Cognitive Radio Networks: Analysis of a Paid-Sharing Approach based on Admission Control Decisions

Claudina Rattaro · Pablo Belzarena

Abstract Cognitive Radio Networks have emerged in the last decade as a solution for two problems: spectrum underutilization and spectrum scarcity. The main idea is to manage the radio spectrum more efficiently, where secondary users (SUs) are allowed to exploit the spectrum holes in primary user's (PUs) frequency bands. We consider a paid-sharing approach where SUs pay for spectrum utilization. A challenging aspect in these mechanisms is how to proceed when a PU needs certain amount of bandwidth and the free capacity is insufficient. We assume a preemptive system where PUs have strict priority over SUs; when a PU arrives to the system and there are not enough free channels to accommodate the new user, one or more SUs will be deallocated. The affected SUs will then be reimbursed, implying some cost for the PUs service provider (SP). This paper bears on the design and analysis of an optimal SU admission control policy; i.e. that maximizes the long-run profit of the SP. We model the optimal revenue problem as a Markov Decision Process and we use dynamic programming and further techniques such as sample-path analysis to characterize properties of the optimal admission control policy. We introduce different changes to one of the best known dynamic programming algorithms incorporating the knowledge of the characterization. In particular, those proposals accelerate the rate of convergence of the algorithm when is applied in the considered context. Our results are validated through numerical examples.

Keywords cognitive radio networks · markov decision process · policy iteration algorithm · modified policy iteration algorithm · dynamic spectrum allocation

C. Rattaro, P. Belzarena
Instituto de Ingeniería Eléctrica, Facultad de Ingeniería, Universidad de la República
Julio Herrera y Reissig 565, ZC 11300
Montevideo, Uruguay
E-mail: {crattaro,belza}@fing.edu.uy

1 Introduction

Nowadays, with the rapid development of wireless communications, the demand for spectrum has been growing dramatically resulting in the spectrum scarcity problem. In spite of this, spectrum utilization measurements have shown that licensed bands are vastly underutilized while unlicensed bands are too crowded [2,20,9,5,12]. Cognitive Radio (CR) has been proposed as a promising technology to solve these problems by an intelligent and efficient dynamic spectrum access [22,16,35]. In the last decade, CR has attracted a lot of attention from both industry (some examples are IEEE 802.22 and 802.11af standards [3,4]) and academy (see for instance [15,11,41,43,31] and the references therein). Moreover, regulators have taken note about this fact and some proposals already exist to regulate the dynamic spectrum allocation [1].

In this paradigm we can identify two classes of users: primary and secondary. Primary users (PUs) are those for which a certain portion of the spectrum has been allocated to (often in the form of contractual quality of service (QoS) guarantees). Secondary users (SUs) are devices which are capable of detecting unused licensed bands and adapt their transmission parameters. The main idea in CRs, in order to improve the spectrum utilization, is that SUs use the licensed resource in the absence of PUs. This dynamic spectrum allocation (DSA), although a rather large number of solutions already exist in the literature, is still one of the main challenges in the design of CR due to the requirement of "peaceful" coexistence of both types of users [10]. There are roughly two different approaches for dynamic spectrum sharing: paid-sharing or free-sharing [28,38,21,7].

Free-sharing spectrum is also known as opportunistic spectrum access. In this context, a SU has to monitor licensed bands and opportunistically transmit (without payments) whenever no primary signal is detected. There is no motivation for PUs to participate in this process because they do not obtain any benefit from it. However, it could be an impulse for regulators in order to improve the spectrum usage. On the other hand, another approach is considering paid-sharing models, where unoccupied channels are allocated to SUs which have to pay to the PU's Service Provider (SP) for the spectrum utilization. The paid-sharing models have drawn more attention recently because they can provide an economic incentive to the SP and/or to its PUs. PU's charge and SU's payments are vital to the PUs-SUs interactions: PUs are motivated to share their own underutilized licensed resource and SUs are persuaded to wisely use the PU's spectrum avoiding an aggressive usage. While there are many problems yet to be solved in this area, one of the most important is how to stimulate the spectrum sharing behavior of PUs. In this context, paid-sharing methods seem to be the most suitable. We thus focus on this kind of mechanisms.

There are different proposals for implementing paid spectrum sharing techniques. These can be grouped into three big classes: dynamic spectrum auctions [42,38,36], pricing [23,44] and admission control mechanisms [33,39,18]. In general, the aim of all the approaches is to maximize the revenue of the pri-

mary SP. While much research has been recently dedicated to paid spectrum sharing methods, most of the works have mainly focused on auction processes based on conventional spectrum auctions, which is not always consistent with the original intention of dynamic spectrum sharing. In this context admission control policies might play a key role. There is no doubt that admission control strategies and pricing rules are strongly related. Prices, SU's demand and admission control boundaries are correlated variables because SU's behavior is sensitive to spectrum utility (costs and benefits) and QoS guarantees. Another challenge in these mechanisms is what to do when a PU needs a channel and there are not enough free channels to satisfy its demand. Some works permit SU's interference over PU's communications and in this context SUs pay the corresponding interference cost to the PUs [40,37]. Other studies assume a channel reservation scheme only for PUs [36]. On the other hand, fewer works contemplate preemptive situations [33] implying a preemption cost for the primary SP. This last alternative of termination model is already defined by FCC in a different scenario (Block D at 700 MHz), therefore, it would represent a natural way to implement that situation.

The focus of our analysis is a paid spectrum sharing method based on admission control decisions over SUs. In particular, we consider a scenario without spatial reuse of channels where if a PU arrives and does not find enough free channels in the system, at least one of the SUs will be deallocated immediately. In other words, a preemptive system is considered where some SUs communications will be aborted whenever a PU needs certain amount of bandwidth and the system has an insufficient number of free bands (free interference between PUs and SUs is considered). In addition to that, the affected SUs (the ones that are deallocated before their services are finished) will be reimbursed. We consider static prices similar to the ones considered in previous articles [33,18,17] and we address the problem of maximizing the total expected discounted revenue of the SP over an infinite horizon. That is to say, the goal is to find the optimal admission control policy for SUs in order to maximize the SP profit. In this work we concentrate on analyzing and characterizing the optimal admission control policies, and also on exploiting their properties to design computational procedures for efficiently finding the parameters of such policies.

To this end, we model the optimal revenue problem as a Markov Decision Process (MDP) where the arrivals of each class are independent Poisson processes with rate λ_1 for PUs and λ_2 for SUs. Besides, the service durations are independent and exponentially distributed with mean μ_1^{-1} and μ_2^{-1} respectively. We also consider that each class of user demands b_i resources, with b_i integer (i.e. a PU requests b_1 channel bands and a SU requests b_2). This problem formulation is much more general than a cognitive radio network analysis, since it can also model many other economic scenarios of dynamic control of queueing systems which consider two different classes of users (or services), one with strict priority, and contemplate preemptive situations with reimbursement, admission control decisions and multi-resource allocation. To

the best of our knowledge, this general admission control problem has not been deeply explored yet in the literature.

In a nutshell the main contribution of this paper is the analysis and characterization of this general asymmetric queueing system and its application to the particular context of cognitive radio networks where PUs and SUs represent the two classes. We use dynamic programming and other techniques such as sample-path to make a complete analysis of the properties of the optimal admission control policies. Not only is the analysis the main contribution, but also we propose many alternatives to improve the dynamic programming algorithms performance based on the system characterization. The proposed alternatives decrease the computational complexity as well as the large running time introduced by methods like value iteration and policy iteration (commonly used to solve MDP problems). In particular, as one of the results of this work, we have obtained a customized version of Modified Policy Iteration Algorithm (MPI [26]) which combines faster calculation with a robust performance. Unlike previous works that study similar models of DSA, the analysis and characterization proposed in this paper are based on more realistic hypotheses.

There are some previous works which contribute in this direction, being the most representative examples [33,34]. In particular, in [33] the authors assume that call durations are exponentially distributed with identical means for both classes. This hypothesis strongly simplifies the problem (it can be analyzed as a 1-D Markov Chain) and does not represent the most common real scenarios. The vast majority of the CR related articles consider different primary and secondary services (e.g. Internet access provided using White Space frequency bands [13]), then the natural situation is to model them with different arrival and service rates as we do in this work. Although in [34] the authors consider preemptive situations and different service rates, they study a symmetric problem where all users (of both classes) have to pay for the spectrum utilization and the service abrupt termination can occur in both classes. Due to these characteristics, their analysis is not totally applicable to our context. On the other hand, an important hypothesis used in the cited articles is to consider that all users (PUs and SUs) request the same amount of bandwidth. Even more, they assume "one user with one channel" representing a particular scenario of the problem. This hypothesis implies that the preemption only occurs when the system is full of users. More general, in our scenario the preemption occurs when there are not enough free resources to satisfy primary demand. Finally, but not less important, our work differs from [33,34] in the payment mechanism. In the preceding works, the reward per user accepted in the system is collected after the SU leaves the system with successful completion of service implying that a SU could earn money without paying anything. In our case, the reward is collected at the moment when the user is accepted in the system (independent if it successfully completes its service or it is deallocated). Even though our formulation is more complex, we consider it more realistic.

The rest of the paper is structured as follows. In section 2 we describe our economic model of dynamic spectrum sharing in CR networks. We introduce

most of the notation used in the paper, in particular, we define the Markov Decision Process. The paper continues in section 3 where we explain our analysis and characterization of the optimal admission policies. In section 4 we propose changes to the Modified Policy Iteration algorithm in order to significantly decrease its running time. In section 5 we include some numerical examples that validate our results. Finally, we conclude and discuss future work in section 6.

2 Model Description

Let us begin by describing our working scenario and introducing the notation, definitions and hypotheses. We assume that the number of resources to be distributed between PUs and SUs is limited then, let us note as C the total number of identical channels. Let $x(t)$ and $y(t)$ be the number of PUs and SUs in the system at time t respectively (in order to simplify the notation, we indistinctly use x and $x(t)$, as well as y and $y(t)$). Let λ_1 and μ_1 be the arrival and leaving rates for PUs respectively (independent Poisson arrivals and exponentially distributed service times). In the same way, λ_2 and μ_2 represent the arrival and leaving rates for SUs. We assume that each PU demands b_1 resources, and analogously each SU requires b_2 channel bands to its transmission. Even more general, we can consider different primary (and/or secondary) demands. It is important to highlight that our analysis is easily scalable to that general situation working with more than two classes of users.

We consider a paid spectrum sharing mechanism where SUs pay to the primary SP for the spectrum utilization. In particular, in order to concentrate the analysis in the explained hypotheses, we consider static prices. Let $R > 0$ be the reward collected for each band when a SU is allowed to exploit the PU's resource (i.e. $b_2 R$ is the collected reward when a SU is accepted). We also consider a preemptive system where PUs have strict priority over SUs. This means that SUs can be removed from the system if there is insufficient free capacity when a PU arrives. In this model, these affected SUs will be reimbursed with $b_2 K$ ($K > 0$), implying a punishment for the SP. We take into account a discount rate $\alpha > 0$, that is, the rewards and costs at time t are scaled by a factor $e^{-\alpha t}$. We work with a discounted model because it is the most analytically tractable and the most widely studied approach, however, our results can be extended to average reward problems. It is important to highlight that similar static pricing models have been used in previous works like [33, 18, 17].

We thus have a Continuous Time MDP (CTMDP) with state space $S = \{(x, y) : 0 \le b_1 x \le C, 0 \le b_2 y \le C, 0 \le b_1 x + b_2 y \le C\}$ where the objective is to maximize the total expected discounted profit over an infinite time horizon applying admission control decisions over SUs, i.e. we want to find the optimal policy π^* that defines the admission control action in each state $s \in S$ maximizing the SP's revenue ($\pi^* : S \to A_s$; A_s is the action space. $A_s = \{0, 1\}$ or $A_s = \{0\}$, depending on $s \in S$, where action 0 corresponds to refusing a SU's

arrival and 1 to admitting. It is easy to note that $A_s = \{0\}$ is the action space for all $s = (x, y) : b_1 x + b_2 y > C - b_2$).

Please note the scalability of our model. For instance if we considered different primary demands (e.g. b_1, b_1'), the MDP would have a greater dimensional state space (e.g. $S = \{(x, x', y) : 0 \le b_1 x \le C, 0 \le b_1' x' \le C, 0 \le b_2 y \le C, 0 \le b_1 x + b_1' x' + b_2 y \le C\}$, where x and x' would represent the number of PUs that use b_1 and b_1' bands respectively). The same idea if we considered different secondary bandwidth requirements. We will not deal with that generalization here, but our results can be easily extended for this general case.

