

Queue-aware Opportunistic Scheduling in Multi-channel Dynamic Spectrum Access Networks

Enas F. Khairullah¹, Mainak Chatterjee¹, and Swades De²

¹ Department of Computer Science, University of Central Florida, Orlando, FL, USA

Email: {enas, mainak}@eecs.ucf.edu

² Department of Electrical Engineering, Indian Institute of Technology Delhi, New Delhi, India

Email: swadesd@ee.iitd.ac.in

Abstract—In this paper, we propose an opportunistic scheduling algorithm for secondary users (SUs) in a dynamic spectrum access network. Scheduling is performed at the beginning of each super-frame where appropriate channel(s) are assigned to the SUs based on the expected throughput that each channel provides. We compute the expected throughput for each channel on specific time slots in a super-frame based on three factors: i) primary channel occupancy, ii) signal-to-interference-and-noise ratio, and iii) back-logged queue length of the SUs. We consider scheduling both on a frame-by-frame basis and also on a slot-by-slot basis. The state transitions are modeled using a discrete-time Markov chain (DTMC) process and their probabilities are calculated. The state transitions occur at the frame/slot boundaries. Multiple SUs are allocated the same channel on the same time slot as long as there is no interference between them. Such non-interfering sets of SUs is found by considering the expected throughput of each user on all candidate channels. In order to maximize spatial and temporal reuse of channels, the non-interfering sets are assigned to as many channels as possible on maximum possible time slots. Performance of the proposed scheduling schemes is validated using simulations where we measure metrics such as throughput, number of slots allocated, fairness, and blocking probability. We also show the efficacy of the queue-aware scheduling by comparing with one that does not consider the queues.

Index Terms—Dynamic spectrum access, cognitive radio, SINR, scheduling, throughput, fairness.

I. INTRODUCTION

In traditional radio spectrum allocation, regulators allocate specific frequency bands for various services under protective licenses. These spectrum allocations are usually long-term and any changes are made under the strict guidance of a governmental agency, for example the Federal Communications Commission (FCC) in the United States. However, there is overwhelming concern about the effective use of the spectrum that has been statically assigned. Studies have shown that more than half of the allocated spectrum is unused, depending on time and location [1]. Intuitively, with the dis-proportionate and time-varying demand, and hence usage of the spectrum, the notion of fixed spectrum assignment to providers is questionable. On the other hand, there is an increasing demand for more spectrum because of emerging wireless services and applications. In order to cater to the growing demands

and improve spectrum utilization, the concept of dynamic spectrum access (DSA) has been proposed [3], [19]. DSA allows unlicensed users (called secondary users (SUs)) to borrow idle spectrum from spectrum license holders (called primary users (PUs)), as long as the PUs are not interfered.

In presence of multiple SUs who try to operate on the same channel(s) that are not used by the PUs, there has to be some policy for channel access. Such contention are resolved by the underlying medium access control (MAC) protocol, wherein the SUs abide by some transmission policies. MAC protocols affect many network and radio functions such as channel sensing, resource allocation, and spectrum sharing [6]. It is worth mentioning that a channel can be used at multiple locations simultaneously as long as the SUs using the common channel do not interfere with each other i.e., spatially they are sufficiently apart. Such constraints are usually represented by a *conflict graph*. Nodes in the graph represent the SUs (and hence ‘nodes’ and ‘users’ are used interchangeably) and an edge represents interference between the nodes that share the edge. Transmissions occur during fixed number of time-slots that constitute a super-frame. The scheduler schedules the slots to the SUs at the beginning of the super-frame. An SU first senses the channel at the beginning of each slot and uses it for transmission only if the channel is idle [17].

In this paper, we propose a scheduling algorithm for a multi-channel DSA network where SUs are assigned channel(s) that are not being used by the PUs. Unlike traditional opportunistic scheduling algorithms (which try to maximize the instantaneous throughput), we not only consider the signal-to-interference-and-noise ratio (SINR) between a transmitter-receiver pair but also consider the number of packets waiting in the queue of the transmitter. That way, we make sure that the SUs make use of the channels to the best possible extent. This also avoids allocation to non-backlogged users. We calculate the expected throughput for each channel and for each time slot in a frame. We do so by considering the state transitions of the queues which we model using a discrete time Markov process. In order to have collision-free transmissions, we use the interference conflict graph and identify the sets of users who do not interfere with each other, and hence can transmit on the same channel and at the same time (slot). Each set of users is assigned one or more channels *and* one or more time slots based on which the overall system throughput is maximized. Through extensive

Enas F. Khairullah is also affiliated with the Department of Information Technology, King Abdulaziz university, Jeddah, Saudi Arabia. This work has been partly supported by the ITRA Media Lab Asia project under Grant no. ITRA/15(63)/Mobile/MBSSCRN/01.

simulation experiments, we show how the proposed frame-by-frame and slot-by-slot scheduling perform with respect to per-user and overall system throughput, fraction of allocated slots, fairness, blocking probability, and average delay. We also show the benefits of scheduling schemes that consider the queue status compared to the schemes that do not.

The rest of the paper is organized as follows. In Section II, discuss the related works. Section III presents the system model. Section IV describes the process of finding the non-interfering sets, allocation process, and the performance evaluation metrics. The simulation results are presented in Section V. Conclusions are drawn in the last section.

II. RELATED WORK

Opportunistic scheduling is a class of scheduling schemes where the physical layer properties (i.e., the wireless channel characteristics) are exploited. For example, in a time division system, the user with the best channel condition is selected for transmission. Opportunistic scheduling strategy exploits multiuser diversity and maximizes the overall throughput, which might be unfair to users with low long-term signal to interference and noise ratio. To strike a balance between throughput and fairness, proportional fair (PF) scheduling has been proposed in [11]. Many other scheduling policies have been proposed and analyzed (see [4] and references therein). Most of these scheduling policies are sensitive to flow-level dynamics [14]. For example, in a PF scheduler, a user is selected based on his current average throughput [11].

