Learning to Compete for Resources in Wireless Stochastic Games

Fangwen Fu and Mihaela 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 to acquire 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 (CSM) 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, interactive learning, multiuser resource management, reinforcement learning, stochastic games, wireless networks.

I. INTRODUCTION

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 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 to achieve high utility in competitive and heterogeneous wireless networks (including cognitive radio networks [1]–[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 an a priori mandated or negotiated rules.

This paper is primarily concerned with the tensions and relationships among autonomous adaptation by secondary (unlicensed) users (SUs), the competition among these users, as well as the interaction of these users with spectrum moderators having their own goals, e.g., making money, imposing fairness rules, ensuring compliance with the Federal Communications Commission (FCC) [1], and local regulations with respect to primary (licensed) users (PUs), etc. Unlike 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, as well as 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 1 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 PUs, 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 environmental disturbances. These types of dynamics are generally modeled as stationary processes. For instance, the use of each channel by the PUs can be modeled as a two-state Markov chain with on-state (the channel is used by PUs) and off-state (the channel is available for the SUs) [7]. The channel conditions can be modeled using a finite-state Markov model [24]. The 73 packet arrival of the source traffic can be modeled as a Poisson process [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 noncollaborative, autonomous, and strategic SUs in the network—work transmitting their traffic—is less well studied to wireless communication networks.

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 multiuser communication.
predict the impact of current actions on future performance and then optimally make their resource bids.

This paper is organized as follows. In Section II, we introduce a stochastic game formulation for multiuser 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 conclusions and future research in Section VIII.

II. STOCHASTIC GAME FORMULATION FOR DYNAMIC MULTIUSER 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), the $M (M \geq N)$ autonomous SUs, each indexed by $i \in \{1, \ldots, M\}$ and 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, we do not discuss the detection methods in this paper but rather rely on the existing literature for this purpose. Instead, we assume that the available TxOps in each channel change over time due to the PUs joining or leaving the network and can be modeled as a two-state Markov chain, as in [7] and [10]. Our goal is to develop a general framework for multiuser interaction in the SN, where users can compete and for 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 176 synchronous slot structure. The time slot has length of $\Delta T = 177$ 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, the channel is in one of the following two states: ON-state (this channel is currently used by the PUs) or OFF-state (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 $y^j_t \in \{0, 1\}$, where $y^j_t = 1$ if the channel is in the ON-state and 0 if it is in the OFF-state. In this paper, we study two cases: OFF-state and ON-state. The TxOp of channel $j$ at time slot $t \in \mathbb{N}$ is denoted by $y^j_t \in \{0, 1\}$, where $y^j_t = 1$ if the channel is in the ON-state and 0 if it is in the OFF-state. In this paper, we consider the ON-state channel model with transition probability $p_j^{	ext{ON}} = p(y^j_{t+1} = 1 | y^j_t = 1)$ and $p_j^{	ext{OFF}} = p(y^j_{t+1} = 0 | y^j_t = 1)$. The TxOp profile of the ON-state channel is represented by $y^j_t = [y^j_1, \ldots, y^j_T]$.

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 only changed 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 easily be extended to more sophisticated models [10], where each SU can simultaneously access multiple channels or the 23 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 slowly change.

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 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 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 \( y^t \). The competition result is \( \vartheta^i_t \), based on which SU \( i \) performs its internal action \( b^i_t \) and obtains the reward \( r^i_t \) at this time slot. After packet transmission, SU \( i \) transits to the next state \( s^{i+1}_t \in S_i \). The conceptual overview of the multi-SU interactions in the repeated auctions is illustrated in Fig. 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 this paper, we interchangeably use the terms “time slot” and “stage.”

We define the stochastic game for SN resource allocation as \( \langle \{S_t, A_t, B_t, O_t, q_t, r_t\}_{t=1}^M, Y \rangle \), where each SU \( i \) is associated with a tuple \( \langle S_t, A_t, B_t, O_t, q_t, r_t \rangle \). Specifically, we have the following.

1) \( Y \) is a finite set of possible TxOps available for SUs. In this paper, \( Y = \{0, 1\}^N \), and \( y^t \in Y \) is the available TxOps at stage \( t \), which is common information for SUs.

2) \( S_t \) is a finite local state space of SU \( i \). We let \( S := \bigcup_{k=1}^N S_k \) be the global state space of all SUs and \( S_{-i} := \bigcup_{k \neq i} S_k \) be the global state space of SUs other than \( i \). At stage \( t \), the global state is denoted by \( s^t = (s^t_1, \ldots, s^t_M) = (s^t_i, s^t_{-i}) \), where \( -i \) represents all the 241 SUs other than \( i \).

3) \( A_i \) is a finite set of external actions performed by SU \( i \) to compete for the available TxOps. The external action \( a^t \) vector at stage \( t \) for all SUs will be \( a^t = (a^1_t, \ldots, a^M_t) \).

4) \( B_t \) is a finite set of internal actions performed by SU \( i \) to determine the packet transmission.

5) \( O_t \) is a finite set of possible output from multi-SU computation. In this paper, the output \( o^t_i \in O_t \), the auction result computed by the CSM for SU \( i \) at stage \( t \). We will specify the form of the output in Section III.

6) \( q_t \) is the state transition probability for SU \( i \). Thus, \( q_t(s^{t+1}_i, y^{t+1} | s^t_i, y^t, o^t_i, b^t_i) \) is the probability that the state \( s^{t+1}_i \) of SU \( i \) transitions from \( s^t_i \) and TxOp \( y^{t+1} \) transits from \( y^t \) to \( y^{t+1} \) if the competition output is \( \vartheta^t_i \) and the internal 265 action is \( b^t_i \). The reason that the transition probability includes the common TxOp \( y^{t+1} \) is because the channel condition transition of SU \( i \) depends on the available 268 TxOp.

7) \( r_t \) is the stage reward (immediate reward) received by SU \( i \), where \( r^t_i : (S_t, O_t, B_t) \rightarrow \mathbb{R} \). It should be noted that 271
the reward function \( r_i \) depends on the competition output and, hence, indirectly depends on the other SUs’ external actions.

To design a stochastic game for the SN with strategic SUs, we have to consider the following: 1) What auction mechanism can be deployed to resolve the competition among SUs; 2) how the dynamic environment experienced by each SU can be modeled; and 3) 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 \( y_i \) and allocates (through polling the SUs) those channels with \( y_i = 1 \) 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 I, 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 computational complexity in the CSM and the communication cost between SUs and CSM.

First, the CSM announces the auction by broadcasting the TxOp \( y_i \). The SUs receive the announcement and determine the external action (i.e., the bid vector) \( a_i = [a_{i1}, \ldots, a_{iN}] \in \mathbb{R}^N \) based on the announced information and their own private information about the environment they experience, which is discussed in detail 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 \( z_i = [z_{i1}, \ldots, z_{iN}] \in \{0,1\}^N \) for each SU \( i \) based on the submitted bids. To compel the SUs to truthfully declare their bids [23], the CSM also computes the payment \( \tau_i \in \mathbb{R}_+ \) 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 to the CSM for the used resources. Hence, the competition output \( \vartheta_i \) in this auction mechanism includes the channel allocation \( z_i \) and the payment \( \tau_i \), i.e., \( \vartheta_i = (z_i, \tau_i) \). The competition output is then transmitted back to the SUs. The computation of the channel allocation \( z_i \) and payment \( \tau_i \) is described as follows.

