On-Demand Multicast Routing in Cognitive Radio Mesh Networks

Hisham M. Almasaeid, Tasneem H. Jawadwala, and Ahmed E. Kamal
Dept. of Electrical and Computer Eng., Iowa State University, Ames, IA 50011, USA
E-mail:{hisham,tasneemj,kamal}@iastate.edu

***Abstract*—Cognitive radio networks (CRN) have emerged as a promising, yet challenging, solution to enhance spectrum utilization, thanks to the technology of cognitive radios. In this work, we consider the multicast routing and channel allocation problem in cognitive radio mesh networks. Due to the potential heterogeneity in channel availability among mesh routers (MRs) and the frequency switching latency, end-to-end delay and throughput degradation could be subject to a significant increase. We propose an on-demand multicast routing and channel allocation algorithm that takes channel heterogeneity and switching latency into consideration. The algorithm aims at reducing the end-to-end delay, and at the same time reducing the degradation of throughput using a dynamic programming approach.**

I. Introduction

The multicast routing problem in multihop cognitive radio networks faces some challenges that are not present in traditional wireless networks. These challenges are mainly caused by the heterogeneity in channel availability at different Secondary Users (SUs), especially at branching points on the multicast tree. This heterogeneity is due to the fact that SUs may observe different sets of available (idle) channels depending on the activity of the Primary Users (PUs) in their vicinity. The heterogeneity leads to:

a) Broadcast deformation: : when an SU has neighbors that do not (all) share a common channel with this SU, it cannot broadcast a data unit to all neighboring SUs in one transmission. Therefore, a broadcast might become a number of multicast transmissions, or in the worst case a number of unicast transmissions. This significantly reduces network capacity and increases end-to-end delay.

b) Switching delay: : another source of capacity wastage and delay increase is channel switching. Assume that SU i receives from SU j and forwards the data to SU k. If i cannot find a common channel with j and k together, then it has to use two different channels for transmission over the two links $j \rightarrow i$ and $i \rightarrow k$. Depending on spectral separation between the two channels, the switching delay could be significant.

In traditional multichannel wireless networks, the use of multiple radio interfaces was widely adopted as a solution to make full use of the capacity provided by the multiple channels and to avoid the switching delay problem [1]. However, it is usually the case in traditional multichannel wireless networks that the same set of channels is available to all nodes in the networks, which is not necessarily the case in cognitive radio networks. Therefore, new solutions are needed for cognitive radio networks. Certainly, utilizing multiple interfaces at SUs can reduce the effect of the channel heterogeneity problem, but new routing and channel allocation strategies are needed. This paper, studies the multicast routing and channel allocation problem in wireless cognitive radio mesh networks.

The rest of this paper is organized as follows. In Section II, we layout the system model and assumptions. The motivation of this work and the problem formulation are presented in Section III. An optimal channel allocation algorithm along a single path is proposed in Section IV. The multicast routing and channel allocation algorithm (OMRA) is then proposed in Section V. In Section VI, we evaluate the performance of the proposed OMRA algorithm. We review some related work in Section VII, and then conclude the paper in Section VIII.

II. System Model

Before we start the discussion about the multicast routing problem, we would like to layout the system model and assumptions. We consider a wireless cognitive mesh network that consists of a number of mesh routers (MRs), each of which manages a set of mesh clients (MCs), and a single gateway that connects the network to the Internet. Any MR can reach the gateway either directly or through multiple hops of MRs. We are only concerned with multicast traffic that originates from the Internet, passes through the gateway, then finally received by the members, SUs, of the multicast session. Thus, we treat the gateway as the source of all multicast traffic.

We assume the existence of a spectrum sensing and management entity (SSME) that provides an SU (MR or MC) with a list of channels which can be used by that specific SU. In addition to the straightforward implementation of having each SU perform the role of the SSME itself (or cooperatively with neighboring SUs), a number of other implementations have been proposed in literature. One of these implementations is to have a wireless sensor network infrastructure that is specifically designed to achieve accurate spectrum sensing and provide SUs with information about spectrum occupancy [2]. We further assume the existence of a common control channel (CCC) (an actual frequency channel [3], or a virtual CCC [4]).