In [8], two optimal schedulers with exponential complexity were proposed. The first one maximized the network throughput while the second one minimized scheduling delay. Then two suboptimal schedulers, referred to as maximum frequency selection (MFS) and probabilistic frequency selection (PFS), were proposed. A queue-aware frame-based opportunistic spectrum scheduling scheme was proposed in [17]. A user uses its queue status and channel conditions to calculate its expected throughput for each channel over the frame. The main goal of that scheduling scheme was to maximize the overall throughput, for which channel assignment to the users were done with the highest expected throughput. The proposed scheme tried to handle the fast changing channel conditions by using the queues. However, this technique increases the overhead of the scheduling process since its slot-based scheduling. Also, there is no re-use to the channel during the frame. Study in [13] also proposed a queuing analytic framework to study the performance of opportunistic spectrum access by cognitive radio users. However, they did not consider the mobility of cognitive radio users across network boundaries. Also, there is no sharing for the frequency. Multi-channel time slot assignment problem in cognitive radio sensor networks was studied in [12] as an integer linear program where the goal is to minimize the number of slots in the schedule and increase the network throughput offered to sensor nodes. A similar system setting was considered in [5], where a dedicated control channel was considered for distributed multichannel access coordination among multiple SUs with possibly heterogeneous quality of service requirements. An Opportunistic centralized Time slot assignment in COgnitive Radio sensor networks

(OTICOR) was proposed in [16] which also used integer linear programming formulation for the MAC-layer scheduling problem and introduced an optimal schedule. Then a distributed heuristic scheduling schema was proposed which showed very close results to the optimal one in terms of schedule length. However, the results were valid for a small network and did not involve mobility of the nodes. A spectrum allocation algorithm for heterogeneous cognitive radio networks is proposed in [2]. However, the study was done under imperfect sensing, where the sensing time had no effect on the system performance.

Study in [18] proposed a Distributed TDMA based MAC protocol (DTMDD) for data dissemination in multi-hop ad-hoc cognitive radio networks. The objectives of this study are ensuring message reachability and avoiding collision among the SUs. An SU selects the best channel based on an intelligent selection strategy. In [8], a scheduling problem called Maximizing the Number of Satisfied Users (MNSU) was proposed to maximize throughput. Two heuristics were proposed: 1) best first resource assignment and 2) resource assignment with partial backtracking. However, this study focused on centralized cognitive radio network and not a distributed one. Research in [15] provided an opportunistic channel access scheme with channel ordering and periodic sensing. The SUs first select the k best primary channels out of N . They sense these k channels for transmission opportunity, and then probe the idle channel for channel quality measurement. However, the reward value, which is the average throughput, is calculated based on the values of threshold and transmission period only.

III. SYSTEM MODEL

We consider a DSA network with multiple SUs as well as PUs randomly scattered over an area. There are U PUs, and each owns a unique channel – thus, there are U channels. SUs rely on the PUs' channel(s) for their communication needs. We consider a time-slotted system where a super-frame is composed of N slots. We assume all the SUs are synchronized through some beacon signaling over some common control channel. Though this widely used assumption is open to criticism, nevertheless we make use of a out-of-band signaling for SUs' synchronization. At the beginning of each super-frame, all SUs sense the channels to determine the status of the PU channel occupancy. We consider perfect sensing without any hidden nodes. If any PU channel is idle, that channel can potentially be used by the SUs in that super-frame. The sensing process also allows the secondary transmitter-receiver pairs to determine their SINR on each channel which we use as a metric to determine the goodness of each channel. Due to the spatial and temporal signal variations, the SINR varies from user to user and also from channel to channel.

Let PU u use channel u . SINR at SU v on channel u is denoted as $SINR_{v,u}$. Note, the interference on channel u is from the PU and the SUs using channel u (as in (3)). SUs may or may not have packets to transmit. Backlogged packets reside in their queues. Irrespective of the packet handling process, packet arrivals are independent for each user and follow batch Bernoulli process with rate of λ packets per unit time. Packet lengths are exponentially distributed with mean l bits.

We consider a cellular-type network with a centralized scheduler which has the ability to compute or learn the SU locations; thus it can compute the interference graph. Each SU v reports $SINR_{v,u}$ and its current queue length to the scheduler. Knowledge of $SINR_{v,u}$ enables the scheduler to calculate the expected throughput of SU v on channel u .

Though the channel with the highest SINR is supposed to be the best channel for user v , allocating the best channel to user v need not maximize the throughput if user v does not have enough packets in the transmission queue. Thus, apart from $SINR_{v,u}$, the current queue size must also be taken into consideration. For example, it might be better to assign a channel to a user who can make better use of the channel, i.e., have enough packets waiting to be transmitted.

The objective of the scheduler is to schedule on frame-by-frame basis, i.e., assign slots in a frame to SUs that maximize the throughput while avoiding any conflict. It may be noted that the PU channel, SINR, and queue status can change in every slot – thus making the decision at the beginning of a frame non-optimal. Therefore, it is important to consider how and with what probability these parameters change. Such considerations do not guarantee optimal solutions, nevertheless, the solutions would be close to optimal, i.e., the solution we would have gotten if we considered slot-by-slot scheduling.

To capture the conflicts (interference) among the SUs, i.e., when they are within each other's transmission range, we use an *interference graph*, where an edge between nodes v and u indicates they are close-by and hence cannot simultaneously use the same channel. However, the channel can be simultaneously used by nodes that are non-conflicting. Such re-use of channel increases the spatial utilization of the spectrum bands.