- After each SU submits the bid vector, the CSM performs two computations, i.e., channel allocation and payment computation. Note that most existing multiuser 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 \( Z_i = [z_{ij}]_{M \times N} \) with \( z_{ij} \) being 1 if channel \( j \) is assigned to SU \( i \), and 0 otherwise. The feasible set of channel assignments is denoted as \( Z_i = \{Z_i | \sum_{j=1}^{N} z_{ij} = y_i, \forall j, \sum_{j=1}^{N} z_{ij} \leq 1, \forall i, z_{ij} \in \{0,1\} \} \). The channel allocation matrix without the presence of SU \( i \) is denoted \( Z_{-i} = [z_{kj}]_{(M-1) \times N} \), and the corresponding feasible set is \( Z_{-i} = \{Z_{-i} | \sum_{k=1, k \neq i}^{M} z_{kj} = 335 y_j, \forall j, \sum_{j=1}^{N} z_{kj} \leq 1 \forall k \neq i, z_{kj} \in \{0,1\} \} \), where \( i = 336 \{1, \ldots, i-1, i+1, \ldots, M \} \). During the first phase, the CSM 337 allocates the channels to SUs based on its adopted fairness rule, 338 e.g., maximizing the total “social welfare,” as

\[
Z_i^{\text{opt}} = \arg \max_{Z_i \in \mathbb{Z}_+^{M \times N}} \sum_{i=1}^{M} \sum_{j=1}^{N} z_{ij} y_{ij}.
\]

If the resources are priced, we will consider in this paper, for illustration, a second-price auction mechanism [19], [23] for 341 determining the tax that needs to be paid by SU \( i \) based on the above optimal channel assignment \( Z_i^{\text{opt}} = [z_{ij}^{\text{opt}}]_{M \times N} \). This 343 tax is equal to

\[
\tau_i = \sum_{k=1, k \neq i}^{M} \sum_{j=1}^{N} z_{kj}^{\text{opt}} y_{kj} - \max_{Z_i \in \mathbb{Z}_+^{M \times N}} \sum_{k=1, k \neq i}^{M} \sum_{j=1}^{N} z_{kj} a_{kj}.
\]

Note that when \( N = 1 \), the generalized auction mechanism 346 presented above becomes the well-known second-price auction 347 [19]. Although the optimization problems in (1) and (2) are discretized optimizations, they can efficiently be solved using 348 linear programming. As argued in [20], the linear optimization 349 problem can be solved in polynomial time, and hence, the CSM only requires limited computational complexity.

The information exchange between the CSM and the SUs is 352 illustrated in Fig. 2. From Fig. 2, we note that, at each 353 stage, the CSM first broadcasts the available TxOps to all the 354 SUs for the auction, and then each SU submits its own bid entry 355 over all the available TxOps. After receiving the bids, the 356 CSM computes the auction results and sends back to the users 357 the channel allocations and the corresponding payments. The 358 signaling required for the auction is most often implemented 359 at the application layer. In the worst case, the amount of 360

\footnote{Note that other fairness solutions than maximizing the social welfare could be adopted, and this will not influence our proposed solution.}
data communicated between the CSM to the SU is equal to 
\((M + 1)N + nN\) bits, where \(n\) is the amount of bits repre-
senting the payment for each SU. The amount of data commu-
nicated 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 with traditional one-stage resource allocation
methods, our proposed auction mechanism has the following
advantages.

1. Unlike traditional centralized resource allocation meth-
ods [30], our proposed auction mechanism is not required
to know the SU’s 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 SU to submit their bid vectors for the avail-
able TxOps. The bid vector computation is performed
by the SU, but not the CSM, based on their utili-

2. Unlike traditional decentralized resource allocation meth-
ods [28] where multiple iterations are required before
convergence, our proposed auction mechanism only re-
quires the SU to submit the bid vectors once. Hence,
our proposed auction mechanism is suitable for online
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 Section I, each SU needs to cope with two
types of “uncertainties,” i.e., disturbances from the environ-
ment and interactions with other SUs. The environment is charac-
terized by packet arrivals from the source (i.e., source/traffic
characterization) connected with the transmitter and the chan-
nel 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 multiuser
interaction proposed in this paper.

For illustration, we assume that each SU 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 extensively been used for
delay-sensitive data transmission, e.g., leaky bucket model 410
for video transmission [25]. The number of packets in the buffer
at time slot \(t\) is denoted as \(x_i^t(0 \leq x_i^t \leq X_i)\). We assume that 412
the packets arrive from the source at the beginning of each time
slot, i.e., \(x_i^t\) is only updated at the beginning of a time slot. The 414
number of packets arriving into the buffer during one time slot
is a random variable independent of the time \(t\) and denoted as 416
\(x_i\). \(x_i\) follows the Poisson distribution with the average arrival 417
rate \(\lambda_i\) packets per second [11]. However, note that the Poisson 418
process is simply used for illustration purposes, and other traffic
models (e.g., renewal process, etc.) can also be used in our 420
framework. The average number of packets arriving during one 421
time slot is equal to \(\lambda_i \Delta T [11]\).

The condition of channel \(j\) experienced by SU \(i\) is rep-
resented by the signal-to-noise ratio (SNR) and denoted as \(\rho_{ij}^t\) (in decibels). When \(y_j^t = 1\), we assume that the channel 423
condition of each channel can be represented by a set of discrete 425
SNR values, i.e., \(\rho_{ij}^t \in \{\sigma_{ij}^1, \ldots, \sigma_{ij}^K\}\). Note that the number of 426
discrete SNR values \(K\) can be determined by SU \(i\) by trading 427
the complexity (a larger \(K\) leads to a larger state space) and 428
the resulting impact on performance. When \(y_j^t = 0\), we set \(\rho_{ij}^t\) 429
equal to \(-\infty\), which means that the channel is unavailable to 430
SU \(i\) at that time. As shown in [24], when \(y_j^t = 1\), the channel 431
condition (in terms of SNR) can also be modeled as a finite-state 432
Markov chain, where the transition from channel condition \(\sigma_{ij}^t\) 433
to time \(t\) to channel condition \(\sigma_{ij}^{t+1}\) at time \(t + 1\) plus takes place with 434
probability \(\pi_{jk}^{-t-k}\). These transition probabilities can easily be 435
estimated by SU \(i\) by repeatedly interacting with the channel. 436
We denote by \(\pi_{jk}^{-t-k}\) the probability that the channel condi-
tion \(\sigma_{ij}^{t+1}\) at time \(t + 1\), knowing that \(y_j^t = 0\) and \(y_j^{t+1} = 1\). 437
The probability that the channel condition transition to \(-\infty\), 438
knowing that \(y_j^{t+1} = 0\), is 0 no matter in what condition the 439
channel \(j\) is at time \(t\). Then, the combination \((y_j^t, \rho_{ij}^t)\) is still a 440
Markov chain with state transition probability as in (3), shown 441
described at the bottom of the page.