According to the previous definitions, the transition rates $q((x, y), (x', y'))$ of the CTMDP are then:

- $q((x, y), (x + 1, y)) = \lambda_1$, if $b_1 x + b_2 y \le C - b_1$
- $q((x, y), (x - 1, y)) = \mu_1 x$
- $q((x, y), (x, y + 1)) = a(x, y)\lambda_2$, if $b_1 x + b_2 y \le C - b_2$
- $q((x, y), (x, y - 1)) = \mu_2 y$
- $q((x, y), (x + 1, y - z)) = \lambda_1$, if $C - b_1 < b_1 x + b_2 y \le C$ and $b_2 y \ge b_1$ (preemption)

where $a(x, y)$ represents the admission control decision in each state, $a(x, y) \in A_s$; and z represents the number of preempted SUs:

$$z = \left[\frac{b_1 x + b_2 y - C + b_1}{b_2} + \mathbb{1}_{\{(b_1 x + b_2 y - C + b_1)\%b_2 \ne 0\}} \right]. \tag{1}$$

In Eq. (1) $[x]$ represents the integer part of x, and $a\%b$ represents the remainder of a when divided by b. Please note that if the system is in state $s = (x, y)$ and a SU arrives, only if $a(x, y) = 1$ will it enter to the system.

Recapitulating, the objective is to find π^*, that is to say, the rule that maps each system state s to its optimal action a, in each decision epoch, maximizing SP's benefits. Using "uniformization" technique (see [26] and [6] for a survey) we can develop a model with constant transition rates and thus the algorithms for discrete time MDP can be used directly. Let $\Gamma + \alpha$ be the uniformization constant; Γ is chosen such that $\Gamma > \gamma(s), \forall s \in S$, where $\gamma(s)$ represents the rate out of a state $s \in S$. In this work $\Gamma = \lambda_1 + \lambda_2 + C(\frac{\mu_1}{b_1} + \frac{\mu_2}{b_2})$, therefore we can define $\lambda_i' = \lambda_i/(\Gamma + \alpha)$ as the probability that the next uniformed transition is an arrival, $\mu_i' = \mu_i/(\Gamma + \alpha)$ a service completion and $\alpha' = \alpha/(\Gamma + \alpha)$ the process termination. In Fig. 1 it is represented a generic state $s = (x, y)$ when the uniformization has already been done, in particular it is a state where the system has enough free capacity for primary demand ($b_1 x + b_2 y \le C - b_1$).

The discrete time MDP (DTMDP) equivalent has the discount factor $\beta = \frac{\Gamma}{\Gamma + \alpha}$ ($0 < \beta < 1$). As we said above, the objective consists in finding π^*, then, the problem can be written as:

$$V^*(s) = \max_{\pi \in \Pi} E\left[\sum_{n=0}^{\infty} \beta^n g(s_n, a_n) | s_0 = s \right], \forall s \in S; \tag{2}$$
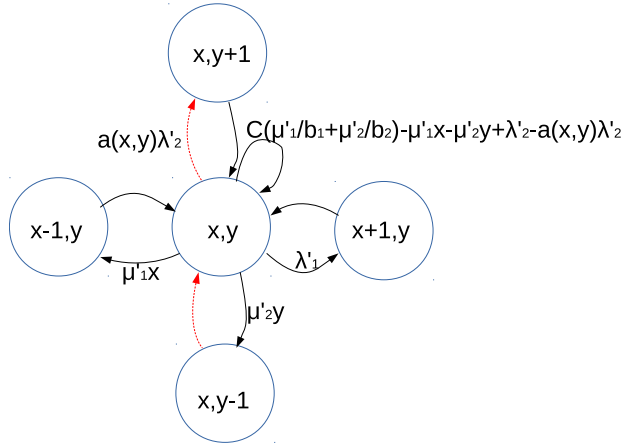
where:

Fig. 1: Discrete time model for a generic state (x, y) obtained after the uniformization process. Observe that this transformation implies "fictitious" transitions from a state to itself. The absorbing state (see Sec. 5.3 in [26]) is not represented.

- $V^*(s)$ is the maximal expected β-discounted reward for the system starting in state s,
- Π represents all the possible policies,
- a_n and s_n are $a(t_n)$ and $s(t_n)$ respectively, and
- $g(s_n, a_n) = b_2 R$, $g(s_n, a_n) = -z b_2 K$ or $g(s_n, a_n) = 0$ depending on (s_n, a_n).

In the next section, we discuss how to solve our optimization problem. In particular, we characterize some properties of the optimal admission control policy to be applied over SUs. Using this characterization we propose some alternatives to improve the performance of the commonly used dynamic programming algorithms.

3 Analysis and Characterization of Optimal Admission Control Policies

In this section we focus on the characterization of the optimal admission control policies in the described context. To this end, we formulate the problem using Dynamic Programming and after that, we prove some properties of the control policies.

3.1 Dynamic Programming Formulation

Dynamic Programming (DP) [8] is a mathematical technique that rests on the *principle of optimality*. It can be used in a variety of contexts and it is usually

based on a recursive formula and initial states. Before expressing the problem using this technique, we make some definitions.

Let $V_n(x, y)$ be the maximal expected discounted profit for the system starting in the state (x, y) when n observation points remain in the horizon. $V_n(x, y)$ is known as the value function of state (x, y) in n-steps. It can be proved that $V^*(x, y) = \lim_{n \to +\infty} V_n(x, y)$, being $V^*(x, y)$ the maximal expected β-discounted reward for the system starting in state (x, y) over an infinite horizon (see Eq. (2)). We refer to [8] for the suitable conditions for that convergence.

Having defined $V_n(x, y)$ we are now able to formulate the problem using DP. After the uniformization process is done, thus working with the DTMDP, the corresponding DP equations are:

For $n = 0$:
$$V_0(x, y) = 0$$

For $n \geq 1$:

− $b_1 x + b_2 y \leq C - b_1$
  $V_n(x, y) = \lambda_1' V_{n-1}(x + 1, y) + \lambda_2' \max\{V_{n-1}(x, y), V_{n-1}(x, y + 1) + b_2 R\} + \mu_1' x V_{n-1}(x - 1, y) + \mu_2' y V_{n-1}(x, y - 1) + (C(\frac{\mu_1'}{b_1} + \frac{\mu_2'}{b_2}) - \mu_1' x - \mu_2' y) V_{n-1}(x, y)$

− $C - b_1 < b_1 x + b_2 y \leq C - b_2,\ b_2 y \geq b_1$
  $V_n(x, y) = \lambda_1'(V_{n-1}(x + 1, y - z) - z b_2 K) + \lambda_2' \max\{V_{n-1}(x, y), V_{n-1}(x, y + 1) + b_2 R\} + \mu_1' x V_{n-1}(x - 1, y) + \mu_2' y V_{n-1}(x, y - 1) + (C(\frac{\mu_1'}{b_1} + \frac{\mu_2'}{b_2}) - \mu_1' x - \mu_2' y) V_{n-1}(x, y)$

− $C - b_2 < b_1 x + b_2 y \leq C,\ b_2 y \geq b_1$
  $V_n(x, y) = \lambda_1'(V_{n-1}(x + 1, y - z) - z b_2 K) + \lambda_2' V_{n-1}(x, y) + \mu_1' x V_{n-1}(x - 1, y) + \mu_2' y V_{n-1}(x, y - 1) + (C(\frac{\mu_1'}{b_1} + \frac{\mu_2'}{b_2}) - \mu_1' x - \mu_2' y) V_{n-1}(x, y)$

− $b_1 x + b_2 y = C,\ b_1 x = C$
  $V_n(x, y) = \lambda_1' V_{n-1}(x, y) + \lambda_2' V_{n-1}(x, y) + \frac{\mu_1'}{b_1} C V_{n-1}(x - 1, y) + \frac{\mu_2'}{b_2} C V_{n-1}(x, y)$

Please note that we have considered $b_1 \geq b_2$ in order to simplify the notation. As we can see, there are four different equations for $n \geq 1$. The first one represents the situation when there are enough idle channels in the system to satisfy primary demand. The second and the third equations are for the preemptive situations, and the last one is for the case when all the channels are used by PUs. Another observation is that the "fictitious" transitions do not affect the total reward of the system. Finally, it is important to remark that the process terminates with probability α' and after that no more profits will be earned [26].

According to the above formulation, it is possible to define the transformation T:

$$V_n(s) = T V_{n-1}(s) = \max_{a \in A_s} f(\lambda_1', \lambda_2', \mu_1', \mu_2', b_1, b_2, R, K, \beta, \pi, V_{n-1}); \quad (3)$$

where $f(.) = E\left[\sum_{j \in S} p(j|s, a)(V_{n-1}(j) + g(s, a))\right]$ and $p(j|s, a)$ are the transition probabilities of the DTMDP. It is important to note that the function

f, that plays a key role in the DP algorithms studied in the next sections, can be deduced from the DP formulation. Due to the fact that A_s is finite for each $s \in S$, the rewards and cost are bounded, and the transition probabilities and costs are stationary, there exists an optimal stationary deterministic policy π^* and therefore

$$V^*(s) = lim_{n \to \infty} T^n V_0(s), \forall s \in S; \qquad (4)$$

where T^n represents the composition of the mapping T with itself n times.

The best known practical algorithms for solving infinite-horizon MDPs based on dynamic programing are: Value Iteration (VI) and Policy Iteration (PI and its several modifications, i.e. Modified Policy Iteration (MPI)) [26]. In [25], there is a complete analysis of these techniques. The rate of convergence of both is strongly related with β (discount factor) and also with the total number of channels considered (C). It is important to highlight that if $\beta < 1$, all the iterative DP algorithms are guaranteed to converge [26].

Following we prove several results to characterize the optimal admission control policy for SUs that maximize the SP profit. This represents one of the main contributions of this paper. This analysis is used, as an example of application, in order to improve the performance of some DP algorithms.

3.2 Optimal control policy analysis and characterization

In this section the structure of the optimal policy is determined. We address the cases $K \geq R$ and $K < R$ separately. For $K \geq R$, we first prove monotonicity properties, then we prove that the admission control boundary is a "switching curve" policy, and finally we demonstrate the concavity of the value function under certain conditions. These properties are analogous to the ones presented in [33,24,30,34] but based on our specific context. For the other case, when the SP's punishment is less than the SU's payments ($K < R$), we prove that the optimal policy is the one that always accepts SUs. This is a strong conclusion because this implies that when $K < R$ an admission control mechanism has no sense (independently of the system parameters $\lambda_i, \mu_i, b_i, \beta$ and C). In the demonstrations we use induction and sample-path arguments [29] [14]. These techniques are used in most of the cited references in Sec. 1 in order to characterize the value function and the optimal policy in a variety of scenarios.

Proposition 1 $V_n(x, y + 1) \leq V_n(x, y)$, $\forall (x, y) : b_1 x + b_2(y + 1) \leq C$, $\forall n$

See appendix A for the proof of the general case ($b_1 \geq 1$ and $b_2 \geq 1$) using induction arguments on n. For better understanding of this result, we explain here the proof using sample-path methods [19] of the particular case: $b_1 = b_2 = 1$. Please note that in this case the preemption only occurs when the system is full of users.

Proof Suppose that we start two processes on the same probabilistic space. Process 1 starting in the state (x, y) and process 2 in $(x, y+1)$. We couple this two systems, in other words, all service and arrival times in one and the other are the same. Suppose the optimal policy of system 2 (π_2^*) is followed in both systems. Let $V_n^{\pi_2^*}(x, y)$ and $V_n^{\pi_2^*}(x, y+1)$ be the value functions of system 1 and 2 using policy π_2^* respectively, so $V_n^{\pi_2^*}(x, y+1) = V_n(x, y+1)$.

Now since both processes use the same policy, as long as system 1 has one less user than system 2, then $V_n^{\pi_2^*}(x, y) = V_n(x, y+1)$ (they see the same arrivals and departures, and, therefore the same rewards and costs). This situation could change only in the following critical case:

– Consider the first time the process 2 enters a state with the number of users equal to C and system 1 has exactly $C-1$ users (this situation will never occur if the extra SU in system 2 leaves the system before the first time that system reaches C users). In this case, if a PU arrives, both systems must accept the new user. System 2 has to remove one of the SUs allocated, thus system 1 will have K more profit than system 2. This implies that $V_n^{\pi_2^*}(x, y) > V_n(x, y+1)$. After this point, the systems will be in the same state and therefore that difference will continue unchanged.