A. Primary Channel Occupancy

We assume that the channel occupancy by PUs follows a two-state Markov process, and the process is identical and independent for each channel. The channel status could be 0 or 1 depending on the PU activity, where 0 means 'idle' and 1 means 'busy'. The channel status in slot n is denoted by C_n . Similar channel occupancy model was used in [13] and [17]. The state transition matrix $P(C_{n+1} | C_n)$ is defined as:

$$P(C_{n+1} | C_n) = \begin{pmatrix} P(0|0) & P(1|0) \\ P(0|1) & P(1|1) \end{pmatrix}$$

where $P(\hat{i} | i)$ represents the transition probability from state i in the current slot to $\hat{i}$ in the next slot. We consider two states, $(i, \hat{i} \in \{0, 1\})$. Also, with U channels from the same number of PUs, we denote the state transition matrix conforming to channel u where $u \in \{1, \dots, U\}$ as $P(C_{n+1}^u | C_n^u)$. As mentioned before, we assume that all SUs sense the channels at the beginning of each super-frame to determine the status. Thus, the transitions probabilities are known to the SUs.

B. SINR

Each SU computes the SINR perceived on the each channel. The interference experienced by an SU on a particular channel is different due to the other users at varying distances who also use that channel. Both PUs and SUs are responsible for

creating the interference. Recall that, on any channel there could be only one PU but multiple SUs that use the channel.

The interference between SU v and PU u on channel u is calculated as [10], [13]:

$$I_{v,u}^p = \frac{P_p}{d_{v,u}^\alpha} \quad (1)$$

where P_p is the transmit power from the PU u , $d_{v,u}$ is the distance between users v and u , and α is the path loss exponent. We do not explicitly consider fading and shadowing.

Denoting the SU transmit power as P_s and the distance between nodes v and n as $d_{v,n}$, the interference between SU v and all other SUs on channel u is calculated as:

$$I_{v,u}^s = \sum_{n \in N_v} \frac{P_s}{d_{v,n}^\alpha} \quad (2)$$

where $n \in N_v$ is a neighbor of node v using channel u .

As a result, interference perceived at SU v on channel u from all interfering sources can be calculated as:

$$I_{v,u} = I_{v,u}^p + I_{v,u}^s \quad (3)$$

With the interference known, we find the $SINR$ for SU v on channel u as:

$$SINR_{v,u} = \frac{\frac{P_s}{d_{v,t}^\alpha}}{n_0 + I_{v,u}} \quad (4)$$

where n_0 is the background noise, and $d_{v,t}$ is the distance between receiver v and transmitter t .

Though $SINR_{v,u} \in (SINR_{min}, SINR_{max})$, where $SINR_{min}$ and $SINR_{max}$ are the minimum and maximum SINR values respectively, we quantize $SINR_{v,u}$ as:

$$SINR_{v,u} \in [S_0, S_1, S_2, \dots, S_{S-1}] \quad (5)$$

where S is the number of quantization levels. With a little abuse of notation for ease of presentation, we write (5) as:

$$SINR_{v,u} \in [0, 1, 2, \dots, S-1] \quad (6)$$

We model the channel quality as a Markov process with S states. The state transition matrix is given by $P(s_{n+1}^{u,v} | s_n^{u,v})$ for every channel u associated with user v . The probability that the channel state changes from quality j in slot n to quality $\hat{j}$ in slot $n+1$ is $p(s_{n+1}^{u,v} = \hat{j} | s_n^{u,v} = j)$, where j and $\hat{j} \in \{0, 1, \dots, S-1\}$.

C. SU Queues

A common assumption in the literature is that all SUs have back-logged traffic, i.e., they have sufficient number of packets waiting in queue for transmission. This need not be true in all cases where some of the SUs might be lightly loaded and might even have empty queues at certain times. Hence it is necessary to check if an SU has enough packets in its queue to fill a slot, or at-least most of it. For example, it does not make sense to allocate resource to an SU who does not have any packets to transmit even if it has favorable channel conditions.

We assume that packet arrivals follow the batch Bernoulli process (BBP) where up to P packets might arrive in a slot. We represent the process as: $\alpha = [\alpha_0, \alpha_1, \dots, \alpha_P]$, where

α_i is the probability that i packets arrive in a slot. One may think of a BBP as a type of a superposition of independent and identically distributed Bernoulli process [9].

The average arrival rate, λ , can be calculated as: $\lambda = \sum_{i=0}^P i \times \alpha_i$. The number of packets in the queue are identically and independently distributed. For any user, the packets are served in a first come first serve manner.

IV. SCHEDULING USING NON-INTERFERING SETS

Based on the PU channel status, SINR, and SU queue, we propose a scheduling technique that maximizes the expected throughput. We also consider the interference conflict graph to identify the SUs who can be assigned the same set of channels to maximize the channel re-use spatially as well as temporally.

A. Expected Throughput

We calculate the expected throughput of each user on each channel for each slot as $T_n(u, v)$ where v is the user, u is the channel and n is the slot. Motivated by [11], we present a state vector $X_n^{u,v} = [s_n^{u,v}, q_n^u]$ for $n = 1, \dots, N$, where $s_n^{u,v}$ is the channel quality (i.e., SINR) for channel u for the user v in slot n , and q_n^u is the queue status of user v at the beginning of slot n . We assume that the status of a channel u in slot n is idle, that means $c_n^u = 0$ and there is no PU activity and any SU can use channel u in slot n . For simplicity, $X_n^{u,v}$ is written as X_n . The probability distribution for the state vector X_n in slot n is denoted as:

$$\Pi_n = [\Pi_{(s_n=i, q_n=k)}] \quad (7)$$

where $i \in \{0, \dots, S-1\}$ and $k \in \{0, \dots, Q\}$, where Q is the maximum queue length. It represents the probability that the values of the states s_n and q_n are i and k , respectively.