To model the dynamics experienced by SU \(i\) at time \(t\) in 442
the SN, we define a “state” \(s_i^t = (y_i^t, \rho_{ij}^t) \in S_i\), where \(\rho_{ij}^t \in [443
(\rho_{ij}^1, \ldots, \rho_{ij}^N)\). The state encapsulates the current buffer state 444
classified as the state of each channel. \(S_i\) is the set of possible 445
states. The total number of possible states for SU \(i\) is equal to 446
\(|S_i| = (X_i + 1) \times (K + 1)^N\). We will show later in this paper 447
that the state information is sufficient for SU \(i\) to compete for 448
resources (make bid vector) at the current time.

\[p(y_j^{t+1}, \rho_{ij}^{t+1} | y_j^t, \rho_{ij}^t) = \begin{cases} 
(1 - p_{ij}^{N}) p_{ij}^{-k}, & \text{if } y_j^t = 1, \quad \rho_{ij}^t = \sigma_{ij}^t, \quad y_j^{t+1} = 1, \quad \rho_{ij}^{t+1} = \sigma_{ij}^k \\
\rho_{ij}^{N} p_{ij}^{-k}, & \text{if } y_j^t = 0, \quad y_j^{t+1} = 1, \quad \rho_{ij}^{t+1} = \sigma_{ij}^k \\
1 - p_{ij}^{N}, & \text{o. w.} 
\end{cases} \]  

(3)
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 $\rho_i^t$. In this paper, we assume that the channel state transition is independent of the buffer state transition. In the above, we describe the transition of the channel state $\rho_i^t$ and the TxOp $y^t$. The buffer state transition is determined by the number of packets arriving and the channel allocation $z_i^t$ as well as the internal action $b_i^t$ during that time slot.

The number of packets transmitted at stage $t$ is denoted by $N_i(s_i^t, z_i^t, b_i^t)$. Given the channel allocation, SU $i$ can adapt its own internal action to maximize the number of transmitted packets, i.e.,

$$ n_i(s_i^t, z_i^t) = \max_{b_i \in B_i} N_i(s_i^t, z_i^t, b_i^t). \quad (4) $$

The optimization can be performed by a cross-layer adaptation algorithm as in [5], [12], and [21]. Since our focus is on the multi-SU interaction, we assume that the internal action will always be performed to maximize the number of transmitted packets. We simply use $n_i(s_i^t, z_i^t)$ 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 $v_i^{t+1} = \min\{(v_i^t - n(s_i^t, z_i^t) + \chi_i, X_i\}$. We define $h_i = v_i^{t+1} - (v_i^t - n(s_i^t, z_i^t))$. Based on the packet arrival model, the buffer state transition probability is computed as in (5), shown at the bottom of the page. The state transition combined with TxOps, given the current resource allocation $z_i^t$, can be computed as

$$ g_i(s_i^{t+1}, y^{t+1}|s_i^t, y^t, z_i^t) = p_i^{\text{buf}}(v_i^{t+1} | v_i^t, z_i^t) \prod_{j=1}^N p_i(y_j^{t+1} | b_j^{t+1}, v_j^t, \rho_j^t). \quad (6) $$

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 $z_i^t$, 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.,

$$ g_i^t(s_i^t, z_i^t) = -(v_i^t - n(s_i^t, z_i^t)) + \chi_i, X_i. $$

The reward at time $t$ for SU $i$ is expressed using the quasi-linear form

$$ r_i(s_i^t, \rho_i^t) = g_i^t + \tau_i^t. $$

Note that the gain $g_i^t$ and payment $\tau_i^t$ depend on the states and bids of all the competing SUs in the SN. Hence, the reward is also rewritten as $r_i(s_i^t, y_i^t, \alpha_i^t)$. 

### 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 the SUs for an infinite number of stages. This assumption is reasonable for applications having a long 500 duration, such as video streaming. In our network setting, we define a history of the stochastic game up to time $t$ as $h^t = \{s^0, y^0, \alpha^0, z^0, \tau^0, \ldots, s^{t-1}, y^{t-1}, \alpha^{t-1}, z^{t-1}, \tau^{t-1}, s^t, y^t\}$ in $\mathcal{H}^t$, which summarizes all previous states, various TxOps, and the actions taken by the SUs as well as the outcomes at 505 each stage of the auction game, and $\mathcal{H}^t$ is the set of all possible 506 histories up to time $t$. However, during the stochastic game, 507 each SU cannot observe the entire history but rather part of 508 the history $h^t$. The observation of SU $i$ is denoted as $o_i^t \in \mathcal{O}_i^t$ 509 and $o_i^t \subset h_i^t$. Note that the current state $s_i^t$ can always be observed, i.e., $s_i^t \in o_i^t$. In this paper, we focus on the 511 external action selection for the SUs. The external action selection 512 for SU $i$ to play the stochastic game is also referred to as a 513 bidding policy $\pi_i^t : \mathcal{O}_i^t \rightarrow A_i$ for SU $i$ at time $t$ and defined 514 as a mapping from the observations up to the time $t$ into the 515 specific action, i.e., $\alpha_i^t = \pi_i^t(o_i^t)$. Furthermore, a policy profile $\pi$, for SU $i$ aggregates the bidding policies about how to play 516 the game over the entire course of the stochastic game, i.e., 518 $\pi_i = (\pi_i^0, \ldots, \pi_i^t, \ldots)$. The policy profile for all the SUs at 519 time slot $t$ is denoted as $\pi^t = (\pi^t_1, \ldots, \pi^t_n) = (\pi_1^t, \ldots, \pi_n^t)$. 520

The policy $\pi_i$ is said to be Markov if the bidding policy $\pi_i^t$ for all $t$, is given the current state $s_i^t$ and current avail- 521 able TxOp $y_i^t$, independent of the states, TxOps, and actions prior to the time $t$, i.e., $\pi_i^t(o_i^t) = \pi_i^t(s_i^t, y_i^t)$. The policy $\pi_i$ is said to be stationary if the bidding policy $\pi_i^t = \pi_i$ for all $t$, i.e., stationary 524 at time $t$. The reward $r_i(s^k, y^k, \alpha^k)$ of the stage $k$ is discounted 525 by the factor $(\alpha^k)^{t-k}$ at time $t$. The factor $\alpha_i(0 \leq \alpha_i < 1)$ is 526 the discount factor determined by a specific application 528 (for instance, for video streaming applications, this factor can 529 be set based on the acceptable delay). The total discounted sum 530 of rewards $Q_i^t(s_i^t, y_i^t, \pi)$ for SU $i$ can be calculated at time $531 t$, starting from the state profile $s_i^t$, assuming that all SUs 532 deploy stationary and Markov policies $\pi = (\pi_1, \pi_\ldots, \pi_n)$, as in (7), 533 shown at the bottom of the next page. The total discounted sum 534 of rewards in (7) consists of two parts: 1) the current 535 stage reward and 2) the expected future reward discounted by 536 $\alpha_i$. Note that SU $i$ cannot independently determine the above value without explicitly knowing the policies and states of other 538 SUs. The SU maximizes the total discounted sum of future 539 rewards to select the bidding policy, which explicitly considers 540

$$ p_i^{\text{buf}}(v_i^{t+1} | v_i^t, z_i^t) = \left\{ \begin{array}{ll} \frac{(\mu_i \Delta T)^{h_i - n(s_i^t, z_i^t)} \sum_{k=h_i}^\infty (\mu_i \Delta T)^k}{h_i!}, & \text{if } 0 \leq h < X_i - (v_i^t - n(s_i^t, z_i^t)) \\ \frac{(\mu_i \Delta T)^{h_i - n(s_i^t, z_i^t)} \sum_{k=h_i}^\infty (\mu_i \Delta T)^k}{h_i!}, & \text{if } h = X_i - (v_i^t - n(s_i^t, z_i^t)) \end{array} \right. \quad (5) $$
The impact of the current bid vector on the expected future rewards. We define the best response $\beta_i$ for SU $i$ to other SUs’ policies $\pi_{-i}$ as

$$\beta_i(\pi_{-i}) = \arg \max_{\pi_i} Q_i^t(s^i, y^i, (\pi_i, \pi_{-i})). \hspace{1cm} (8)$$

