iteration, it makes a function evaluation. Together with the information from earlier evaluations, it reduces the minimum interval where the optimal point is known to lie. This interval is referred to as *interval of uncertainty*. Under the following criteria of optimality, Fibonacci search is optimal for searching the maximum of a unimodal function. If the number of function evaluations is fixed in advance, Fibonacci search finishes with the largest ratio of initial size of interval of uncertainty to its final size [20].

In our case, function evaluations, i.e., measurements are conducted for discrete values of price. Therefore, we utilize a discrete version of Fibonacci search (also known as lattice search)[21]. While Fibonacci search might fail to converge when the function is multimodal, `MTP` converges to a local maximum of $R_{max}(u)$. We manage this by taking advantage of the fact that $R_{max}(u)$ is the maximum of unimodal functions $R_T(u)$ for $1 \leq T \leq C$. Algorithm 1 provides a pseudo-code for `MTP` which we next explain.

In the i^{th} iteration, where $i \geq 0$, `MTP` attempts to maximize the unimodal function $R_{T_i^*}(u)$ the same way as Fibonacci search would do. Here, T_i^* represents the *active threshold* in iteration i which we calculate in the following manner. Let S be the set of prices for which measurements have been obtained so far, i.e., if $u \in S$, then we know the corresponding arrival rate $\lambda_s(u)$. For $u \in S$, we can then calculate $R_T(u)$ for all values of T and deduce the value of $R_{max}(u)$ as well. Let $u_i^* = \arg\max_{u \in S}(R_{max}(u))$ be the price which yields the maximum profit observed so far. Then, $T_i^* = \arg\max_{1 \leq T \leq C}(R_T(u_i^*))$ is the optimal threshold for the price u_i^* and $R_{T_i^*}(u_i^*) = R_{max}(u_i^*)$ is the maximum profit calculated so far. In every iteration, `MTP` makes measurements for a new test price. At the end of every iteration, the active threshold is updated according to these new measurements.

`MTP` chooses the new test price according to Fibonacci numbers. Fibonacci numbers are defined such that $F_k = F_{k-1} + F_{k-2}$ where $F_0 = 0$ and $F_1 = 1$. Let, $\hat{\mathbb{U}}^i$ be the interval of uncertainty in the i^{th} iteration. `MTP` requires that the initial interval of uncertainty $\hat{\mathbb{U}}^0$ contains exactly $F_m + 1$ prices where m is the smallest integer which satisfies $|\mathbb{U}| \leq F_m + 1$. Recall that $\mathbb{U}$ is the set of all possible prices. In order to comply with this condition, we insert $F_m + 1 - |\mathbb{U}|$ fictitious prices to the end of the price series. We assume that the fictitious prices are all equal to u_{max}.

Algorithm 1 Measurement-based Threshold Pricing (`MTP`)

Calculate m and construct $\hat{\mathbb{U}}_0$
$u_a^0 \leftarrow u_{F_{m-2}}^0$
$u_b^0 \leftarrow u_{F_{m-1}}^0$
Make measurements for u_a^0 and u_b^0
$u_0^* = \arg\max_{u \in S}(R_{max}(u))$
$T_0^* = \arg\max_{1 \leq T \leq C}(R_T(u_0^*))$
for $i = 0$ to $m - 4$ **do**
  if $R_{T_i^*}(u_a^i) \geq R_{T_i^*}(u_b^i)$ **then**
    $\hat{\mathbb{U}}^{i+1} = [u_0^i, u_b^i]$
    Make measurements for u_a^{i+1}
  else
    $\hat{\mathbb{U}}^{i+1} = [u_a^i, u_{F_{m-i}}^i]$
    Make measurements for u_b^{i+1}
  end if
  $u_{i+1}^* = \arg\max_{u \in S}(R_{max}(u))$
  $T_{i+1}^* = \arg\max_{1 \leq T \leq C}(R_T(u_{i+1}^*))$
end for
return u_{m-3}^* and T_{m-3}^*

Let u_j^i represent the j^{th} price in $\hat{\mathbb{U}}^i$, where $j \geq 0$. Then, $\hat{\mathbb{U}}^0 = \{u_0^0, u_1^0, u_2^0, .., u_{F_m}^0\}$ which naturally contains all local optima of $R_{max}(u)$. In every iteration, the size of the interval of uncertainty is reduced such that $|\hat{\mathbb{U}}^i| = F_{m-i} + 1$ i.e., $\hat{\mathbb{U}}^i = \{u_0^i, u_1^i, u_2^i, ..., u_{F_{m-i}}^i\} = [u_0^i, u_{F_{m-i}}^i]$. `MTP` reduces $\hat{\mathbb{U}}^i$ by comparing $R_{T_i^*}(u)$ for two internal test prices, $u_{F_{m-i-2}}^i$ and $u_{F_{m-i-1}}^i$. For the sake of simpler notation, we denote these prices as u_a^i and u_b^i, respectively.

