Learning to Compete for Resources in Wireless Stochastic Games

Fangwen Fu, and Miheala van der Schaar, *Senior Member, IEEE*

***Abstract*— In this paper, we model the various users in a wireless network (e.g. cognitive radio network) as a collection of selfish, autonomous agents that strategically interact in order to acquire the dynamically available spectrum opportunities. Our main focus is on developing solutions for wireless users to successfully compete with each other for the limited and time-varying spectrum opportunities, given experienced dynamics in the wireless network. To analyze the interactions among users given the environment disturbance, we propose a stochastic game framework for modeling how the competition among users for spectrum opportunities evolves over time. At each stage of the stochastic game, a central spectrum moderator auctions the available resources and the users strategically bid for the required resources. The joint bid actions affect the resource allocation and hence, the rewards and future strategies of all users. Based on the observed resource allocations and corresponding rewards, we propose a best response learning algorithm that can be deployed by wireless users to improve their bidding policy at each stage. The simulation results show that by deploying the proposed best response learning algorithm, the wireless users can significantly improve their own bidding strategies and hence, their performance, in terms of both the application quality and the incurred cost for the used resources.**

***Index Terms*— Delay-Sensitive Transmission, Multi-user Resource Management, Wireless Networks, Interactive Learning, Reinforcement Learning, Stochastic Games.**

I. INTRODUCTION

The Dynamic resource management in heterogeneous wireless networks is a challenging problem [3]. The wireless stations and radio systems that must coexist in such a network differ in their individual utility functions, transmission actions, resource demands, and capabilities. Thus, various levels of strategic[1] interaction and adaptation are necessary to cope with the widely varying dynamics. In this paper we focus on synthesizing new, dynamic and informationally- decentralized resource management mechanisms for achieving high utility in competitive and heterogeneous wireless networks (including cognitive radio networks [1][2][3]). Specifically, our focus is on designing associated communication algorithms that enable self-interested, autonomous wireless stations to strategically compete for the available spectrum resources in either ISM bands [1] or bands shared with licensed users, according to a priori mandated or negotiated rules.

Our paper is primarily concerned with the tensions and relationships among autonomous adaptation by secondary (unlicensed) users (SUs), the competition among these users, the interaction of these users with spectrum moderators having their own goals, e.g. making money, imposing fairness rules, ensuring compliance to FCC [1], and local regulations with respect to primary (licensed) users (PUs) etc. Unlike the previous works on resource management [6][21][26], our main focus is on discussing how users can adapt, predict, learn and determine how they compete for the time-varying resources, and how they select the associated transmission strategies, given the experienced "dynamics".

In wireless networks, these dynamics can be categorized into two types: one is the *disturbance due to the "environment"* and the other is the *impact caused by competing users*. The disturbance due to the environment results from variations (uncertainties) of the wireless channels or source (e.g. multimedia) characteristics. For example, the stochastic behavior of the primary users, the time-varying channel conditions experienced by the SUs and the time-varying source traffic that needs to be transmitted by the SUs can be considered as environment disturbances. These types of dynamics are generally modeled as stationary processes. For instance, the usage of each channel by the primary users can be modeled as a two-state Markov chain with ON (the channel is used by PUs) and OFF (the channel is available for the SUs) states [7]. The channel conditions can be modeled using a finite state Markov model [24]. The packet arrival of the source traffic can be modeled as a Poisson process[2] [11].

Conventionally, wireless stations have only considered these environment disturbances when adapting their cross-layer strategies [12] for delay-sensitive transmission. The other type of dynamics - the impact from competing users, which is due to the non-collaborative, autonomous and strategic SUs in the network transmitting their traffic - is less well studied in wireless communication networks.

Manuscript received August 28, 2007; revised April, 27, 2008. This work was supported in part by the National Science Foundation under Grant XXX and in part by the University of California under the UC MICRO Program.

F. Fu and M. van der Schaar are with the Department of Electrical Engineering, University of California at Los Angeles, Los Angeles, CA 90095 USA (e-mail: {fwfu, mihaela}@ee.ucla.edu).

---

[1] By strategic users we mean users which are not price-takers, and do not have an a-priori consensus on resources allocation.

[2] Other packet arrival models can also been used in our proposed framework.

The goal of this paper is to provide solutions and associated metrics that can be used by an autonomous SU to analyze and predict the outcome of various dynamic interactions among competing SUs in dynamic multi-user communication systems and, based on this forecast, adapt and optimize its transmission strategy. In our considered wireless networks, the SUs are modeled as rational and strategic. We model the spectrum management as a stochastic game [22] in which the SUs simultaneously and repeatedly make their own resource bids. The competition for the dynamic resources is assisted by a central coordinator (similar to that in existing wireless LAN standards such as 802.11e HCF [13]). We refer to this coordinator as the central spectrum moderator (CSM). The role of the CSM is to allocate resources to the SUs based on pre-determined utility maximization rule[3].

In this paper, to explicitly consider the strategic behavior of the autonomous SUs and the informationally-decentralized nature of the competition for wireless resources, we assume that the CSM deploys an auction mechanism for dynamically allocating resources. Auction theory has been extensively studied in economics [19] and it has also been recently applied to network resource allocation [4][5][6]. Note that the role of the CSM [4] in our resource management game for our considered wireless networks will be kept minimal. Unlike alternative existing solutions [21], the CSM will not require knowledge of the private information of the users and will not perform complex computations for deciding the resource allocation. Its only role will be the implementation of the spectrum etiquette rules as in [8], and ensuring that the available spectrum holes are auctioned among users. In order to capture the network dynamics, we allow the CSM to *repeatedly* auction the available spectrum opportunities based on the PUs' behaviors. Meanwhile, each SUs is allowed to strategically adapt its bidding strategy based on information about the available spectrum opportunities, its source and channel characteristics, and the impact of the other SUs bidding actions.

Using this stochastic wireless allocation framework, we develop a learning methodology for SUs to improve their policies for playing the auction game, i.e. the policies for generating the bids to compete for the available resources. Specifically, during the repeated multi-user interaction, the SUs can observe partial historic information of the outcome of the auction game, through which the SUs can estimate the impact on their future rewards and then adopt their best response in order to effectively compete for the channel opportunities. The estimation of the impact on the expected future reward can be performed using different types of interactive learning [18]. In this paper, we focus on reinforcement learning [17][27] because this allows the SUs to improve their bidding strategy based only on the knowledge of their own past received payoffs, without knowing the bids or payoffs of the other SUs. Our proposed best response learning algorithm is inspired from the Q-learning for the single agent interacting with environment. Unlike Q-learning, the proposed best response learning explicitly considers the interactions and coupling among SUs in the wireless network. By deploying the best response learning algorithm, the SUs can strategically predict the impact of current actions on future performance and then optimally make their resource bids.

The paper is organized as follows. In Section II, we introduce a stochastic game formulation for multi-user interaction in wireless networks. In Section III, we show how a one-stage auction mechanism can be used to divide the spectrum allocation among strategic SUs. In Section IV, we present the state definition, state transition model and stage reward function for the SUs in the stochastic game. In Section V, we discuss the bidding strategies of the SUs for playing the stochastic game. In Section VI, we propose a best response learning approach for the SUs to predict their future rewards based on the observed historic information. In Section VII, we present the simulation results, followed by the conclusions and future research in Section VIII.

II. STOCHASTIC GAME FORMULATION FOR DYNAMIC MULTI-USER INTERACTION

We consider a spectrum consisting of N channels, each indexed by $j \in \{1,\ldots,N\}$. The N wireless channels are originally licensed to a primary network (PN) whose users (i.e. PUs) exclusively access the channels. In the secondary network (SN), M $(M \geq N)$ autonomous SUs, each indexed by $i \in \{1,\ldots,M\}$, transmitting delay-sensitive data compete for the spectrum opportunities released by the PUs in these N channels. Although the available transmission opportunities (TxOps) for SUs depend on the access patterns of PUs and the detection systems [2], we do not discuss the detection methods in this paper, but rather rely on the existing literature for this purpose [3]. Instead, we assume that the available TxOps in each channel change over time due to the primary users joining or leaving the network and can be modeled as a two-state Markov chain as in [7][10]. Our goal is to develop a general framework for multi-user interaction in the SN, where users can compete for the dynamically available TxOps. Moreover, we also aim to provide solutions for SUs to improve their strategies for playing the repeated resource management game, by considering their past interactions with other SUs.

The communications of the PUs are assumed to follow a synchronous slot structure. The time slot has length ΔT seconds. We assume that during each time slot, each channel is either exclusively occupied by PUs or that there is no PU accessing the channel [7][10]. Hence, during each time slot,

---

[3] Other fairness rules can also be deployed in the CSM such as air-time fairness, utility-based fairness, etc. [12]

[4] It should be noted that this approach can also allow for multiple CSMs to manage the spectrum, by dividing their responsibilities fairly, e.g. based on their geo-location or frequency band in which they are operating, or by competing against each other for the number of SUs that will associated with them.

the channel is in one of the following two states: ON (this channel is currently used by the PUs) or OFF (this channel is not used by the PUs and hence, the SUs can use this channel). Note that if this is an unlicensed band, the channel will always be in the OFF mode, and can be utilized by the SUs at all times. The TxOp of channel j at time slot $t \in \mathbb{N}$ is denoted by $y_j^t \in \{0,1\}$, with y_j^t being 0 if the channel is in the ON state, and being 1 if it is in the OFF state. In this paper, the TxOp y_j^t of channel j is modeled by a two-state Markov chain with transition probability $p_j^{FN} = p\left(y_j^{t+1} = 0 \mid y_j^t = 1\right)$ and $p_j^{NF} = p\left(y_j^{t+1} = 1 \mid y_j^t = 0\right)$. The TxOp profile of the N channels is represented by $\boldsymbol{y}^t = \left[y_1^t, ..., y_N^t\right]$.

As in [13], we assume that a polling-based medium access protocol is deployed in the SN, which is arbitrated by a CSM. The polling policy is changed only at the start of every time slot. For simplicity, we assume that each SU can access a single channel and that each channel can be accessed by a single SU within the time slot. The SUs can switch the channels only when crossing time slots. Note that this simple medium access model used for illustration in this paper can be easily extended to more sophisticated models [10], where each SU can simultaneously access multiple channels or the channels are being shared by multiple SUs etc. When using this time division channel access, we assume that the wireless users deploy constant transmission power and experience no interference. Furthermore, we assume that the wireless users move slowly and thus, their experienced channel conditions change slowly.

During each time slot, an SU needs to first determine how to compete with the other SUs for the time-varying TxOps. This represents its external actions, since they determine the interaction between this SU and the other SUs and determine the amount of resources allocated to that SU. The external actions at time slot t are denoted by $a_i^t \in A_i$, where A_i is the set of possible external actions available to SU i. Based on the allocated resources, the SU determines how to transmit its traffic (application layer data) by selecting the various strategies at the different layers of the OSI stack (e.g. through cross-layer adaptation [12]). These actions are referred to as internal actions, since they only determine the SU's utility at the current time. The internal actions at time slot t are denoted by $b_i^t \in B_i$, where B_i is the set of possible internal actions available to SU i. In this paper, we propose an auction mechanism deployed in the CSM. Hence, the external action a_i^t of SU i is the bid it submits to CSM. The auction mechanism will be detailed in Section III. The environment experienced by an SU i can be characterized by its current "state" $s_i^t \in S_i$ which will be discussed in Section IV. At each time slot t, SU i generates the external action a_i^t to compete for the TxOps $\boldsymbol{y}^t$. The competition result is ϑ_i^t, based on which SU i performs its internal action b_i^t and obtains the reward r_i^t at this time slot. After the packet transmission, SU i transits to the next state $s_i^{t+1} \in S_i$. The conceptual overview of the multi-SUs interactions in the repeated auctions is illustrated in Figure 1. The repeated competition among the SUs can be modeled as a stochastic game [16][22]. The time slot corresponds to the term "stage", which is commonly used in stochastic games. In the remainder of the paper, we use the terms "time slot" and "stage" interchangeably.

We define the stochastic game for the SN resource allocation as $\left\langle \langle S_i, A_i, B_i, O_i, q_i, r_i \rangle_{i=1}^M, \mathcal{Y} \right\rangle$, where each SU i is associated with a tuple $\langle S_i, A_i, B_i, O_i, q_i, r_i \rangle$. Specifically,

- $\mathcal{Y}$ is finite set of possible TxOps available for SUs. In this paper, $\mathcal{Y}=\{0,1\}^N$ and $\boldsymbol{y}^t \in \mathcal{Y}$ is the available TxOps at stage t which is a common information for SUs.
- S_i is a finite local state space of SU i. We let $S := \prod_{k=1}^N S_k$ be the global state space of all SUs and $S_{\text{-}i} := \prod_{k \neq i} S_k$ be the global state space of SUs other than i. At stage t, the global state is denoted by $\boldsymbol{s}^t = \left(s_1^t, \cdots, s_M^t\right) = \left(s_i^t, \boldsymbol{s}_{-i}^t\right)$ where $-i$ represents all the SUs other than i.
- A_i is a finite set of external actions performed by SU i to compete for the available TxOps. The external action vector at stage t for all SUs will be $\boldsymbol{a}^t = \left(a_1^t, \cdots, a_M^t\right)$.
- B_i is a finite set of internal actions performed by SU i to determine the packet transmission.
- O_i is a finite set of possible output from multi-SU competition. In this paper, the output $\vartheta_i^t \in O_i$ is the auction result computed by the CSM for SU i at stage t. We will give the specific form of the output in Section III.
- q_i is the state transition probability for SU i. Thus, $q_i\left(s_i^{t+1}, \boldsymbol{y}^{t+1} \mid s_i^t, \boldsymbol{y}^t, \vartheta_i^t, b_i^t\right)$ is the probability that the state of SU i transits from s_i^t to s_i^{t+1} and the TxOp transits from $\boldsymbol{y}^t$ to $\boldsymbol{y}^{t+1}$ if the competition output is ϑ_i^t and the internal action is b_i^t. The reason that the transition probability includes the common TxOp $\boldsymbol{y}^t$ is because the channel condition transition of SU i depends on the available TxOp.
- r_i is the stage reward (immediate reward) received by SU i, where $r_i : (S_i, O_i, B_i) \mapsto \mathbb{R}$. It should be noted that the reward function r_i depends on the competition output and hence, indirectly depends on other SUs' external actions.