Considering that the reward obtained in this way for system 1 will not be greater than its optimal reward $(V_n^{\pi_2^*}(x, y) \leq V_n(x, y))$, and examining that the initial state was arbitrary considered, the result follows.   □

As we have explained, we study the behavior of the system for all values of K and R. Then, Propositions 2, 3 y 4 are for the case where $K \geq R$, that is, when the reimbursement obtained by an affected SU is greater than its payment. This scenario plays a key role in motivating the SU's participation in the spectrum sharing (in particular, it is very important when the SU's service interruption causes a big damage affecting the QoS of their communications). On the other hand, Proposition 5 is for the complementary case $(R > K)$. In this particular situation, if the objective is to maximize the SP's profit, we prove that the optimal policy is the one which defines $a(x, y) = 1, \forall (x, y)/b_1 x + b_2 y \leq C - b_2$.

Proposition 2 *if $K \geq R$, $V_n(x, y+1) - V_n(x, y) \geq -b_2 K$, $\forall (x, y) : b_1 x + b_2(y+1) \leq C$, $\forall n$*

The proof of the general case is located in appendix B. As in the previous proposition, we present here the proof for the same particular case using sample-path arguments.

Proof Considering system 1 starting in (x, y) and system 2 in $(x, y+1)$. In this case, we assume that system 1 follows its optimal admission policy (π_1^*) and system 2 imitates all decisions of system 1 whenever it is feasible. We can identify two critical cases which may alter the difference between the profit of the systems.

When $x + y + 1 = C$, in other words, when system 2 reaches C users:

- If a SU arrives, system 2 will reject it. System 1 can accept or reject, it will depend on its optimal policy π_1^*. If the SU is accepted in system 1, the inequality holds only if $K \geq R$. After that, the difference of the profit of both systems will not change because the systems will be in the same state. This implies that if $K \geq R$, then $V_n^{\pi_1^*}(x, y+1) - V_n(x,y) \geq -K$
- If a PU arrives, both systems have to accept it. System 2 will remove one of the secondary users paying K, so the equality holds. After that, the two will be coupled (they will be in the same state).

Due to the fact that $V_n^{\pi_1^*}(x, y+1) \leq V_n(x, y+1)$, the result is proved. $\quad\square$

Proposition 3 *if $K \geq R$, $V_n(x+1, y+1) - V_n(x+1, y) \leq V_n(x, y+1) - V_n(x,y)$, $\forall(x,y) : b_1(x+1) + b_2(y+1) \leq C$, $\forall n$*

The detailed proof can be found in appendix C and it is based on induction arguments on n. Intuitively, we can interpret this last result as: we expect that it should be more difficult to accept SUs when there are more PUs in the system. This gives us the idea that the admission control boundary that maximize the SP profit is from a special form, it is a "switching curve".

Definition 1 Switching curve: For every y, we define a level $l(y)$ such that when the system is in state (x, y) decision 1 is taken if and only if $x \leq l(y)$ and 0 otherwise. The mapping y with $l(y)$ is called a "switching curve".

In a particular case, if $\mu_1 = \mu_2$ we have $V_n(x+1, y+1) - V_n(x+1, y) = V_n(x, y+2) - V_n(x, y+1)$, $\forall(x,y) : b_1(x+1) + b_2(y+1) \leq C$, $\forall n$. This means that the admission control boundary only depends on the number of busy channels at the system.

Following, we prove the concavity property of the value function. This would lead to monotony of the thresholds. In addition, the concavity will be a key in the proposals of the next section.

Proposition 4 *if $K \geq R$, $V_n(x, y) - V_n(x, y+1) \leq V_n(x, y+1) - V_n(x, y+2)$, $\forall(x,y) : b_1 x + b_2(y+2) \leq C$, $\forall n$*

Its proof can be found in appendix D. In the same way as in the previous case, the proof is based on induction arguments.

It is important to note that sample path analysis might do the last two demonstrations in the same way as in the first propositions, but would be more tedious.

Proposition 5 *if $R \geq K$ and let π be an admission control policy which satisfies that $a(x,y) = 1$ $\forall(x,y)/b_1 x + b_2 y < C$ and $a(x,y) = 0$ in other cases, then π is the optimal one.*

Proof Working in the particular case $b_1 = b_2 = 1$, using sample-path arguments we prove that $V_n^{\pi}(x,y) \geq V_n^{\pi'}(x,y) \forall \pi' \neq \pi$ (π' represents a policy that has at least one state $(x,y)/x + y < C$ where $a(x,y) = 0$). Considering two

systems starting in the same state (x, y), system 1 following policy π and system 2 policy π'. It is easy to note that the number of PUs will always be the same in both systems, they could only differ in the number of SUs. If system 2 is in state (x', y') then system 1 will be in state $(x', y' + w), w \geq 0$.

While a preemptive situation does not occur, $V_n^\pi(x, y) - V_n^{\pi'}(x, y) \geq wR$. On the other hand, considering the times when the process 1 enters a state with the number of users equal to C $(x' + y' + w = C)$. In this situation the worst-case is when at least w successive PU arrivals occurs (that implies wK of punishment for the SP). Then, $V_n^\pi(x, y) - V_n^{\pi'}(x, y) \geq w(R - K)$. Considering that the the initial state was arbitrary considered and $R \geq K$, the desired result is proved. It is possible to prove the general case $(b_1 \neq b_2)$ using induction arguments in the same way as in previous propositions.   □

The results were proven for all $n \geq 0$, in particular they are true for the limit $n \to \infty$. Since the control space and the state space are finite, the results can also be extended to the average profit case [34, 26].

4 Variants of Modified Policy Iteration Algorithm

The characterization presented in the previous section represents one of the main contributions of the paper. Now, we propose a possible application. In particular, we use those properties in order to improve the performance of a specific DP algorithm: the Modified Policy Iteration. The goal in this work is to present a new version of that algorithm which running time drastically decreases respect to the original one when it is applied to our problem. We consider the case when $K \geq R$ because in the other case the optimal control is already known.

First of all, we introduce the DP algorithm that we have used and optimized. In the rest of the section, we present some alternatives in order to modify the algorithm using the knowledge of the characterization.

4.1 Modified Policy Iteration algorithm

As aforementioned, Policy Iteration (PI) and Value Iteration (VI) are the most usual DP mechanisms for solving MDPs. Comparing PI with VI, the first one is desirable to be used in practice because of its finite-time convergence to the optimal policy. That is the main reason why in this work we have chosen PI. Even more, we have worked with Modified Policy Iteration (MPI) algorithm [26] that is an approximate version of PI.

Policy iteration (PI) mainly consists of two iteration steps: policy evaluation and policy improvement (see Alg. 1). The most tricky part is the Policy Evaluation (Step 4 showed in the pseudo-code of Alg. 1); it consists in solving a system of linear equations. The most common method to work it out is the Gaussian elimination, but when you have a model with q states, this requires on the order of q^3 multiplications and divisions. So, for a large q (in our case

---

Algorithm 1 Policy Iteration (PI)

Require: C, λ_1', λ_2', μ_1', μ_2', b_1, b_2, β, R, K
1: choose an arbitrary policy π'
2: **repeat**
3:    $\pi := \pi'$;
4:    compute the value function of the policy π:
      $V_\pi(s) = f(\lambda_1', \lambda_2', \mu_1', \mu_2', b_1, b_2, R, K, \beta, \pi, V_\pi(s))$, $\forall s \in S$;
5:    improve the policy based on $V(s) = V_\pi(s)$:
      $\pi' = \arg\max_\pi f(\lambda_1', \lambda_2', \mu_1', \mu_2', b_1, b_2, R, K, \beta, \pi, V(s))$, $\forall s \in S$;
6: **until** $\pi = \pi'$
7: **return** π

---

q is related with C, b_1 and b_2), obtaining an exact value function for a specific policy might be computationally prohibitive.

There are many proposals of this algorithm implementation. In particular, in [26,27] the authors proposed the MPI algorithm to improve the efficiency of PI when there is a large state space involved. In MPI the idea is to implement the "policy evaluation step" similar to the VI algorithm with the only difference that the value functions are computing for a fixed policy instead of computing the maximum of the value function for all possible actions in each state. It is demonstrated that VI finds a stationary policy that is ϵ-optimal within a finite number of iterations (ϵ is related with the stop criterion). Therefore, due to the fact that this implementation of MPI combines features of policy iteration and value iteration, the policy obtained is also ϵ-optimal.

Definition 2 ϵ-optimal policy: Let $V^*(i)$ be the maximum expected total discounted reward over an infinite horizon starting in state i. It is well known that V^* satisfies the Bellman equation. Then, a policy π is ϵ-optimal for $\epsilon > 0$ if $V^\pi \geq V^* - \epsilon$.

Numerical results reported in [25,27] suggest that MPI is more efficient than either value iteration or policy iteration in practice. In this paper we have worked with that MPI algorithm definition. Its convergence is demonstrated in several works like [26,27].

Following we present some alternatives in order to modify the MPI algorithm using the knowledge of the previous sections. One proposal is based on that it is not necessary to consider all actions in each state while improving the policy. Another consists in picking a subset of states in order to apply the "policy evaluation step". In addition, at the end of this section, we also present a way to modify the MPI to obtain a linear approximation of the optimal admission control boundary.

4.2 Modification of "policy improvement step"

We incorporate the information given in Prop. 3 and 4 in order to reformulate the Step 5 (policy improvement) of the MPI algorithm. In this Step the idea is to improve the policy at each state of the system, therefore the original MPI

algorithm has to evaluate it in the whole state space. Our proposal consists in evaluating the policy improvement only in few states. In other words, the idea is to consider a subset of the policies (i.e. the feasible policies). In the original MPI algorithm, the total number of policies to be considered is $\approx 2^q$ (letting q the total number of states), in our proposal we only consider the policies whose boundaries represent a "switching curve" verifying Prop. 3 and 4.

Remembering that $a(x,y) \in \{0,1\} \ \forall (x,y)/b_1 x + b_2 y \leq C - b_2$ and $a(x,y) = 0$ in other cases (where 1 means to accept and 0 to reject). According to Prop. 3, we can infer that if $a(x^*, y^*) = 1$ for a particular state (x^*, y^*) then $\forall x < x^*, a(x, y^*) = 1$. In addition, please note that using Prop. 4, if $a(x^*, y^*) = 0$ then $\forall y > y^*, a(x^*, y) = 0$.

We present in Alg. 2 an alternative way to implement the Step 5 of the MPI using the above explanation.

Algorithm 2 New version of Policy improvement step

1: **for** $y := 0 : [C/b_2] - 1$ **do**
2:    **if** $(y = 0)$ **then**
3:       $x := [C/b_1] - 1$;
4:    **else**
5:       $x := x_b$;
6:    **end if**
7:    out:=false;
8:    **while** not out **do**
9:       $a(x,y) := \arg\max_{A_{(x,y)}} f(\lambda_1', \lambda_2', \mu_1', \mu_2', b_1, b_2, R, K, \beta, \pi, V(x,y))$;
10:      **if** $(a(x,y) = 1)$ **then**
11:         out:=true
12:         $\forall x' < x$ SET $a(x', y) := 1$ (all the states $(x', y) : x' < x$ have the same admission control decision as state (x, y));
13:      **else**
14:         $x := x - 1$;
15:      **end if**
16:    **end while**
17:    $x_b := x$;
18: **end for**
19: $\pi' := \{a(x,y), \forall (x,y)\}$;
20: **return** π'

4.3 Modification of "policy evaluation step"

In [26] there is a detailed demonstration of the non-necessity of determining the exact value function $(V_n(s))$ to identify an improved policy. The fundamental concept behind is to make a finite number of iterations in order to obtain an approximation of $V_n(s) \forall s \in S$. It represents the essential part of the MPI algorithm which differentiates it from PI. The convergence in this scenario is also proven in [26]. For a better understanding, a pseudo code of the "policy evaluation step" (from original MPI) is presented in Alg. 3. We have used it as a pattern to compare with our new versions.