III. Motivation and Problem Definition

To explain the motivation behind this work, consider the example shown in Figure 1. The set besides each SU in Figure 1.(a) represents the list of channels available to that SU. Two

This research was supported in part by the National Science Foundation under grants CNS-0626822 and CNS-0721453.

TABLE I
A DYNAMIC PROGRAM FOR OPTIMAL CHANNEL ALLOCATION ALONG $\mathcal{R}$.

$$f^*(|\mathcal{R}|) = \min_{m \in \mathcal{L}_{|\mathcal{R}|} \cap \mathcal{L}_{|\mathcal{R}|-1}} L_m^{|\mathcal{R}|,|\mathcal{R}|-1} + f^*(|\mathcal{R}|-1, m) \quad (1)$$

$$f^*(i,m) = \min_{n \in \mathcal{L}_i \cap \mathcal{L}_{i-1}} \left(C_{n,m}^{i-1,i} + f^*(i-1,n) \right) \quad (2)$$

$$f^*(1,m) = 0, \ \forall m \in \mathcal{L}_1 \quad (3)$$

different channel assignments are presented in Figures 1.(b) and 1.(c) to explain the effect of channel assignment on the throughput and end-to-end delay of multicast traffic. Let us investigate the total time that the forwarding node f needs to relay the multicast data to the multicast receiver-nodes r_1 and r_2, after receiving it from the multicast source s. Assume that the switching delay between two channels operating at central frequencies f_1 and f_2 is a linear function denoted by $d_{sw}(f_1, f_2)$. Let $d_{sw}(f_1, f_2) = \alpha |f_1 - f_2|$, where α is the tuning speed (in *seconds*/*Hz*) of the spectrum processor. Also, let $L_n^{i,j}$ be the packet transmission time from MR i to MR j on channel n. This will depend on the packet size and *achievable* channel bit-rate, which in turn depends on the transmission power, the channel bandwidth, the coding and modulation schemes, and the link quality between i and j. To keep the example simple, assume that a packet transmission on any channel takes the same time of L, i.e., $L_n^{i,j} = L \ \forall i, j, n$. Then, the total relay time at node f for case (b) is $L + \alpha$; α to switch from channel 3 to channel 4, and L for one transmission on channel 4 to r_1 and r_2. For case (c), on the other hand, the total relay time $= d_{sw}(1,4) + L + d_{sw}(4,5) + L = 2L + 4\alpha$.

Let τ_b and τ_c denote the throughput of node f in case b and c respectively. Then, $\tau_b^{-1} = (L) + L + \alpha + (\alpha) = 2L + 2\alpha$, and $\tau_c^{-1} = (L) + 2L + 4\alpha + (4\alpha) = 3L + 8\alpha$. The term (L) in both of the formulas above represent the time to receive the packet from s, and the terms (α) and (4α) represent the switching time from the last channel used for transmission back to the channel used for reception for cases (b) and (c) respectively. For 10 $Mbps$ channel rate, 1500 byte packet size, and $1ms$ baseband switching delay, the ratio τ_c/τ_b evaluates to $\approx 38\%$, which is a significant reduction of f's throughput.

On the other hand, the end-to-end delay is also affected. For example, in Figure 1, it will take the multicast data $2L + \alpha$ to reach r_1 and r_2 in case (b). However, the end-to-end delay is $2L + 3\alpha$ to r_1 and $3L + 4\alpha$ to r_2 in case (c). Based on the practical values used before, the ratio of the end-to-end in case (c) to case (b) is 158.82% for r_1 and 223.53% for r_2.

The numbers above highlight the need for an on-demand routing approach. The purpose of this study is to propose such an approach for data multicast between MRs only, as the problem of transmitting the multicast data from an MR to its MCs was studied by the authors in [5].

IV. OPTIMAL CHANNEL ASSIGNMENT ON A ROUTE

In this section, we address the problem of optimal channel assignment along a single path. We then extend the solution to include the multicast routing problem.