To design a stochastic game for the SN with strategic SUs, we have to consider: (i) what auction mechanism can be deployed to resolve the competition among the SUs; (ii) how the dynamic environment experienced by each SU can be modeled; and (iii) how the SUs can forecast the impact of their bids made at the current time, on their future performance.

III. AUCTION MECHANISM – ONE STAGE RESOURCE ALLOCATION

In this paper, we assume that the CSM is aware of the TxOp $\boldsymbol{y}^t$ and allocates (through polling the SUs) those channels with $y_j^t = 1$ to the SUs. To efficiently allocate the available resources (opportunities), the CSM needs to collect information about the SUs [21]. However, as mentioned in Section **Error! Reference source not found.**, in a wireless network, the information is decentralized, and thus, the information exchange between the SUs and the CSM needs to be kept limited due to the incurred communication cost. On the other hand, the SUs competing with each other are selfish and strategic, and hence, the information they hold is private and they may not desire to reveal this information to the CSM. Therefore, one of our key interests in this paper is to determine what information should be exchanged between the SUs and the CSM, and how this information should be exchanged. In the following, we present an auction mechanism for dynamically coordinating the interactions among SUs and discuss the computation complexity in the CSM and the communication cost between SUs and CSM.

First, the CSM announces the auction by broadcasting the TxOp $\boldsymbol{y}^t$. The SUs receive the announcement and determine the external action (i.e. the bid vector) $a_i^t = \left[a_{i1}^t, \ldots, a_{iN}^t\right] \in \mathbb{R}^N$ based on the announced information and their own private information about the environment they experience, which is discussed in details in Section IV. Subsequently, each SU submits the bid vector to the CSM. After receiving the bid vectors from the SUs, the CSM computes the channel allocation $\boldsymbol{z}_i^t = \left[z_{i1}^t, \ldots, z_{iN}^t\right] \in \{0,1\}^N$ for each SU i based on the submitted bids. To compel the SUs to declare their bids truthfully [23], the CSM also computes the payment $\tau_i^t \in \mathbb{R}_-$ that the SUs have to pay for the use of resources during the current stage of the game. The negative value of the payment means the absolute value that SU i has to pay the CSM for the used resources. Hence, the competition output ϑ_i^t in this auction mechanism includes the channel allocation $\boldsymbol{z}_i^t$ and the payment τ_i^t, i.e. $\vartheta_i^t = \left(\boldsymbol{z}_i^t, \tau_i^t\right)$. The competition output is then transmitted back to the SUs. The computation of the channel allocation $\boldsymbol{z}_i^t$ and payment τ_i^t is described as follows.

After each SU submits the bid vector, the CSM performs two computations: (i) channel allocation and (ii) payment computation. Note that most existing multi-user wireless resource allocation solutions can be modeled as such repeated auctions for resources. If the resources are priced or the users may lie about their resource needs, taxes associated with the resource usage will need to be imposed [14]. Otherwise, these taxes can be considered to be zero throughout the paper.

We denote the channel allocation matrix $Z^t = [z_{ij}^t]_{M \times N}$ with z_{ij}^t being 1 if channel j is assigned to SU i, otherwise 0. The feasible set of channel assignments is denoted as $\mathcal{Z}^t = \{Z^t \mid \sum_{i=1}^M z_{ij}^t = y_j^t, \forall j, \sum_{j=1}^N z_{ij}^t \le 1, \forall i, z_{ij}^t \in \{0,1\}\}$. The channel allocation matrix without the presence of SU i is denoted $Z_{-i}^t = [z_{kj}^t]_{(M-1)\times N}$ and the corresponding feasible set is $\mathcal{Z}_{-i}^t = \{Z_{-i}^t \mid \sum_{k=1,k\neq i}^M z_{kj}^t = y_j^t, \forall j, \sum_{j=1}^N z_{kj}^t \le 1, \forall k \neq i, z_{kj}^t \in \{0,1\}\}$, where $-i = \{1, \ldots, i-1, i+1, \ldots, M\}$. During the first phase, the CSM allocates the channels to SUs based on its adopted fairness rule, e.g. maximizing the total "social welfare"[5]:

$$Z^{t,opt} = \arg\max_{Z^t \in \mathcal{Z}^t} \sum_{i=1}^M \sum_{j=1}^N z_{ij}^t a_{ij}^t \,. \tag{1}$$

If the resources are priced, we will consider in this paper, for illustration, a second price auction mechanism [19][23] for determining the tax that needs to be paid by SU i based on the above optimal channel assignment $Z^{t,opt} = [z_{ij}^{t,opt}]_{M\times N}$. This tax equals:

$$\tau_i^t = \sum_{k=1,k\neq i}^M \sum_{j=1}^N z_{kj}^{t,opt} a_{kj}^t - \max_{Z_{-i}^t \in \mathcal{Z}_{-i}^t} \sum_{k=1,k\neq i}^M \sum_{j=1}^N z_{kj}^t a_{kj}^t \,. \tag{2}$$

Note that when $N = 1$, the generalized auction mechanism presented above becomes the well-known second price auction [19]. Although the optimization problems in Eqs. (1) and (2) are discrete optimizations, they can be efficiently solved using linear programming. As argued in [20], the linear optimization problem can be solved in polynomial time and hence, the CSM requires only limited computational complexity.

The information exchange between the CSM and the SUs is illustrated in Figure 2. From Figure 2, we note that, at each stage, the CSM first broadcasts the available TxOps to all the SUs for the auction and then, each SU submits its own bid vector over all the available TxOps. After receiving the bids, the CSM computes the auction results and sends back to the users the channel allocations and the corresponding payments. The signaling required for the auction is most often implemented at the application layer. In the worst case, the amount of data communicated between the CSM to the SUs equals $(M+1)N + nN$ bits, where n is the amount of bits representing the payment for each SU. The amount of data communicated by each SU to the CSM is $n'N$ bits, where n' is the amount of bits representing the bid submitted to the CSM on each channel.

Compared to traditional one-stage resource allocation methods, our proposed auction mechanism has the following advantages:

---

[5] Note that other fairness solutions than maximizing the social welfare could be adopted and this will not influence our proposed solution.

- Unlike traditional centralized resource allocation methods [30], our proposed auction mechanism is not required to know the SUs' utility functions or preferences, which is often the private information of the users and is not common knowledge. In fact, our auction mechanism only requires the SUs to submit their bid vectors for the available TxOps. The bid vector computation is performed by the SUs, but not the CSM, based on their utilities, preferences, action sets, experienced environment characteristics etc.
- Unlike traditional decentralized resource allocation methods [28] where multiple iterations are required before convergence, our proposed auction mechanism only requires the SUs to submit the bid vectors once. Hence, our proposed auction mechanism is suitable for on-line resource management. Moreover, we do not assume as in [29] that users are price-takers and that there is consensus about what is a fair distribution of the resources. Instead, in the proposed framework, users are strategic and are able to determine their own bid vectors for resources based on their knowledge, utilities, preferences etc.

IV. USER MODELING IN THE STOCHASTIC GAME FRAMEWORK

A. *Definition of SU States*

As discussed in the introduction, each SU needs to cope with two types of "uncertainties": disturbances from the environment and interactions with other SUs. The environment is characterized by the packet arrivals from the source (i.e. source/traffic characterization) connected with the transmitter and the channel conditions. In this section, we will illustrate how these disturbances can be modeled. However, note that other models of the environment existing in the literature can be adopted. The use of a specific model will only affect the performance of the proposed solution, and not the general framework for multi-user interaction proposed in this paper.

For illustration, we assume that each SU i maintains a buffer with limited size X_i, which can be interpreted as a time window that specifies which packets are considered for transmission at each time based on their delay deadlines. Expired packets are dropped from the buffer. This model has been extensively used for delay-sensitive data transmission, e.g. leaky bucket model for video transmission [25]. The number of packets in the buffer at time slot t is denoted as x_i^t ($0 \le x_i^t \le X_i$). We assume that the packets arrive from the source at the beginning of each time slot, i.e. x_i^t is updated only at the beginning of a time slot. The number of packets arriving into the buffer during one time slot is a random variable independent of the time t and denoted as χ_i. χ_i follows the Poisson distribution with the average arrival rate $\overline{\chi}_i$ packets/second [11]. However, note that the Poisson process is simply used for illustration purposes and other traffic models (e.g. renewal process, etc.) can also be used in our framework. The average number of packets arriving during one time slot equals $\overline{\chi}_i \Delta T$ [11].

The condition of channel j experienced by SU i is represented by the Signal-to-Noise Ratio (SNR) and denoted as ρ_{ij}^t in dB. When $y_j^t = 1$, we assume that the channel condition of each channel can be represented by a set of discrete SNR values, i.e. $\rho_{ij}^t \in \{\sigma_{ij}^1, ..., \sigma_{ij}^K\}$. Note that the number of discrete SNR values, K, can be determined by SU i by trading-off the complexity (larger K leads to a larger state space) and the resulting impact on the performance. When $y_j^t = 0$, we set ρ_{ij}^t equal to $-\infty$ which means that the channel is unavailable to SUs at that time. As shown in [24], when $y_j^t = 1$, the channel condition (in terms of SNR) can also be modeled as a finite-state Markov chain, where the transition from channel condition σ_{ij}^l at time t to channel condition σ_{ij}^k at time $t+1$ takes place with probability $p_{ij}^{l \to k}$. These transitions probabilities can be easily estimated by SU i, by repeatedly interacting with the channel. We denote by $p_{ij}^{-\infty \to k}$ the probability that the channel condition is σ_{ij}^k at time $t+1$, knowing that $y_j^t = 0$ and $y_j^{t+1} = 1$. The probability that the channel condition transition to $-\infty$, knowing that $y_j^{t+1} = 0$, is 1 on matter what condition the channel j is at time t. Then the combination $\left(y_j^t, \rho_{ij}^t\right)$ is still a Markov chain with state transition probability as follows:

$$
p\left(y_j^{t+1}, \rho_{ij}^{t+1} \mid y_j^t, \rho_{ij}^t\right) =
\begin{cases}
\left(1 - p_j^{FN}\right) p_{ij}^{l \to k} & \mathit{if}\ y_j^t = 1, \rho_{ij}^t = \sigma_{ij}^l, y_j^{t+1} = 1, \rho_{ij}^{t+1} = \sigma_{ij}^k \\
p_j^{NF} p_{ij}^{-\infty \to k} & \mathit{if}\ y_j^t = 0, y_j^{t+1} = 1, \rho_{ij}^{t+1} = \sigma_{ij}^k \\
p_j^{FN} & \mathit{if}\ y_j^t = 1, \rho_{ij}^t = \sigma_{ij}^l, y_j^t = 0 \\
1 - p_j^{NF} & o.w.
\end{cases}
\tag{3}
$$

To model the dynamics experienced by SU i at time t in the SN, we define a "state" $s_i^t = (v_i^t, \boldsymbol{\rho}_i^t) \in \mathcal{S}_i$, where $\boldsymbol{\rho}_i^t = \left(\rho_{i1}^t, \cdots, \rho_{iN}^t\right)$. The state encapsulates the current buffer state as well as the state of each channel. $\mathcal{S}_i$ is the set of possible states[6]. The total number of possible states for SU i equals $|\mathcal{S}_i| = (X_i + 1) \times (K + 1)^N$. We will show later in this paper that the state information is sufficient for SU i to compete for resources (make bid vector) at the current time.

B. *State Transition and Stage Reward*

We will now discuss the state transition process. Remember that the state of SU i includes the buffer state v_i^t and the channel state $\boldsymbol{\rho}_i^t$. In this paper, we assume that the channel state transition is independent of the buffer state transition. In

---

[6] We assume that the channel state and the transmission buffer independently evolve as time goes by.

the above, we describe the transition of the channel state $\boldsymbol{\rho}_i^t$ and the TxOp $\boldsymbol{y}^t$. The buffer state transition is determined by the number of packets arriving and the channel allocation $\boldsymbol{z}_i^t$ as well as the internal action b_i^t during that time slot. The number of packets transmitted at stage t is denoted by $\mathcal{N}_i\left(s_i^t, \boldsymbol{z}_i^t, b_i^t\right)$. Given the channel allocation, SU i can adapt its own internal action to maximize the number of transmitted packets, i.e.

$$n_i\left(s_i^t, \boldsymbol{z}_i^t\right) = \max_{b_i^t \in B_i} \mathcal{N}_i\left(s_i^t, \boldsymbol{z}_i^t, b_i^t\right) \tag{4}$$

The optimization can be performed by a cross-layer adaptation algorithm as in [5][12][21]. Since our focus is on the multi-SU interaction, we assume that the internal action will be always performed in order to maximize the number of transmitted packets. We simply use $n_i\left(s_i^t, \boldsymbol{z}_i^t\right)$ to represent the number of transmitted packets and omit the internal actions in the following notations.