Algorithm 3 Original Policy evaluation Step (from MPI Algorithm)

Require: $\epsilon > 0$, m_n, V_n, π (n represents one iteration of Alg. 1);
1: $k := 0$;
2: $u_n^k(s) := \max_\pi f(\lambda_1', \lambda_2', \mu_1', \mu_2', b_1, b_2, R, K, \beta, \pi, V_n(s)), \forall s \in S$;
3: **if** $(\left\| u_n^k - V_n \right\| < \epsilon)$ **then**
4:     MPI finishes and the result is π;
5: **else**
6:     **if** $(k = m_n)$ **then**
7:        go to step 13;
8:     **else**
9:        $u_n^{k+1}(s) := f(\lambda_1', \lambda_2', \mu_1', \mu_2', b_1, b_2, R, K, \beta, \pi, u_n^k(s)), \forall s \in S$;
10:       $k := k + 1$;
11:       go to step 6;
12:     **end if**
13:     $V_{n+1} := u_n^{m_n}$;
14:     **return** V_{n+1}
15: **end if**

Please note that the existence of the parameter m_n in Alg. 3 represents the main characteristic of the original "policy evaluation step" from MPI. It is demonstrated that the algorithm converges for any order sequence $\{m_n\}$ and the rate of its convergence is related with $\{m_n\}$. Optimal choice of the sequence $\{m_n\}$ remains an open question. In this paper we have chosen $m_n = m, \forall n$ and its value has remained unchanged in all the simulations we did for a performance comparison (both in the original MPI and in the new proposals).

In this subsection (and also in Subsec. 4.5) we propose heuristic modifications to this step of the algorithm. These alternatives are concentrated in reducing the number of system states that participate directly in "policy evaluation step". With that in mind, we introduce into the algorithm the information that the value function is concave for a fixed value of PU (Prop. 4: $V_n(x,y) - V_n(x,y+1) \leq V_n(x,y+1) - V_n(x,y+2), \forall(x,y)/b_1 x + b_2(y+2) \leq C$, $\forall n$). This property allows us to apply linear interpolation (when the difference $V_n(x,y) - V_n(x,y+1)$ is small) reducing the computational complexity of the algorithm. The idea is to sacrifice the precision of the value determination without affecting the result of the algorithm. In other words, the new value determination will be applied only on certain states ($S' \subseteq S$, including in S' the states where the difference $V_n(x,y) - V_n(x,y+1)$ is higher) and for the others, it will be estimated using piecewise linear interpolation.

It is important to highlight that the computational complexity of linear interpolation is one multiplication and two additions per sample of output. On the other hand, $u_n^{k+1} = f(\lambda_1', \lambda_2', \mu_1', \mu_2', b_1, b_2, R, K, \beta, \pi, u_n^k(s))$, in most of the states, implies: ten multiplications and seven additions.

The question is how to chose the subset S' and how many elements to include. Observing the DP formulation, it is easy to note that the value function of each state only depends on the value function (in the previous iteration) of its "neighbor states". However, it is a recursive dependency because the value function of the neighbors of an specific state also depends on their neighbors' functions and so on. In this context, we propose that the subset S' must in-

---

Algorithm 4 New version of Policy Evaluation Step

Require: $\epsilon > 0$, m_n, V_n, π

1: $k := 0$;
2: $u_n^k(s) := \max_\pi f(\lambda_1', \lambda_2', \mu_1', \mu_2', b_1, b_2, R, K, \beta, \pi, V_n(s)), \forall s \in S$;
3: **if** $(\left\| u_n^k - V_n \right\| < \epsilon)$ **then**
4:     MPI finishes and the result is π;
5: **else**
6:     **if** $(k = m_n)$ **then**
7:         go to step 14;
8:     **else**
9:         $u_n^{k+1}(s) := f(\lambda_1', \lambda_2', \mu_1', \mu_2', b_1, b_2, R, K, \beta, \pi, u_n^k(s)), \forall s \in S'$;
10:        With the results of the previous step, use linear interpolation to estimate $u_n^{k+1}(s) \forall s \in S \setminus S'$;
11:        $k := k + 1$;
12:        go to step 6;
13:    **end if**
14:    $V_{n+1} := u_n^{m_n}$;
15:    **return** V_{n+1}
16: **end if**

---

clude the states that are located near the admission control (AC) boundary and also their nearest neighbors. We do not know a priori where is going to be located the AC boundary, but, we can infer it by using the results of the next two subsections. In Alg. 4 a pseudo code of the explained alternative is presented. Please note that the policy evaluation is executed only in $s \in S'$.

How do we choose the states to be considered in S'? To this end, we define two new parameters (K^* and L) and we proceed in the following way:

- choose the parameters $K^* \in \mathbb{N}$ and $L \in \mathbb{N}$.
- randomly choose a set of states $I = \{(x, y)\}$ such that $b_1 x + b_2 y < K^*$ and the number of elements of I is L.
- finally $S' = \{I \bigcup \{(x, y) : K^* \leq b_1 x + b_2 y \leq C\}\}$

The result of this proposal is a sub-optimal and its performance will be associated with the parameters K^* ($0 \leq K^* \leq C$) and L ($0 \leq L \leq \frac{K^*/b_2(K^*/b_1+1)}{2}$). In particular, because of the concavity of $V_{n_x}(y)$[1], its piecewise linear estimation (the segments) will be always under the corresponding MPI's value function. As a consequence, the admission boundary obtained by this modification will be more conservative than the original MPI's solution (this characteristic might be important when the goal is to guarantee certain level of QoS to SUs: a smaller interruption probability). It is important to remark that choosing appropriate values of K^* and L, the interpolation error could be as small as you want, therefore the results could be optimal (or nearly optimal).

Which are good options of these parameters? This is an open question but following we list some important considerations. The smaller is K^*, more accurate will be the value determination, but the running time will have a smaller impact in its reduction. On the other hand, for larger values of K^*, although the algorithm running time will decrease, the admission control bound-

---

[1] $V_{n_x}(y)$ is $V_n(x, y)$ for fixed x.

ary might be not optimal. The opposite effect occurs with L: the larger value of L is considered, the more precision will be reached, but the less impact we will have in the running time. As a conclusion, the values of these new parameters are related with the system parameters (λ_1', λ_2', μ_1', μ_2', b_1, b_2, R, K and β). In the next section we propose a heuristic method for K^*; once K^* is chosen, a boundary to L is implicitly determined.

Using the proposals of Subsec. 4.2 and 4.3 we have developed a new version of MPI called *newMPI*. In Sec. 5 we present some simulation experiments and a performance comparison between MPI and *newMPI*.

4.4 Heuristic method for choosing K^*

Based on [33], in this subsection we propose a heuristic method for choosing the parameter K^* defined before. We want to know which area (i.e. which set of states) is most likely to contain the optimal AC border. The authors in [33] studied a similar problem (preemption with reimbursement in a cognitive scenario) with different hypotheses: call durations are exponentially distributed with identical means ($\mu_1 = \mu_2 = \mu$), the reward per SU accepted is collected after the SU leaves the system with successful completion of its service, and each user demand is one channel, independently of the class ($b_1 = b_2 = 1$).

According to the first hypothesis, that strongly simplifies the problem without representing a realistic scenario, they demonstrated that the optimal admission control decision only depends on the total number of occupied channels, in other words the admission control boundary is a line with equation $x + y - T^* = 0$. More general, we demonstrated in Prop. 3 that this result is still valid when $b_1 \neq b_2$, then, the admission control boundary when $\mu_1 = \mu_2$ is $b_1 x + b_2 y - T^* = 0$.

As a consequence, it is possible to use a birth-death 1D MDP with state space $S_{1d} = \{i \in \mathbb{N} | 0 \leq i \leq C\}$ to determine the optimal threshold (T^*). Adapting their model in order to satisfy our assumptions (in particular the difference with their second and third hypothesis) the problem is reduced to solve the next optimization problem:

$$\max_T b_2 R \lambda_2 \left(1 - \sum_{T \leq i \leq C} \pi_T(i) \right) - K\lambda_1 \sum_{C-b_1+1 \leq j \leq C} (j - C + b_1)\pi_T(j) + K b_1 \lambda_1 E\left(\frac{\lambda_1}{\mu_1}, j \right)$$
(5)

where $\pi_T(i), 0 \leq i \leq C$ is the steady state probability with threshold T ($T \leq C - b_2$) and $E(\frac{\lambda_1}{\mu_1}, C)$ corresponds to Erlang-B formula.

Due to the fact that the above formulation can be used only when the mean service time of both classes are identical, using the idea of [30] we can transform our general system to be in that particular context. Our proposal consists of obtaining a μ-scaled system with parameters λ_1^s, λ_2^s, μ, b_1, b_2, K, and R, and after solving Eq. (5) an idea of the location of the admission control boundary is possible to obtain.
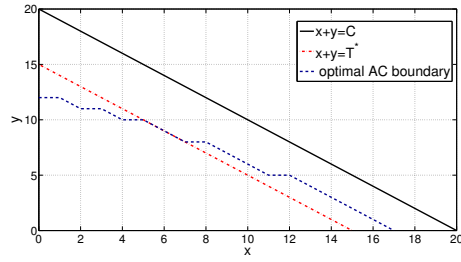
(a) parameters: $\lambda_1 = 20$, $\lambda_2 = 5$, $\mu_1 = 1$, $\mu_2 = 4$, $\mu = 1$, $b_1 = b_2 = 1$, $C = 20$, $R = 1$, $K = 10$.
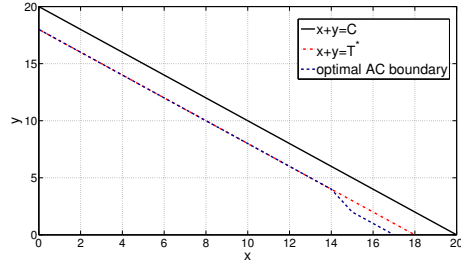
(b) parameters: $\lambda_1 = 20$, $\lambda_2 = 5$, $\mu_1 = 3$, $\mu_2 = 4$, $\mu = 3$, $b_1 = b_2 = 1$, $C = 20$, $R = 1$, $K = 10$.

Fig. 2: Performance comparison in order to find the area that is most likely to contain the optimal AC boundary.

The μ-scaled system has the following parameters: $\mu_1^s = \mu_2^s = \mu$, $\lambda_1^s = \frac{\lambda_1 \mu}{\mu_1}$ and $\lambda_2^s = \frac{\lambda_2 \mu}{\mu_2}$. Observe that $\frac{\lambda_1^s}{\mu} = \frac{\lambda_1}{\mu_1}$ and $\frac{\lambda_2^s}{\mu} = \frac{\lambda_2}{\mu_2}$.

In Fig. 2a and Fig. 2b there are two simulated examples that show the optimal admission control boundary obtained using MPI and the line obtained applying the results of our adaptation of the model of [33] (Eq. (5)) to the scaled system. In both examples, the threshold T^* gives an estimation of the location of the optimal boundary, therefore this result can be used in order to choose an appropriate value of K^* in our proposal of *newMPI*. For instance, in the example of Fig. 2a being conservative a possible value of K^* is 10, while in the other one $K^* = 15$ can be a reasonable option.

4.5 Linear approximation of MPI

In order to continue improving the computational efficiency, we use the results of Prop. 3 and 4 to obtain another alternative version of Modified Policy Iteration algorithm (*linMPI*). It returns a linear approximation of the optimal admission boundary. Because of its characteristics, the optimal AC boundary achieved by MPI could be approximated by a line with equation $y = ax + b$ (see for example Figs. 2a and 2b). In this new proposal, the idea is to reach

a policy that its reward is nearly optimal implying less computational effort than *newMPI* and MPI.

Prop. 3 is used in order to optimize Step 5 of MPI in a similar way as we described before and Prop. 4 to optimize the running time of Step 4 of MPI. In this new proposal, the policy evaluation defined in the original MPI is executed only in a set of the system states: $S^* = \{(x,y) : x \leq k^* \text{ or } y \leq k^*\} \bigcup \{I\}$ where I represents a group of randomly chosen states of $S \setminus \{(x,y) : x \leq k^* \text{ or } y \leq k^*\}$. Using the concavity property, the value functions of the not considered states are obtained by linear interpolation based on the value of $s \in S^*$. Note that the parameters to be defined are k^* and L (as in the previous explanation, L represents the number of states chosen randomly). Advantage: we do not need to know the possible location of the optimal policy.