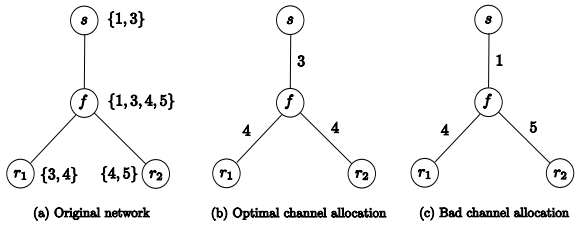

Fig. 1. An example that illustrates the effect of channel assignment on the throughput and end-to-end delay of multicast traffic.

A. Dynamic programming approach for channel assignment

Given a route (or path) $\mathcal{R}$ that consists of $|\mathcal{R}|$ MRs, numbered from 1 to $|\mathcal{R}|$, and the data flows from MR $|\mathcal{R}|$ to MR 1, i.e, MR $|\mathcal{R}|$ is the source and MR 1 is the destination. The objective is to allocate a channel to each link along $\mathcal{R}$ such that the end-to-end delay is minimized. A formal definition of a dynamic program consisting of *stages*, *states*, *transitions*, and *transition cost* is now developed. MRs along a path are mapped into stages, available channels to MRs into states, and channel assignments into transitions. Let,

- $\mathcal{L}_i$ be the set of channels available at MR i.
- $C_{n,m}^{i-1,i}$ be the cost of making a transition from state n at stage $i-1$ to state m at stage i. For a single path, this cost is given as $C_{n,m}^{i-1,i} = d_{sw}(n,m) + L_n^{i,i-1}$. The cost will be slightly different for multicast routing (see Section V).
- $f^*(i,m)$ be the delay under the optimal solution starting from state m at stage i and ending at stage 1.
- $f^*(|\mathcal{R}|)$ be the minimum end-to-end delay under the optimal solution (channel allocation) for path $\mathcal{R}$.

Choosing state m at stage i ($i < |\mathcal{R}|$) means that MR i receives data from MR $i+1$ on channel m. Finally, the dynamic program is outlined in Table I. Equation (1) finds the optimal channel assignment, i.e., the one with the minimum end-to-end delay, along the entire route. Equation (2), on the other hand, finds the optimal channel assignment starting from MR i and ending at MR 1, such that MR i receives from MR $i+1$ on channel m. A boundary condition that gives all channel assignments equal cost of "0" at MR 1 is defined in (3).

B. Distributed optimal allocation

We devise a distributed implementation based on the above dynamic program formulation, that can be used in cognitive radio networks. To setup a connection from the source MR $|\mathcal{R}|$ to a destination MR 1 along the path $\mathcal{R}$, two phases are executed; the *forward phase* and the *backward phase*. In the forward phase, which starts from the destination and ends at the source, the dynamic program proposed earlier is used to calculate the optimal cost (end-to-end delay) along the route. In the backward phase, which starts from the source and ends at the destination, the channels to be used at each link to obtain the calculated end-to-end delay are identified at each MR along the route. The assumption that the destination starts the channel allocation process is suitable for multicast routing. In the case of multicast routing, the source MR, i.e., the gateway, sends a JOIN_REQ packet to the destination MR (the MR that should join a multicast session) on any available route. Then, the destination MR starts a search process to find the optimal

path to the gateway. That is why we assume the forward phase is initiated by the destination MR (for more details see Section V). The forward phase operates as follows,

- The destination calculates the value $f(1,m)\ \forall m \in \mathcal{L}_1$ (see Table I) and sends it in one packet to the next hop along the path, i.e., MR 2, through the CCC. We call this packet the *allocation cost packet (ACP)*. Let $\mathbf{ACP^{(i)}}$ denote the ACP packet sent by MR i to MR $i+1$, and $\mathbf{ACP^{(i)}}{\cdot}cost(m)$ is a field of $\mathbf{ACP}^{(i)}$ used to carry the cost of using channel m for the link $(i \rightarrow i+1)$. The source MR, i.e., MR 1, fills the fields of its ACP packet as follows:

$$\mathbf{ACP^{(1)}}{\cdot}cost(m) = 0, \qquad \forall m \in \mathcal{L}_1 \quad (4)$$

- Then, upon receiving $\mathbf{ACP^{(i-1)}}$ $(i < |\mathcal{R}|)$, MR i prepares its ACP packet as follows:

$$\mathbf{ACP^{(i)}}{\cdot}cost(m) = \min_{n \in \mathcal{L}_i \cap \mathcal{L}_{i-1}} C_{n,m}^{i-1,i} + \mathbf{ACP^{(i-1)}}{\cdot}cost(n) \quad (5)$$