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 that each SU $i$ follows to compete for the channel opportunities is illustrated in Fig. 3. In this procedure, the bidding strategy $\pi_i^t$ 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 to improve 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 (similar 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 $u^t_{ij}$ over the channel $j$, which captures the benefit derived when using that channel. The preference $u^t_{ij}$ 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 $u^t_{ij}^{\text{opt}}$ that SU $i$ can take on channel $j$ at time $t$ is the bid maximizing $u^t_{ij}$ in the auction discussed in Section III. The optimal bid that SU $i$ can make is $u^t_{ij}^{\text{opt}} = u^t_{ij}$, i.e., the optimal bid for SU $i$ is to announce 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 $u^t_{ij}$ in the context of the 576 stochastic game model. Using the channel $j$, SU $i$ obtains the immediate gain $g_i^t(s^i, y^i, e_j)$ by transmitting the pack-ets in its buffer, where $e_j$ indicates that channel $j$ is allocated to SU $i$ during the current time slot. SU $i$ then moves into the next state $s^t_i + 1$ from which it may obtain the future reward $Q_i^t + 1(s^t_i + 1, y^t + 1, \pi)$. On the other hand, if no channel is assigned to SU $i$, it receives the 578 immediate gain $g_i^t(s^i, y^i, 0)$ and then moves into the next state $s^t_i + 1$, from which it may obtain the future reward $Q_i^t + 1(s^t_i + 1, y^t + 1, \pi)$. We define a feasible set of channel assignments to SU $i$’s opponents (given SU $i$’s channel allocation $z^t_i$) as $Z^t_{ij}(z^t_i)$, with $Z^t_{ij}(z^t_i) = \{z^t_{ij} \mid \sum_{k=1}^{M} z^t_{kj} = y^t - z^t_i \forall j, \sum_{k=1}^{M} z^t_{kj} < 1 \forall k \neq i \}$. The preference over the current state can then be computed as

$$u^t_{ij}(s^i, y^i) = g_i^t(s^i, y^i, e_j) + \alpha_i \sum_{y^t+1 \in \mathbb{G}} \left[ q_i(s^t_i + 1, y^t + 1 | s^i, y^i, e_j) \sum_{Z^t_{ij}(e_j)} \left( \prod_{k=1}^{M} q_i(s^t_k + 1, y^t + 1 | s^i_k, y^i, z^t_k, e_j) Q_i^t + 1(s^t_i + 1, y^t + 1, \pi) \right) - q_i(s^t_i + 1, y^t + 1 | s^i, y^i, 0) \sum_{Z^t_{ij}(e_j)} \left( \prod_{k=1}^{M} q_i(s^t_k + 1, y^t + 1 | s^i_k, y^i, z^t_k, e_j) Q_i^t + 1(s^t_i + 1, y^t + 1, \pi) \right) \right] \right]. \hspace{1cm} (9)$$

The expected stage reward at time $t$ is

$$Q_i^t(s^i, y^i, \pi) = \sum_{k=t}^{\infty} (\alpha_i)^{k-t} r_i(s^k, y^k, \pi(s^k, y^k)) = r_i(s^t, y^t, \pi(s^t, y^t)) \text{ stage reward at time } t$$

$$+ \alpha_i \sum_{y^t+1 \in \mathbb{G}} \left( \prod_{k=1}^{M} q_i(s^t_k + 1, y^t + 1 | s^i_k, y^i, z^t_k, \pi(s^t, y^t)) \right) \sum_{Z^t_{ij}(e_j)} \left( \prod_{k=1}^{M} q_i(s^t_k + 1, y^t + 1 | s^i_k, y^i, z^t_k, e_j) Q_i^t + 1(s^t_i + 1, y^t + 1, \pi) \right) \text{ expected future reward}$$

$$= \left( g_i^t(s^t_i, y^t, z^t_i(\pi(s^t, y^t))) + \tau_i^t(\pi(s^t, y^t)) \right) \text{ stage reward at time } t$$

$$+ \alpha_i \sum_{y^t+1 \in \mathbb{G}} \left( \prod_{k=1}^{M} q_i(s^t_k + 1, y^t + 1 | s^i_k, y^i, z^t_k, \pi(s^t, y^t)) \right) \sum_{Z^t_{ij}(e_j)} \left( \prod_{k=1}^{M} q_i(s^t_k + 1, y^t + 1 | s^i_k, y^i, z^t_k, e_j) Q_i^t + 1(s^t_i + 1, y^t + 1, \pi) \right) \text{ expected future reward} \hspace{1cm} (7)$$
Fig. 3. Procedure for SU $i$ to play the auction game at time slot $t$.

From this equation, it is clear that the true value $u_{t}^{ij}$ depends not only on its own current state $s_{t}^{i}$ but also on the other SUs’ states $s_{t}^{j}, j \neq i$, the channel allocations $z_{t}^{j}(e_{j})$ to the other users when channel $j$ is assigned to SU $i$, $z_{t}^{j}(0)$ when SU $i$ is not assigned to any channel, and the state transition models $q_{k}(s_{t}^{i+1}, y_{t}^{i+1} | s_{t}^{i}, y_{t}^{i}) \forall k$. 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 $u_{t}^{ij}(s_{t}, y_{t})$.

Without knowing the other SUs’ states and state transition models, SU $i$ cannot derive its optimal bidding strategy $a_{t}^{i, \text{opt}} = u_{t}^{i}(s_{t}^{i}, y_{t}^{i})$. However, if SU $i$ chooses the bid vector by only maximizing the immediate reward $g_{t}^{i} + \tau_{t}^{i}$, i.e., the total discounted sum of reward degenerates in $Q_{t}^{i}(s_{t}^{i}, y_{t}^{i}, \pi) = g_{t}^{i}(s_{t}^{i}, y_{t}^{i}, z_{t}^{i}(\pi(s_{t}^{i}, y_{t}^{i}))) + \tau_{t}^{i}(\pi(s_{t}^{i}, y_{t}^{i}))$ by setting $\alpha_{i} = 0$. Then, the preference over channel $j$ becomes $u_{t}^{ij}(s_{t}^{i}, y_{t}^{i}) = g_{t}^{i}(s_{t}^{i}, y_{t}^{i}, e_{j}) - g_{t}^{i}(s_{t}^{i}, y_{t}^{i}, 0)$. Now, since $u_{t}^{ij}$ only depends on the state $s_{t}^{i}$, SU $i$ can compute both the optimal bid vector and 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 $\pi_{i}^{\text{myopic}}$.

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 $\alpha_{i}$. 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?