The algorithm starts with an initialization step in which m is calculated and $\hat{\mathbb{U}}^0$ is constructed. Then, measurements for u_a^0 and u_b^0 are obtained. The initialization step ends with the calculation of u_0^* and T_0^*.

Each iteration starts by comparing the value of $R_{T_i^*}(u_a^i)$ and $R_{T_i^*}(u_b^i)$. If $R_{T_i^*}(u_a^i) \geq R_{T_i^*}(u_b^i)$, then we have $R_{T_i^*}(u_a^i) \geq R_{T_i^*}(u_b^i) \geq R_{T_i^*}(u_{F_{m-i}}^i)$. Since $R_{T_i^*}(u)$ is a unimodal function, the optimal price for $R_{T_i^*}(u)$ can not be in $[u_{b+1}^i, u_{F_{m-i}}^i]$. Therefore, the interval of uncertainty is reduced to $\hat{\mathbb{U}}^{i+1} = [u_0^i, u_b^i]$. For the next iteration we need $u_a^{i+1} \in S$ and $u_b^{i+1} \in S$. Since $u_b^{i+1} = u_a^i$ (i.e., $u_{F_{m-i-1}}^{i+1} = u_{F_{m-i-2}}^i$), we only need to make measurements for u_a^{i+1}.

If $R_{T_i^*}(u_a^i) < R_{T_i^*}(u_b^i)$, we have $R_{T_i^*}(u_0^i) \leq R_{T_i^*}(u_a^i) < R_{T_i^*}(u_b^i)$. Due to similar arguments to those in the previous case, the interval of uncertainty is reduced to $\hat{\mathbb{U}}^{i+1} = [u_a^i, u_{F_{m-i}}^i]$. In this case $u_a^{i+1} = u_b^i$ and we make a measurements for u_b^{i+1}.

At the end of each iteration, u_{i+1}^* and T_{i+1}^* are updated for the use of next iteration. The algorithm terminates after $m-3$ iterations (when $i = m - 4$), and returns u_{m-3}^* and T_{m-3}^*, which are local optimal price and threshold of $R_{max}(u)$, as proven next.

We start with the following lemma.

Lemma 4.3: When the active threshold is changed to $T_i^* \neq T_{i-1}^*$, the optimal price for $R_{T_i^*}(u)$ is in $\hat{\mathbb{U}}^i$.

Proof: Assume that the active threshold is changed to T_i^* due to the measurements for price u_a^i. This means, $R_{T_i^*}(u_a^i)$

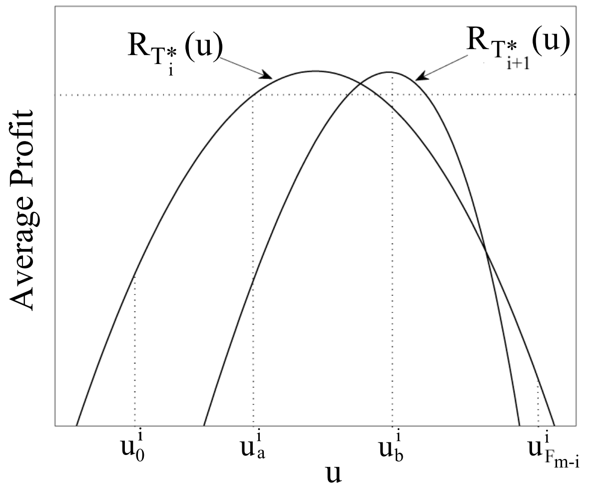


Fig. 2. Sample iteration of MTP where measurements are made for u_b^i and the interval of uncertainty is reduced from $[u_0^i, u_{F_{m-i}}^i]$ to $[u_a^i, u_{F_{m-i}}^i]$

is the maximum profit calculated so far, i.e., $R_{T_i^*}(u_a^i) > R_{T_i^*}(u_0^i)$ and $R_{T_i^*}(u_a^i) > R_{T_i^*}(u_{F_{m-i}}^i)$ Since $u_0^i < u_a^i < u_{F_{m-i}}^i$ and $R_{T_i^*}(u)$ is unimodal, the optimal price for $R_{T_i^*}(u)$ must be in $\hat{\mathbb{U}}^i$. Same arguments are true if the measurements had been conducted for u_b^i. ∎

Theorem 4.4: MTP converges to a local optimum of $R_{max}(u)$ in $m-3$ iterations and requires $m-1$ measurement windows, where $m = \min_k\{k : |\mathbb{U}| \leq F_k + 1\}$.