- Finally, when the source MR, i.e, MR $|\mathcal{R}|$, receives the ACP packet of MR $|\mathcal{R}|-1$, it calculates $f^*(|\mathcal{R}|)$ as follows:

$$f^*(|\mathcal{R}|) = \min_{n \in \mathcal{L}_{|\mathcal{R}|} \cap \mathcal{L}_{|\mathcal{R}|-1}} L_n^{|\mathcal{R}|,|\mathcal{R}|-1} + \mathbf{ACP^{(|\mathcal{R}|-1)}}{\cdot}cost(n) \quad (6)$$

The backward phase, on the other hand, operates as follows,

- The source MR identifies the optimal channel on the link between $|\mathcal{R}|$ and $|\mathcal{R}|-1$. Let this channel be $k^*_{|\mathcal{R}|,|\mathcal{R}-1|}$, then

$$k^*_{|\mathcal{R}|,|\mathcal{R}|-1} = \operatorname*{argmin}_{n \in \mathcal{L}_{|\mathcal{R}|} \cap \mathcal{L}_{|\mathcal{R}|-1}} L_n^{|\mathcal{R}|,|\mathcal{R}|-1} + \mathbf{ACP^{(|\mathcal{R}|-1)}}{\cdot}cost(n) \quad (7)$$

  The source node $|\mathcal{R}|$ informs the previous node on the route, i.e, $|\mathcal{R}-1|$, about the identified channel.
- Then, at each MR i on the route, for $1 < i < |\mathcal{R}|$, the channel $k^*_{i,i-1}$ is identified as follows,

$$k^*_{i-1,i} = \operatorname*{argmin}_{n \in \mathcal{L}_i \cap \mathcal{L}_{i-1}} \left(C_{n,k^*_{i,i+1}}^{i-1,i} + \mathbf{ACP^{(i-1)}}{\cdot}cost(n) \right) \quad (8)$$

- Finally, the destination, i.e., MR 1, receives, from MR 2, information about the channel assigned to the link $(2 \rightarrow 1)$.

V. Multicast Routing: Challenges and Solutions

In this section, we use the dynamic program developed in Section IV for the case of a single path as a building block to design an on-demand multicast routing and channel allocation algorithm. A common multicast routing design in wireless mesh networks is the tree-based structure, in which a multicast tree originates from the source of the multicast session and reaches every member of that session. We adopt this structure in this work. We introduce a decentralized dynamic tree construction algorithm by which an MR may attach itself to an existing multicast tree (or be the first in a new one) while jointly minimizing the end-to-end delay and throughput wastage at MRs along the selected route.

To understand the complexity of the problem, consider an MR, say i, that wants to join a multicast session. For i to find the route with the minimum end-to-end delay that connects it to the existing multicast tree, it must inspect all possible routes. This inspection is more complicated in cognitive radio networks than it is in traditional wireless networks because of the fact that *"longer paths, in terms of number of hops, do not necessarily impose longer delays because of the channel switching delay"*, i.e., the search domain may include the entire network. Thus, we need a systematic way to constrict the search domain and achieve a near optimal solution.

A. Finding the minimum hop distance (level) of MRs

To find the shortest hop-count distance,i.e., *level*, from the gateway to every other MR in the network, the gateway starts a search process on the CCC. Each MR i maintains a local level counter l_i, which is used to maintain the shortest hop distance to the gateway. This counter is initialized to ∞, except for the gateway for which the level is initialized to 0. For an MR j in level l_j, define the following:

- *Neighbors of* j: the set of MRs that are within the transmission range of MR j, even if they do not share a common data channel with j. This set is denoted by $\mathcal{N}(j)$.
- *Parents of* j: the set of MRs in level l_j-1 that MR j can reach, which is denoted by $\mathcal{P}(j)$.
- *Children of* j: the set of MRs in level l_j+1 which can reach MR j , and is denoted by $\mathcal{C}(j)$.