The transitions among the states occur at the slot boundaries with probability $P(x_{n+1} | x_n)$, also written as $P(s_{n+1} = \hat{i}, q_{n+1} = \hat{k} | s_n = i, q_n = k)$, where $\hat{i} \in \{0, \dots, S-1\}$, and $\hat{k} \in \{0, \dots, Q\}$. All previous states variables are discrete values and modeled with Markov chain as above.

As a result, we can model the evaluation of the state vector x_n using a discrete-time Markov process with the state transition matrix as: $P(X_{n+1} | X_n) =$

$$\begin{pmatrix} P(0,0|0,0) & P(0,1|0,0) & \dots & P(S-1,Q|0,0) \\ P(0,0|0,1) & P(0,1|0,1) & \dots & P(S-1,Q|0,1) \\ \vdots & \vdots & \ddots & \vdots \\ P(0,0|S-1,Q) & P(0,1|S-1,Q) & \dots & P(S-1,Q|S-1,Q) \end{pmatrix}$$

The transition probability is written as:

$$\begin{aligned} P(\hat{i}, \hat{k} | i, k) &= P(s_{n+1} = \hat{i}, q_{n+1} = \hat{k} | s_n = i, q_n = k) \\ &= \frac{P(s_{n+1} = \hat{i}, q_{n+1} = \hat{k}, s_n = i, q_n = k)}{P(s_n = i, q_n = k)} \\ &= \frac{P(q_{n+1} = \hat{k} | s_n = i, q_n = k, s_{n+1} = \hat{i})}{P(s_n = i, q_n = k)} \quad (8) \\ &\quad \times P(s_n = i, q_n = k, s_{n+1} = \hat{i}) \\ &= P(q_{n+1} = \hat{k} | s_n = i, q_n = k, s_{n+1} = \hat{i}) \\ &\quad \times P(s_{n+1} = \hat{i} | s_n = i, q_n = k) \end{aligned}$$

We can reformulate eqn. (8) as:

$$P(\hat{i}, \hat{k} | i, k) = P(q_{n+1} = \hat{k} | s_n = i, q_n = k) \times P(s_{n+1} = \hat{i} | s_n = i) \quad (9)$$

since the states variables of s_{n+1} and q_{n+1} are independent from one another. This is because the queue length also depends on the arrival process which is no way related to the channel quality that governs the packet departure process.

The value of $P(s_{n+1} = \hat{i} | s_n = i)$ is calculated as discussed in section III-B, while the value of $P(q_{n+1} = \hat{k} | s_n = i, q_n = k)$ is calculated based on the queue status, q_{n+1} , which depends on i) q_n , ii) the number of arriving packets at the queue, a_n , in slot n , and iii) the number of packets departing, d_n , in slot n . Thus, we get:

$$q_{n+1} = q_n - d_n + a_n \quad (10)$$

Though a good SINR would allow many packets to be transmitted from the queue, but an SU might not have that many packets. Thus, the number of packets that can be transmitted is limited by the channel capacity or the current queue length. If t_n denotes that maximum number of packets that can be potentially transmitted as per Shannon's law, then the number of packets departing in slot n is given as:

$$d_n = \min(q_n, t_n) \quad (11)$$

The expected throughput $T_n(u, v)$ on a channel u and SU v in slot n is calculated from equations (7) and (11) as:

$$T_n = \sum_{i,k} \Pi_{(s_n=i, q_n=k)} d_n \quad (12)$$

For each channel u , the expected throughput $T_n(u, v)$ for all SUs M in all slots N can be written in one $M \times N$ matrix. Similarly, for each SU, the expected throughput $T_n(u, v)$ on all channels U in all slots N is written in one $U \times N$ matrix.

B. Finding Non-Interfering Sets

With randomly deployed nodes, each SU experiences varying SINR on vacant PU channels. The aim of the proposed scheduling is to allocate time slots to the SUs which experience high SINR and have sufficient queued packets to transmit.

For allocating channels to the SUs that offers the best throughput, we recall that the same channel cannot be allocated to users that are within the interfering range of each other. However, multiple non-interfering users can simultaneously transmit on the *same* channel. Therefore, we identify sets of non-interfering users which we call *Non-Interfering Sets* (NIS). The problem of finding such non-interfering sets in an arbitrary network is known to be NP-complete.

In addition to grouping the non-conflicting users, we try group them in a set such away that they would also maximize the system throughput. Thus, we propose an algorithm to find the NISs that also maximize the throughput.

Algorithm: As mentioned before, we have an expected throughput matrix for each channel for each user for each slot. We start with the entire network represented by G . We find the user (represented by 'node' in a graph) that can achieve the highest expected throughput for first slot. We call that node

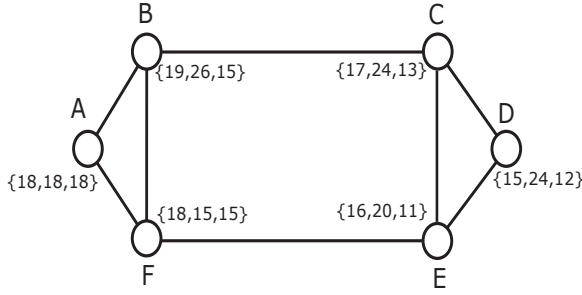


Fig. 1: Finding the non-interfering sets. $NIS_1 = (B, D)$ uses channel c_1 ; $NIS_2 = (B, E)$ uses channel c_2 ; $NIS_1 = (A, C)$ uses channel c_3 . B belongs to two NISs; F belongs to none.

n_1 and place it in the first non-interfering set, represented by NIS_1 . All neighbors of n_1 in the interference conflict graph cannot be placed NIS_1 . Therefore, we remove all such nodes and their edges from G to obtained G' .