Due to the fact that we are looking for a line as an approximation of the AC boundary, in this case the Step 5 is only evaluated over the states that satisfy $x = 0$ or $y = 0$ (we only need two points to determine the line $y = ax + b$). That is the reason of the parameter k^*. After having done many simulations, we conclude that using the pair $(k^* = \frac{C}{10}, L = \frac{C/b_1 C/b_2}{4})$ the results obtained are nearly optimal (see Sec. 5).

In this alternative, the policy π is translated as a pair of parameters $\pi = (a, b)$. Therefore, the stop criterion of the *linMPI* is when $a = a'$ and $b = b'$, letting $\pi = (a, b)$ and $\pi' = (a', b')$ be two policies en two different and consecutive iterations of the algorithm (see Alg. 5 where it is explained the policy improvement proposed).

Algorithm 5 New version of Policy improvement step - linMPI

1: $y := 0$; $x := [C/b_1] - 1$ (or $x := 0$; $y := [C/b_2] - 1$);
2: out:=false;
3: **while** not out **do**
4:     compute $a(x, y)$ improving the policy at state (x, y); (evaluate Step 5 of PI)
5:     **if** $(a(x, y) = 1)$ **then**
6:         out:=true;
7:         $-b/a := x$ (or $b := y$)
8:     **else**
9:         $x := x - 1$ (or $y := y - 1$);
10:     **end if**
11: **end while**
12: $\pi' := (a, b)$;
13: **return** π'

On the other hand, "policy evaluation step" is implemented in a similar way to *newMPI* but considering the defined set S^*. In Sec. 5 we show some examples of the performance of *linMPI*.

5 Simulated Experiments and Results

In order to test the proposed algorithms we consider different simulation experiments. In the first case we evaluate the *newMPI* algorithm and finally, the last experiments corresponds to *linMPI* proposal.

5.1 Performance tests of *newMPI*

As we explained before, we developed a new version of MPI called *newMPI* using the proposals of Subsec. 4.2 and 4.3. We expect to get the optimal admission control boundary reducing the total CPU-time consumed in comparison to the MPI pattern.

  After having tested the algorithm with both proposed modification versus the original MPI, we have identified that in most cases the number of states evaluated in "Policy Improvement Step" in the *newMPI* represents less than 5% of the whole space.

  For instance, in Fig. 3 it is shown an example where *newMPI* is applied. In this example, we have tested the new version of the algorithm for different values of parameters K^* and L, and only when $K^* = 95$ (and $L = \frac{K^*(K^*+1)}{4}$) there is a minimal difference between the optimal AC boundary and the AC obtained in the *newMPI* (for $K^* < 95$ the resulted boundaries are the same as the optimal one). In particular in Fig. 3b there is a performance comparison showing $\frac{T_{newMPI}}{T_{MPI}}$ for different values of K^* and L. T_{MPI} and T_{newMPI} represent the running times of the original MPI algorithm and of our new version respectively. It is clear that the running time has a big impact as bigger is K^*. For example for $K^* = 85$ (and $L = 1827$) the running time of the new version is less than $0.5 T_{MPI}$ and the AC boundary obtained is the optimal.

  It is important to remark that a practical determination of K^* value can be done using the results of Subsec. 4.4 or Subsec. 4.5. The proposed methods in these subsections give us "instantaneously" the knowledge of where is going to be located the optimal admission control boundary. Knowing that, a reasonable value of K^* will be if the boundary is totally located in the area $b_1 x + b_2 y - K^* \geq 0$. The parameter L is related with K^* because it represents the number of states to be evaluated that are not considered when K^* is chosen. In the experiments we have chosen $L = \frac{K^*(K^*+1)}{4}$, this value could be chosen in a different way but note that it plays an important role in minimizing the interpolation error (or minimizing the algorithm running time).

5.2 Performance tests of *linMPI*

In Fig. 4 we present some numerical examples of two different systems. T_{linMPI} and T_{MPI} represent the algorithm's running time of *linMPI* and MPI respectively. It is showed that the linear AC boundaries are good approximations to the optimal ones in both cases. Using $k^* = C/10$ and $L = C^2/4$, we can see that the running time of the new version decreases by 75%.
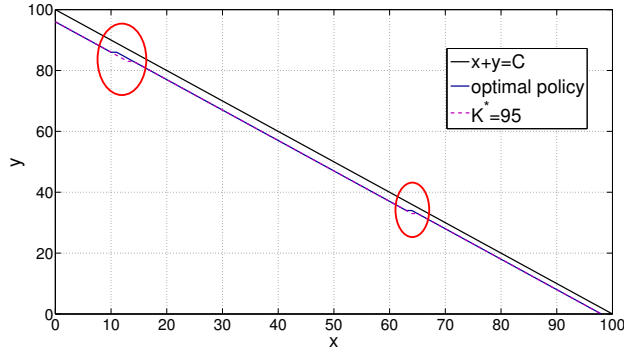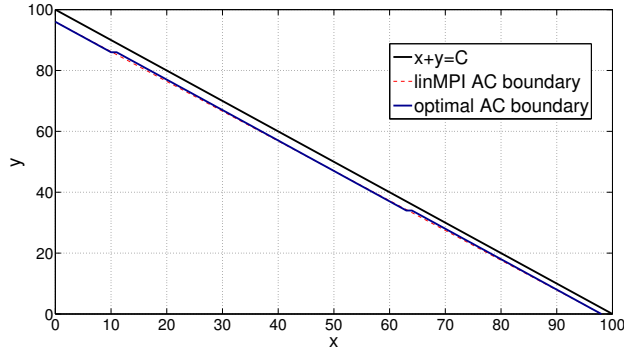
(a) Parameters: $\lambda_1 = 200$, $\lambda_2 = 300$, $\mu_1 = 3$, $\mu_2 = 1$, $b_1 = b_2 = 1$, $C = 100$, $R = 1$, $K = 3$, $\beta = 0.9474$. The circles show the differences between the AC boundary obtained with $newMPI$ using $K^* = 95$ and $L = 2280$ and the optimal boundary. In terms of SP's profit that difference is insignificant. As we explained before, the AC boundary obtained by $newMPI$ is the optimal or is more conservative.
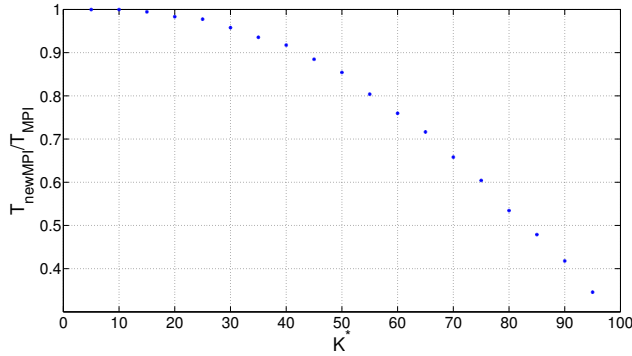

(b) Relation between running time of $newMPI$ and MPI algorithm for different values of K^* and L. For each value of (K^*, L) we have made a hundred of experiments. The mean values of each group of parameters is the one that is showed in the graphic

Fig. 3: Comparison between MPI and $newMPI$ (improving Steps 4 and 5). L is chosen proportional to K^*: $L = \frac{K^*(K^*+1)}{4}$.

In order to test how closely to the optimal are the approximations of the admission control boundary, we have made several experiments (n) with both boundaries (the optimal and its approximation using $linMPI$) and compute the profit of the SP. Each experiment consists in one realization of the continuous time markov chain using the appropriate AC boundary. In each transition the discount profit of the SP is computed. In Table 1 the results are summarized for both cases. Based on these results we can conclude that the reward of the approximated boundary has a good accuracy.

(a) Case 1. parameters: $\lambda_1 = 200$, $\lambda_2 = 300$, $\mu_1 = 3$, $\mu_2 = 1$, $b_1 = b_2 = 1$, $C = 100$, $R = 1$, $K = 3$, $\beta = 0.9474$. $\frac{T_{linMPI}}{T_{MPI}} = 0.25$.
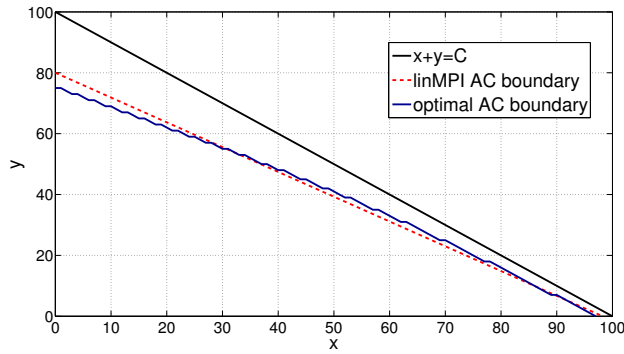
(b) Case 2. parameters: $\lambda_1 = 200$, $\lambda_2 = 500$, $\mu_1 = 3$, $\mu_2 = 0.5$, $b_1 = b_2 = 1$, $C = 100$, $R = 1$, $K = 3$, $\beta = 0.9953$. $\frac{T_{linMPI}}{T_{MPI}} = 0.2$

Fig. 4: Optimal Boundaries MPI vs linMPI. Parameters: $k^* = C/10$, $L = C^2/4$

Table 1: Examples considering cases of Fig. 4. Confidence Interval: $n = 30$, confidence level= 0.95.

Case	Method	Reward Confidence Interval
1	MPI	[5.75, 6.90]
	linMPI	[5.77, 6.92]
2	MPI	[39.90 43.06]
	linMPI	[39.95 43.14]

6 Conclusions and Future Work

We have studied a general asymmetric queueing system and we have applied the results to the cognitive radio network context. In particular we have characterized a paid-sharing approach as a spectrum allocation mechanism where

SUs pay for the spectrum utilization and preemptive situations are also contemplated. In this model when a PU arrives to the system and the free capacity is insufficient, at least one SU will be deallocated and reimbursed for its service interruption implying some cost for the primary SP.

We have proposed different ways to obtain the admission control policy for SUs that maximize the long-run discount profit of the SP. Different techniques have been used implying a wholeness in the analysis of the problem. We have considered a general model with different demands and different service rates between PUs and SUs. We have modeled the optimal revenue problem as a MDP. As one of the main contribution, we have characterized many properties of the structure of the optimal admission control policy. These results can be used for different purposes. As an example of application, we have proposed different alternatives to change the Modified Policy Iteration algorithm incorporating the knowledge of the characterization. The new versions show drastically shorter running-times than the original MPI algorithm. It is important to remark that all the contributions have been evaluated through extensive sets of simulations.

This paper presents a specific application of the problem centered in wireless cognitive network area. It would be interesting to look for other possible problems that could be modeled in a similar way, in the same or other application areas, to analyze how the same analysis could be applied and what would be the results we could obtain.

Finally, the next stage in our line of research would be to incorporate to our problem: dynamic pricing features like for example the ones defined in [32] and QoS requirements to SU's communications.

A Proof of Proposition 1

Inequality holds for $n = 0$ by definition. Assuming the inequality holds for $n - 1$, we show that it holds for n. We divide the problem in two cases: preempted and non-preempted situations.

Proof — case 1: $b_1 x + b_2(y + 1) \leq C - b_1$;

$$
\begin{align}
V_n(x, y + 1) = \lambda_1' V_{n-1}(x + 1, y + 1) + \tag{6} \\
\lambda_2' \max\{V_{n-1}(x, y + 1), V_{n-1}(x, y + 2) + b_2 R\} + \tag{7} \\
\mu_1' x V_{n-1}(x - 1, y + 1) + \tag{8} \\
\mu_2' y V_{n-1}(x, y) + \tag{9} \\
\mu_2' V_{n-1}(x, y) + \tag{10} \\
\left(C\left(\frac{\mu_1'}{b_1} + \frac{\mu_2'}{b_2}\right) - \mu_1' x - \mu_2'(y + 1)\right) V_{n-1}(x, y + 1) \tag{11}
\end{align}
$$

$$V_n(x,y) = \lambda_1' V_{n-1}(x+1,y) + \tag{12}$$
$$\lambda_2' \max\{V_{n-1}(x,y), V_{n-1}(x,y+1) + b_2 R\} + \tag{13}$$
$$\mu_1' x V_{n-1}(x-1,y) + \tag{14}$$
$$\mu_2' y V_{n-1}(x,y-1) + \tag{15}$$
$$\mu_2' V_{n-1}(x,y) + \tag{16}$$
$$\left(C \left(\frac{\mu_1'}{b_1} + \frac{\mu_2'}{b_2} \right) - \mu_1' x - \mu_2'(y+1) \right) V_{n-1}(x,y) \tag{17}$$

Please note that $(6) \leq (12)$, $(8) \leq (14)$, $(9) \leq (15)$, $(11) \leq (17)$ are proved directly by induction assumption. On the other hand $(10) = (16)$.