The evolution of the buffer state is captured by the equation $v_i^{t+1} = \min\{(v_i^t - n\left(s_i^t, \boldsymbol{z}_i^t\right))^+ + \chi_i^t, X_i\}$. We define $h = v_i^{t+1} - (v_i^t - n\left(s_i^t, \boldsymbol{z}_i^t\right))^+$. Based on the packet arrival model, the buffer state transition probability is computed as

$$p_i^{buf}(v_i^{t+1} \mid v_i^t, \boldsymbol{z}_i^t) =$$
$$\begin{cases} \dfrac{(\mu_i \Delta T)^h e^{-\mu_i \Delta T}}{h!}, & if\ 0 \le h < X_i - (v_i^t - n\left(s_i^t, \boldsymbol{z}_i^t\right))^+ \\ \displaystyle\sum_{k=h}^{\infty} \dfrac{(\mu_i \Delta T)^k e^{-\mu_i \Delta T}}{k!}, & if\ h = X_i - (v_i^t - n\left(s_i^t, \boldsymbol{z}_i^t\right))^+ \end{cases} . \tag{5}$$

The state transition combined with TxOps, given current resource allocation $\boldsymbol{z}_i^t$, can be computed as

$$q_i(s_i^{t+1}, \boldsymbol{y}^{t+1} | s_i^t, \boldsymbol{y}^t, \boldsymbol{z}_i^t) =$$
$$\underbrace{p_i^{buf}(v_i^{t+1} \mid v_i^t, \boldsymbol{z}_i^t)}_{\text{buffer state transition}} \underbrace{\prod\nolimits_{j=1}^{N} p\left(y_j^{t+1}, \rho_{ij}^{t+1} \mid y_j^t, \rho_{ij}^t\right)}_{\text{channel state transition}}, \tag{6}$$

where the first term represents the buffer state transition, which is independent of the second term of the channel state transition.

Based on the channel allocation $\boldsymbol{z}_i^t$, the SU transmits the available packets in the buffer. In the next time slot, new packets arrive into the buffer. Newly incoming packets may lead to packets already existing in the buffer being dropped, whenever the buffer is full or their delay deadline has passed. Clearly, the performance of the application (e.g. video quality) improves when fewer packets are lost. Hence, we can interpret a negative value of the number of lost packets as the stage gain, which is denoted by g_i^t, i.e $g_i^t\left(s_i^t, \boldsymbol{z}_i^t\right) = -\left((v_i^t - n_i\left(s_i^t, \boldsymbol{z}_i^t\right))^+ + \chi_i^t - X_i\right)^+$. The reward at time t for SU i is expressed using the quasi-linear form $r_i\left(s_i^t, \vartheta_i^t\right) = g_i^t + \tau_i^t$. Note that the gain g_i^t and payment τ_i^t depend on the states and bids of all the competing SUs in the SN. Hence, the reward is also rewritten as $r_i\left(\boldsymbol{s}^t, \boldsymbol{y}^t, \boldsymbol{a}^t\right)$.

V. BIDDING STRATEGY FOR PLAYING THE STOCHASTIC GAME

A. *Best Response Bidding Policy*

In the SN, we assume that the stochastic game is played by all SUs for an infinite number of stages. This assumption is reasonable for applications having a long duration, such as video streaming. In our network setting, we define a history of the stochastic game up to time t as $\boldsymbol{h}^t = \{\boldsymbol{s}^0, \boldsymbol{y}^0, \boldsymbol{a}^0, \boldsymbol{z}^0, \boldsymbol{\tau}^0, ..., \boldsymbol{s}^{t-1}, \boldsymbol{y}^{t-1}, \boldsymbol{a}^{t-1}, \boldsymbol{z}^{t-1}, \boldsymbol{\tau}^{t-1}, \boldsymbol{s}^t, \boldsymbol{y}^t\}$ $\in \boldsymbol{\mathcal{H}}^t$, which summarizes all previous states, available TxOps, and the actions taken by the SUs as well as the outcomes at each stage of the auction game and $\boldsymbol{\mathcal{H}}^t$ is the set of all possible history up to time t. However, during the stochastic game, each SU i cannot observe the entire history, but rather part of the history $\boldsymbol{h}^t$. The observation of SU i is denoted as $o_i^t \in \boldsymbol{\mathcal{O}}_i^t$ and $\boldsymbol{o}_i^t \subset \boldsymbol{h}^t$. Note that the current state $\boldsymbol{s}_i^t$ can be always observed, i.e. $\boldsymbol{s}_i^t \in \boldsymbol{o}_i^t$. In this paper, we focus on the external action selection for the SUs. The external action selection for SU i to play the stochastic game is also referred to as a bidding policy $\pi_i^t : \boldsymbol{\mathcal{O}}_i^t \mapsto A_i$ for SU i at the time t and defined as a mapping from the observations up to the time t into the specific action, i.e. $\boldsymbol{a}_i^t = \pi_i^t(\boldsymbol{o}_i^t)$. Furthermore, a policy profile $\boldsymbol{\pi}_i$ for SU i aggregates the bidding policies about how to play the game over the entire course of the stochastic game, i.e. $\boldsymbol{\pi}_i = (\pi_i^0, ..., \pi_i^t, ...)$. The policy profile for all the SUs at time slot t is denoted as $\boldsymbol{\pi}^t = \left(\pi_1^t, ..., \pi_M^t\right) = \left(\pi_i^t, \boldsymbol{\pi}_{-i}^t\right)$.

The policy $\boldsymbol{\pi}_i$ is said to be Markov if the bidding policy π_i^t for $\forall t$ is, given the current state $\boldsymbol{s}_i^t$ and current available TxOp $\boldsymbol{y}^t$, independent of the states, TxOps and actions prior to the time t, i.e. $\pi_i^t(\boldsymbol{o}_i^t) = \pi_i^t(\boldsymbol{s}_i^t, \boldsymbol{y}^t)$. The policy $\boldsymbol{\pi}_i$ is said to be stationary, if the bidding policy $\pi_i^t = \pi_i$ for $\forall t$. The reward $r_i\left(\boldsymbol{s}^k, \boldsymbol{y}^k, \boldsymbol{a}^k\right)$ of the stage k is discounted by factor $(\alpha_i)^{k-t}$ at time t. The factor $\alpha_i\,(0 \le \alpha_i < 1)$ is the discounted factor determined by a specific application (for instance, for video streaming applications, this factor can be set based on the tolerable delay). The total discounted sum of rewards $Q_i^t(\boldsymbol{s}^t, \boldsymbol{y}^t, \boldsymbol{\pi})$ for SU i can be calculated at time t starting from the state profile $\boldsymbol{s}^t$, assuming that all SUs deploy stationary and Markov policies $\boldsymbol{\pi} = (\pi_i, \boldsymbol{\pi}_{-i})$, as:

$$\begin{aligned}
&Q_i^t(\boldsymbol{s}^t,\boldsymbol{y}^t,\boldsymbol{\pi}) = \textstyle\sum_{k=t}^{\infty}(\alpha_i)^{k-t} r_i(\boldsymbol{s}^k,\boldsymbol{y}^k,\boldsymbol{\pi}(\boldsymbol{s}^k,\boldsymbol{y}^k)) \\
&= \underbrace{r_i(\boldsymbol{s}^t,\boldsymbol{y}^t,\boldsymbol{\pi}(\boldsymbol{s}^t,\boldsymbol{y}^t))}_{\text{stage reward at time t}} + \\
&\underbrace{\alpha_i \sum_{\substack{\boldsymbol{s}^{t+1}\in\boldsymbol{S}\\ \boldsymbol{y}^{t+1}\in\{0,1\}^N}} \left\{\begin{array}{c} \displaystyle\prod_{k=1}^{M} q_k(s_k^{t+1},\boldsymbol{y}^{t+1}\mid s_k^t,\boldsymbol{y}^t,\boldsymbol{z}_k^t(\boldsymbol{\pi}(\boldsymbol{s}^t,\boldsymbol{y}^t))\times \\ Q_i^{t+1}(\boldsymbol{s}^{t+1},\boldsymbol{y}^{t+1},\boldsymbol{\pi}) \end{array}\right\}}_{\text{expected future reward}} \\
&= \{\underbrace{g_i^t(s_i^t,\boldsymbol{y}^t,\boldsymbol{z}_i^t(\boldsymbol{\pi}(\boldsymbol{s}^t,\boldsymbol{y}^t))) + \tau_i^t(\boldsymbol{\pi}(\boldsymbol{s}^t,\boldsymbol{y}^t))}_{\text{stage reward at time t}} + \\
&\underbrace{\alpha_i \sum_{\substack{\boldsymbol{s}^{t+1}\in\boldsymbol{S}\\ \boldsymbol{y}^{t+1}\in\{0,1\}^N}} \left\{\begin{array}{c} \displaystyle\prod_{k=1}^{M} q_k(s_k^{t+1},\boldsymbol{y}^{t+1}\mid s_k^t,\boldsymbol{y}^t,\boldsymbol{z}_k^t(\boldsymbol{\pi}(\boldsymbol{s}^t,\boldsymbol{y}^t))\times \\ Q_i^{t+1}(\boldsymbol{s}^{t+1},\boldsymbol{y}^{t+1},\boldsymbol{\pi}) \end{array}\right\}}_{\text{expected future reward}}, \qquad (7)
\end{aligned}$$

The total discounted sum of rewards in Eq. (7) consists of two parts: (i) the current stage reward and (ii) the expected future reward discounted by α_i. Note that SU i cannot independently determine the above value without explicitly knowing the policies and states of other SUs. The SU maximizes the total discounted sum of future rewards in order to select the bidding policy, which explicitly considers the impact of the current bid vector on the expected future rewards.

We define the *best response* β_i for SU i to other SUs' policies $\boldsymbol{\pi}_{-i}$ as

$$\beta_i(\boldsymbol{\pi}_{-i}) = \arg\max_{\pi_i} Q_i^t(\boldsymbol{s}^t,\boldsymbol{y}^t,(\pi_i,\boldsymbol{\pi}_{\text{-}i})) \qquad (8)$$

The central issue in our stochastic game is how the best response policies can be determined by the SUs. In the repeated auction mechanism discussed in Section III, the procedure each SU i follows to compete for the channel opportunities is illustrated in Figure 3. In this procedure, the bidding strategy π_i^t is continuously improved by the "bidding strategy improvement" module. In Section V.B, we discuss the challenges involved in building such a module, and in Section VI we develop a best response learning algorithm that can be used for improving the bidding strategy.

B. *Challenges for Selecting the Best Response Bidding Policy*

Recall that during each time slot, the CSM announces an auction based on the available TxOps and then SUs bid for the resources. To enable the successful deployment of this resource auction mechanism, we can prove, similarly to our prior work in [21], that SUs have no incentive to misrepresent their information, i.e. they adhere to the "truth telling" policy. We assume that at each time slot t, SU i has preference u_{ij}^t over the channel j, which capture the benefit derived when using that channel. The preference u_{ij}^t is interpreted as the benefit obtained by SU i when using channel j, compared to the benefit when this channel is not used. Note that this benefit also includes the expected future rewards. The optimal bid $a_{ij}^{t,opt}$ that SU i can take on the channel j at time t is the bid maximizing the net benefit $u_{ij}^t + \tau_i^t$. In auction discussed in Section III, the optimal bid that SU i can make is $a_{ij}^{t,opt} = u_{ij}^t$, i.e. the optimal bid for SU i is to announces its true preference to the CSM [21]. The proof is omitted here due to space limitations, since it is similar to that in [21]. The payment made by SU i is computed by the CSM based on the inconvenience incurred by other SUs due to SU i during that time slot [23].

Next, we define the preference u_{ij}^t in the context of the stochastic game model. Using the channel j, SU i obtains the immediate gain $g_i^t\left(s_i^t,\boldsymbol{y}^t,\boldsymbol{e}_j\right)$ by transmitting the packets in its buffer, where $\boldsymbol{e}_j$ indicates that channel j is allocated to SU i during the current time slot. SU i then moves into next state $\boldsymbol{s}_i^{t+1}$ from which it may obtain the future reward $Q_i^{t+1}\left(\boldsymbol{s}^{t+1},\boldsymbol{y}^{t+1},\boldsymbol{\pi}\right)$. On the other hand, if no channel is assigned to SU i, it receives the immediate gain $g_i^t\left(s_i^t,\boldsymbol{y}^t,\boldsymbol{0}\right)$ and then moves into the next state s_i^{t+1} from which it may obtain the future reward $Q_i^{t+1}\left(\boldsymbol{s}^{t+1},\boldsymbol{y}^{t+1},\boldsymbol{\pi}\right)$. We define a feasible set of channel assignments to SU i's opponents, given SU i's channel allocation $\boldsymbol{z}_i^t$, as $\boldsymbol{\mathcal{Z}}_{\text{-}i}^t(\boldsymbol{z}_i^t)$, with $\boldsymbol{\mathcal{Z}}_{\text{-}i}^t(\boldsymbol{z}_i^t) = \{Z_{-i}^t \mid \sum_{k=1,k\neq i}^{M} z_{kj}^t = y_j^t - \boldsymbol{z}_i^t, \forall j,$
$\sum_{j=1}^{N} z_{kj}^t \leq 1, \forall k \neq i,\, z_{kj}^t \in \{0,1\}\}$.