Proof: We first prove the first part of the theorem. In the last iteration $i = m-4$, the interval of uncertainty is reduced to $\hat{\mathbb{U}}^{m-3}$ which contains $|\hat{\mathbb{U}}^{m-3}| = 4$ different prices, and measurements are conducted for the only price in $\hat{\mathbb{U}}^{m-3}$ which has not been yet tested (either u_a^{m-3} or u_b^{m-3}). Finally, u_{m-3}^* and T_{m-3}^* are calculated. Even though T_{m-3}^* could be different from T_{m-4}^* (the last active threshold), u_{m-3}^* is the optimal price of $R_{T_{m-3}^*}(u)$ due to Lemma 4.3 and the fact that u_{m-3}^* is the best performing price in $\hat{\mathbb{U}}^{m-3}$. u_{m-3}^* is a local optimum of $R_{max}(u)$ because it is the optimal price of $R_{T_{m-3}^*}(u)$ and T_{m-3}^* is the optimal threshold for u_{m-3}^*.

As for the second part of the theorem, MTP makes new measurements in every iteration. Together with the initial two, the algorithm requires $m-1$ measurement windows. ∎

In Fig. 2, we give a graphical representation of a sample MTP iteration. In this example, the initial active threshold is T_i^*, the interval of uncertainty is $\hat{\mathbb{U}}_i = [u_0^i, u_{F_{m-i}}^i]$ and measurements are made for u_b^i. As a result of these measurements, the active threshold will be changed to T_{i+1}^* ($T_i^* \neq T_{i+1}^*$) because $R_{T_{i+1}^*}(u_b^i)$ is the maximum profit calculated so far. Therefore, the interval of uncertainty is reduced according to $R_{T_{i+1}^*}(u)$ to $\hat{\mathbb{U}}_{i+1} = [u_a^i, u_{F_{m-i}}^i]$.

In conclusion to this section, we note that the number of measurement windows required by MTP is the same as in the Fibonacci search which is $\log_\phi(|\mathbb{U}|) + O(1)$ where $\phi = (1+\sqrt{5})/2$ is the golden ratio [22]. MTP can easily be adapted to converge to the global maximum of $R_{max}(u)$. To do so, the threshold should be fixed throughout the MTP algorithm. This should be repeated for all possible thresholds $1 \leq T \leq C$. However, in this case, the algorithm would require $C(\log_\phi(|\mathbb{U}|) + O(1))$ measurement windows instead.

Algorithm 2 Adaptive Measurement-Based Threshold Policy (AMTP)

```
ũ ← u* − γΔu
Make measurements for ũ
if R_max(ũ) > R_max(u*) then
    Set direction of exploration d = −1
else
    ũ ← u* + γΔu
    Make measurements for ũ
    Set direction of exploration d = 1
end if
while R_max(ũ − dγΔ) ≤ R_max(ũ) do
    ũ ← ũ + dγΔu
    Make measurements for ũ
end while
Apply MTP on the interval [min(ũ, ũ − 2dγΔu), max(ũ, ũ − 2dγΔu)]
return u* and T*
```

V. Adaptive Measurement-based Threshold Policy (AMTP)

In this section, we demonstrate that the MTP algorithm can be used to adapt the operating price and threshold when the arrival rate of PUs and the demand function of SUs change over time. We refer to the extended algorithm as Adaptive Measurement-based Threshold Policy (AMTP). This algorithm is invoked periodically or when a significant change in PU or SU arrival rate is detected. As in Section IV, in order to prove convergence, we assume that measurements of the arrival rates are exact.

One straightforward way to adapt MTP is to restart it on the entire interval of prices $[0, u_{max}]$ whenever needed. However, the profit loss due to testing prices that are far away from the optimal one is highly undesirable. We address this issue by combining *hill climbing* [23] and MTP. By using hill climbing, AMTP first searches for an interval (typically much smaller than $[0, u_{max}]$) that contains a local maximum of $R_{max}(u)$. Then, it runs MTP on this interval in order to converge to the local maximum. Algorithm 2 provides a pseudo-code of AMTP.

Let u^* be the operating price before AMTP starts and $\tilde{u}$ be the price that AMTP is exploring in order to find an interval of prices that contains a local maximum of $R_{max}(u)$. Recall that Δu is the price granularity. AMTP starts by making exploratory measurements for $\tilde{u} = u^* - \gamma\Delta u$ where $\gamma \in \mathbb{Z}^+$. The value of γ determines how aggressively the algorithm explores new prices. Large γ means more exploration while small γ means more exploitation since the operating price stays around u^*.

If $R_{max}(u^* - \gamma\Delta u) > R_{max}(u^*)$ then AMTP sets the direction of exploration to $d = -1$ i.e., exploration is performed by decreasing the operating price in increments of $\gamma\Delta u$. If $R_{max}(u^* - \gamma\Delta u) \leq R_{max}(u^*)$ then the direction of exploration is set to $d = 1$ i.e., exploration is performed by increasing the operating price in increments of $\gamma\Delta u$. The exploration phase ends when the new operating price $\tilde{u}$ does not provide further improvement over the previous price $\tilde{u} - d\gamma\Delta u$ (i.e., $R_{max}(\tilde{u} - d\gamma\Delta u) > R_{max}(\tilde{u})$). When the exploration phase ends, MTP is applied on the interval $[\min(\tilde{u}, \tilde{u} - 2d\gamma\Delta u), \max(\tilde{u}, \tilde{u} - 2d\gamma\Delta u)]$, and it returns updated values for u^* and T^*, which are local optimal price and threshold of $R_{max}(u)$.