It is worth pointing out that reachability between a pair of MRs requires them to be within the communication range of each other and to share a common data channel. The search process starts by having the gateway broadcasts a control packet, that we call LEVEL_UPDATE packet, with a *"level"* field initialized to 0, and a *"sender"* field initialized to 0 (the ID of the gateway MR) over the CCC.

Every MR, other than the gateway, runs the procedure shown in Algorithm 1. The algorithm guarantees that the level counters remain up-to-date for all MRs. The first *if*-statement (line 3) updates the level counter l_i (using the equation in line 6) upon receiving a LEVEL_UPDATE message from some MR $j \in \mathcal{N}(i)$. If the value of l_i is different from the current one, using the aforementioned equation, then all MRs in $\mathcal{N}(i)$ are notified about the new value (using $SendLevelUpdate(\cdot)$ as in line 7), and l_i is updated (line 8). If the channel availability at MR i, represented by the set of available channels $\mathcal{L}_i$, changes, then the second *if*-statement (line 9) will be executed. In that statement, MR i will first provide its neighbors, i.e., $\mathcal{N}(i)$, with the updated $\mathcal{L}_i$ (line 10). Then, it updates its level counter l_i and informs its neighbors about any change in l_i as shown in lines 12 and 13. Finally, the third *if*-statement (line 14) updates level l_i upon receiving an updated channel availability from some MR $j \in \mathcal{N}(i)$, and informs nodes in $\mathcal{N}(i)$ if l_i changes.

B. Multicast Routing Algorithm

The gateway sends a JOIN_REQ packet over the CCC to an MR(s) that should join a multicast group. This packet should contain enough information to identify the multicast session. This packet is sent over the CCC. Upon receiving this packet, the MR needs to find the path that connects it to the multicast tree of the intended session with the minimum end-to-end delay. The search for this shortest path involves allocating channels to the links of that path that were not allocated channels before, while keeping the bandwidth wastage due to channel switching as low as possible.

Before proposing the algorithm, we need to define a cost metric that can jointly represent the delay (transmission and switching) and bandwidth wastage. For the single path case in Section IV, the cost metric $C_{n,m}^{i-1,i}$ included the switching delay and transmission time only, because we did not address existing flows that pass through an MR. Let $\overline{\mathcal{L}}(i) \subseteq \mathcal{L}_i$ be the set of channels used to handle (i.e., receive or transmit) the flows (unicast or multicast) that are served by MR i. Then, we define the function $\Delta_i(\cdot)$ that takes as an argument a set of channels $\mathbf{S}$, and returns the maximum possible switching delay between any pair of channels in $\mathbf{S} \cup \overline{\mathcal{L}}_i$. Therefore,

$$\Delta_i(\mathbf{S}) = d_{sw}(\max_{m \in \mathbf{S} \cup \overline{\mathcal{L}}_i} m, \min_{n \in \mathbf{S} \cup \overline{\mathcal{L}}_i} n) \quad (9)$$

Equation (9) is based on the assumption that the cognitive radio remains tuned to the last used channel. If a different radio management policy is used (like having the cognitive radio always tuned to a particular channel during idle times), a different definition of the $\Delta_i(\cdot)$ function is needed. By including the $\Delta_i(\cdot)$ function into the cost metric $C_{n,m}^{j,i}$, channels closer to the ones already allocated to some incoming or outgoing links of MR i will be preferable over others that are farther away in the spectrum. Therefore, the nodal delay due to switching between flows served by the MR on different channels will be less, which intuitively means less bandwidth wastage.

The proposed on-demand multicast routing and channel allocation (OMRA) algorithm is outlined in Algorithm 2, and is based on the forward and backward phases of the distributed implementation of the dynamic program proposed in Section IV-B. When the MR that is supposed to join a multicast session, say MR i, receives a JOIN_REQ packet from the gateway, it prepares for each MR $j \in \mathcal{P}(i)$ an ACP packet as described in Section IV-B (equation (4)) and sends it to j over the CCC. Then, every other MR that receives an ACP packet, from one of its children, does the same. In other words, it prepares an ACP packet for each of its parents as described in Section IV-B (equation (5)) and sends it over the CCC. Therefore, the gateway will receive multiple ACP packets that originated at MR i, each of which corresponds to a distinct path between i and the gateway. The gateway then chooses the path with the minimum cost and initiates the backward phase.