The preference over the current state can be then computed as

$$\begin{aligned}
&u_{ij}^t\left(\boldsymbol{s}^t,\boldsymbol{y}^t\right) = \\
&\left[\begin{array}{l} g_i^t\left(s_i^t,\boldsymbol{y}^t,\boldsymbol{e}_j\right) + \\ \alpha_i \displaystyle\sum_{\substack{\boldsymbol{s}^{t+1}\in\boldsymbol{S}\\ \boldsymbol{y}^{t+1}\in\{0,1\}^N}} \left[\begin{array}{l} q_i\left(s_i^{t+1},\boldsymbol{y}^{t+1}\mid s_i^t,\boldsymbol{y}^t,\boldsymbol{e}_j\right)\times \\ \displaystyle\sum_{Z_{-i}^t\in\boldsymbol{\mathcal{Z}}_{-i}^t(\boldsymbol{e}_j)} \left[\begin{array}{l} \displaystyle\prod_{k=1}^{M} q_k\left(s_k^{t+1},\boldsymbol{y}^{t+1}\mid s_k^t,\boldsymbol{y}^t,\boldsymbol{z}_k^t\right)\times \\ Q_i^{t+1}\left(\boldsymbol{s}^{t+1},\boldsymbol{y}^{t+1},\boldsymbol{\pi}\right) \end{array}\right] \end{array}\right] \end{array}\right] \\
&-\left[\begin{array}{l} g_i^t\left(s_i^t,\boldsymbol{y}^t,\boldsymbol{0}\right) + \\ \alpha_i \displaystyle\sum_{\substack{\boldsymbol{s}^{t+1}\in\boldsymbol{S}\\ \boldsymbol{y}^{t+1}\in\{0,1\}^N}} \left[\begin{array}{l} q_i\left(s_i^{t+1},\boldsymbol{y}^{t+1}\mid s_i^t,\boldsymbol{y}^t,\boldsymbol{0}\right)\times \\ \displaystyle\sum_{Z_{-i}^t\in\boldsymbol{\mathcal{Z}}_{-i}^t(\boldsymbol{0})} \left[\begin{array}{l} \displaystyle\prod_{k=1}^{M} q_k\left(s_k^{t+1},\boldsymbol{y}^{t+1}\mid s_k^t,\boldsymbol{y}^t,\boldsymbol{z}_k^t\right)\times \\ Q_i^{t+1}\left(\boldsymbol{s}^{t+1},\boldsymbol{y}^{t+1},\boldsymbol{\pi}\right) \end{array}\right] \end{array}\right] \end{array}\right]. \qquad (9)
\end{aligned}$$

From this equation, it is clear that the true value u_{ij}^t depends on its own current state $\boldsymbol{s}_i^t$, but also the other SUs' states $\boldsymbol{s}_{-i}^t$, the channel allocations $\boldsymbol{\mathcal{Z}}_{-i}^t\left(\boldsymbol{e}_j\right)$ to the other users when channel j is assigned to SU i, $\boldsymbol{\mathcal{Z}}_{-i}^t(\boldsymbol{0})$ when SU i is not assigned to any channel, and the state transition models $q_k\left(s_k^{t+1},\boldsymbol{y}^{t+1}\mid s_k^t,\boldsymbol{y}^t,\boldsymbol{z}_k^t\right),\forall k$. However, the other SUs' states, the channel allocations and the state transition models of other SUs are not known to SU i, and it is thus impossible for each SU to determine its preference $u_{ij}^t\left(\boldsymbol{s}^t, \boldsymbol{y}^t\right)$.

Without knowing the other SUs' states and state transition models, SU i cannot derive its optimal bidding strategy $a_{ij}^{t,opt} = u_{ij}^t\left(\boldsymbol{s}^t, \boldsymbol{y}^t\right)$. However, if SU i chooses the bid vector by only maximizing the immediate reward $g_i^t + \tau_i^t$, i.e. the total discounted sum of reward degenerates in $Q_i^t(\boldsymbol{s}^t, \boldsymbol{y}^t, \boldsymbol{\pi}) = g_i^t(s_i^t, \boldsymbol{y}^t, \boldsymbol{z}_i^t(\boldsymbol{\pi}(\boldsymbol{s}^t, \boldsymbol{y}^t))) + \tau_i^t(\boldsymbol{\pi}(\boldsymbol{s}^t, \boldsymbol{y}^t)$ by setting $\alpha_i = 0$. Then, the preference over channel j becomes $u_{ij}^t\left(\boldsymbol{s}^t, \boldsymbol{y}^t\right) = g_i^t\left(s_i^t, \boldsymbol{y}^t, \boldsymbol{e}_j\right) - g_i^t\left(s_i^t, \boldsymbol{y}^t, \mathbf{0}\right)$. Since now u_{ij}^t only depends on the state s_i^t, SU i can compute both the optimal bid vector as well as the optimal bidding policy. We refer to this optimal bidding policy as the "myopic" policy, since it only takes the immediate reward into consideration and ignores the future impact. The myopic policy is referred to as π_i^{myopic}. To solve the difficult problem of optimal bidding policy selection when $\alpha_i \neq 0$, an SU needs to forecast the impact of its current bidding actions on the expected future rewards discounted by α_i. The forecast can be performed using learning from its past experiences.

VI. INTERACTIVE LEARNING FOR PLAYING THE RESOURCE MANAGEMENT GAME

A. *How to Evaluate Learning Algorithms?*

In Section V.B, it was shown that an SU needs to know other SUs' states and state transition models in order to derive its own optimal bidding policy. This coupling among SUs is due to the shared nature of the wireless resources. However, an SU cannot exactly know the other SUs' models and private information in the wireless networks. Thus, to improve the bidding policy, an SU can only predict the impacts of dynamics (uncertainties) caused by the competing SUs based on its observations from past auctions. In this paper, we propose a learning algorithm for predicting these impacts. We define a learning algorithm $\mathcal{L}_i$ for SU i as a function taking the observation $\boldsymbol{o}_i^t$ as input and having the bidding policy π_i^t as output.

Before developing a learning algorithm, we first discuss how to evaluate the performance of a learning algorithm in terms of its impact on the SU's reward. Unlike existing multi-agent learning research, which is aimed at achieving converge to an equilibrium point for the interacting agents, we develop learning algorithms based on the performance of the bidding strategy on the SU's reward. We denote a bidding policy generated by the learning algorithm $\mathcal{L}_i$ as $\pi_i^{\mathcal{L}_i}$. An SU will learn in order to improve its bidding policy and its rewards from participating in the auction game. The performance of the bidding strategy π_i is defined as the time average reward that SU i obtains in a time window with length T when it adopts π_i:

$$\mathcal{V}^{\pi_i}(T) = \frac{1}{T}\sum_{k=1}^{T} r_i^k \tag{10}$$

Using this definition, the performance of two learning algorithms can be easily compared. For instance, given two algorithm $\mathcal{L}_i'$ and $\mathcal{L}_i''$, if $\mathcal{V}^{\pi_i^{\mathcal{L}_i'}} > \mathcal{V}^{\pi_i^{\mathcal{L}_i''}}$, then we say that learning algorithm $\mathcal{L}_i'$ is better than $\mathcal{L}_i''$.

B. *What Information to Learn from?*

First let us consider what information the SU can observe while playing the stochastic game in our SN. As shown in Figure 1, at the beginning of time slot t, the SUs submit the bids $a_i^t, \forall i$. Then, the CSM returns the channel allocation $z_i^t, \forall i$ and $\tau_i^t, \forall i$. If SU i is not allowed to observe the bids, the channel allocations and payments for other SUs, then the observation of SU i becomes $\boldsymbol{o}_i^t = \{\boldsymbol{s}_i^0, \boldsymbol{y}^0, a_i^0, \boldsymbol{z}_i^0, \boldsymbol{\tau}_i^0, \ldots, \boldsymbol{s}_i^{t-1}, \boldsymbol{y}^{t-1}, a_i^{t-1}, \boldsymbol{z}_i^{t-1}, \boldsymbol{\tau}_i^{t-1}, \boldsymbol{s}_i^t, \boldsymbol{y}^t\}$. If the information is exchanged among SUs or broadcasted and overheard by all SUs, the observed information by SU i becomes $\boldsymbol{o}_i^t = \{\boldsymbol{s}_i^0, \boldsymbol{y}^0, \boldsymbol{a}^0, \boldsymbol{z}^0, \boldsymbol{\tau}^0, \ldots, \boldsymbol{s}_i^{t-1}, \boldsymbol{y}^{t-1}, \boldsymbol{a}^{t-1}, \boldsymbol{z}^{t-1}, \boldsymbol{\tau}^{t-1}, \boldsymbol{s}_i^t, \boldsymbol{y}^t\}$. Now, the problem that needs to be solved by SU i is how it can improve its own policy for playing the game by learning from the observation $\boldsymbol{o}_i^t$. In this paper, we assume that SU i observes the information $\boldsymbol{o}_i^t = \{\boldsymbol{s}_i^0, \boldsymbol{y}^0, a_i^0, \boldsymbol{z}_i^0, \boldsymbol{\tau}_i^0, \ldots, \boldsymbol{s}_i^{t-1}, \boldsymbol{y}^{t-1}, a_i^{t-1}, \boldsymbol{z}_i^{t-1}, \boldsymbol{\tau}_i^{t-1}, \boldsymbol{s}_i^t, \boldsymbol{y}^t\}$.

C. *What to Learn?*

In Section VI.A, we introduced learning as a tool to predict the impacts of dynamics and hence, improve the bidding policy. However, a key question is what needs to be learned. Recall that the optimal bidding policy for SU i is to generate a bid vector that represents its preferences for using different channels. From Eq. (9), we can see that SU i needs to learn: (i) the state space of other SUs, i.e. $\boldsymbol{S}_{-i}$; (ii) the current state of other SUs, i.e. $\boldsymbol{s}_{-i}^t$; (iii) the transition probability of other SUs, i.e. $\prod_{k\neq i} q_k\left(s_k^{t+1}, \boldsymbol{y}^{t+1} \mid s_k^t, \boldsymbol{y}^t, \boldsymbol{z}_k^t\right)$; (iv) the resource allocation $\boldsymbol{\mathcal{Z}}_{\text{-}i}^t(\boldsymbol{e}_j), \forall j$ and $\boldsymbol{\mathcal{Z}}_{\text{-}i}^t(\mathbf{0})$; and (v) the discounted sum of rewards $Q_i^{t+1}\left(\boldsymbol{s}^{t+1}, \boldsymbol{y}^{t+1}, \boldsymbol{\pi}\right)$.

However, SU i can only observes the information $\boldsymbol{o}_i^t = \{\boldsymbol{s}_i^0, \boldsymbol{y}^0, a_i^0, \boldsymbol{z}_i^0, \boldsymbol{\tau}_i^0, \ldots, \boldsymbol{s}_i^{t-1}, \boldsymbol{y}^{t-1}, a_i^{t-1}, \boldsymbol{z}_i^{t-1}, \boldsymbol{\tau}_i^{t-1}, \boldsymbol{s}_i^t, \boldsymbol{y}^t\}$ from which SU i cannot accurately infer the other SUs' state space and transition probability. Moreover, capturing the exact information about other SUs requires heavy computational and storage complexity. Instead, we allow SU i to classify the space $\boldsymbol{S}_{-i}$ into H_i classes each of which is represented by a representative state $\tilde{s}_{-i,h}, h \in \{1, \ldots, H_i\}$. We discuss how the space $\boldsymbol{S}_{-i}$ is decomposed in Section VI.D. By dividing the state space $\boldsymbol{S}_{-i}$, the transition probability $\prod_{k\neq i} q_k\left(s_k^{t+1}, \boldsymbol{y}^{t+1} \mid s_k^t, \boldsymbol{y}^t, \boldsymbol{z}_k^t\right)$ is approximated by