In the next lemma, we prove the convergence of AMTP.

Lemma 5.1: AMTP converges to a local maximum of $R_{max}(u)$.

Proof: When the exploration phase ends, the prices $\tilde{u} - 2d\gamma\Delta u$, $\tilde{u} - d\gamma\Delta u$ and $\tilde{u}$ form a three point pattern i.e., $R_{max}(\tilde{u}) \leq R_{max}(\tilde{u} - d\gamma\Delta u)$ and $R_{max}(\tilde{u} - 2d\gamma\Delta u) < R_{max}(\tilde{u} - d\gamma\Delta u)$. Therefore, there must be at least one local maximum of $R_{max}(u)$ in the interval $[\min(\tilde{u}, \tilde{u} - 2d\gamma\Delta u), \max(\tilde{u}, \tilde{u} - 2d\gamma\Delta u)]$. When we apply MTP on this interval, it returns a local maximum of $R_{max}(u)$ from Theorem 4.4. ■

Next, we provide two heuristic methods to improve the performance of AMTP. Even though these practices are not required for the convergence of AMTP, they significantly increase the average online profit during the exploration phase of AMTP. First, we numerically observed that prices which are close to each other tend to have the same optimal threshold. Thus, when conducting measurements with AMTP, we set the current threshold to the optimal threshold of the previously explored price. Second, when we encounter negative profit at the end of a measurement window, we stop admitting SUs while keep observing their arrival rate according to the operating price set by AMTP. We start re-admitting SUs once we detect PU and SU arrival rates for which positive profit is possible. This method prevents the system to be stuck in long negative profit streaks, and thus increases the overall profit.

VI. Numerical results

A. Pricing with unknown demand function

In this section, we evaluate the performance of MTP with finite measurement windows. The arrival rate of SUs $\lambda_s(u)$ is estimated by dividing the total number of SU arrivals within a measurement window by the length (in time) of that window. Then, the value of $R_{max}(u)$ is computed using Eqs. (3) and (4).

We consider the interval $\mathbb{U} = [0, 10]$ as the set of available prices where discrete prices are $\Delta u = 10^{-3}$ monetary units apart. Therefore, the total number of possible prices is $|\mathbb{U}| = 10001$. It can be verified that, in this case, $m = 21$ because $F_{21} = 10946$. It means that MTP terminates in $m - 3 = 18$ iterations and uses $m - 1 = 20$ measurement windows.

In Figures 3 and 4, we show the average profit ($R_T(u)$) corresponding to the current test price and the current active threshold T_i^*. We run the algorithm 100 times and take sample average of the average profits. We also display 95% confidence interval for these samples for each measurement window. Recall that we define the mean call length to be one time unit, which is typically in the order of a few minutes [18].

In Fig. 3, we assume that SUs request access to the spectrum with a rate of $\lambda_s(u) = 10(\frac{10-u}{10})^{1/2}$ calls per time unit, which is unknown to MTP. PUs arrive with a known constant rate of $\lambda_p = 8$ calls per time unit. We use a 10 time units measurement window. The figure shows that MTP converges within 4% of the optimal threshold policy and within 5% of the optimal occupancy-based policy after 5 iterations. Note that, at some iterations, MTP selects test prices such that the average profit drops, as seen in the fourth iteration of Fig. 3.