Section V-B shows that an SU needs to know the other SUs’ states and state transition models to derive its own optimal bidding policy. This coupling among SUs is due to the shared nature of wireless resources. However, an SU cannot exactly know the other SUs’ models and private information in 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 $L_{i}$ for SU $i$ as a function taking the observation $\alpha_{i}^{t}$ as input and having the 631 bidding policy $\pi_{i}^{t}$ 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 multiagent 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 \( L_i \) as \( \pi_i^T \). An SU will learn to improve its bidding policy and its rewards from participating in the auction game. The performance of the bidding strategy \( \pi_i \) is defined as the average reward that SU \( i \) obtains in a time window with length \( T \) when it adopts \( \pi_i \), i.e.,

\[
\mathcal{V}^{\pi_i}(T) = \frac{1}{T} \sum_{k=1}^{T} r_k^i.
\]  

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

### 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 Fig. 1, at the beginning of time slot \( t \), the SUs submit the bids \( a_i^t \). Then, the CSM returns the channel allocations \( z_i^t \) \( \forall i \) 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 \( o_i^t = \{ s_i^t, y_i^t, a_i^t, z_i^0, \tau_i^0, \ldots, s_i^{t-1}, y_i^{t-1}, a_i^{t-1}, z_i^{t-1}, \tau_i^{t-1}, s_i^t, y_i^t \} \). If the information is exchanged among SUs or broadcasted and overheard by all SUs, the observed information by SU \( i \) becomes \( o_i^t = \{ s_i^t, y_i^t, a_i^t, 0, z^0_i, \tau_i^0, \ldots, s_i^{t-1}, y_i^{t-1}, a_i^{t-1}, z_i^{t-1}, \tau_i^{t-1}, s_i^t, y_i^t \} \). 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 \( o_i^t \). In this paper, we assume that SU \( i \) observes the information \( o_i^t = \{ s_i^t, y_i^t, a_i^t, z_i^t, \tau_i^t, s_i^0, y_i^0, a_i^0, z_i^0, \tau_i^0, \ldots, s_i^{t-1}, y_i^{t-1}, a_i^{t-1}, z_i^{t-1}, \tau_i^{t-1}, s_i^t, y_i^t \} \).

### C. What to Learn?

In Section VI-A, we introduce 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 (9), we can see that SU \( i \) needs to learn the following: 1) the state space of other SUs, i.e., \( S_i \); 2) the current state of other SUs, i.e., \( s_i^t \); 3) the transition probability of other SUs, i.e., \( \prod_{k \neq i} q_k(s_k^{t+1}, y_k^t | s_k^t, y_k^t, z_k^t) \); 4) the resource allocations \( Z_i^t(\cdot) \) \( \forall i \) and \( Z_i^t(\cdot) \) \( \forall i \); and 5) the discounted sum of rewards \( Q_i^t(s_i^t, y_i^t, \pi) \). However, SU \( i \) can only observe the information \( o_i^t = \{ s_i^t, y_i^t, a_i^t, 0, z^0_i, \tau_i^0, \ldots, s_i^{t-1}, y_i^{t-1}, a_i^{t-1}, z_i^{t-1}, \tau_i^{t-1}, s_i^t, y_i^t \} \) 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 \( S_i \) into \( H_i \) classes, each of which is represented by a representative state \( \hat{s}_{i,h} \), \( h \in \{1, \ldots, H_i\} \). We discuss how the space \( S_i \) is decomposed in Section VI-D. By dividing the state space \( S_i \), the transition probability \( \prod_{k \neq i} q_k(s_k^{t+1}, y_k^t | s_k^t, y_k^t, z_k^t) \) is approximated by \( q_i(s_i^{t+1}, y_i^t | s_i^t, y_i^t, z_i^t) \), where \( s_i^t \) and \( y_i^t \) are the representative states of the classes to which \( s_i^t \) and \( y_i^t \) belong. This approximation is performed by aggregating all the other SUs’ states into one representative state and assuming that the transition depends on the resource allocation \( z_i^t \). The transition probability approximation is also discussed in Section VI-D. The discounted sum of rewards \( Q_i^t(s_i^t, y_i^t) \) is approximated by \( V_i^t \) \( \forall i \). Now, the problem that needs to be solved by SU \( i \) is how it can improve its own policy from past observations: 1) how the space \( S_i \) is classified; 2) the transition probability \( q_i(s_i^{t+1}, y_i^t | s_i^t, y_i^t, z_i^t) \); and 3) the approximate future rewards \( V_i^{t+1}(s_i^{t+1}, y_i^{t+1}) \).

### 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 \( S_i \):** As discussed in Section VI-B, only \( o_i^t = \{ s_i^t, y_i^t, a_i^t, 0, z^0_i, \tau_i^0, \ldots, s_i^{t-1}, y_i^{t-1}, a_i^{t-1}, z_i^{t-1}, \tau_i^{t-1}, s_i^t, y_i^t \} \) are observed. From the auction mechanism presented in Section III, we know that the value of 714...
We assume that the maximum absolute tax is $\Gamma$. We split the range $[0, \Gamma]$ into $[\Gamma_0, \Gamma_1), [\Gamma_1, \Gamma_2), \ldots, [\Gamma_{H-1}, \Gamma_H]$ with $0 = \Gamma_0 \leq \Gamma_1 \leq \cdots \leq \Gamma_{H-1} = \Gamma$. Here, we assume that the values 726 of $\{\Gamma_1, \ldots, \Gamma_{H-1}\}$ are equally located in the range of $[0, \Gamma]$. 727 (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 $t$.

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

$$\tilde{s}_{i,t}^t = h, \text{ if } |\tau_{i,t}^t| \in [\Gamma_{h-1}, \Gamma_h).$$

2) If the resource allocation $z_{i,t}^t = 0$ but $y_{i,t}^t \neq 0$, the tax is 0. In this case, we cannot use the tax to predict network congestion. However, we can infer that the congestion is more severe than the minimum bid for those available channels, i.e., $\min_{j:t^y_j \neq 0}\{a_{i,j}^t\}$. This is because, in this current stage of the auction game, only SU $i'$ with $a_{i',j}^t \geq a_{i,j}^t$ can obtain channel $j$, which indicates that $|\tau_{i',t}^t| \geq \min_{j:t^y_j \neq 0}\{a_{i,j}^t\}$ if SU $i$ is allocated any channel. Then, the representative state of the other SUs is chosen as

$$\tilde{s}_{i,t}^t = h, \text{ if } \min_{j:t^y_j \neq 0}\{a_{i,j}^t\} \in [\Gamma_{h-1}, \Gamma_h).$$

3) If the resource allocation $z_{i,t}^t = 0$ and $y_{i,t}^t = 0$, there is no interaction among the SUs in this time slot. Hence, $\tilde{s}_{i,t}^t = \tilde{s}_{i,t}^{t-1}$.

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

2) Estimating the Transition Probability: To estimate the transition probability, SU $i$ maintains a table $F$ with size $H_i \times 747 H_i \times (N + 1)$. Each entry $f_{h',h'';i,j}$ in the table $F$ represents the number of transitions from state $s_{i,t}^t = h'$ to state $s_{i,t}^{t+1} = h''$ when the resource allocation $z_{i,t}^t$ is chosen as $e_j$ (or $0$ if $j = 0$). It is 750 clear that $H_i$ will significantly influence the complexity and 751 memory requirements, etc., of SU $i$. The update of $F$ is simply 752 based on the observation $o_{i,t}^t$ and the state classification in the 753 above section. Then, we use the frequency to approximate the 754 transition probability [15], i.e.,

$$q_{i,t}^t(s_{i,t}^{t+1} = h' | s_{i,t}^t = h'', e_j) = \frac{f_{h',h'';i,j}}{\sum_{h'\prime} f_{h',h'';i,j}}. \quad (14)$$

3) Learning the Future Reward: By classifying the state space $S_{i,t}$ and estimating the transition probability, SU $i$ can now forecast the value of the average future reward $V_{i,t}^{t+1}(s_{i,t}^{t+1}, s_{i,t+1}^{t+1})$ using learning. Equation (7) can be approximated by (15), shown at the bottom of the page.

$$\text{Equation (7) can be approximated by (15), shown at the bottom of the page.}$$

Similar to the Q-learning established in [17], we also use 761 the received rewards to update the estimation of future rewards. 762 However, the main difference between our proposed algorithm 763 and Q-learning is that our solution explicitly considers the 764 impacts of other SUs’ bidding actions through the state clas- 765 sifications and transition probability approximation.