$q_{-i}\left(\tilde{s}_{-i}^{t+1},\boldsymbol{y}^{t+1} \mid \tilde{s}_{-i}^{t},\boldsymbol{y}^{t},\boldsymbol{z}_i^{t}\right)$, where $\tilde{s}_{-i}^{t}$ and $\tilde{s}_{-i}^{t+1}$ are the representative states of the classes that $\boldsymbol{s}_{-i}^{t}$ and $\boldsymbol{s}_{-i}^{t+1}$ belong to. This approximation is performed by aggregating all other SUs' states into one representative state and assuming that the transition depends on the resource allocation $\boldsymbol{z}_i^t$. The transition probability approximation is also discussed in Section VI.D. The discounted sum of rewards $Q_i^{t+1}\left(\boldsymbol{s}^{t+1},\boldsymbol{y}^{t+1},\boldsymbol{\pi}\right)$ is approximated by $V_i^{t+1}\left(\left(s_i^{t+1},\tilde{s}_{-i}^{t+1}\right),\boldsymbol{y}^{t+1}\right)$. Note that the classification on the state space $\boldsymbol{S}_{-i}$ and approximation of the transition probability and discounted sum of rewards affects the learning performance. Hence, a user can tradeoff an increased complexity for an increased performance. After the classification, the preference computation can be approximated as

$$u_{ij}^t\left(\left(s_i^t,\tilde{s}_{-i}^t\right),\boldsymbol{y}^t\right)=\left[\begin{array}{l}g_i^t\left(s_i^t,\boldsymbol{y}^t,\boldsymbol{e}_j\right)+\\ \alpha_i\displaystyle\sum_{\substack{\left(s_i^{t+1},\tilde{s}_{-i}^{t+1}\right)\in\left(S_i,\tilde{\boldsymbol{S}}_{-i}\right)\\ \boldsymbol{y}^{t+1}\in\{0,1\}^N}}\left[\begin{array}{l}q_i\left(s_i^{t+1},\boldsymbol{y}^{t+1}\mid s_i^t,\boldsymbol{y}^t,\boldsymbol{e}_j\right)\times\\ q_{-i}\left(\tilde{s}_{-i}^{t+1},\boldsymbol{y}^{t+1}\mid \tilde{s}_{-i}^t,\boldsymbol{y}^t,\boldsymbol{e}_j\right)\times\\ V_i^{t+1}\left(\left(s_i^{t+1},\tilde{s}_{\text{-}i}^{t+1}\right),\boldsymbol{y}^{t+1}\right)\end{array}\right]\end{array}\right]-\left[\begin{array}{l}g_i^t\left(s_i^t,\boldsymbol{y}^t,\boldsymbol{0}\right)+\\ \alpha_i\displaystyle\sum_{\substack{\left(s_i^{t+1},\tilde{s}_{-i}^{t+1}\right)\in\left(S_i,\tilde{\boldsymbol{S}}_{-i}\right)\\ \boldsymbol{y}^{t+1}\in\{0,1\}^N}}\left[\begin{array}{l}q_i\left(\tilde{s}_i^{t+1},\boldsymbol{y}^{t+1}\mid s_i^t,\boldsymbol{y}^t,\boldsymbol{0}\right)\times\\ q_{-i}\left(\tilde{s}_{-i}^{t+1},\boldsymbol{y}^{t+1}\mid \tilde{s}_{-i}^t,\boldsymbol{y}^t,\boldsymbol{0}\right)\times\\ V_i^{t+1}\left(\left(s_i^{t+1},\tilde{s}_{\text{-}i}^{t+1}\right),\boldsymbol{y}^{t+1}\right)\end{array}\right]\end{array}\right] \quad (11)$$

In this setting, to find the approximated preference and thus, the approximated optimal bidding policy, we need to learn the following from the past observations: (i) how the space $\tilde{\boldsymbol{S}}_{-i}$ is classified; (ii) the transition probability $q_{-i}\left(\tilde{s}_{-i}^{t+1},\boldsymbol{y}^{t+1} \mid \tilde{s}_{-i}^{t},\boldsymbol{y}^{t},\boldsymbol{z}_i^{t}\right)$; (iii) the approximated future rewards $V_i^{t+1}\left(\left(s_i^{t+1},\tilde{s}_{\text{-}i}^{t+1}\right),\boldsymbol{y}^{t+1}\right)$.

D. *How to Learn?*

In this section, we develop a learning algorithm to estimate the terms listed in Section VI.C.

1) *Decomposition of the space $\boldsymbol{S}_{-i}$*

As discussed in Section VI.B, only $\boldsymbol{o}_i^t=\{\boldsymbol{s}_i^0,\boldsymbol{y}^0,a_i^0,\boldsymbol{z}_i^0,\boldsymbol{\tau}_i^0,\ldots,\boldsymbol{s}_i^{t-1},\boldsymbol{y}^{t-1},a_i^{t-1},\boldsymbol{z}_i^{t-1},\boldsymbol{\tau}_i^{t-1},\boldsymbol{s}_i^t,\boldsymbol{y}^t\}$ are observed. From the auction mechanism presented in Section III, we know that the value of the tax τ_i^t is computed based on the inconvenience that SU i causes to the other SUs. In other words, a higher value of $\left|\tau_i^t\right|$ indicates that the network is more congested[7]. Based on the bid vector $\boldsymbol{b}_i^t$, the channel allocation $\boldsymbol{z}_i^t$ and the tax τ_i^t, SU i can infer the network congestion and thus, indirectly, the resource requirements of the competing SUs. Instead of knowing the exact state space of other SUs, SU i can classify the space $\boldsymbol{S}_{-i}$ as follows.

We assume the maximum absolute tax is Γ. We split the range $[0,\Gamma]$ into $[\Gamma_0,\Gamma_1),[\Gamma_1,\Gamma_2),\ldots,\left[\Gamma_{H_i-1},\Gamma_{H_i}\right]$ with $0=\Gamma_0\le\Gamma_1\le\cdots\le\Gamma_{H_i}=\Gamma$. Here, we assume that the values of $\left\{\Gamma_1,\ldots,\Gamma_{H_i-1}\right\}$ are equally located in the range of $[0,\Gamma]$. (Note that more sophisticated selection for these values can be deployed, and this forms an interesting area of future research.)

We need to consider three cases to determine the representative state $\tilde{s}_{-i}^t$ at time t.

(i) If the resource allocation $\boldsymbol{z}_i^t\neq\boldsymbol{0}$, then the representative state of other SUs is chosen as

$$\tilde{s}_{-i}^t=h,\ if\ \left|\tau_i^t\right|\in[\Gamma_{h-1},\Gamma_h). \quad (12)$$

(ii) If the resource allocation $\boldsymbol{z}_i^t=\boldsymbol{0}$ but $\boldsymbol{y}^t\neq\boldsymbol{0}$, the tax is 0. In this case, we cannot use the tax to predict the network congestion. However, we can infer that the congestion is more severe than the minimum bid for those available channels, i.e. $\min_{j\in\{l:y_l^t\neq 0\}}\left\{a_{ij}^t\right\}$. This is because, in this current stage of the auction game, only SU i' with $a_{i'j}^t\ge a_{ij}^t$ can obtain channel j which indicates that $\left|\tau_i^t\right|\ge\min_{j\in\{l:y_l^t\neq 0\}}\left\{a_{ij}^t\right\}$, if SU i is allocated any channel. Then the representative state of other SUs is chosen as

$$\tilde{s}_{-i}^t=h,\ if\ \min_{j\in\{l:y_l^t\neq 0\}}\left\{a_{ij}^t\right\}\in[\Gamma_{h-1},\Gamma_h) \quad (13)$$

(iii) If the resource allocation $\boldsymbol{z}_i^t=\boldsymbol{0}$ and $\boldsymbol{y}^t=\boldsymbol{0}$, there is no interaction among the SUs in this time slot. Hence, $\tilde{s}_{-i}^t=\tilde{s}_{-i}^{t-1}$.

2) *Estimating the transition probability*

To estimate the transition probability, SU i maintains a table F with size $H_i\times H_i\times(N+1)$. Each entry $f_{h',h'',j}$ in the table F represents the number of transitions from state $\tilde{s}_{-i}^t=h''$ to $\tilde{s}_{-i}^{t+1}=h'$ when the resource allocation $\boldsymbol{z}_i^t=\boldsymbol{e}_j$ (or $\boldsymbol{0}$ if $j=0$). It is clear that H_i will influence significantly the complexity and memory requirements etc. of SU i. The update of F is simply based on the observation $\boldsymbol{o}_i^t$ and the state classification in the above section. Then, we use the frequency to approximate the transition probability [15], i.e.

$$q_{-i}\left(\tilde{s}_{-i}^{t+1}=h'\mid\tilde{s}_{-i}^t=h'',\boldsymbol{e}_j\right)=\frac{f_{h',h'',j}}{\sum_{h'}f_{h',h'',j}} \quad (14)$$

3) *Learning the future reward*

By classifying the state space $\boldsymbol{S}_{-i}$ and estimating the transition probability, SU i can now forecast the value of the average future reward $V_i^{t+1}\left(\left(s_i^{t+1},\tilde{s}_{\text{-}i}^{t+1}\right),\boldsymbol{y}^{t+1}\right)$ using

---

[7] When the CSM deploys a mechanism without tax for the resource management, the space classification for other SUs can also be done based on the announced information and corresponding resource allocation.

learning. Eq. (7) can be approximated by

$$Q_i^t\left(\left(s_i^t,\tilde{s}_{-i}^t\right),\boldsymbol{y}^t,\boldsymbol{\pi}\right) \doteq \left\{\begin{array}{l} g_i^t(s_i^t,\boldsymbol{y}^t,\boldsymbol{z}_i^t(\boldsymbol{\pi}(\boldsymbol{s}^t,\boldsymbol{y}^t))) + \tau_i^t(\boldsymbol{\pi}(\boldsymbol{s}^t,\boldsymbol{y}^t)) + \\ \alpha_i \displaystyle\sum_{\substack{\left(s_i^{t+1},\tilde{s}_{-i}^{t+1}\right)\in\left(S_i,\tilde{\boldsymbol{S}}_{-i}\right)\\ \boldsymbol{y}^{t+1}\in\{0,1\}^N}} \left\{\begin{array}{l} q_i\left(s_i^{t+1},\boldsymbol{y}^{t+1}\mid s_i^t,\boldsymbol{y}^t,\boldsymbol{z}_i^t(\boldsymbol{\pi}(\boldsymbol{s}^t,\boldsymbol{y}^t)\right)\times \\ q_{-i}\left(\tilde{s}_{-i}^{t+1},\boldsymbol{y}^{t+1}\mid \tilde{s}_{-i}^t,\boldsymbol{y}^t,\boldsymbol{z}_i^t(\boldsymbol{\pi}(\boldsymbol{s}^t,\boldsymbol{y}^t))\right)\times \\ V_i^{t+1}\left(\left(s_i^{t+1},\tilde{s}_{\text{-}i}^{t+1}\right),\boldsymbol{y}^{t+1}\right) \end{array}\right\} \end{array}\right\} \quad (15)$$

Similar to the Q-learning established in [17], we also use the received rewards to update the estimation of future rewards. However, the main difference between our proposed algorithm and Q-learning is that our solution explicitly considers the impacts of other SUs' bidding actions through the state classifications and transition probability approximation.

We use a 3-dimensional table to store the value $V_i\left((s_i,\tilde{s}_{\text{-}i}),\boldsymbol{y}\right)$ with $s_i \in S_i, \tilde{s}_{-i} \in \tilde{\boldsymbol{S}}_{-i}$. The total number of entries in V_i is $|S_i|\times H_i \times 2^N$. SU i updates the value of $V_i\left((s_i,\tilde{s}_{\text{-}i}),\boldsymbol{y}\right)$ at time t according to the following rules:

$$V_i^t\left((s_i,\tilde{s}_{\text{-}i}),\boldsymbol{y}\right) = \begin{cases} (1-\gamma_i^t)V_i^{t-1}\left((s_i,\tilde{s}_{\text{-}i}),\boldsymbol{y}\right) + \\ \qquad \gamma_i^t Q_i^t\left((s_i,\tilde{s}_{-i}),\boldsymbol{y},\boldsymbol{\pi}\right) & \mathit{if}\ (s_i^t,\tilde{s}_{\text{-}i}^t) = (s_i,\tilde{s}_{\text{-}i}),\boldsymbol{y}^t = \boldsymbol{y} \\ V_i^{t-1}\left((s_i,\tilde{s}_{\text{-}i}),\boldsymbol{y}\right) & \mathit{otherwise} \end{cases} \quad (16)$$

where $\gamma_i^t \in [0,1)$ is a learning rate factor satisfying $\sum_{t=1}^{\infty}\gamma_i^t = \infty$ and $\sum_{t=1}^{\infty}\left(\gamma_i^t\right)^2 < \infty$ [17], In summary, the learning procedure that is developed for an SU is shown in Table 1.

E. Complexity of Learning

In section III, we have discussed the computation complexity incurred by the CSM and the communication cost between the CSM and the SUs. In this section, we further quantify the complexity of learning in terms of the computational and storage burden. We use the "flop" (floating-point operation) as a measure of complexity, which will provide us an estimation of the computational complexity required for performing the learning algorithm. Also, based on this, we can determine how the complexity grows with the increasing number of SUs [20]. At each stage, the SU performs the classification of other SUs' states, which, in the worst case, requires a number of "flops" of approximately N. The number of "flops" for estimating the transition probability of other SUs' states as in Eq. (14) is approximately (H_i+1). The number of "flops" for learning the future reward is approximately $(2|S_i|H_i+6)$. Therefore, the total number of "flops" incurred by the SU is $N + H_i + 2|S_i|H_i + 7$, from which we can note that the complexity of learning for each SU is proportional to the number of possible states of that SU and the number of classes in which the other SUs' state space is decomposed.

To perform the learning algorithm, the SU needs to store 2 tables (i.e. transition probability table and state-value table) which have totally $\left(H_i^2(N+1) + 2^N|S_i|H_i\right)$ entries. We also note that the storage complexity is also proportional to the number of possible states of that SU and the number of classes in which the other SUs' state space is decomposed.

VII. SIMULATION RESULTS

In this section, we aim at quantifying the performance of our proposed stochastic interaction and learning framework. We assume that the SUs compete for the available spectrum opportunities in order to transmit delay-sensitive multimedia data. First, we compare the performance of various bidding strategies. Next, we quantify the performance of our proposed learning algorithm in various network environments. We will present here only several illustrative examples. However, the same observations can be obtained using a larger number of SUs or channels.