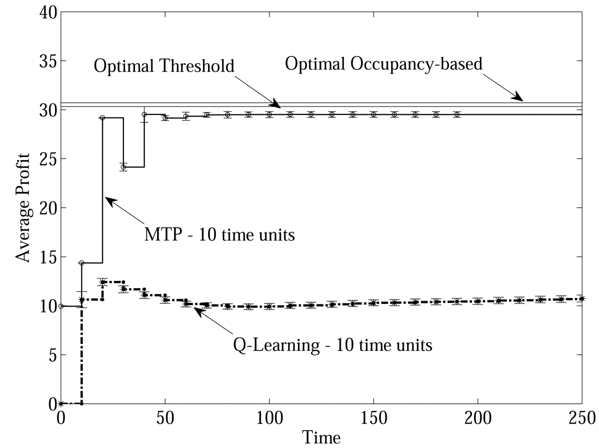


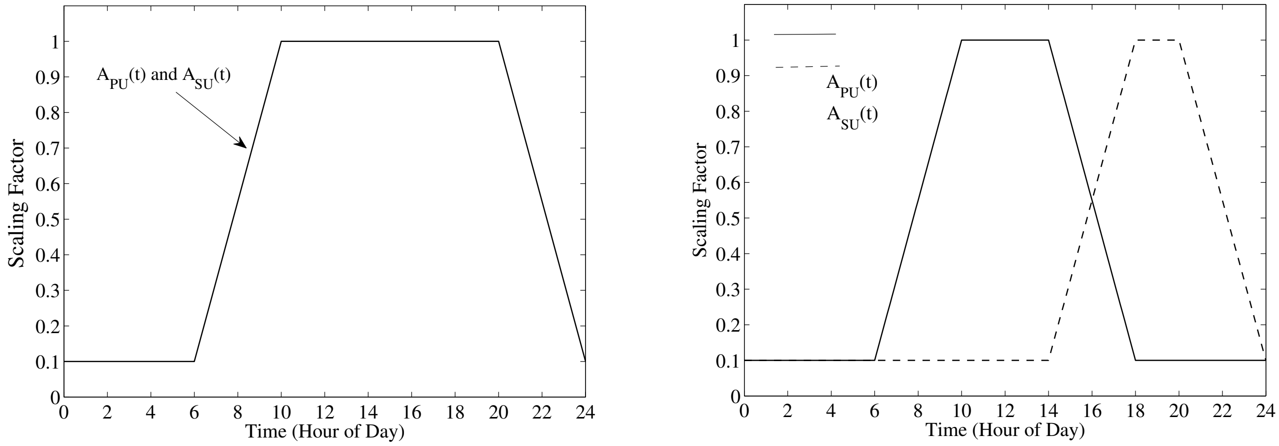
Fig. 3. Performance comparison of MTP and Q-Learning with a measurement window of 10 time units, $\lambda_s(u) = 10(\frac{(10-u)_+}{10})^{1/2}$, $\lambda_p = 8$ calls per time unit, $K = 100$, and $C = 20$. Average of 100 runs with 95% confidence interval. $\Delta u = 10^{-3}$, $\mathbb{U} = [0, 10]$ and $|\mathbb{U}| = 10001$. As a reference, the figure also shows the average profit achieved with the optimal occupancy-based policy and the optimal threshold policy, both of which require knowledge of the demand function.

In Fig. 4, we run MTP for the linear demand function $\lambda_s(u) = (10 - u)_+$ and different measurement windows (note the logarithmic scale of the time axis). We observe that for 1 time unit measurement window, MTP comes within 10% neighborhood of optimal pricing in just 5 time units. Naturally, the longer the measurement window, the closer MTP comes to the optimal threshold policy.

B. Comparison between MTP and Q-Learning

In this section, we compare the performance of MTP to that of Q-Learning. Q-Learning is a powerful tool to compute the optimal policy of a Markov decision process without knowing its transition probabilities. Through measurements, Q-Learning aims to find the optimal action for each state. In our case, a state corresponds to a channel occupancy level and an action corresponds to a price.

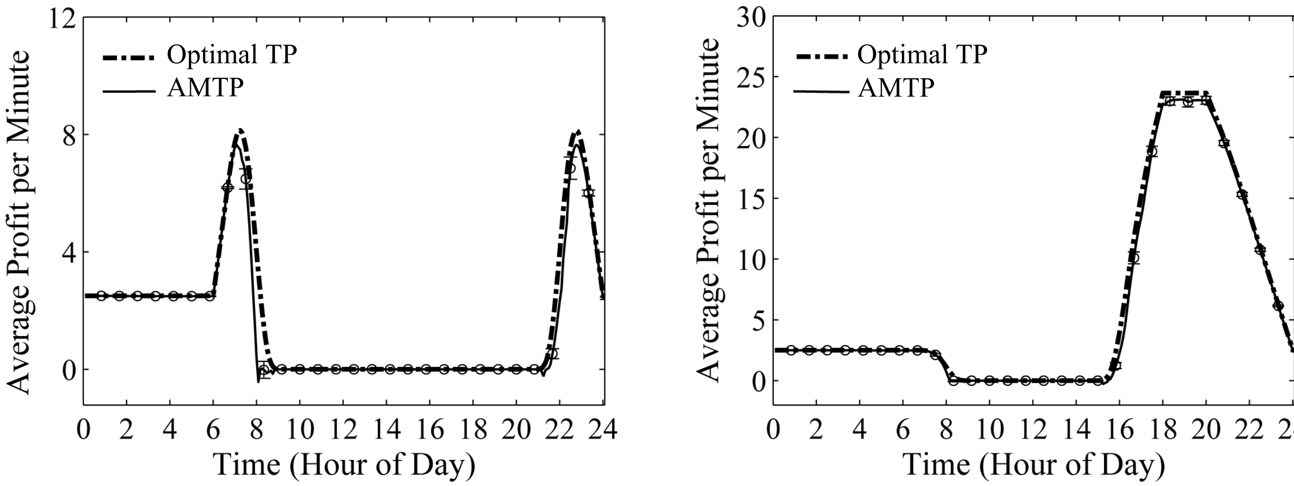
We implement an average reward Q-Learning algorithm using value iteration, following the procedure described in [17, Section 3.2]. Each possible state-price pair has a corresponding Q-value. All the Q-values are initialized to zero and prices are chosen uniformly at random at the beginning. The algorithm is run for 9500 iterations, where an iteration is defined as a state transition. When the algorithm terminates after the maximum number of iterations, the collection of prices that yield the maximum Q-value at each state is considered to be the desired price vector. The rate of convergence is mainly



(a) PU and SU arrival rates change similarly.