Following we show the justification of $(7) \leq (13)$:

1. If $(\max\{V_{n-1}(x,y+1), V_{n-1}(x,y+2) + b_2 R\} = V_{n-1}(x,y+1)$ and $\max\{V_{n-1}(x,y), V_{n-1}(x,y+1) + b_2 R\} = V_{n-1}(x,y))$ or $(\max\{V_{n-1}(x,y+1), V_{n-1}(x,y+2) + b_2 R\} = V_{n-1}(x,y+2) + b_2 R$ and $\max\{V_{n-1}(x,y), V_{n-1}(x,y+1) + b_2 R\} = V_{n-1}(x,y+1) + b_2 R)$, the inequality is a direct consequence of induction hypothesis.
2. If $(\max\{V_{n-1}(x,y+1), V_{n-1}(x,y+2) + b_2 R\} = V_{n-1}(x,y+1)$ and $\max\{V_{n-1}(x,y), V_{n-1}(x,y+1) + b_2 R\} = V_{n-1}(x,y+1) + b_2 R)$, it is simple to note the validity of the demonstration because $R > 0$.
3. If $(\max\{V_{n-1}(x,y+1), V_{n-1}(x,y+2) + b_2 R\} = V_{n-1}(x,y+2) + b_2 R$ and $\max\{V_{n-1}(x,y), V_{n-1}(x,y+1) + b_2 R\} = V_{n-1}(x,y))$, then we have that $V_{n-1}(x,y) > V_{n-1}(x,y+1) + b_2 R \geq V_{n-1}(x,y+2) + b_2 R$.

– case 2: $C - b_1 < b_1 x + b_2(y+1) \leq C$;
   According to Eq. (1), we define the number of preempted SUs (z and z') as
   $z = \left[\frac{b_1 x + b_2(y+1) - C + b_1}{b_2} + \mathbb{1}_{\{(b_1 x + b_2(y+1) - C + b_1)\%b_2 \neq 0\}} \right]$ and
   $z' = \max \left\{ \left[\frac{b_1 x + b_2 y - C + b_1}{b_2} + \mathbb{1}_{\{(b_1 x + b_2 y - C + b_1)\%b_2 \neq 0\}} \right], 0 \right\}$; we have:

$$V_n(x,y+1) = \lambda_1' (V_{n-1}(x+1,y+1-z) - z b_2 K) + \tag{18}$$
$$\lambda_2' \max\{V_{n-1}(x,y+1), V_{n-1}(x,y+2) + b_2 R\} + \tag{19}$$
$$\mu_1' x V_{n-1}(x-1,y+1) + \tag{20}$$
$$\mu_2' y V_{n-1}(x,y) + \tag{21}$$
$$\mu_2' V_{n-1}(x,y) + \tag{22}$$
$$\left(C \left(\frac{\mu_1'}{b_1} + \frac{\mu_2'}{b_2} \right) - \mu_1' x - \mu_2'(y+1) \right) V_{n-1}(x,y+1) \tag{23}$$

$$V_n(x,y) = \lambda_1' (V_{n-1}(x+1,y-z') - z' b_2 K) + \tag{24}$$
$$\lambda_2' \max\{V_{n-1}(x,y), V_{n-1}(x,y+1) + b_2 R\} + \tag{25}$$
$$\mu_1' x V_{n-1}(x-1,y) + \tag{26}$$
$$\mu_2' y V_{n-1}(x,y-1) + \tag{27}$$
$$\mu_2' V_{n-1}(x,y) + \tag{28}$$
$$\left(C \left(\frac{\mu_1'}{b_1} + \frac{\mu_2'}{b_2} \right) - \mu_1' x - \mu_2'(y+1) \right) V_{n-1}(x,y) \tag{29}$$

On one hand, we have $(22) = (28)$; and $(20) \leq (26)$, $(21) \leq (27)$, $(23) \leq (29)$ are proved directly with the hypothesis of induction. On the other hand, the proof of $(19) \leq (25)$

is analogous to the previous case[2]. Finally, please note that $z' = z - 1$ and $K > 0$, then the inequality $(18) \leq (24)$ is proved.

B Proof of Proposition 2

Proof Inequality holds for $n = 0$ by definition. Assuming the inequality holds for $n - 1$, we show that it holds for n. We divide the problem in two cases: preempted and non-preempted situations.

– case 1: $b_1 x + b_2(y + 1) \leq C - b_1$;

$$
\begin{aligned}
V_n(x, y + 1) - V_n(x, y) = {} & \lambda'_1 (V_{n-1}(x + 1, y + 1) - V_{n-1}(x + 1, y)) + \\
& \lambda'_2 \max\{V_{n-1}(x, y + 1), V_{n-1}(x, y + 2) + b_2 R\} - \\
& \lambda'_2 \max\{V_{n-1}(x, y), V_{n-1}(x, y + 1) + b_2 R\} + \\
& \mu'_1 x (V_{n-1}(x - 1, y + 1) - V_{n-1}(x - 1, y)) + \\
& \mu'_2 y (V_{n-1}(x, y) - V_{n-1}(x, y - 1)) + \\
& \mu'_2 V_{n-1}(x, y) - \mu'_2 V_{n-1}(x, y) + \\
& \left(C \left(\frac{\mu'_1}{b_1} + \frac{\mu'_2}{b_2} \right) - \mu'_1 x - \mu'_2(y + 1) \right) (V_{n-1}(x, y + 1) - V_{n-1}(x, y)) \\
\geq {} & -b_2 K
\end{aligned}
$$

Using induction hypothesis we have $V_n(x, y + 1) - V_n(x, y) \geq (1 - \lambda'_2 - \alpha')(-b_2 K) - \mu'_2 (V_{n-1}(x, y + 1) - V_{n-1}(x, y)) + \lambda'_2 (\max\{V_{n-1}(x, y + 1), V_{n-1}(x, y + 2) + b_2 R\} - \max\{V_{n-1}(x, y), V_{n-1}(x, y + 1) + b_2 R\})$. We can observe that the second term is always positive or zero (being zero the worst case). Then, if $K \geq R$ we can prove that $\lambda'_2 (\max\{V_{n-1}(x, y + 1), V_{n-1}(x, y + 2) + b_2 R\} - \max\{V_{n-1}(x, y), V_{n-1}(x, y + 1) + b_2 R\}) \geq -\lambda'_2 b_2 K$. This proves the result $\forall \alpha$.

– case 2: $C - b_1 < b_1 x + b_2(y + 1) \leq C$;
  In the same way as in the previous case, we define z and z' as
  $z = \left[\frac{b_1 x + b_2(y + 1) - C + b_1}{b_2} + \mathbb{1}_{\{(b_1 x + b_2(y + 1) - C + b_1) \% b_2 \neq 0\}} \right]$ and
  $z' = \max \left\{ \left[\frac{b_1 x + b_2 y - C + b_1}{b_2} + \mathbb{1}_{\{(b_1 x + b_2 y - C + b_1) \% b_2 \neq 0\}} \right], 0 \right\}$; we have:

$$
\begin{aligned}
V_n(x, y + 1) - V_n(x, y) = {} & \lambda'_1 (V_{n-1}(x + 1, y + 1 - z) - z b_2 K - V_{n-1}(x + 1, y - z') + z' b_2 K) + \\
& \lambda'_2 \max\{V_{n-1}(x, y + 1), V_{n-1}(x, y + 2) + b_2 R\} - \\
& \lambda'_2 \max\{V_{n-1}(x, y), V_{n-1}(x, y + 1) + b_2 R\} + \\
& \mu'_1 x (V_{n-1}(x - 1, y + 1) - V_{n-1}(x - 1, y)) + \\
& \mu'_2 y (V_{n-1}(x, y) - V_{n-1}(x, y - 1)) + \\
& \mu'_2 V_{n-1}(x, y) - \mu'_2 V_{n-1}(x, y) + \\
& \left(C \left(\frac{\mu'_1}{b_1} + \frac{\mu'_2}{b_2} \right) - \mu'_1 x - \mu'_2(y + 1) \right) (V_{n-1}(x, y + 1) - V_{n-1}(x, y)) \\
\geq {} & -b_2 K
\end{aligned}
$$

We can observe that $z' = z - 1$ and using the same arguments as in case 1, the proof is completed. When we have to evaluate the terms that implies the decision (the maximization), we have used that $K \geq R$.

---

[2] If the number of busy channels is greater than $C - b_2$, the admission control decision is known, then the number of combinations to evaluate is fewer than in case 1

C Proof of Proposition 3

Proof Inequality holds for $n = 0$ by definition. Assuming the inequality holds for $n - 1$, we show that it holds for n. We divide the problem in two cases: preempted and non-preempted situations.

– case 1: $b_1(x+1) + b_2(y+1) \leq C - b_1$;

$$
\begin{aligned}
V_n(x+1, y+1) - V_n(x+1, y) = {} & \lambda_1'(V_{n-1}(x+2, y+1) - V_{n-1}(x+2, y)) + & (30) \\
& \lambda_2'(\max\{V_{n-1}(x+1, y+1), V_{n-1}(x+1, y+2) + b_2 R\} - \\
& \max\{V_{n-1}(x+1, y), V_{n-1}(x+1, y+1) + b_2 R\}) + & (31) \\
& \mu_1' x(V_{n-1}(x, y+1) - V_{n-1}(x, y)) + & (32) \\
& \mu_2' y(V_{n-1}(x+1, y) - V_{n-1}(x+1, y-1)) + & (33) \\
& \mu_1'(V_{n-1}(x, y+1) - V_{n-1}(x, y)) + & (34) \\
& \left(C \left(\frac{\mu_1'}{b_1} + \frac{\mu_2'}{b_2} \right) - \mu_1'(x+1) - \mu_2'(y+1) \right) \\
& (V_{n-1}(x+1, y+1) - V_{n-1}(x+1, y)) & (35)
\end{aligned}
$$

$$
\begin{aligned}
V_n(x, y+1) - V_n(x, y) = {} & \lambda_1'(V_{n-1}(x+1, y+1) - V_{n-1}(x+1, y)) + & (36) \\
& \lambda_2'(\max\{V_{n-1}(x, y+1), V_{n-1}(x, y+2) + b_2 R\} - \\
& \max\{V_{n-1}(x, y), V_{n-1}(x, y+1) + b_2 R\}) + & (37) \\
& \mu_1' x(V_{n-1}(x-1, y+1) - V_{n-1}(x-1, y)) + & (38) \\
& \mu_2' y(V_{n-1}(x, y) - V_{n-1}(x, y-1)) + & (39) \\
& \mu_1'(V_{n-1}(x, y+1) - V_{n-1}(x, y)) + & (40) \\
& \left(C \left(\frac{\mu_1'}{b_1} + \frac{\mu_2'}{b_2} \right) - \mu_1'(x+1) - \mu_2'(y+1) \right) \\
& (V_{n-1}(x, y+1) - V_{n-1}(x, y)) & (41)
\end{aligned}
$$

$(30) \leq (36)$, $(32) \leq (38)$, $(33) \leq (39)$, $(35) \leq (41)$ are proved directly by induction hypothesis. In addition, we have $(34) = (40)$.