A. Various Bidding Strategies for Dynamic Multi-user Interaction

In this section, we highlight the merits of the stochastic game framework proposed in Section II by comparing the performance of different SUs, which deploy different bidding strategies. The SUs are required to submit the bid vector on the available channels. The SUs can deploy different bidding strategies to generate their bid vector:

1. Fixed bidding strategy π_i^{fixed}: this strategy generates a constant bid vector during each stage of the auction game, irrespective of the state that SU i is currently in and of the states other SUs are in. In other words, π_i^{fixed} does not consider any of the dynamics defined in Section IV.
2. Source-aware bidding strategy π_i^{source}: this strategy generates various bid vectors by considering the dynamics in source characteristics (based on the current buffer state), but not the channel dynamics.
3. Myopic bidding strategy π_i^{myopic}: this strategy takes into account the disturbance due to the environment as well as the impact caused by other SUs, as discussed in Section V.B. However, it does not consider the impact on the future rewards.
4. Bidding strategy based on best response learning $\pi_i^{\mathcal{L}_i}$: This strategy is produced using the learning algorithm proposed in Section VI. $\pi_i^{\mathcal{L}_i}$ considers the two types of dynamics defined in Section IV, and the interaction impact on the future reward.

In terms of the required information, the above bidding strategies are illustrated in Figure 4. For instance, the fixed bidding strategy π_i^{fixed} does not require information about SU i's state or other SUs' states. The source-aware bidding strategy π_i^{buff} considers the source characteristics based on the current buffer state. However, the myopic bidding strategy π_i^{myopic} requires full information about SU i's state. The bidding strategy based on best response learning $\pi_i^{\mathcal{L}_i}$ also requires information about the states of other SUs.

In this simulation, we consider the SN as an extension of WLANs with spectral agile capability [9]. In the following, we first simulate the case that two SUs compete for the channel opportunities and then extend to the case with multiple (five) SUs.

1) *Competition among two SUs for channel opportunities*

We first consider a simple illustrative network with two SUs competing for the available TxOps. The packet arrivals of the SUs are modeled using a Poisson process with the same average arrival rate of 1Mbps. For illustration simplicity, the channel condition of SU 1 (SU 2) on each channel takes only three values ($K = 3$), which are 18dB, 23dB and 26dB. The transition probabilities are $p_{ij}^{0\to1} = p_{ij}^{0\to2} = 0.4, p_{ij}^{0\to3} = 0.2$, $p_{1j}^{l\to1} = p_{1j}^{l\to2} = 0.4, p_{1j}^{l\to3} = 0.2$, $\forall i, j, l$. The transition probability of the availability of the channels to the SUs are $p_j^{NF} = p_j^{FN} = 0.5$. For illustration simplicity, the environment parameters experienced by the two SUs are the same. The length of the time slot ΔT is 10^{-2} s.

In this simulation, we consider five scenarios. In scenario (1), both SU 1 and 2 deploy the fixed bidding strategy π_1^{fixed}. In scenario (2)~(5), SU 1 deploys the fixed bidding strategy π_1^{fixed}, source-aware bidding strategy π_1^{source}, myopic bidding strategy π_1^{myopic} and best response learning based bidding strategy $\pi_1^{\mathcal{L}_1}$, respectively, and SU 2 always deploys the myopic bidding strategy π_2^{myopic}. The discounted factor for the best response learning algorithm is set to 0.8. As discussed in Section IV.B, the stage reward is defined as $r_i^t = (g_i^t + \tau_i^t)$, with $(-g_i^t - \tau_i^t)$ being the number of packet lost plus the tax charged by the CSM (note that $\tau_i^t \le 0$). This can be interpreted as the cost incurred at each stage. Similar to Eq. (10), we use the average cost over the time window $T = 1000$ to evaluate the performance of the bidding strategies. Hence, the lower the average cost, the better the performance of the bidding strategy is. The packet loss rate, average tax and cost per time slot are presented in Table 2. The accumulated packet loss and cost of SU 1 for the five scenarios are plotted in Figure 5(a) and (b), respectively.

From this simulation, comparing scenario 2 with scenario 1, we observe that when SU 2 deploys the myopic strategy against SU 1 which adopted the fixed bidding strategy, SU 2 reduces its average cost by around 42% and the average packet loss rate by around 16.6%. This significant improvement is because SU 2 can value the channel opportunities more accurately by modeling and considering its experienced dynamics, i.e. source characteristics, channel conditions and availability.

In scenario 3, SU 1 improves its bidding strategy (i.e. it deploys now a source-aware bidding strategy) by partially considering its experienced environment, i.e. SU 1 generates its bid vector by only considering the source dynamics though its current buffer state. Compared to scenario 2, if SU 1 considers more information about its own state, it can further reduce its packet loss rate by an average of 4.5% and an average cost by around 5.4%. This observation verifies that the information about the SU's state improves the bidding strategy.

In scenario 4, SU 1 deploys a myopic bidding strategy which is more advanced than the source-aware bidding strategy since it considers both types of dynamics defined in Section IV (including the dynamics regarding to the source characteristics, channel conditions, and channel availability, and the interaction with other SUs in the auction mechanism). The significant improvement in terms of packet loss rate (13% reduced) and average cost (25% reduced), compared to scenario 2, indicates that the myopic bidding strategy provides the optimal bid vector when only current benefits are considered as shown in Section V.B.

In scenario 5, SU 1 improves further the bidding strategy using the best response learning algorithm developed in Section VI. Using learning, SU 1 reduces the packet loss rate to 15.14% and the average cost to 1.7428 (11.8% lower compared to scenario 4). This significant improvement is due to the ability of the SU to learning and forecast the future impact of its current actions.

It is also worth to note that the reduction of the packet loss rate of SU 1 in scenarios 2~5 comes from two parts: one is the advanced bidding strategies, which allows the SU to take into consideration more information about its own states and the other SUs' states and, based on this, better forecast the impact of various actions, and the other one is the increase in the amount of resources consumed by SU 1 which corresponds to higher tax charged by the CSM, as shown in Table 2.

We further note that the bidding strategy deployed by SU 1 will affect the performance of SU 2. For example, comparing scenario 2 with scenario 4, the fixed bidding strategy of SU 1 in scenario 2 leads to a lower average cost (15% reduced) for SU 2. This is because SU 1 uses a fixed bidding strategy, which does not account for the dynamic changes in its environment, while SU 2 minimizes its current cost (the number of packets lost plus the tax) based on its current state. However, when comparing scenario 5 with scenario 4, SU 1 using learning not only improves its prediction of the current environment dynamics but also better predicts the impact on the future cost based on the observations. The improvement leads to higher resource allocation (hence, incurring higher tax, see in Table 2) for SU 1, thereby resulting in worse performance for SU 2 (i.e. the average cost is increased by 22.2%).

2) *Multiple SUs competition for channel opportunities*

In this simulation, we consider five SUs competing for the available TxOps in the WLAN-like SN. The packet arrivals of all the five SUs are modeled using a Poisson process with the same average arrival rate of 1Mbps. The number of channels is 3 and the channel condition of all the five SUs on each channel takes only three values ($K = 3$), which are 18dB, 23dB and 26dB. The transition probabilities are $p_{ij}^{0\to 1} = p_{ij}^{0\to 2} = 0.4, p_{ij}^{0\to 3} = 0.2$, $p_{1j}^{l\to 1} = p_{1j}^{l\to 2} = 0.4,$, $p_{1j}^{l\to 3} = 0.2, \forall i, j, l$. The parameters of the model of the availability of the channels to the SUs are $p_j^{NF} = 0.7, p_j^{FN} = 0.3$. The length of the time slot ΔT is also 10^{-2} s. Similar parameters are used for the five SUs in order to clearly illustrate the performance differences obtained based on the different strategies.

In this simulation, we consider only two scenarios. In scenario (1), all SUs deploy a myopic bidding strategy $\pi_i^{myopic}, i = 1,2,...,5$, while in scenario (2), SU 5 deploys the multi-user learning-based bidding strategy $\pi_5^{\mathcal{L}_5}$ with the discount factor of 0.5 and the other SUs deploy the myopic bidding strategy $\pi_i^{myopic}, i = 1,...,4$. The packet loss rate and cost per time slot incurred by the SUs are presented in Table 3. The accumulated packet loss and cost of SU 5 for the five scenarios are plotted in Figure 6(a) and (b), respectively.

Similar to the two-SU network, SU 5 significantly reduces the packet loss rate by 14.6% and average cost by 16.1% by adopting the best response learning-based bidding strategy. Figure 6 (a) and (b) further verify the improvement of the performance for SU 1. However, other SUs' performances are decreased, as they need now to compete against a learning SU (i.e. SU 5), which is able to make better bids for the available resources.

B. *Multi-user Learning and Delay Impact in a Wireless Test-Bed*

To validate the performance of the multi-user learning and the impact of various delays in a realistic network setting, we considered two SUs competing for the available transmission opportunities in our 802.11a-enabled wireless test-bed [31]. The channel condition experienced by the SUs varied between 10dB to 30dB, and we represented this variation using 10 states ($K = 10$). The parameters of the TxOp model are $p_j^{NF} = 0.6, p_j^{FN} = 0.4$. The length of the time slot ΔT is also 10^{-2} s. The SUs stream the delay-sensitive video traffic (e.g. the Mobile sequence encoded using an H.264 video encoder) to their own destinations with an average data rate of 1.5Mbps. We compare three scenarios. In scenario (1), both SUs deploy a myopic bidding strategy $\pi_i^{myopic}, i = 1,2$. In scenario (2), SU 1 deploys the learning-based bidding strategy $\pi_1^{\mathcal{L}_1}$ with a discount factor of 0.5 and SU 2 deploys a myopic strategy π_2^{myopic}. In scenario (3), both SUs deploy the learning-based bidding strategy $\pi_i^{\mathcal{L}_i}, i = 1,2$. In the above three scenarios, the video applications are considered to tolerate a delay[8] of 533 ms, which is used in some real-time video streaming applications. In scenario (4), SU 1 deploys the learning-based bidding strategy $\pi_1^{\mathcal{L}_1}$ with a discount factor of 0.5 and SU 2 deploys a myopic strategy π_2^{myopic}. However, in this scenario, SU 1 streams the video sequence which can only tolerate a delay of 266ms, which is typical for video conferencing applications.

Table 4 shows the average video quality in terms of Peak Signal to Noise Ratio (PSNR)[9] and incurred cost for both SUs under various scenarios. Comparing scenario (2) with scenario (1), we observe that the SU using the learning-based bidding strategy improves the received video quality by 2.2dB and reduces the incurred cost by 9.3%. However, as the performance of SU 1 improves, this also results in worse performance for SU 2. This observation is similar to the results in Section VII.A.1) and has the same explanation.

In scenario (3), both SUs deploy the learning-based bidding strategies and are able to better predict the impact of their current bidding actions on the future cost based on their observations. Thus, compared to scenario 1, the performance of both SUs has been improved: SU 1 (SU 2) increases by 1 dB (1.2dB) in terms of PSNR and reduces its cost by 4.3% (4.0%). Compared to scenario (2), if SU 2 also deploys the learning-based approach, then SU 2 also observes its estimated future reward and will increase its bid, thereby reducing the performance of SU 1. From Table 4, we note that the PSNR of SU 1 is decreased by 1.2dB, while the PSNR of SU 2 is increased by 2dB. We also observe that the cost of SU 1 is increased by around 5.6%, while the cost of SU is decreased by 9.1%.

In scenario (4), since SU 1 streams a video application with a lower delay deadline, it has to bid more to ensure that the packets with stringent delay deadline are transmitted to the destination and hence, SU 1 incurs a higher transmission cost (41% increased) compared to scenario 2. Although SU 1 bids more for the limited available resources, the video quality of SU 1 is reduced by 1.8dB due to its stringent delay deadline. Interestingly, the stringent delay deadline of the SU 1's application also increases the transmission cost of SU 2 and also reduces its video quality. This is because the higher bid of SU 1 on the limited resources automatically increases the bid of SU 2.

C. *Learning with Imperfect Information*

In this section, we consider that SU 1 deploys the learning-based bidding strategy and SU 2 deploys the myopic strategy. The environment parameters are the same as in Section VII.B. To quantify the impact of imperfect information about the environment on the SUs' performance, we assume that SU 1 has the transition probability of TxOps, $p_j^{NF} = 0.55, p_j^{FN} = 0.45$, which is slightly different from the true one (i.e. $p_j^{NF} = 0.6, p_j^{FN} = 0.4$). Table 5 shows the PSNRs and corresponding cost of both SUs when SU 1 has

---

[8] During the simulations, for simplicity, we assume that the packets within one Group of Picture (GOP) have the same delay deadline.

[9] PSNR is a widely adopted metric to objectively measure the video quality. A PSNR difference of 1 dB is significant, and can be seen by an untrained human observer.

perfect or imperfect information about the TxOps.

From Table 5, we observe that an inaccurate model of TxOps reduces the performance of SU 1 (i.e. the PSNR decreases by 0.3dB and increases the cost by 4.2%). We further note that this will also affect the performance of SU 2. In this simulation, the PSNR of SU 2 is reduced by 0.2dB and the cost is increased by 3.5%. This performance loss can be explained as follows: since SU 1 has an inaccurate model about the available TxOps, it may generate a suboptimal bid vector at each stage, which will accordingly result in a suboptimal allocation (TxOps and payment) among the SUs. This suboptimal allocation will also lead to the performance loss of other SUs. Hence, it is essential for the users to learn and accurately predict their environment.