We use a 3-D table to store the value $V_i((s_i, \tilde{s}_{-i}), y)$ with 776 $s_i \in S_i, \tilde{s}_{-i} \in \tilde{S}_{-i}$. The total number of entries in $V_i$ is $|S_i| \times 768 H_i \times 2^N$. SU $i$ updates the value of $V_i((s_i, \tilde{s}_{-i}), y)$ at time 769 $t$ according to the rules in (16), shown at the bottom of the 770 page, where $\gamma_i \in [0, 1)$ is a learning rate factor satisfying 771 $\sum_{i=1}^\infty \gamma_i = \infty$, and $\sum_{i=1}^\infty \gamma_i^2 < \infty$ [17]. In summary, the 772 learning procedure that is developed for an SU is shown in 773 Table I.

E. Complexity of Learning

In Section III, we have discussed the computation complexity incurred by the CSM and the communication cost between 777 the CSM and the SUs. In this section, we further quantify 778 the complexity of learning in terms of the computational and 779 storage burden. We use a floating-point operation (“flop”) as a 780 measure of complexity, which will provide us an estimation of 781

$$Q_i^t(s_{i,t}^t, \tilde{s}_{i,t}^{t-1}), y^t, \pi) = \left\{ \begin{array}{ll} g_i^t(s_{i,t}^t, y^t, z_{i,t}^t(\pi(s_{i,t}^t, y^t))) + \tau_{i,t}^t(\pi(s_{i,t}^t, y^t)) + \alpha_i \sum_{{(s_{i,t}^{t+1}, z_{i,t}^{t+1}) \in (S_i, S_{-i}) \cap (s_{i,t}^{t+1}, y^{t+1} \in \{0, 1\}^N) \cap (s_{i,t}^{t+1}, z_{i,t}^{t+1}) \in (S_i, S_{-i})}} q_i(s_{i,t}^{t+1}, y^{t+1}|s_{i,t}^t, y^t, z_{i,t}^t(\pi(s_{i,t}^t, y^t))) V_{i,t}^{t+1}((s_{i,t}^{t+1}, \tilde{s}_{i,t}^{t+1}), y^{t+1}) \end{array} \right\} \quad (15)$$

$$V_i^t((s_i, \tilde{s}_{-i}), y) = \begin{cases} (1 - \gamma_i) V_i^{t-1}((s_i, \tilde{s}_{-i}), y) + \gamma_i Q_i^t((s_i, \tilde{s}_{-i}), y, \pi), & \text{if } (s_{i,t}^t, \tilde{s}_{i,t}^{t-1}) = (s_i, \tilde{s}_{-i}), \quad y^t = y \\ V_i^{t-1}((s_i, \tilde{s}_{-i}), y), & \text{otherwise} \end{cases} \quad (16)$$
TABLE I
LEARNING PROCEDURE

| Initializing: $V_0^i((s_i, \hat{s}_i), y) \leftarrow 0$ for all possible states $s_i \in S_i$, $\hat{s}_i \in \hat{S}_i$. |
| Learning: | At time $t$, SU $i$: |
| a. | Observes the current state $s_i^t$ and $y^t$; |
| b. | Chooses an action $a_i^t = [n_{i1}, \ldots, n_{iN}]$ as computed in Eq. (11) by replacing $V_{t+1}^i((s_i^{t+1}, \hat{s}_i^{t+1}), y^{t+1})$ with $V_{t+1}^i((s_i^{t+1}, \hat{s}_i^{t+1}), y^{t+1})$, and then submits it to the CSM; |
| c. | Computes the representative state $\hat{s}_i^{t+1}$ as in Section VI.D.1 and update the transition probability as in Section VI.D.2; |
| d. | Updates the expected total discounted sum of the rewards $Q_i^t((s_i^t, \hat{s}_i^t), y^t, \pi)$ as in Eq. (15); |
| e. | Updates the learning rate factor $\gamma^t$, according to Eq. (16). |

---

the computational complexity required to perform the learning algorithm. In addition, based on this, we can determine how complexity grows with the increasing number of SUs [20]. At each stage, SU performs the classification of other the SUs’ states, which, in the worst case, requires a number of “flops” of approximately $N$. The number of “flops” to estimate the transition probability of other SUs’ states as in (14) is approximately $(H_i + 1)$. The number of “flops” to learn 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 two tables (i.e., transition probability table and state value table), which, in total, have $(H_i^2(N + 1) + 2^N|S_i|H_i)$ 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 available spectrum opportunities 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 only present here several illustrative examples. However, the same observations can be obtained using a larger number of SUs or channels.

A. Various Bidding Strategies for Dynamic Multiuser 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 $\pi_i^{fixed}$: 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, $\pi_i^{fixed}$ does not consider any of the dynamics defined in Section IV.

2) Source-aware bidding strategy $\pi_i^{source}$: 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 $\pi_i^{myopic}$: 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 future rewards.

4) Bidding strategy based on best-response learning $\pi_i^{L}$. This strategy is produced using the learning algorithm proposed in Section VI. $\pi_i^{L}$ considers the two types of dynamics defined in Section IV and the interaction impact on future reward.

In terms of required information, the above bidding strategies are illustrated in Fig. 4. For instance, the fixed bidding strategy $\pi_i^{fixed}$ does not require information about SU $i$’s state or other SUs’ states. The source-aware bidding strategy $\pi_i^{source}$
TABLE II

<table>
<thead>
<tr>
<th>Bidding Strategies</th>
<th>Packet loss rate (%)</th>
<th>Average tax</th>
<th>Average cost</th>
<th>Packet loss rate (10%)</th>
<th>Average tax</th>
<th>Average cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scenario 1</td>
<td>$\pi_1^{\text{fixed}}, \pi_2^{\text{fixed}}$</td>
<td>32.53</td>
<td>0.4875</td>
<td>2.8966</td>
<td>31.05</td>
<td>0.5095</td>
</tr>
<tr>
<td>Scenario 2</td>
<td>$\pi_1^{\text{fixed}}, \pi_2^{\text{myopic}}$</td>
<td>34.36</td>
<td>0.1222</td>
<td>2.6337</td>
<td>14.39</td>
<td>0.5495</td>
</tr>
<tr>
<td>Scenario 3</td>
<td>$\pi_1^{\text{source}}, \pi_2^{\text{myopic}}$</td>
<td>29.83</td>
<td>0.3147</td>
<td>2.4915</td>
<td>18.11</td>
<td>0.6048</td>
</tr>
<tr>
<td>Scenario 4</td>
<td>$\pi_1^{\text{myopic}}, \pi_2^{\text{myopic}}$</td>
<td>21.55</td>
<td>0.4669</td>
<td>1.9767</td>
<td>19.55</td>
<td>0.3763</td>
</tr>
<tr>
<td>Scenario 5</td>
<td>$\pi_1^{\text{non}}, \pi_2^{\text{myopic}}$</td>
<td>15.14</td>
<td>0.6923</td>
<td>1.7428</td>
<td>27.29</td>
<td>0.4197</td>
</tr>
</tbody>
</table>

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

In this simulation, we consider five scenarios. In scenario 1, both SU 1 and SU 2 deploy the fixed bidding strategy $\pi_1^{\text{fixed}}$. In scenarios 2–5, SU 1 deploys the fixed bidding strategy $\pi_1^{\text{fixed}}$, source-aware bidding strategy $\pi_1^{\text{source}}$, myopic bidding strategy $\pi_1^{\text{myopic}}$, and best-response learning-based bidding strategy $\pi_1^{\text{L}}$, respectively, and SU 2 always deploys the myopic bidding strategy $\pi_2^{\text{myopic}}$. 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 $r_t^i = (g_t^i + \tau_t^i)$, 874 where $g_t^i + \tau_t^i$ being the number of packet lost plus the tax $\tau$ charged by the CSM (note that $\tau_t^i \leq 0$). This can be interpreted as the cost incurred at each stage. Similar to (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. The packet loss rate, average tax, and cost per time slot are presented in Table II. The accumulated packet loss and cost of SU 1 for the five scenarios are plotted in Fig. 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 more accurately value the channel opportunities 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 with 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 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 with 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 further improves 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 with scenario 4). This significant improvement is due to the ability of the SU to learn and forecast the future impact of its current actions.