Suppose, the throughput of n_1 was maximized for some channel, say c_1 . We follow the same process of finding the node with the highest expected throughput for first slot for the same channel (c_1) in G' (say n_2) and place it in NIS_1 . Note that, no node in G' interferes with NIS_1 . We further reduce the G' and continue to find nodes that would be belong to NIS_1 till the graph reduces to a null set. Thus, all the nodes belonging to non-interfering set NIS_1 do not interfere with each other and can transmit on channel c_1 in first slot.

With channel c_1 allocated to NIS_1 , we find the second non-interfering set from the remaining channels $U - c_1$, where U is the total channel set. We find the channel from $U - c_1$ that offers the maximum throughput. Suppose, that channel is c_2 . We continue as before to find NIS_2 which is allocated c_2 .

Note, it may happen that NIS_1 and NIS_2 can have some or all of the nodes common. However, since the two sets use different channels, having common node(s) between the two non-interfering sets is allowable. Continuing similarly, we get U non-interfering sets – each having a unique channel. Note that, some nodes may not belong to any non-interfering set.

An Illustrative Example: Let us consider a network with 6 nodes labeled A through F and 3 channels, as shown in Fig. 1. The expected throughput on each channel in a slot is shown next to a node. For example, C has the expected throughput on channels 1, 2, and 3 respectively 17, 24, and 13.

Using the proposed algorithm, we form the non-interfering sets. With 3 channels, the non-interfering sets formed are: $\{B, D\}$, $\{B, E\}$, and $\{A, C\}$. It may be noted that, B belongs to two non-interfering sets, whereas F does not belong to any. Though F is not allocated any channel during this slot, it may be assigned channel(s) in the other slots in the super-frame.

C. Allocation Process

Channels are allocated to the nodes on slot-by-slot basis. The number of available channels are allocated to the equal number of non-interfering sets. In the first slot, NIS_1 is allocated channel c_1 that maximizes its throughput. Then, channel c_2 is allocated to the NIS_2 , and so on, until all channels are allocated to all the non-interfering sets. The same

process is repeated for all slots in a super-frame. As a result, a node may be assigned more than one channel in the same slot. Also, multiple nodes are allocated the same channel. Thus, the process increases the overall throughput.

D. Metrics

In order to measure performance of the proposed algorithms, we use five metrics: throughput, number of slots allocated, fairness among users, delay, and blocking probability.

Throughput: The system is considered to have M users and U channels. We consider a large period of time consisting of T super-frames (each consisting of N slots) and quantify throughput during this time. Obviously, the total number of slots is $N \times T$. Setting $T \rightarrow \infty$ gives the steady state average.

We define and study the throughput in three different ways.

Definition 1: Average throughput of user v per slot for all channels is denoted by t_v is defined as:

$$t_v = \sum_{l=1}^T \sum_{k=1}^N \sum_{u=1}^U \frac{t_{v,u,k,l}}{n \times T} \quad (13)$$

where, $t_{v,u,k,l}$ is the throughput of user v on channel u in slot k of super-frame l .

Definition 2: Average throughput per slot due to all users and all channels denoted by S_{slot} is given as:

$$S_{slot} = \sum_{v=1}^M \sum_{l=1}^T \sum_{k=1}^N \sum_{u=1}^U \frac{t_{v,u,k,l}}{N \times T} = \sum_{v=1}^m t_v \quad (14)$$

Definition 3: Average throughput per user/per slot is denoted by $S_{user/slot}$ is given as:

$$S_{user/slot} = \frac{\sum_{v=1}^m t_v}{M} = \frac{S_{slot}}{M} \quad (15)$$

Number of slots allocated: Denote the total number of slots allocated to user v for all channels as c_v for T super-frames.

The fraction of slots allocated to user v is given as:

$$f_v = \frac{c_v}{N \times T} \quad (16)$$

The average fraction of slots allocated to all users is:

$$\bar{f} = \frac{1}{M} \sum_{v=1}^M f_v \quad (17)$$

Fairness among users: In order to determine fair resource allocation to the SUs, we use the popularly-used Jain's fairness index. With the total number of slots allocated to user v is c_v , Jain's fairness index is obtained as:

$$\mathcal{J}(c_1, c_2, \dots, c_M) = \frac{(\sum_{v=1}^M c_v)^2}{M \sum_{v=1}^M c_v^2} \quad (18)$$

The index varies from $\frac{1}{M}$ (worst case) to 1 (best case).

Delay: The overall system delay is obtained by absorbing the Markov chain. Packets arrive at the queues of all SUs with an arrival rate of λ packets per unit time. The queue length varies

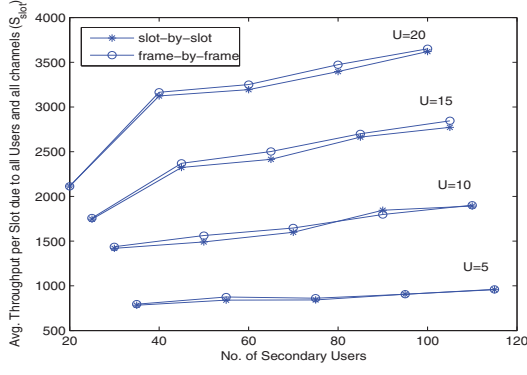


Fig. 2: Average SU system throughput per slot on all channels.

based on the number of packets that arrive and the number of packets that depart the queue. The delay incurred by a packet is the difference between the time when it enters and leaves the queue. We denote $D_{p,v}$ as the delay for packet p for user v . Average system delay is obtained as:

$$\text{Avg. System Delay} = \frac{\sum_{v=1}^M \sum_{\forall p} D_{p,v}}{N_p} \quad (19)$$

where N_p is the total number of packets sent from all users.

Blocking probability: It is defined as the probability of a user not getting a slot for transmission. In other words, the blocked users are the ones who do not belong to any non-interfering set. We find the long term fraction of those users.