D. *Impact of various dynamics on learning*

In Section VII.A, we demonstrate that the best response learning algorithm improves the bidding strategy, thereby leading to a reduced packet loss rate and average cost. In this simulation, we further investigate how various dynamics impact the learning algorithm proposed in Section VI.D. Specifically, we compare the learning performance under different channel dynamics, i.e. various available spectrum opportunities for the SUs as discussed in Section II. The source characteristics and channel conditions experienced by the SUs are kept the same as in Section VII.A.1). We consider three types of channel dynamics corresponding to scenarios 1~3. The transition probabilities of the TxOps for all three scenarios are listed in Table 6. In each scenario, we compare two cases: in the first one, both SUs deploy myopic bidding strategies, and in the second one, SU 1 deploys best response learning-based bidding strategy, while SU 2 still uses the myopic bidding strategy.

Table 7 shows the average packet loss rate and cost experienced by the SUs under various channel dynamics. Interestingly, we observe from these results that even though the learning algorithm reduces the packet loss rate, it does not reduce the cost associated with SU 1, when the channel resources are abundant as in scenario 1. As the resources become increasingly scarce, the learning algorithm helps SU 1 to simultaneously reduce the packet loss rate and cost, e.g. in scenario 2 and 3. This observation can be explained as follows: when the resources are abundant, the cost (including the packet loss and tax) is small, i.e. the “value” of the channel is limited, and hence, the learning-based bidding strategy does not significantly benefit. On the other hand, when the resources are scarce, the bid vectors of the SUs in the current time slot will significantly affect the transition of their states through the channel allocation comparing to the case when the resources are abundant. For example, if an SU makes low bids as compared to other SUs, it might have no resources (channels) allocated to it when resources are scarce (i.e. the SN is congested). In this case, the learning-based bidding strategy will carefully plan the bid by considering the future impact and thus, it is able to successfully improve the performance of SU 1 in terms of reducing the average cost.

VIII. CONCLUSIONS AND FUTURE RESEARCH

In this paper we model the dynamic resource allocation problem as a “stochastic game” played among strategic SUs. At each stage of the game, the CSM deploys a generalized second price auction mechanism to allocate the available spectrum resource. The SUs are allowed to simultaneously and independently make bid decision on that resource by considering their current states, experienced environment as well as the estimated future reward. To improve the bid decision at each stage, we propose a best response learning algorithm to predict the possible future reward at each state. The simulation results show that our proposed learning algorithm can significantly improve the SUs’ performance.

We note that, the constraint of the perfect information about the available wireless resources can be relaxed for the case when the CSM and the wireless users do not have the perfect information about the available resources. In this case, the wireless users can estimate and build a belief about the available resource. Hence, the stochastic game model can be extended to partially observably stochastic games [32]. This is one of our interesting future research topics. We also note that, we can allow the wireless users to adapt their transmission power, which will lead to different interference levels to other users. In this case, the wireless users compete with each other for lower interference levels incurred by other users [6], instead of competing for the transmission time. This can also be formulated as a stochastic game and similar learning algorithms can be developed. This forms another interesting topic of our future research. This forms another interesting topic of our future research. Our future work also includes analyzing the performance of SNs where multiple SUs are deploying various learning strategies and protocols.

REFERENCES

[1] Federal Communications Commission, “Spectrum Policy Task Force,” *Rep. ET Docket* No. 02-135, Nov. 2002.

[2] S. Haykin, “Cognitive radio: Brain-empowered wireless communications,” *IEEE J. Sel. Areas Commun.*, vol. 23, no. 2, Feb. 2005.

[3] F. A. Ian, W.Y. Lee, M.C. Vuran, and S. Mohanty, “NeXt generation/dynamic spectrum access/cognitive radio wireless network: a survey,” Computer Networks, vol 50, no. 13, Sept. 2006.

[4] C. Kloeck, H. Jaekel, and F. Jondral, “Auction Sequence as a new resource allocation mechanism,” *Proceedings of VTC’05*, Dallas, Sept. 2005.

[5] F. Fu, A. R. Fattahi, and M. van der Schaar, "Game-theoretic paradigm for resource management in spectrum agile wireless networks," in *Proc. 2006 IEEE Int. Conf. Multimedia & Expo (ICME 06)*, 2006, pp. 873-876.

[6] J. Huang, R. Berry and M. L. Honig, "Auction-based Spectrum Sharing", *ACM Mobile Networks and Applications Journal (MONET)*, vol. 11, no. 3, pp. 405-418, June 2006.

[7] Y. Xing, R. Chandramouli, and C. M. Cordeiro, “Price dynamics in a secondary spectrum access market ,” *IEEE J. Sel. Areas Commun.*, April 2007.

[8] L. Berlemann, S. Mangold, G.R. Hiertz and B.H. Walke, “Policy defined spectrum sharing and medium access for cognitive radios*”, Journal of Communications, Academy Publishers*, Vol. 1, Issue 1, April 2006.

[9] C. T. Chou, S. Shankar N, H. Kim and K, Shin, “What and how much to gain by spectrum agility?” *IEEE J. Sel. Areas Commun.*, vol. 25, no. 3, pp. 576-588, Apr. 2007.

[10] S. Shankar, C.T. Chou, K. Challapali, and S. Mangold, “Spectrum agile radio: capacity and QoS implications of dynamic spectrum assignment,” *Global Telecommunications Conference*, Nov. 2005.

[11] D. Bertsekas, and R. Gallager, “Data networks,” Prentice Hall, Inc. Upper Saddle River, NJ, 1987.

[12] M. van der Schaar and S. Shankar, “Cross-layer wireless multimedia transmission: Challenges, principles and new paradigms,” *IEEE Wireless Communications Magazine*, Aug 2005.

[13] “IEEE 802.11e/D5.0, wireless medium access control (MAC) and physical layer (PHY) specifications: Medium access control (MAC) enhancements for Quality of Service (QoS), draft supplement,” June 2003.

[14] R. W. Lucky, “Tragedy of the commons,” *IEEE Spectrum*, vol. 43, No. 1, pp. 88, Jan 2006.

[15] R. G. Gallager, “Discrete stochastic processes,” Kluwer Academic Publishers, 1996.

[16] L. S. Shapley, “Stochastic games,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 39, 1095-1100, 1953.

[17] C. Watkins, and P. Dayan, “Q-learning.” *Technical Note, Machine Learning*, vol. 8, 279-292, 1992.

[18] M. Bowling, and M. Veloso, “Rational and convergent learning in stochastic games,” *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI)* , 1021--1026, August 2001.

[19] P. Klemperer, “Auction theory: A guide to the literature,” J. Economics Surveys, vol. 13, no. 3, pp. 227-286, Jul. 1999.

[20] S. P. Boyd, and L. Vandenberghe, “Convex optimization,” Cambridge University Press, 2004.

[21] F. Fu, and M. van der Schaar, “Non-collaborative resource management for wireless multimedia applications using mechanism design,” *IEEE Transaction on Multimedia*, vol. 9, no. 4, pp. 851-868, Jun. 2007.

[22] D. Fudenberg, and D. K. Levine, “The theory of learning in games,” Cambridge, MA: MIT Press, 1999.

[23] M. Jackson, “Mechanism theory,” *In the Encyclopedia of Life Support Systems*, 2003.

[24] Q. Zhang, and S.A. Kassam, “Finite-state Markov model for Rayleigh fading channels,” *IEEE Transaction on Communications*, vol. 47, no. 11, Nov. 1999.

[25] A. Ortega, “Variable Bit-rate Video Coding”, *in Compressed Video over Networks , M.-T. Sun and A.~R. Reibman, Eds,* pp.343-382, Marcel Dekker, New York, NY, 2000.

[26] S. Lal, and E.S. Sousa, “Distributed resource allocation for DS-CDMA-based multimedia ad hoc wireless LANs,” *IEEE J. Sel. Areas Commun*., Vol. 17, No. 5, 947 – 967, May 1999.

[27] R. S. Sutton, and A. G. Barto, “Reinforcement learning: an introduction,” Cambridge, MA:MIT press, 1998.

[28] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, 'Layering as optimization decomposition: A mathematical theory of network architectures', *Proceedings of the IEEE*, vol. 95, no. 1, pp. 255-312, January 2007.

[29] F. Kelly, A. Maulloo, and D. Tan. “Rate control for communication networks: shadow prices, proportional fairness and stability,” *Operational Research Society*, vol. 49, pp.237-252, 1998.

[30] X. Zhu, P. Agrawal, J. P. Singh, T. Alpcan, and B. Girod, "Rate Allocation for Multi-User Video Streaming over Heterogenous Access Networks", *Proc. ACM Multimedia, (MM-07*), September 2007.

[31] D. Krishnaswamy, and J. Vicente, “Scalable adaptive wireless networks for multimedia in the proactive enterprise,” Intel Technology J. [Online.] Available: http://developer.intel.com/technology/itj/2004/volume08issue04/art04_scalingwireless/p01_abstract.htm.

[32] D. S. Bernstein, E. A. Hansen, S. Zilberstein and C. Amato, “Dynamic Programming for Partially Observable Stochastic Games,” *Proceedings of the AAAI Spring Symposium on Bridging the Multi-Agent and Multi-Robotic Research Gap*, Stanford, California, March 2004.

Fangwen Fu received his bachelor and master degrees from Tsinghua University, Beijing, China, in 2002 and 2005, respectively. He is currently working toward a Ph.D. degree in the Department of Electrical Engineering at University of California, Los Angeles (UCLA). During the summer of 2006, he worked as an intern in IBM T.J. Watson Research Center, New York. His research interests include wireless multimedia streaming, resource management for networks and systems, applied game theory, video processing and analysis.

Mihaela van der Schaar received both the M.S. and Ph.D. degrees from Eindhoven University of Technology, Eindhoven, The Netherlands, in 1996 and 2001, respectively. Prior to joining the UCLA Electrical Engineering Department faculty on July 1st, 2005, she was between 1996 and June 2003 a senior researcher at Philips Research in the Netherlands and USA, where she led a team of researchers working on multimedia coding, processing, networking, and streaming algorithms and architectures. From January to September 2003, she was also an Adjunct Assistant Professor at Columbia University. From July 1st, 2003 until July 1st, 2005, she was an Assistant Professor in the Electrical and Computer Engineering Department at University of California, Davis.

Prof. van der Schaar has published extensively on multimedia communications, networking, architectures, systems, compression and processing, and holds 30 granted US patents and several more pending. Since 1999, she has been an active participant in the ISO Motion Picture Expert Group (MPEG) standard, to which she has made more than 50 contributions and for which she has received three ISO recognition awards. She also chaired the ad-hoc group on MPEG-21 Scalable Video Coding for three years, and co-chaired the MPEG ad-hoc group on Multimedia Test-beds. She was a guest editor of the EURASIP Special Issue on Multimedia over IP and Wireless Networks, and was the general chair of Picture Coding Symposium 2004, the oldest conference on image/video coding. She is a senior member of IEEE, and was also elected as a Member of the Technical Committee on Multimedia Signal Processing, as well as the Technical Committee on Image and Multiple Dimensional Signal Processing of the IEEE Signal Processing Society. She was an Associate Editor of the IEEE Transactions on Multimedia and the SPIE Electronic Imaging Journal from 2002-2005. Currently, she is an Associate Editor of the IEEE Transactions on Circuits and System for Video Technology, of the IEEE Signal Processing Letters, and of the newly founded IEEE Signal Processing Society e-Newsletter.