(b) PU and SU arrival rates change differently.

Fig. 5. Time of day change.



(a) PU and SU arrival rates vary similarly.

(b) PU and SU arrival rates vary differently.

Fig. 6. Average profit per minute of AMTP (re-started every hour). 100 runs with 95% confidence interval. System parameters: Measurement window is 5 mins, $\gamma = 10$, $C = 20$, $K = 100$, $\lambda_p = 8$, $\lambda_s(u) = (10 - u)_+$, $\Delta u = 10^{-2}$,$\mathbb{U} = [0, 10]$ and $|\mathbb{U}| = 1001$.

related to the step-size used in the algorithm. According to [24], the step-size rule $\mu_k = \frac{a}{b+k}$ gives the best convergence rate among numerous candidates. Here, μ_k represents the step-size in iteration k of the algorithm. In our simulations, we set $a = 5000$ and $b = 10000$.

Simulations are run for different number of channels C. For small values of C, e.g., one or two channels, Q-Learning achieves an average profit that is close to the profit of the optimal occupancy-based policy. However, as C increases, the performance of Q-Learning degrades due to its slow convergence speed. Thus, for $C = 20$ channels, Fig. 3 illustrates that the pricing policy obtained using Q-Learning loses about 65% of the profit achievable with the optimal occupancy-based policy. For the same system parameters and the same simulation time interval, MTP achieves much higher profit than Q-Learning.

C. *Pricing with time-varying user demand*

In this section, we simulate AMTP over the course of a day. We assume that one time unit (i.e., the average call length) is 2.5 minutes, which is compatible with the observations of [18]. We use 5 minutes measurement windows. Possible discrete prices are $\Delta u = 10^{-2}$ monetary units apart in the interval $[0, 10]$. The exploration parameter of AMTP is $\gamma = 10$.

We consider PU and SU arrival rates which scale with time dependent scaling factors $A_{PU}(t)$ and $A_{SU}(t)$, respectively. The resulting arrival rates are $A_{PU}(t)\lambda_p$ and $A_{SU}(t)\lambda_s(u)$. We consider two scenarios: In the first scenario, SU and PU arrival rates change similarly (i.e., $A_{PU}(t) = A_{SU}(t)$), peaking between 10AM and 8PM, as seen in Figure 5(a). At this time of the day, the trend is a piecewise linear representation of the real data traces observed in [18]. Second, we consider a hypothetical scenario, shown in Figure 5(b), that

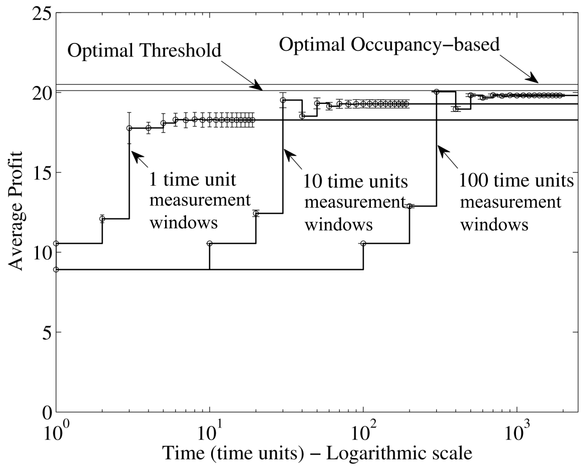


Fig. 4. Performance of MTP with measurement windows 1,10 and 100 time units, $\lambda_s(u) = (10 - u)_+$, $\lambda_p = 8$, $K = 100$, and $C = 20$. Average of 100 runs with 95% confidence interval. $\Delta u = 10^{-3}$, $\mathbb{U} = [0, 10]$ and $|\mathbb{U}| = 10001$.

may be more desirable from the point of view of secondary spectrum access; PU and SU arrival rates have opposite trends. In this scenario, PUs could represent business applications that peak during the business hours and SUs could represent personal applications that peak during the evening hours.