It is also worth noting that the reduction of packet loss rate of SU 1 in scenarios 2–5 comes from two parts: One is the advanced bidding strategies, which allow 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; the other one is the increase in the amount of resources consumed by SU 1, which corresponds to a higher tax charged by the CSM, as shown in Table II.

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 II) 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 1 Mb/s. The number of channels is 3, and the channel condition of all the five SUs on each 948 channel takes only three values (K = 3), which are 18, 23, 949 and 26 dB. The transition probabilities are \( p_{ij}^{l} = p_{ij}^{r} = 0.4 \), \( p_{ij}^{l} = 0.2 \), \( p_{ij}^{r} = 0.2 \forall i, j \). The 951 parameters of the model of the availability of the channels to 952 the SUs are \( p_{ij}^{NF} = 0.7 \) and \( p_{ij}^{FN} = 0.3 \). The length of the time 953 slot \( \Delta T \) is also \( 10^{-2} \) s. Similar parameters are used for the five 954 SUs to clearly illustrate the performance differences obtained based on the different strategies.

In this simulation, we consider only two scenarios. In sce- 955 nario 1, all SUs deploy a myopic bidding strategy \( \pi^{\text{myopic}}_{i}, i = 1, 2, \ldots, 5 \), whereas in scenario 2, SU 5 deploys the multiuser 959 learning-based bidding strategy \( \pi^{\text{LBS}}_{i} \) with the discount factor \( \gamma = 0.5 \), and the other SUs deploy the myopic bidding strategy \( \pi^{\text{myopic}}_{i}, i = 1, \ldots, 4 \). The packet loss rate and cost per time slot are incurred by the SUs are presented in Table III. The accumulated packet loss and cost of SU 5 for the five scenarios are plotted in Fig. 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. Fig. 6(a) and (b) further verifies the improvement of the performance for SU 1. However, the other SUs’ performances are decreased as they now need to compete against a learning SU (i.e., SU 5), which is able to make better bids for the available resources.

### Table III: Performance of SU 1–5 with Various Bidding Strategies in the Five SU Networks

<table>
<thead>
<tr>
<th>SU 1</th>
<th>SU 2</th>
<th>SU 3</th>
<th>SU 4</th>
<th>SU 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Packet Loss Rate (%)</td>
<td>21.14</td>
<td>19.99</td>
<td>22.05</td>
<td>21.37</td>
</tr>
<tr>
<td>Average cost</td>
<td>1.2002</td>
<td>1.1666</td>
<td>1.2123</td>
<td>1.1949</td>
</tr>
<tr>
<td>Packet Loss Rate (%)</td>
<td>25.03</td>
<td>24.20</td>
<td>25.72</td>
<td>26.02</td>
</tr>
<tr>
<td>Average cost</td>
<td>1.2992</td>
<td>1.2993</td>
<td>1.3338</td>
<td>1.3568</td>
</tr>
<tr>
<td>Packet Loss Rate (%)</td>
<td>25.03</td>
<td>24.20</td>
<td>25.72</td>
<td>26.02</td>
</tr>
<tr>
<td>Average cost</td>
<td>1.2992</td>
<td>1.2993</td>
<td>1.3338</td>
<td>1.3568</td>
</tr>
<tr>
<td>Packet Loss Rate (%)</td>
<td>25.03</td>
<td>24.20</td>
<td>25.72</td>
<td>26.02</td>
</tr>
<tr>
<td>Average cost</td>
<td>1.2992</td>
<td>1.2993</td>
<td>1.3338</td>
<td>1.3568</td>
</tr>
</tbody>
</table>

In scenario 5, SU 1 further improves 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 with scenario 4). This significant improvement is due to the ability of the SU to learn and forecast the future impact of its current actions.

It is also worth noting that the reduction of 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; the other one is the increase in the amount of resources consumed by SU 1, which corresponds to a higher tax charged by the CSM, as shown in Table II.

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 II) 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 1 Mb/s. The number of channels is 3, and the channel condition of all the five SUs on each 948 channel takes only three values (K = 3), which are 18, 23, and 26 dB. The transition probabilities are \( p_{ij}^{l} = p_{ij}^{r} = 0.4 \), \( p_{ij}^{l} = 0.2 \), \( p_{ij}^{r} = 0.2 \forall i, j \). The 951 parameters of the model of the availability of the channels to 952 the SUs are \( p_{ij}^{NF} = 0.7 \) and \( p_{ij}^{FN} = 0.3 \). The length of the time 953 slot \( \Delta T \) is also \( 10^{-2} \) s. Similar parameters are used for the five 954 SUs to clearly illustrate the performance differences obtained based on the different strategies.

In this simulation, we consider only two scenarios. In sce- 955 nario 1, all SUs deploy a myopic bidding strategy \( \pi^{\text{myopic}}_{i}, i = 1, 2, \ldots, 5 \), whereas in scenario 2, SU 5 deploys the multiuser 959 learning-based bidding strategy \( \pi^{\text{LBS}}_{i} \) with the discount factor \( \gamma = 0.5 \), and the other SUs deploy the myopic bidding strategy \( \pi^{\text{myopic}}_{i}, i = 1, \ldots, 4 \). The packet loss rate and cost per time slot are incurred by the SUs are presented in Table III. The accumulated packet loss and cost of SU 5 for the five scenarios are plotted in Fig. 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. Fig. 6(a) and (b) further verifies the improvement of the performance for SU 1. However, the other SUs’ performances are decreased as they now need to compete against a learning SU (i.e., SU 5), which is able to make better bids for the available resources.

B. Multiuser Learning and Delay Impact in a Wireless Test Bed

To validate the performance of multiuser learning and the impact of various delays in a realistic network setting, we considered two SUs competing for the available TxOps in our 802.11a-enabled wireless test bed [31]. The channel condition experienced by the SUs varied between 10 and 30 dB, and we represented this variation using ten states \( K = 10 \). The parameters of the TxOp model are \( p_{ij}^{NF} = 0.6 \) and \( p_{ij}^{FN} = 0.4 \). The length of the time slot \( \Delta T \) 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 1.5 Mb/s. We compare three 986 scenarios. In scenario 1, both SUs deploy a myopic bidding strategy \( \pi^{\text{myopic}}_{i}, i = 1, 2 \). In scenario 2, SU 1 deploys the learning-based bidding strategy \( \pi^{\text{LBS}}_{i} \) with a discount factor of 0.5, and SU 2 deploys a myopic strategy \( \pi^{\text{myopic}}_{i} \). In scenario 3, both SUs deploy the learning-based bidding strategy \( \pi^{\text{LBS}}_{i}, i = 991, 992 \).
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 \(\pi^L_1\) with a discount factor of 0.5, and SU 2 deploys a myopic strategy \(\pi^\text{myopic}_2\). However, in this scenario, SU 1 streams a video sequence that can only tolerate a delay of 266 ms, which is typical for video conferencing applications.