V. SIMULATION MODEL AND RESULTS

In this section, we first evaluate the performance of our proposed queue-aware scheduling strategies under the standard performance metrics, namely, throughput, user-level fairness, blocking probability, and system delay. Subsequently, we conduct comparative performance evaluation with respect to the conventional strategy, namely, queue-unaware scheduling.

In order to evaluate performance of the proposed algorithms, we conduct extensive simulation experiments. We consider up to 120 nodes which were randomly scattered over a 100×100 m square grid. Various topologies from sparsely connected to densely connected were simulated. Number of channels were varied from 5 to 20, each with a bandwidth of 2 MHz. Two nodes are considered neighbors of each other if their mutual distance is within the transmitting range of 20 m. The SINR for a transmitter-receiver pair is assumed to be uniformly distributed between 0 dBW and 20 dBW. Packets arrive at the user queues following batch Bernoulli process with an average rate of 250 packets every unit time, unless otherwise specified.

A. Performance of Queue-Aware Scheduling Strategies

We study the performance of the proposed scheduling in two different ways depending on how the non-interfering sets are created: i) at the beginning of each super-frame, which we refer to as frame-by-frame scheduling, and ii) at the beginning of each slot, which we refer to as slot-by-slot scheduling. For the simulations, we consider a frame to consist of 10 slots.

Throughput: Fig. 2 shows the average throughput per slot due to all users and all channels, (S_{slot}). Fig. 3 shows the

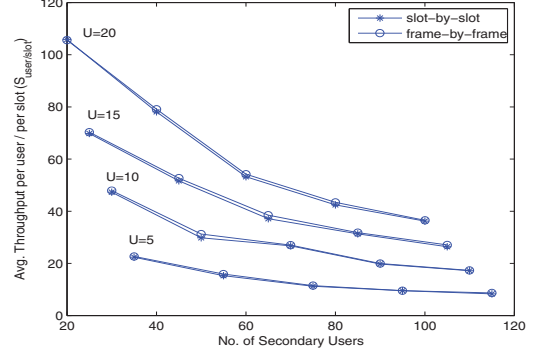


Fig. 3: Average throughput per user/per slot ($S_{user/slot}$).

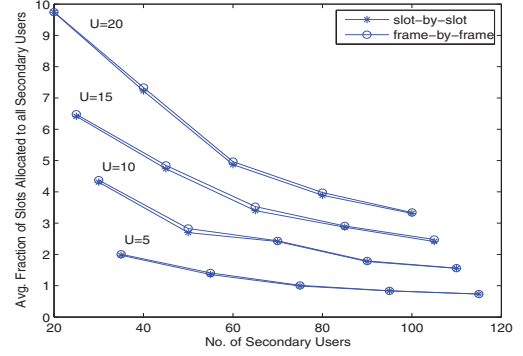


Fig. 4: Average fraction of slots allocated.

average throughput per user/per slot due to all users and all channels, ($S_{user/slot}$). As expected, with more users, the total system throughput increases. However, the throughput of each user decreases with increasing number of users. More users lead to many non-interfering sets, and as a result the available bandwidth in each slot is divided among more users, which decreases the average throughput per user per slot.

Overall, the proposed algorithm variants have the ability to satisfy more users and also attain higher throughput via the use of the Markov model discussed earlier. Also, the ability to use the same channel in the same slot by multiple users allows better spatial reuse of the channels.

Slot-by-slot scheduling yields better performance than frame-by-frame scheduling most of the time. This is primarily because the scheduler is able to make many short-term decisions with less error than making decisions for all the slots in a frame. Of course, this enhanced performance comes at the cost of more frequent schedule cycles.

Generally, the proposed methods have an ability to satisfy more users and yield more system throughput compared with [13] and [17]. Since in [17], a channel is assigned only to one user, there is no reuse of the channel within the super-frame. In [13], a channel is assigned only to one user in slot, so there is no re-use of the channel within the slot.

Fig. 4 exhibits the average fraction of slots allocated to all users for increasing number of users. With more users, the fraction of slots allocated decreases as the available bandwidth is shared among them. More number of channels mean more

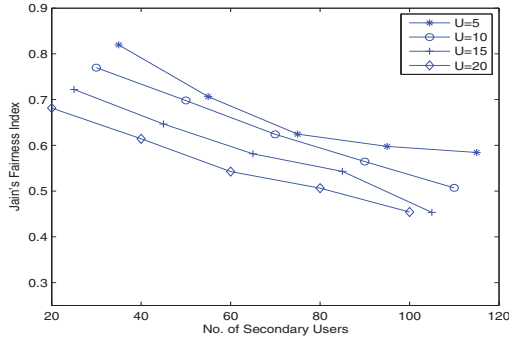


Fig. 5: Jain's Fairness Index for slot-by-slot scheduling.

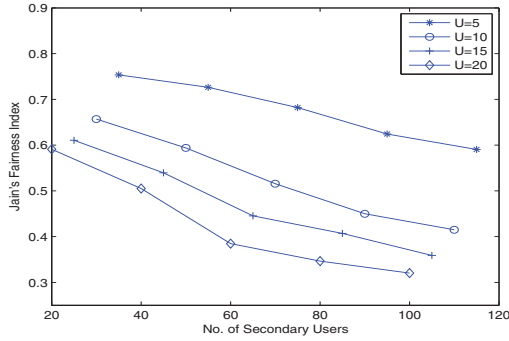


Fig. 6: Jain's Fairness Index for frame-by-frame scheduling.