In Figure 6 we compare the performance of AMTP to that of the optimal threshold policy (which knows the demand function in advance), assuming a linear demand function. In these examples, we assume that, at the beginning (midnight), the system is calibrated such that the operating price and threshold are optimal. This is a reasonable assumption since [18] observes that the nighttime arrival rates are relatively stable and does not change significantly between the days of the week. This makes predicting the arrival rates and calibrating AMTP relatively easy. We periodically restart AMTP every hour. In the figures, we take a snapshot of the system every 5 minutes and look at the current operating price and threshold of the optimal threshold policy and AMTP. The dotted line shows the average profit of the optimal threshold policy according to current PU and SU arrival rates. The solid line shows the average profit of the AMTP with the same arrival rates.

To give a perspective on how much profit might be lost due to adaptation, we consider the example in Figure 6(a). In this example, the optimal threshold policy obtains 2504 monetary units in a day, while AMTP obtains 2227 monetary units. For the example in Figure 6(b), these values are 9096 and 8722, respectively (AMTP loses only 4% of the total profit in that case).

VII. Conclusion

In this paper, we investigated pricing of secondary spectrum access with unknown demand function. We proposed a new on-line algorithm, called Measurement-based Threshold Pricing (MTP). We proved that MTP provably converges to a local maximum of $R_{max}(u)$, the threshold policy profit function which depends on the price only. In addition, MTP returns the corresponding optimal threshold.

In our numerical studies, we observed that $R_{max}(u)$ is generally unimodal and whenever it is multimodal, the local maximum profits are very close to each other. Under mild assumptions on the demand function, we proved that despite the possible multimodality of $R_{max}(u)$, MTP converges to a local optimum with the same number of measurements as if $R_{max}(u)$ were unimodal, which is $\log_\phi(|\mathbb{U}|) + O(1)$. For example, with $|\mathbb{U}| = 10001$ different prices, MTP requires only 20 measurement windows.

The performance of MTP depends on the length of the measurement window. In our numerical result, we observed that with measurement windows of 10 time units, where one time unit corresponds to the average call length, MTP comes in only 5 iterations within 4% of the average profit of the optimal threshold policy, which knows the demand function. We also showed that MTP achieves much better performance than Q-Learning.This result is non-obvious since Q-Learning aims to mimic the optimal occupancy-based pricing policy, while MTP aims to mimic the optimal threshold pricing policy, which has slightly inferior performance. However, MTP needs to optimize only two parameters, compared to C parameters (i.e., a different price for each occupancy level) for Q-Learning.

Finally, we introduced an adaptive version of MTP, called AMTP. We provided numerical simulations of AMTP for realistic time-of-the-day arrival rate trends. The simulations showed that AMTP closely tracks the time-varying optimal price and threshold.

Future work could focus on the development of alternative methods for mimicking the optimal threshold policy. Such methods could accommodate statistical fluctuations due to finite measurement windows as an intrinsic part of their design. Similarly, future work could explore scenarios where the quality of different channels may vary and, as a result, the willingness of SUs to join them may differ.

In summary, this work provides robust and practical methods to manage secondary spectrum access. The demonstrated performance of these methods, under minimal assumptions on the underlying environment, shows promise to facilitate spectrum reforms in achieving their full potential.

References

[1] H. Mutlu, M. Alanyali, and D. Starobinski, "On-line pricing of secondary spectrum access with unknown demand function and call length distribution," in *Proceedings of IEEE INFOCOM*, 2010.

[2] "Promoting efficient use of spectrum through elimination of barriers to the development of secondary markets," http://hraunfoss.fcc.gov/edocs_public/attachmatch/FCC-04-167A1.pdf, September 2004, second Report and Order, Order on Reconsideration, and Second Further Notice of Proposed Rulemaking (FCC 04-167).

[3] "Meet the eBay of wireless spectrum. Spectrum Bridge automates, webifies wireless deals," http://www.networkworld.com/news/2008/091208-wireless-spectrum.html, September 12 2008, news Release.

[4] M. Buddhikot, "Understanding dynamic spectrum access: Models, taxonomy and challenges," in *New Frontiers in Dynamic Spectrum Access Networks (DySPAN), 2007 IEEE Symposium on*, April 2007.

[5] I. C. Paschalidis and J. N. Tsitsiklis, "Congestion-dependent pricing of network services," *IEEE/ACM Transactions on Networking*, vol. 8, no. 2, pp. 171–184, April 2000.