Figures and tables

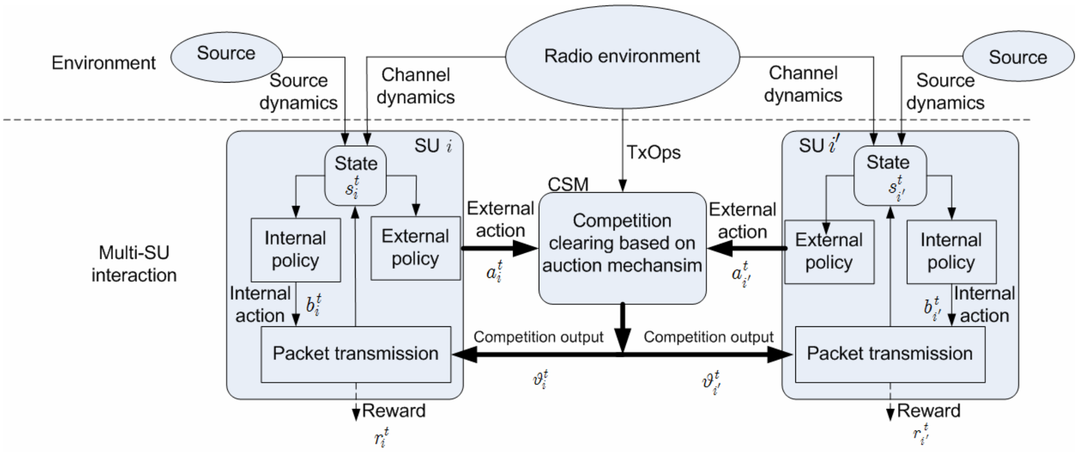


Figure 1. Conceptual overview of the multi-SU interaction in the SN

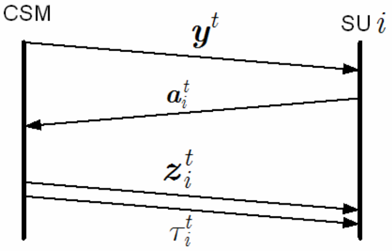


Figure 2. Information exchange between the CSM and SU i

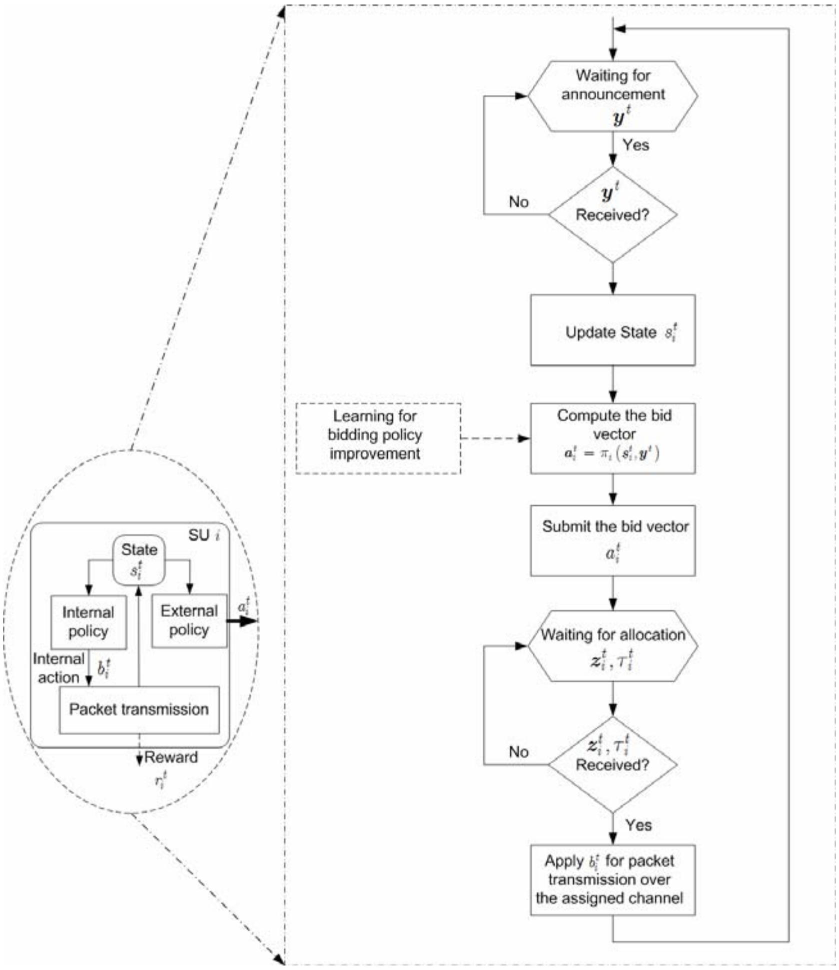

SU i
State s_i^t
Internal policy
External policy
a_i^t
Internal action b_i^t
Packet transmission
Reward r_i^t
Waiting for announcement y^t
Yes
No
y^t Received?
Update State s_i^t
Learning for bidding policy improvement
Compute the bid vector $a_i^t = \pi_i(s_i^t, y^t)$
Submit the bid vector a_i^t
Waiting for allocation z_i^t, τ_i^t
No
z_i^t, τ_i^t Received?
Yes
Apply b_i^t for packet transmission over the assigned channel

Figure 3. The procedure for SU i to play the auction game at time slot t

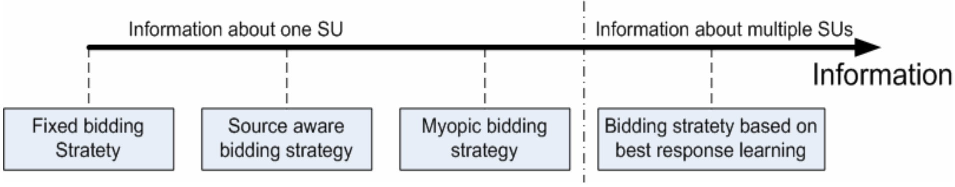
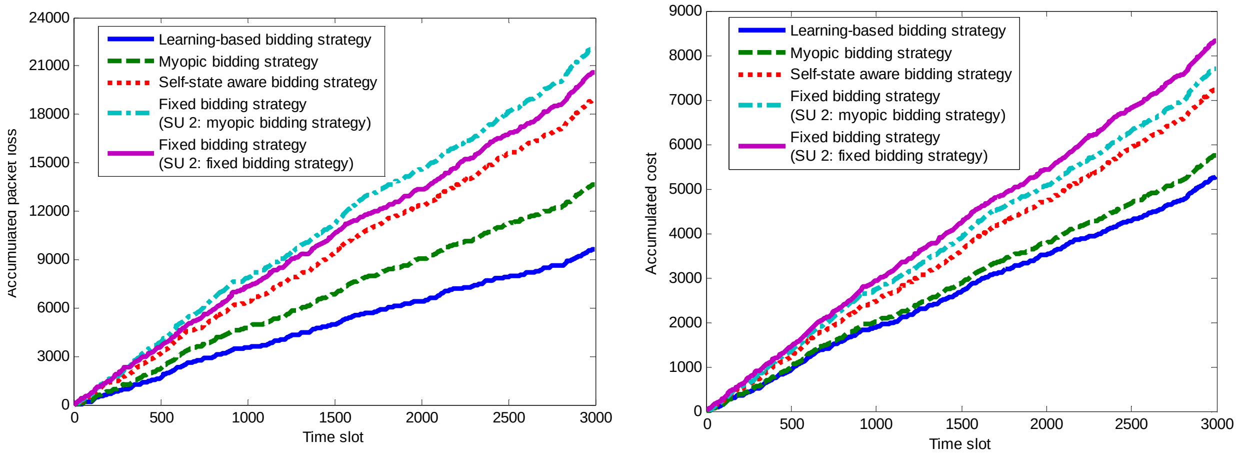
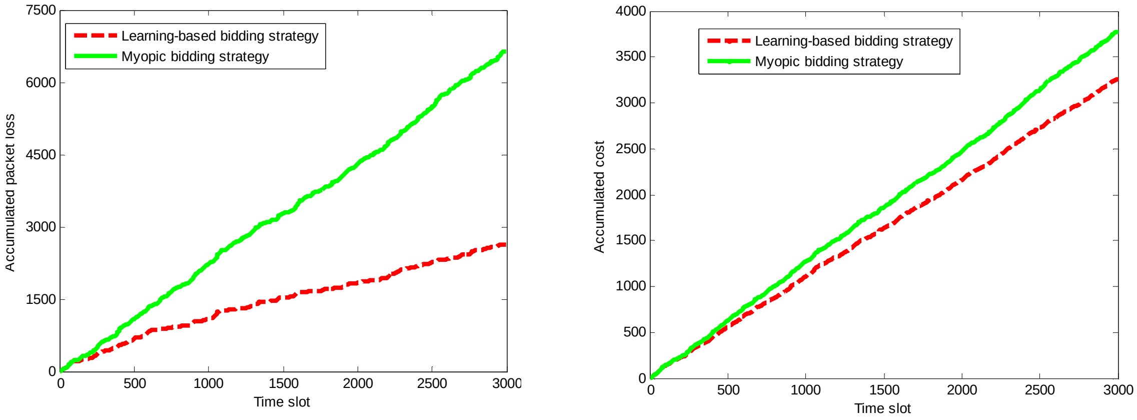

Information about one SU
Information about multiple SUs
Information
Fixed bidding Stratety
Source aware bidding strategy
Myopic bidding strategy
Bidding stratety based on best response learning

Figure 4. The illustration of bidding strategies based on the required information



(a) (b)

Figure 5. The accumulated packet loss and cost of SU 1 in the five scenarios, (a) accumulated packet loss over the time slot; (b) accumulated cost over the time slot



(a) (b)

Figure 6. The accumulated packet loss and cost of SU 5 in the two scenarios, (a) accumulated packet loss over the time slot; (b) accumulated cost over the time slot

Table 1. Learning Procedure

```
Initializing: $V_i^0((s_i, \tilde{s}_{-i}), \boldsymbol{y}) \Leftarrow 0$ for all possible states $s_i \in S_i$, $\tilde{s}_{-i} \in \tilde{\boldsymbol{S}}_{-i}$.
Learning:
    At time $t$, SU $i$:
        b. Observes the current state $s_i^t$ and $\boldsymbol{y}^t$;
        c. Chooses an action $a_i^t = [u_{i1}^t, \dots, u_{iN}^t]$ as computed in Eq. (11) by replacing
           $V_i^{t+1}((s_i^{t+1}, \tilde{s}_{-i}^{t+1}), \boldsymbol{y}^{t+1})$ with $V_i^{t-1}((s_i^{t+1}, \tilde{s}_{-i}^{t+1}), \boldsymbol{y}^{t+1})$, and then submits it to the CSM;
        d. Receives the allocation $\boldsymbol{z}_i^t$ and payment $\tau_i^t$;
        e. Computes the representative state $\tilde{s}_{-i}^t$ as in Section VI.D.1) and update the transition probability
           as in Section VI.D.2);
        f. Computes the expected total discounted sum of the rewards $Q_i^t((s_i^t, \tilde{s}_{-i}^t), \boldsymbol{y}^t, \boldsymbol{\pi})$ as in Eq. (15);
        g. Updates the future reward table $V_i^t((s_i, \tilde{s}_{-i}), \boldsymbol{y})$ at the state $(\boldsymbol{s}_i^t, \tilde{s}_{-i}^t)$ and TxOp $\boldsymbol{y}^t$ using the
           learning rate factor $\gamma_i^t$, according to Eq. (16).
```

Table 2. Performance of SU 1 and 2 with various bidding strategies in the two SUs network

	Bidding Strategies	SU 1			SU 2		
		Packet loss rate (%)	Average tax	Average cost	Packet loss rate (10%)	Average tax	Average cost
Scenario 1	$\pi_1^{fixed}, \pi_2^{fixed}$	32.53	0.4875	2.8966	31.05	0.5095	2.6104
Scenario 2	$\pi_1^{fixed}, \pi_2^{myopic}$	34.36	0.1222	2.6337	14.39	0.5495	1.5105
Scenario 3	$\pi_1^{source}, \pi_2^{myopic}$	29.83	0.3147	2.4915	18.11	0.6048	1.6116
Scenario 4	$\pi_1^{myopic}, \pi_2^{myopic}$	21.55	0.4669	1.9767	19.55	0.3763	1.7837
Scenario 5	$\pi_1^{\mathcal{L}_1}, \pi_2^{myopic}$	15.14	0.6923	1.7428	27.29	0.4197	2.2967

Table 3. Performance of SU 1~5 with various bidding strategies in the five SUs network

	SU 1		SU 2		SU 3		SU 4		SU 5	
	Packet Loss Rate (%)	Average cost	Packet Loss Rate (%)	Average cost	Packet Loss Rate (%)	Average cost	Packet Loss Rate (%)	Average cost	Packet Loss Rate (%)	Average cost
1	21.14	1.2002	19.99	1.1666	22.05	1.2123	21.37	1.1949	24.17	1.3101
2	25.03	1.2992	24.20	1.2993	25.72	1.3338	26.02	1.3568	9.56	1.0988

Table 4. Performance of SU 1 and 2 with various bidding strategies in the more realistic network

	Bidding strategies	SU 1		SU 2	
		PSNR (dB)	Average cost	PSNR (dB)	Average cost
Scenario 1	$\pi_1^{myopic}, \pi_2^{myopic}$	30.8	5.8951	30.7	5.8845
Scenario 2	$\pi_1^{\mathcal{L}_1}, \pi_2^{myopic}$	33.0	5.3449	29.9	6.2236
Scenario 3	$\pi_1^{\mathcal{L}_1}, \pi_2^{\mathcal{L}_2}$	31.8	5.6493	31.9	5.6536
Scenario 4	$\pi_1^{\mathcal{L}_1}, \pi_2^{myopic}$	31.2	7.5439	29.2	6.6748

Table 5. Performance comparison between the scenarios whether SU 1 has perfect information or not

	Bidding strategies	SU 1		SU 2	
		PSNR (dB)	Average cost	PSNR (dB)	Average cost
Scenario 1 (SU 1 has perfect information)	$\pi_1^{\mathcal{L}_1}, \pi_2^{myopic}$	33.0	5.3449	30.7	6.2236
Scenario 2 (SU 1 has imperfect information)	$\pi_1^{\mathcal{L}_1}, \pi_2^{myopic}$	32.7	5.5685	30.5	6.4385

Table 6. Channel availability probability

	Channel 1			Channel 2		
	p_1^{NF}	p_1^{FN}	Number of opportunities	p_2^{NF}	p_2^{FN}	Number of opportunities
Scenario 1	0.8	0.2	3502	0.8	0.2	3498
Scenario 2	0.5	0.5	2490	0.5	0.5	2462
Scenario 3	0.4	0.6	1960	0.4	0.6	1968

Table 7. Average packet loss rate and cost for the SUs under various resource constraints

		SU 1		SU 2	
		Packet loss rate	Average cost	Packet loss rate	Average cost
Scenario 1	$\pi_1^{myopic}, \pi_2^{myopic}$	3.08	0.2678	2.90	0.2844
	$\pi_1^{\mathcal{L}_1}, \pi_2^{myopic}$	2.69	0.3092	4.17	0.4110
Scenario 2	$\pi_1^{myopic}, \pi_2^{myopic}$	21.36	1.8954	23.85	1.7471
	$\pi_1^{\mathcal{L}_1}, \pi_2^{myopic}$	14.54	1.6764	30.67	2.1744
Scenario 3	$\pi_1^{myopic}, \pi_2^{myopic}$	45.01	3.6283	45.42	3.8289
	$\pi_1^{\mathcal{L}_1}, \pi_2^{myopic}$	35.21	3.2590	56.44	4.5162