Table IV shows the average video quality in terms of peak SNR (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 2.2 dB 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-A1 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 with scenario 1, the performance of both SUs has improved: SU 1 (SU 2) increases by 1 dB (1.2 dB) 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 IV, we note that the PSNR of SU 1 is decreased by 1.2 dB, whereas the PSNR of SU 2 is increased by 2 dB. We also observe that the cost of SU 1 is increased by 54% around 5.6%, whereas the cost of SU is decreased by 99%. In scenario 4, since SU 1 streams a video application with a lower delay deadline, it has to bid more to ensure that packets with stringent delay deadline are transmitted to the destination, and hence, SU 1 incurs a higher transmission cost (41% increased) compared with scenario 2. Although SU 1 bids more for the limited available resources, the video quality of SU 1 is reduced by 1.8 dB 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 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 SUs’ performance, we assume that SU 1 has the transition probability of \(\pi^L_j\) (i.e., \(p^{NF}_j = 0.55\) and \(p^{FN}_j = 0.45\)), which is slightly different from the true one (i.e., \(p^{NF}_j = 0.6\) and \(p^{FN}_j = 0.4\)). Table V shows the PSNRs and corresponding cost of both SUs when SU 1 has perfect or imperfect information about the TxOps.

From Table V, we observe that an inaccurate model of TxOps reduces the performance of SU 1 (i.e., the PSNR decreases by...
TABLE V

<table>
<thead>
<tr>
<th></th>
<th>Bidding strategies</th>
<th>SU 1</th>
<th>SU 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( \pi^p_1 ), ( \pi^p_2 )</td>
<td>33.0</td>
<td>5.3449</td>
</tr>
<tr>
<td>Scenario 2 (SU 1 has imperfect information)</td>
<td>( \pi^p_1 ), ( \pi^p_2 )</td>
<td>32.7</td>
<td>5.5685</td>
</tr>
</tbody>
</table>

TABLE VI

<table>
<thead>
<tr>
<th>Channel Availability Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Channel 1</td>
</tr>
<tr>
<td>( p^N )</td>
</tr>
<tr>
<td>Scenario 1</td>
</tr>
<tr>
<td>Scenario 2</td>
</tr>
<tr>
<td>Scenario 3</td>
</tr>
</tbody>
</table>

TABLE VII

<table>
<thead>
<tr>
<th></th>
<th>SU 1</th>
<th>SU 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Packet loss rate</td>
<td>Average cost</td>
</tr>
<tr>
<td>Scenario 1</td>
<td>( \pi^s_1 ), ( \pi^s_2 )</td>
<td>3.08</td>
</tr>
<tr>
<td>Scenario 2</td>
<td>( \pi^s_1 ), ( \pi^s_2 )</td>
<td>2.69</td>
</tr>
<tr>
<td>Scenario 3</td>
<td>( \pi^s_1 ), ( \pi^s_2 )</td>
<td>21.36</td>
</tr>
</tbody>
</table>

1.046 dB 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 0.2 dB, 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 demonstrated 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. We consider three types of channel dynam-ics corresponding to scenarios 1–3. The transition probabilities
of TxOps for all three scenarios are listed in Table VI. In each
scenario, we compare two cases. In the first case, both SUs
deploy myopic bidding strategies, and in the second case, SU
1 deploys the best-response learning-based bidding strategy,
while SU 2 still uses the myopic bidding strategy.

Table VII shows the average packet loss rate and cost ex-perienced by the SUs under various channel dynamics. In-

TABLE VII

<table>
<thead>
<tr>
<th></th>
<th>SU 1</th>
<th>SU 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Packet loss rate</td>
<td>Average cost</td>
</tr>
<tr>
<td>Scenario 1</td>
<td>( \pi^s_1 ), ( \pi^s_2 )</td>
<td>3.08</td>
</tr>
<tr>
<td>Scenario 2</td>
<td>( \pi^s_1 ), ( \pi^s_2 )</td>
<td>2.69</td>
</tr>
<tr>
<td>Scenario 3</td>
<td>( \pi^s_1 ), ( \pi^s_2 )</td>
<td>45.01</td>
</tr>
</tbody>
</table>

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 scenarios 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 compared with 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 allocated to it when resources are scarce (i.e., the SNR 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. CONCLUSION AND FUTURE RESEARCH

In this paper, we have modeled the dynamic resource alloca-
tion problem as a “stochastic game” played among strategic SUs. At each stage of the game, the CSM deploys a general-ized second-price auction mechanism to allocate the available spectrum resource. The SUs are allowed to simultaneously and independently make bid decisions on that resource by 1102
1102 considering their current states, experienced environment, and
1103 estimated future reward. To improve the bid decision at each
1104 stage, we propose a best-response learning algorithm to predict
1105 the possible future reward at each state. The simulation results
1106 show that our proposed learning algorithm can significantly
1107 improve the SUs’ performance.

1108 We note that the constraint of the perfect information about
1109 the available wireless resources can be relaxed for the case
1110 when the CSM and wireless users do not have perfect infor-
1111 mation about the available resources. In this case, the wire-
1112 less users can estimate and build a belief about the available
1113 resource. Hence, the stochastic game model can be extended
1114 to partially observably stochastic games [32]. This is one of
1115 our interesting future research topics. We also note that we
1116 can allow the wireless users to adapt their transmission power,
1117 which will lead to different interference levels to other users.

1118 In this case, the wireless users compete with each other for
1119 lower interference levels incurred by other users [6] instead
1120 of competing for the transmission time. This can also be for-
1121 mulated as a stochastic game, and similar learning algorithms
1122 can be developed. This forms another interesting topic of our
1123 future research. Our future work also includes analyzing the
1124 performance of SNs, where multiple SUs are deploying various
1125 learning strategies and protocols.

1126

REFERENCES

1129 applications: Shadow prices, proportional fairness and stability,” in
1132 for multimedia in the proactive enterprise,” Int. J. Network Manag., vol. 15,
1134 [4] F. Kelly, A. Maulloo, and D. Tan, “Rate control for communication net-
1138 for multi-user video streaming over heterogeneous access networks,” in
1140 [6] F. Kelly, A. Maulloo, and D. Tan, “Rate control for communication net-
1144 for multi-user video streaming over heterogeneous access networks,” in
1146 [8] F. Kelly, A. Maulloo, and D. Tan, “Rate control for communication net-
1150 for multi-user video streaming over heterogeneous access networks,” in

Fangwen Fu received the B.S. and M.S. degrees from Tsinghua University, 1215 AQ3 Beijing, China, in 2002 and 2005, respectively. He is currently working toward the Ph.D. degree with the Department of Electrical Engineering, University of California at Los Angeles.

During the summer of 2006, he was an Intern with the IBM T. J. Watson Research Center, Yorktown Heights, NY. His research interests include wireless multimedia streaming, resource management for networks and systems, applied game theory, and video processing and analysis.


She is currently an Associate Professor with the Department of Electrical Engineering, University of California, Los Angeles. She is the holder of 30 granted U.S. patents and three ISO awards. Her research interests are in multimedia communications, networking, processing, and systems.

Dr. van der Schaar received the National Science Foundation Career Award in 2004, the Best Paper Award from IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY in 2005, the Okawa Foundation Award in 2006, the IBM Faculty Award in 2005, 2007, and 2008, and the Most Cited Paper Award from EURASIP: Image Communications in 2006.