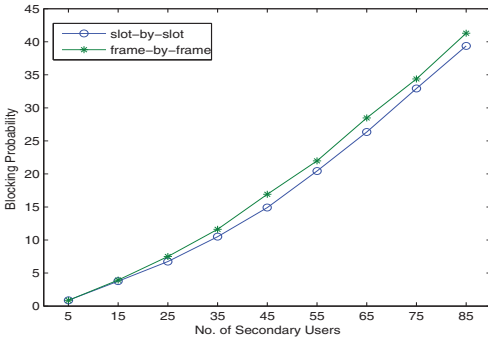


Fig. 7: Blocking probability.

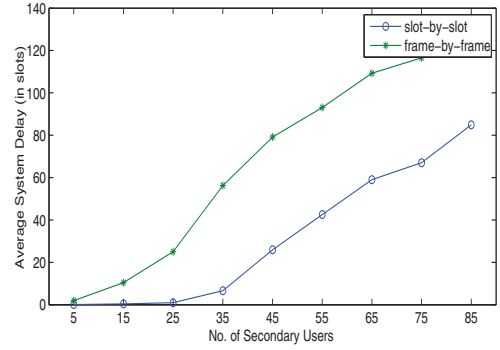


Fig. 8: Average system delay (in slots) versus number of SUs.

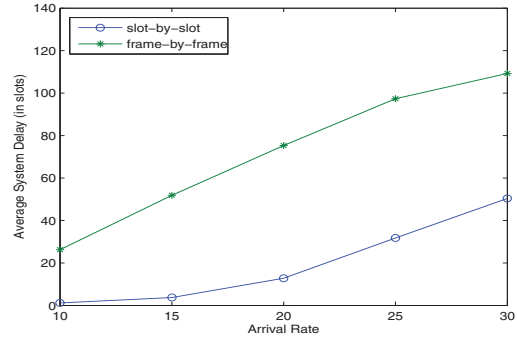


Fig. 9: Average system delay (in slots) versus arrival rates.

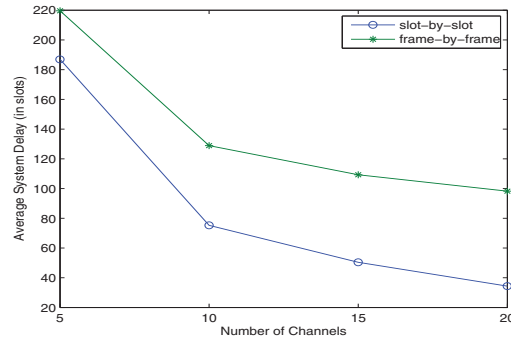


Fig. 10: Average system delay versus number of channels.

slots allocated to the users. Again, both scheduling methods manage to assign slots to all users.

Fairness: To maximize the system throughput, users with low SINR are not assigned any slot. Thus, with more number of users the fairness decreases, as demonstrated in Figs. 5 and 6, where the effect of number of channels is also shown.

Blocking probability: Fig. 7 shows the blocking probability when $\lambda = 30$ and number of channels being 15. More number of users means more conflict, as a result the number of user that do not belong to any independent set is increasing, causing higher blocking probability. These users are blocked and do not get any chance to transmit in a super-frame. The blocking performance is slightly better for slot-by-slot scheduling as non-interfering sets are calculated after every slot, allowing more users to be accommodated.

Delay: Figs. 8, 9, and 10 show the delay performance with

varying number of available channels, arrival rates, and number of users. In Fig. 8, $\lambda = 30$ and number of channels =15, while in Fig 9, λ is varied between 10 packets per unit time to 30 packets per unit time, number of channels was set at 15 with number of users being 80. In Fig 10, number of channels are varied between 5 to 20, with $\lambda = 30$, and number of users as 80. As expected, with more users and higher packet arrival rates, the average system delay increases, while with more channels the average system delay is decreases. It has to be noted that the better delay performance of the slot-by-slot scheduling comes at the cost of more frequent scheduling.

B. Performance Comparison with Queue-Unaware Approach

We now compare the queue-aware scheduling with queue-unaware scheduling in terms of the total number of allocated slots to all SUs and the throughput each gets. Since our

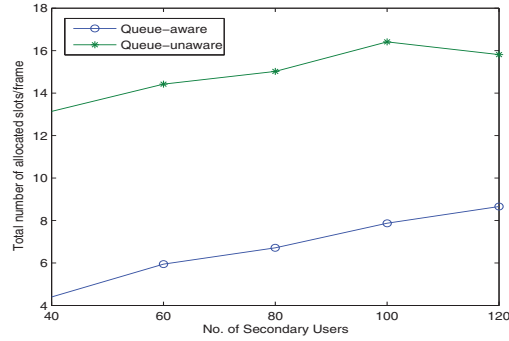


Fig. 11: Comparison of total number of slots allocated.

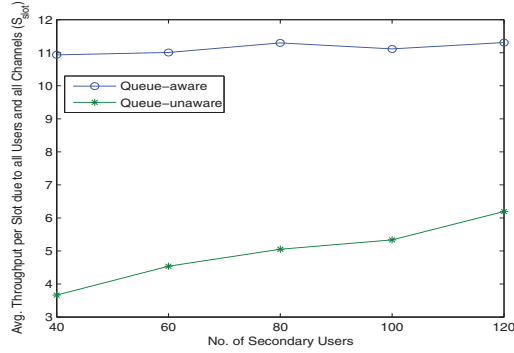


Fig. 12: Comparison of average system throughput per slot.

performance studies have clearly demonstrated that the overall performance with slot-by-slot allocation granularity is better than the frame-by-frame allocation, in comparative study of our proposed queue-aware allocation with queue-unaware allocation we only consider slot-by-slot allocation strategy.

When the queue status is not accounted for scheduling purposes, the scheduler might allocate slots to the SUs who do not have queued packets, thereby wasting resource. This is shown in Fig. 11, where the queue-unaware scheduler unnecessarily allocates more slots than the queue-aware scheduler. On the other hand, the queue-aware scheduler only assigns channels/slots to the users with packets in their queue. As a result, the system throughput is better, as shown in Fig. 12.