The last inequality requires a degree of algebraic manipulation:
$\lambda_2'(\max\{V_{n-1}(x, y+1), V_{n-1}(x, y+2) + b_2 R\} - \max\{V_{n-1}(x, y), V_{n-1}(x, y+1) + b_2 R\}) =$
$\lambda_2'(\max\{V_{n-1}(x, y+2) - V_{n-1}(x, y+1) + b_2 R, 0\} - \max\{V_{n-1}(x, y) - V_{n-1}(x, y+1), b_2 R\}) =$
$\lambda_2'(\max\{V_{n-1}(x, y+2) - V_{n-1}(x, y+1) + b_2 R, 0\} + \min\{-V_{n-1}(x, y) + V_{n-1}(x, y+1), -b_2 R\})$
$\geq$
$\lambda_2'(\max\{V_{n-1}(x+1, y+2) - V_{n-1}(x+1, y+1) + b_2 R, 0\} + \min\{-V_{n-1}(x+1, y) + V_{n-1}(x+1, y+1), -b_2 R\}) =$
$\lambda_2'(\max\{V_{n-1}(x+1, y+1), V_{n-1}(x+1, y+2) + b_2 R\} - \max\{V_{n-1}(x+1, y), V_{n-1}(x+1, y+1) + b_2 R\})$

– case 2: $C - b_1 < b_1(x+1) + b_2(y+1) \leq C$;
Defining z, z', z^+ and z^* according to Eq. (1):
$z = \left[\frac{b_1(x+1) + b_2(y+1) - C + b_1}{b_2} + \mathbb{1}_{\{(b_1(x+1) + b_2(y+1) - C + b_1)\%b_2 \neq 0\}} \right]$,
$z' = \max \left\{ \left[\frac{b_1(x+1) + b_2 y - C + b_1}{b_2} + \mathbb{1}_{\{(b_1(x+1) + b_2 y - C + b_1)\%b_2 \neq 0\}} \right], 0 \right\}$,
$z^+ = \max \left\{ \left[\frac{b_1 x + b_2(y+1) - C + b_1}{b_2} + \mathbb{1}_{\{(b_1 x + b_2(y+1) - C + b_1)\%b_2 \neq 0\}} \right], 0 \right\}$ and

$$z^* = \max\left\{\left[\frac{b_1 x + b_2 y - C + b_1}{b_2} + \mathbb{1}_{\{(b_1 x + b_2 y - C + b_1)\%b_2 \neq 0\}}\right], 0\right\},$$

$$
\begin{aligned}
V_n(x+1, y+1) - V_n(x+1, y) =\ & \lambda_1'(V_{n-1}(x+2, y+1-z) - zb_2 K) - \\
& \lambda_1'(V_{n-1}(x+2, y-z') - z'b_2 K) + && (42) \\
& \lambda_2'(\max\{V_{n-1}(x+1, y+1), V_{n-1}(x+1, y+2) + b_2 R\} - \\
& \max\{V_{n-1}(x+1, y), V_{n-1}(x+1, y+1) + b_2 R\}) + && (43) \\
& \mu_1' x(V_{n-1}(x, y+1) - V_{n-1}(x, y)) + && (44) \\
& \mu_2' y(V_{n-1}(x+1, y) - V_{n-1}(x+1, y-1)) + && (45) \\
& \mu_1'(V_{n-1}(x, y+1) - V_{n-1}(x, y)) + && (46) \\
& \left(C\left(\frac{\mu_1'}{b_1} + \frac{\mu_2'}{b_2}\right) - \mu_1'(x+1) - \mu_2'(y+1)\right) && (47) \\
& (V_{n-1}(x+1, y+1) - V_{n-1}(x+1, y))
\end{aligned}
$$

$$
\begin{aligned}
V_n(x, y+1) - V_n(x, y) =\ & \lambda_1'(V_{n-1}(x+1, y+1-z^+) - z^+ b_2 K) - \\
& \lambda_1(V_{n-1}(x+1, y-z^*) - z^* b_2 K) + && (48) \\
& \lambda_2'(\max\{V_{n-1}(x, y+1), V_{n-1}(x, y+2) + b_2 R\} && (49) \\
& - \max\{V_{n-1}(x, y), V_{n-1}(x, y+1) + b_2 R\}) + \\
& \mu_1' x(V_{n-1}(x-1, y+1) - V_{n-1}(x-1, y)) + && (50) \\
& \mu_2' y(V_{n-1}(x, y) - V_{n-1}(x, y-1)) + && (51) \\
& \mu_1'(V_{n-1}(x, y+1) - V_{n-1}(x, y)) + && (52) \\
& \left(C\left(\frac{\mu_1'}{b_1} + \frac{\mu_2'}{b_2}\right) - \mu_1'(x+1) - \mu_2'(y+1)\right) && (53) \\
& (V_{n-1}(x, y+1) - V_{n-1}(x, y))
\end{aligned}
$$

(42) $\leq$ (48) is valid according to Prop. 2 and observing that $z' = z - 1$ and $z^* = z^+ - 1$. The other inequalities can be demonstrated in the same way as in the previous case. In particular, we have to be careful in the proof of (43) $\leq$ (49) in the scenarios where the number of allocated channels is greater $C - b_2$.

D Proof of Proposition 4

Proof Inequality holds for $n = 0$ by definition. Assuming the inequality holds for $n - 1$, we show that it holds for n. We divide the problem in two cases: preempted and non-preempted situations.

– case 1: $b_1 x + b_2(y+2) \leq C - b_1$;

$$
\begin{aligned}
V_n(x, y) - V_n(x, y+1) =\ & \lambda_1'(V_{n-1}(x+1, y) - V_{n-1}(x+1, y+1)) + && (54) \\
& \lambda_2'(\max\{V_{n-1}(x, y), V_{n-1}(x, y+1) + b_2 R\} && (55) \\
& - \max\{V_{n-1}(x, y+1), V_{n-1}(x, y+2) + b_2 R\}) + \\
& \mu_1' x(V_{n-1}(x-1, y) - V_{n-1}(x-1, y+1)) + && (56) \\
& \mu_2' y(V_{n-1}(x, y-1) - V_{n-1}(x, y)) + && (57) \\
& \mu_2' V_{n-1}(x, y) - \mu_2' V_{n-1}(x, y+1) + && (58) \\
& \left(C\left(\frac{\mu_1'}{b_1} + \frac{\mu_2'}{b_2}\right) - \mu_1' x - \mu_2'(y+2)\right) && (59) \\
& (V_{n-1}(x, y) - V_{n-1}(x, y+1))
\end{aligned}
$$

$$V_n(x, y+1) - V_n(x, y+2) = \lambda_1'(V_{n-1}(x+1, y+1) - V_{n-1}(x+1, y+2)) + \quad (60)$$
$$\lambda_2'(\max\{V_{n-1}(x, y+1), V_{n-1}(x, y+2) + b_2R\} \quad (61)$$
$$- \max\{V_{n-1}(x, y+2), V_{n-1}(x, y+3) + b_2R\}) +$$
$$\mu_1'x(V_{n-1}(x-1, y+1) - V_{n-1}(x-1, y+2)) + \quad (62)$$
$$\mu_2'y(V_{n-1}(x, y) - V_{n-1}(x, y+1)) + \quad (63)$$
$$\mu_2'V_{n-1}(x, y) - \mu_2'V_{n-1}(x, y+1) + \quad (64)$$
$$\left(C\left(\frac{\mu_1'}{b_1} + \frac{\mu_2'}{b_2}\right) - \mu_1'x - \mu_2'(y+2)\right) \quad (65)$$
$$(V_{n-1}(x, y+1) - V_{n-1}(x, y+2))$$

The relations $(54) \leq (60)$, $(56) \leq (62)$, $(57) \leq (63)$ and $(59) \leq (65)$ are direct consequences of the induction hypothesis. The (58) and (64) terms cancel out each other.

Next we show that $(61) \geq (55)$:
$\lambda_2'(\max\{V_{n-1}(x, y+1), V_{n-1}(x, y+2) + b_2R\} - \max\{V_{n-1}(x, y+2), V_{n-1}(x, y+3) + b_2R\}) = \lambda_2'(\max\{V_{n-1}(x, y+1) - V_{n-1}(x, y+2), b_2R\} - \max\{0, V_{n-1}(x, y+3) - V_{n-1}(x, y+2) + b_2R\}) = \lambda_2'(\max\{V_{n-1}(x, y+1) - V_{n-1}(x, y+2), b_2R\} + \min\{0, -V_{n-1}(x, y+3) + V_{n-1}(x, y+2) - b_2R\}) \geq \lambda_2'(\max\{V_{n-1}(x, y) - V_{n-1}(x, y+1), b_2R\} + \min\{0, -V_{n-1}(x, y+2) + V_{n-1}(x, y+1) - b_2R\}) = \lambda_2'(\max\{V_{n-1}(x, y) - V_{n-1}(x, y+1), b_2R\} - \max\{0, V_{n-1}(x, y+2) - V_{n-1}(x, y+1) + b_2R\}) = \lambda_2'(\max\{V_{n-1}(x, y), V_{n-1}(x, y+1) + b_2R\} - \max\{V_{n-1}(x, y+1), V_{n-1}(x, y+2) + b_2R\})$

– case 2: $C - b_1 < b_1x + b_2(y+2) \leq C$;
Defining z, z' and z^* according to Eq. (1):
$z = \left[\frac{b_1x + b_2(y+2) - C + b_1}{b_2} + \mathbb{1}_{\{(b_1x + b_2(y+2) - C + b_1)\%b_2 \neq 0\}}\right]$,
$z' = \max\left\{\left[\frac{b_1x + b_2(y+1) - C + b_1}{b_2} + \mathbb{1}_{\{(b_1x + b_2(y+1)y - C + b_1)\%b_2 \neq 0\}}\right], 0\right\}$ and
$z^* = \max\left\{\left[\frac{b_1x + b_2y - C + b_1}{b_2} + \mathbb{1}_{\{(b_1x + b_2y - C + b_1)\%b_2 \neq 0\}}\right], 0\right\}$,

$$V_n(x, y) - V_n(x, y+1) = \lambda_1'(V_{n-1}(x+1, y-z^*) - z^*b_2K) - \quad (66)$$
$$\lambda_1'(V_{n-1}(x+1, y+1-z') - z'b_2K)$$
$$\lambda_2'(\max\{V_{n-1}(x, y), V_{n-1}(x, y+1) + b_2R\} \quad (67)$$
$$- \max\{V_{n-1}(x, y+1), V_{n-1}(x, y+2) + b_2R\}) +$$
$$\mu_1'x(V_{n-1}(x-1, y) - V_{n-1}(x-1, y+1)) + \quad (68)$$
$$\mu_2'y(V_{n-1}(x, y-1) - V_{n-1}(x, y)) + \quad (69)$$
$$\mu_2'V_{n-1}(x, y) - \mu_2'V_{n-1}(x, y+1) + \quad (70)$$
$$\left(C\left(\frac{\mu_1'}{b_1} + \frac{\mu_2'}{b_2}\right) - \mu_1'x - \mu_2'(y+2)\right) \quad (71)$$
$$(V_{n-1}(x, y) - V_{n-1}(x, y+1))$$

$$
\begin{aligned}
V_n(x, y+1) - V_n(x, y+2) = \; & \lambda_1'\left(V_{n-1}(x+1, y+1-z') - z'b_2K\right) - && (72)\\
& \lambda_1'\left(V_{n-1}(x+1, y+2-z) - zb_2K\right) + \\
& \lambda_2'(\max\{V_{n-1}(x, y+1), V_{n-1}(x, y+2) + b_2R\} - && (73)\\
& \max\{V_{n-1}(x, y+2), V_{n-1}(x, y+3) + b_2R\}) + \\
& \mu_1'x(V_{n-1}(x-1, y+1) - V_{n-1}(x-1, y+2)) + && (74)\\
& \mu_2'y(V_{n-1}(x, y) - V_{n-1}(x, y+1)) + && (75)\\
& \mu_2'V_{n-1}(x, y) - \mu_2'V_{n-1}(x, y+1) + && (76)\\
& \left(C\left(\frac{\mu_1'}{b_1} + \frac{\mu_2'}{b_2}\right) - \mu_1'x - \mu_2'(y+2)\right) && (77)\\
& (V_{n-1}(x, y+1) - V_{n-1}(x, y+2))
\end{aligned}
$$

(66) $\leq$ (72) is true according to Prop. 2 (in particular, the result of Prop. 2 is used in the limit case: $z' = z^* = 0$). The relations (68) $\leq$ (74), (69) $\leq$ (75) and (71) $\leq$ (77) are direct consequences of the induction hypothesis, and (70) = (76) is trivial. In order to analyze the inequality (67) $\leq$ (73) the same arguments as case 1 can be applied. It is important to highlight that if the number of busy resources in the system is greater than $C - b_2$, the admission control decision has no sense (no SU will be accepted). So, if for example $b_1x + b_2(y+2) > C - b_2$ (and $b_1x + b_2(y+1) \leq C - b_2$), we can demonstrate the inequality in the following way: $\lambda_2'(\max\{V_{n-1}(x, y+1), V_{n-1}(x, y+2) + b_2R\} - V_{n-1}(x, y+2)) = \lambda_2'(\max\{V_{n-1}(x, y+1) - V_{n-1}(x, y+2), b_2R\}) \geq \lambda_2'(\max\{V_{n-1}(x, y) - V_{n-1}(x, y+1), b_2R\}) = \lambda_2'(\max\{V_{n-1}(x, y), V_{n-1}(x, y+1) + b_2R\} - V_{n-1}(x, y+1))$, then
- if $\max\{V_{n-1}(x, y+1), V_{n-1}(x, y+2) + b_2R\} = V_{n-1}(x, y+1)$ the proof is completed.
- if $\max\{V_{n-1}(x, y+1), V_{n-1}(x, y+2) + b_2R\} = V_{n-1}(x, y+2) + b_2R$, then $\lambda_2(\max\{V_{n-1}(x, y), V_{n-1}(x, y+1) + b_2R\} - V_{n-1}(x, y+1)) \geq \lambda_2(\max\{V_{n-1}(x, y), V_{n-1}(x, y+1) + b_2R\} - V_{n-1}(x, y+2) - b_2R)$ and the proof is over.