[6] I. C. Paschalidis and Y. Liu, “Pricing in multiservice loss networks: static pricing, asymptotic optimality, and demand substitution effects,” *IEEE/ACM Transactions on Networking*, vol. 10, no. 3, pp. 425–438, June 2002.
[7] X. Lin and N. B. Shroff, “Simplification of network dynamics in large systems,” *IEEE/ACM Transactions on Networking*, vol. 13, no. 4, pp. 813–826, 2005.
[8] H. Mutlu, M. Alanyali, and D. Starobinski, “Spot pricing of secondary spectrum access in wireless cellular networks,” *IEEE/ACM Transactions on Networking*, vol. 17, no. 6, pp. 1794–1804, 2009.
[9] N. Gans and S. Savin, “Pricing and capacity rationing for rentals with uncertain durations,” *Management Science*, vol. 53, no. 3, pp. 390–407, March 2007.
[10] A. Al-Daoud, M. Alanyali, and D. Starobinski, “Secondary pricing of spectrum in cellular CDMA networks,” in *2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, 2007, pp. 535–542.
[11] H. Mutlu, “Spot pricing of secondary access to wireless spectrum,” *Ph.D. Dissertation, Boston University*, 2010.
[12] A. O. Ercan, J. Lee, S. Pollin, and J. M. Rabaey, “A revenue enhancing stackelberg game for owners in opportunistic spectrum access,” in *IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, 2008, pp. 1–8.
[13] E. Campos-Nanez and S. Patek, “On-line tuning of prices for network services,” in *Proceedings of IEEE INFOCOM*, 2003, pp. 1231–1241.
[14] O. Besbes and A. Zeevi, “Dynamic pricing without knowing the demand function: Risk bounds and near optimal algorithms,” *Operations Research*, Accepted for publication, 2008.
[15] M. Alanyali, A. Al Daoud, and D. Starobinski, “Profitability of dynamic spectrum provision for secondary use,” in *New Frontiers in Dynamic Spectrum Access Networks (DySPAN), 2011 IEEE Symposium on*, May 2011, pp. 136 –145.
[16] D. Bertsekas, *Dynamic Programming and Optimal Control*. Athena Scientific, 2005.
[17] A. Gosavi, “Reinforcement learning: A tutorial survey and recent advances,” *INFORMS Journal on Computing*, vol. 21, no. 2, pp. 178–192, 2009.
[18] J. B. D. Willkomm, S. Machiraju and A. Wolisz, “Primary users in cellular networks: A large-scale measurement study,” in *IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, 2008, pp. 1–11.
[19] S. Ziya, H. Ayhan, and R. Foley, “Relationships among three assumptions in revenue management,” *Operations Research*, vol. 52, no. 5, pp. 804–809, September-October 2004.
[20] J. Kiefer, “Sequential minimax search for a maximum,” *Proceedings of the American Mathematical Society*, vol. 4, no. 3, pp. 502–506, 1953.
[21] D. Wilde, *Optimum seeking methods*. Prentice-Hall, 1964.
[22] E. D. Demaine and S. Langerman, “Optimizing a 2d function satisfying unimodality properties,” *Lecture Notes in Computer Science, Algorithms ESA*, vol. 3669, pp. 887–898, 2005.
[23] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Prentice Hall, 2009.
[24] A. Gosavi, “On step sizes, stochastic shortest paths, and survival probabilities in reinforcement learning,” in *Winter Simulation Conference*, 2008, pp. 525–531.

Huseyin Mutlu received his B.S. degree from Middle East Technical University, Ankara, Turkey in 2003 and M.S. degree from Northeastern University, Boston, MA in 2005 both in electrical engineering. He received his Ph.D. degree from Boston University, Electrical and Computer Engineering Department in 2010.

His research interests are primarily in pricing and management of wireless communication systems.

Murat Alanyali received his Ph.D. degree in Electrical and Computer Engineering from the University of Illinois at Urbana-Champaign in 1996. He held positions at Bell Laboratories, Holmdel, NJ, during 1996-1997, and at the Department of Electrical and Electronics Engineering at Bilkent University, Ankara, Turkey, from 1998 to 2002. He is presently an Associate Professor of Electrical and Computer Engineering at Boston University. His research interests are in communication networks and distributed algorithms. Dr. Alanyali is a recipient of the National Science Foundation (NSF) CAREER award in 2003.

David Starobinski is Professor of Electrical and Computer Engineering at Boston University, with a joint appointment in the Division of Systems Engineering. He received his Ph.D. in Electrical Engineering from the Technion-Israel Institute of Technology, in 1999. In 1999-2000, he was a visiting post-doctoral researcher in the EECS department at UC Berkeley. In 2007-2008, he was on sabbatical at EPFL (Switzerland), where he was an invited Professor.

Dr. Starobinski received a CAREER award from the U.S. National Science Foundation (2002), an Early Career Principal Investigator (ECPI) award from the U.S. Department of Energy (2004), the best paper award at the WiOpt 2010 conference, and the 2010 BU ECE Faculty Teaching Award. He is currently an Associate Editor of the IEEE/ACM Transactions on Networking. His research interests are in the modeling, performance evaluation, and security analysis of communication networks.

Aylin Turhan received her B.S. degree in Electrical and Electronics Engineering from Middle East Technical University, Ankara, Turkey in 2010 and her M.S. degree in Computer Engineering from Boston University, Boston, MA in 2012.

Her research focuses mainly on optimal control and pricing in wireless communication networks. She is a student member of IEEE, IEEE Communication Society and IEEE Women in Engineering.