VI. CONCLUSIONS

In this paper, we proposed a queue aware frame-by-frame and slot-by-slot scheduling for DSA networks. We used the primary channel occupancy, channel quality, and the queue status to create the discrete-time Markov chain to estimate the maximum expected throughput for each channel on every slot. We maximized the spatial and temporal re-use of channels by allowing multiple conflict-free SUs to use the same channel on the same slots. We conduct simulation experiments to validate the proposed algorithm for various network setting and metrics. For all the metrics used, the performance of slot-by-slot scheduling was better than frame-by-frame scheduling. We also show the benefits of queue-aware scheduling in achieving better per-user throughput.

REFERENCES

- [1] General Survey of Radio Frequency Bands (30 MHz to 3 GHz): Vienna, Virginia, September 1-5, 2009. Available at: <http://www.sharespectrum.com/papers/spectrum-reports/>
- [2] N. H. Adam and M. Abdel-Hafez, "Spectrum sharing with QoS awareness in cognitive radio networks, *International Wireless Communications and Mobile Computing Conference (IWCMC)*, pp. 445-451, Sep. 2016.
- [3] I. F. Akyildiz, W. Y. Lee, M. C. Vuran, and S. Mohanty, "NeXt generation/ dynamic spectrum access/ cognitive radio wireless networks: A survey, *Elsevier Computer Networks*, vol. 50, pp. 2127-2159, Sep. 2006.
- [4] S. Borst, "User-level performance of channel-aware scheduling algorithms in wireless data networks, in *Proc. IEEE Infocom*, San Francisco, CA, USA, pp. 636-647, Mar. 2003.
- [5] S. Debroy, S. De, and M. Chatterjee, "Contention based multichannel MAC protocol for distributed cognitive radio networks," *IEEE Transactions on Mobile Computing*, vol. 13, no. 12, pp. 2749-2762, Dec. 2014.
- [6] A. Domenico, E. Strinati, and M. Benedetto, "A survey on MAC strategies for cognitive radio networks, *IEEE Communications Surveys and Tutorials*, vol. 14, no. 1, First Quarter 2012.
- [7] D. Gozupek and F. Alagoz, "Throughput and delay optimal scheduling in cognitive radio networks under interference temperature constraints, *Journal of Communications and Networks*, vol. 11, no. 2, pp. 148-156, Apr. 2009.
- [8] D. Gozupek, E. Eraslan, and F. Alagoz, "Throughput satisfaction-based scheduling for cognitive radio networks, *IEEE Transactions on Vehicular Technology*, vol. 61, no. 9, pp. 4079-4094, Sep. 2012.
- [9] O. Hashida, Y. Takahashi, and S. Shimogawa, "Switched bath Bernoulli process (SBBP) and the discrete-time SBBP/G/1 queue with application to statistical multiplexer performance," *IEEE Journal on Selected Areas in Communications*, vol. 9, no. 3, pp. 394-401, Apr. 1991.
- [10] E. F. Khairullah and M. Chatterjee, "Multi-bid auctions for channel allocation in multi-channel dynamic spectrum access networks, *European Conference on Networks and Communications (EuCNC)*, Athens, pp. 282-287, June 2016.
- [11] H. Kushner and P. Whiting, "Asymptotic properties of proportional-fair sharing algorithms, in *Proc. 40th Annual Allerton Conference on Communications, Control, and Computing*, Monticello, IL, USA, pp. 1-9, Oct. 2002.
- [12] O. Mabrouk, H. Idoudi, I. Amdouni, R. Soua, P. Minet, and L. Saidane, "OTICOR: Opportunistic time slot assignment in cognitive radio sensor networks, in *Proc. IEEE International Conference on Advanced Information Networking and Applications (AINA)*, Victoria, Canada, pp. 790-797, May 2014.
- [13] M. Rashid, M. Hossain, E. Hossain, and V. Bhargava, "Opportunistic spectrum scheduling for multiuser cognitive radio: A queueing analysis, *IEEE Transactions on Wireless Communications*, vol. 8, no. 10, pp. 5259-5269, Oct. 2009.
- [14] S. Sengupta, M. Chatterjee, S. Ganguly, R. Izmailov, "Exploiting MAC flexibility in WiMAX for media streaming," *IEEE WoWMoM 2005*, pp. 338-343.
- [15] K. K. Sharma and A. Trivedi, "An Opportunistic channel access scheme with channel ordering for cognitive radio network, in *Proc. 5th International Conference on Communication Systems and Network Technologies (CSNT)*, Gwalior, India, pp. 354-358, Apr. 2015.
- [16] M. Thoppian, S. Venkatesan, R. Prakash, and R. Chandrasekaran, "MAC-layer scheduling in cognitive radio based multi-hop wireless networks in *Proc. IEEE WoWMoM: International Symposium on a World of Wireless, Mobile, and Multimedia Networks*, Buffalo, NY, USA, pp. 1-10, Jun. 2006.
- [17] V. Tumuluru, P. Wang, and D. Niyato, "An Opportunistic Spectrum Scheduling Scheme for Multi-Channel Cognitive Radio Networks, in *Proc. Vehicular Technology Conference - Fall*, Ottawa, Canada, pp. 1-5, Sep. 2010.
- [18] H. Yarnagula, S. Deka, and N. Sarma, "Distributed TDMA based MAC protocol for data dissemination in ad-hoc Cognitive Radio networks, in *Proc. IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, Chennai, India, pp. 1-6, Dec. 2013.
- [19] Q. Zhao and B. M. Sadler, "A Survey of Dynamic Spectrum Access, *IEEE Signal Processing Magazine*, vol. 24, no. 3, pp. 79-89, May 2007.