Acknowledgments

References

1. Promoting efficient use of spectrum through elimination of barriers to the development of secondary markets - report and order and further notice of proposed rulemaking, wt docket no. 00-230. Tech. rep., Federal Communications Commission (FCC) (2003)
2. Nprm 04-186 unlicensed operation in the tv broadcast bands. Tech. rep., Federal Communications Commission (FCC) (2004)
3. Ieee standard for information technology– local and metropolitan area networks– specific requirements– part 22: Cognitive wireless ran medium access control (mac) and physical layer (phy) specifications: Policies and procedures for operation in the tv bands. IEEE Std 802.22-2011 pp. 1–680 (2011). DOI 10.1109/IEEESTD.2011.5951707
4. Ieee standard for information technology - telecommunications and information exchange between systems - local and metropolitan area networks - specific requirements - part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 5: Television white spaces (tvws) operation. IEEE Std 802.11af-2013 (Amendment to IEEE Std 802.11-2012, as amended by IEEE Std 802.11ae-2012, IEEE Std 802.11aa-2012, IEEE Std 802.11ad-2012, and IEEE Std 802.11ac-2013) pp. 1–198 (2014). DOI 10.1109/IEEESTD.2014.6744566

5. Akyildiz, I.F., Lee, W.Y., Vuran, M.C., Mohanty, S.: Next generation/dynamic spectrum access/cognitive radio wireless networks: A survey. Computer Networks **50**(13), 2127 – 2159 (2006). DOI 10.1016/j.comnet.2006.05.001

6. Alagoz, O., Ayvaci, M.U.: Uniformization in Markov Decision Processes. John Wiley & Sons, Inc. (2010). DOI 10.1002/9780470400531.eorms0934

7. Backens, J., Xin, C., Song, M.: A novel protocol for transparent and simultaneous spectrum access between the secondary user and the primary user in cognitive radio networks. Computer Communications **69**, 98 – 106 (2015). DOI 10.1016/j.comcom.2015.07.004

8. Bertsekas, D.P.: Dynamic Programming and Optimal Control, 2nd edn. Athena Scientific (2000)

9. Blaschke, V., Jaekel, H., Renk, T., Kloeck, C., Jondral, F.: Occupation measurements supporting dynamic spectrum allocation for cognitive radio design. In: Cognitive Radio Oriented Wireless Networks and Communications, 2007. CrownCom 2007. 2nd International Conference on, pp. 50–57 (2007). DOI 10.1109/CROWNCOM.2007.4549771

10. Conti, M., Chong, S., Fdida, S., Jia, W., Karl, H., Lin, Y.D., Mhnen, P., Maier, M., Molva, R., Uhlig, S., Zukerman, M.: Research challenges towards the future internet. Computer Communications **34**(18), 2115 – 2134 (2011). DOI 10.1016/j.comcom.2011.09.001

11. Cui, C., Man, H., Wang, Y., Liu, S.: Optimal cooperative spectrum aware opportunistic routing in cognitive radio ad hoc networks. Wireless Personal Communications **91**(1), 101–118 (2016). DOI 10.1007/s11277-016-3447-x. URL http://dx.doi.org/10.1007/s11277-016-3447-x

12. Das, D., Das, S.: A survey on spectrum occupancy measurement for cognitive radio. Wireless Personal Communications **85**(4), 2581–2598 (2015). DOI 10.1007/s11277-015-2921-1. URL http://dx.doi.org/10.1007/s11277-015-2921-1

13. De Nardis, L., Holland, O.: Deployment Scenarios for Cognitive Radio, pp. 49–116. Springer International Publishing, Cham (2014). DOI 10.1007/978-3-319-04022-6_2. URL http://dx.doi.org/10.1007/978-3-319-04022-6_2

14. El-Taha, M., Jr., S.S.: Sample-Path Analysis of Queueing Systems. Springer US (1999). DOI 10.1007/978-1-4615-5721-0

15. Hassan, M.S., Abusara, A., Din, M.S.E., Ismail, M.H.: On efficient channel modeling for video transmission over cognitive radio networks. Wireless Personal Communications **91**(2), 919–932 (2016). DOI 10.1007/s11277-016-3504-5. URL http://dx.doi.org/10.1007/s11277-016-3504-5

16. Haykin, S.: Cognitive radio: brain-empowered wireless communications. Selected Areas in Communications, IEEE Journal on **23**(2), 201–220 (2005). DOI 10.1109/JSAC.2004.839380

17. Jiang, C., Duan, L., Huang, J.: Joint spectrum pricing and admission control for heterogeneous secondary users. In: Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt), 2014 12th International Symposium on, pp. 497–504 (2014). DOI 10.1109/WIOPT.2014.6850338

18. Kim, B., Ko, S.S.: Optimal admission control and state space reduction in two-class preemptive loss systems. ETRI Journal **37**(5), 917–921 (2015). DOI 10.4218/etrij.15.0114.0348. URL http://dx.doi.org/10.4218/etrij.15.0114.0348

19. Liu, Z., Nain, P., Towsley, D.: Sample path methods in the control of queues. Queueing Systems **21**(3), 293–335 (1995). DOI 10.1007/BF01149166

20. McHenry, M.A., McCloskey, D., Roberson, D., MacDonald, J.T.: Nsf spectrum occupancy measurements project summary. Tech. rep., Shared Spectrum Co., and IIT Wireless Interference Lab Illinois Institute of Technology (2005)

21. McMenamy, J., Macaluso, I., Marchetti, N., Doyle, L.: A methodology to help operators share the spectrum through an enhanced form of carrier aggregation. In: Dynamic Spectrum Access Networks (DYSPAN), IEEE International Symposium on, pp. 334–345 (2014). DOI 10.1109/DySPAN.2014.6817810

22. Mitola, J., Maguire G.Q., J.: Cognitive radio: making software radios more personal. IEEE Personal Communications **6**(4), 13–18 (1999). DOI 10.1109/98.788210

23. Niyato, D., Hossain, E.: Competitive pricing for spectrum sharing in cognitive radio networks: Dynamic game, inefficiency of nash equilibrium, and collusion. IEEE

Journal on Selected Areas in Communications **26**(1), 192–202 (2008). DOI 10.1109/JSAC.2008.080117

24. Ormeci, E., van der Wal, J.: Admission policies for a two class loss system with general interarrival times. Stochastic Models **22**(1), 37–53 (2006). DOI 10.1080/15326340500481721

25. Pashenkova, E., Rish, I., Dechter, R.: Value iteration and policy iteration algorithms for markov decision problem. Tech. rep., Department of Information and Computer Science, University of California at Irvine (1996)

26. Puterman, M.L.: Markov Decision Processes: Discrete Stochastic Dynamic Programming, 1st edn. John Wiley & Sons, Inc., New York, NY, USA (1994)

27. Puterman, M.L., Shin, M.C.: Modified policy iteration algorithms for discounted markov decision problems. Management Science **24**(11), 1127–1137 (1978). DOI 10.1287/mnsc.24.11.1127

28. Raiss-El-Fenni, M., El-Azouzi, R., El-Kamili, M., Ibrahimi, K., Bouyakhf, E.H.: Dynamic spectrum allocation with admission control based on cognitive radio for qos support in multiple wireless network. EURASIP Journal on Wireless Communications and Networking **2012**(1), 1–13 (2012). DOI 10.1186/1687-1499-2012-296

29. Ross, S.M.: Stochastic Processes, 2nd edn. John Wiley & Sons (1996)

30. Savin, S.V., Cohen, M.A., Gans, N., Katalan, Z.: Capacity management in rental businesses with two customer bases. Operations Research **53**(4), 617–631 (2005). DOI 10.1287/opre.1040.0193

31. Sucasas, V., Radwan, A., Mumtaz, S., Rodriguez, J.: Effect of noisy channels in mac-based ssdf counter-mechanisms for 5g cognitive radio networks. In: 2015 International Symposium on Wireless Communication Systems (ISWCS), pp. 736–740 (2015). DOI 10.1109/ISWCS.2015.7454447

32. Turhan, A., Alanyali, M., Starobinski, D.: Dynamic pricing of preemptive service for elastic demand. In: 2012 IEEE 51st IEEE Conference on Decision and Control (CDC), pp. 2084–2089 (2012). DOI 10.1109/CDC.2012.6426304

33. Turhan, A., Alanyali, M., Starobinski, D.: Optimal admission control of secondary users in preemptive cognitive radio networks. In: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt), 10th International Symposium on, pp. 138–144 (2012)

34. Ulukus, M.Y., Gullu, R., Ormeci, L.: Admission and termination control of a two class loss system. Stochastic Models **27**(1), 2–25 (2011). DOI 10.1080/15326349.2011.542719

35. Wang, B., Liu, K.: Advances in cognitive radio networks: A survey. Selected Topics in Signal Processing, IEEE Journal of **5**(1), 5–23 (2011). DOI 10.1109/JSTSP.2010.2093210

36. Wang, X., Li, Z., Xu, P., Xu, Y., Gao, X., Chen, H.H.: Spectrum sharing in cognitive radio networks – an auction-based approach. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on **40**(3), 587–596 (2010). DOI 10.1109/TSMCB.2009.2034630

37. Wang, Z.q., Jiang, L.g., He, C.: Price-based power control algorithm in cognitive radio networks based on monotone optimization. Journal of Shanghai Jiaotong University (Science) **20**(6), 654–659 (2015). DOI 10.1007/s12204-015-1673-0

38. Wu, G., Ren, P., Du, Q.: Dynamic spectrum auction with time optimization in cognitive radio networks. In: IEEE Vehicular Technology Conference (VTC Fall), pp. 1–5 (2012). DOI 10.1109/VTCFall.2012.6398966

39. Xie, L., Xiang, J., Zhang, Y.: Revenue-based admission control for cognitive radio cellular systems. In: Communications and Networking in China, 2008. ChinaCom 2008. Third International Conference on, pp. 1200–1204 (2008). DOI 10.1109/CHINACOM.2008.4685241

40. Yu, H., Gao, L., Li, Z., Wang, X., Hossain, E.: Pricing for uplink power control in cognitive radio networks. Vehicular Technology, IEEE Transactions on **59**(4), 1769–1778 (2010)

41. Zhan, S.C., Chang, S.C.: Double auction design for short-interval and heterogeneous spectrum sharing. IEEE Transactions on Cognitive Communications and Networking **2**(1), 83–94 (2016). DOI 10.1109/TCCN.2016.2570799

42. Zhang, Y., Lee, C., Niyato, D., Wang, P.: Auction approaches for resource allocation in wireless systems: A survey. IEEE Communications Surveys Tutorials **15**(3), 1020–1041 (2013). DOI 10.1109/SURV.2012.110112.00125
43. Zhao, Q., Wang, B., Li, X.: Two classes of improved quickest spectrum sensing approaches in cognitive radio. Wireless Personal Communications **91**(2), 755–769 (2016). DOI 10.1007/s11277-016-3494-3. URL http://dx.doi.org/10.1007/s11277-016-3494-3
44. Zhu, Y., Sun, W., Yu, J., Liu, T., Li, B.: Distributed spectrum sharing in cognitive radio networks: A pricing-based decomposition approach. International Journal of Distributed Sensor Networks **2014** (2014). DOI 10.1155/2014/262137