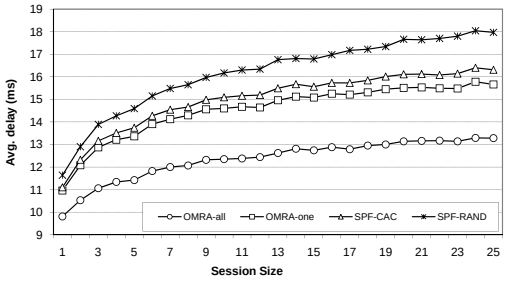

Fig. 2. Average end-to-end delay for a single session with 4 MHz spacing.

VI. Performance Evaluation

To evaluate the performance of the proposed routing algorithm, we study a cognitive radio mesh network of N MRs organized in a grid topology, and deployed in a square area of $A \times A$ meters. One of these N MRs is a gateway to the Internet, and is placed at the upper-left corner of the network field. Each MR can physically reach only the MRs in its left-hand, right-hand, upper, and lower cells in the grid (i.e.,

Algorithm 1: Level Evaluation and Reconfiguration

1 For an MR i (other than the gateway):
2 **begin**
3 **if** *a LEVEL_UPDATE message is received* **then**
4 j = LEVEL_UPDATE.*sender*;
5 l_j = LEVEL_UPDATE.*level*;
6 $l_i^{new} = \begin{cases} l_j + 1 & \text{if } \mathcal{L}_i \cap \mathcal{L}_j \neq \emptyset \text{ and } l_i > l_j + 1 \\ l_i & \text{otherwise} \end{cases}$
7 `SendLevelUpdate(`i,l_i,l_i^{new}`)`;
8 $l_i = l_i^{new}$;
9 **if** $\mathcal{L}_i$ *changes* **then**
10 Provide all MRs in $\mathcal{N}(i)$ with the new set $\mathcal{L}_i$;
11 $l_i^{new} = \min_{j \in \mathcal{N}(i) | \mathcal{L}_i \cap \mathcal{L}_j \neq \emptyset} \begin{cases} \infty & \text{if } \mathcal{P}(j) = \{i\} \\ l_j + 1 & \text{otherwise} \end{cases}$
12 `SendLevelUpdate(`i,l_i,l_i^{new}`)` ;
13 $l_i = l_i^{new}$;
14 **if** *an updated* $\mathcal{L}_j$ *for some* $j \in \mathcal{N}(i)$ *is received* **then**
15 Find l_i^{new} as described in line 12;
16 `SendLevelUpdate(`i,l_i,l_i^{new}`)`;
17 $l_i = l_i^{new}$;
18 **end**
19 `SendLevelUpdate(`i,l_i^{old},l_i^{new}`)`
20 **if** $l_i^{new} \neq l_i^{old}$ **then**
21 Create a new LEVEL_UPDATE message;
22 Set LEVEL_UPDATE.*sender*=i;
23 Set LEVEL_UPDATE.*level*=l_i^{new};
24 Broadcast LEVEL_UPDATE over the CCC;
25 **end**

Algorithm 2: The OMRA algorithm

Input : Multicast group ID g, an MR i
Output : A multicast route with channel allocation.
1 **if** i *is the MR that has received the JOIN_REQ* **then**
2 **foreach** $j \in \mathcal{P}(i)$ **do**
3 Create a new ACP packet, $\mathbf{ACP^{(i)}}$;
4 $\mathbf{ACP^{(i)}} \cdot cost(m) = \Delta(\{m\}),\ \forall m \in \mathcal{L}_i$;
5 Send $\mathbf{ACP^{(i)}}$ to MR j;
6 **else if** i *is the gateway* **then**
7 Find the optimal path from all received ACP's using (6);
8 Start the backward phase (see Section IV-B);
9 **else if** $i \in \mathcal{T}(g)$ **then** `/* On the multicast tree` $\mathcal{T}(g)$ `*/`
10 **upon** *receiving* $\mathbf{ACP^{(j)}}$ *from some* $j \in \mathcal{C}(i)$ **do**
11 Create a new ACP packet, $\mathbf{ACP^{(i)}}$;
12 Let k be the parent of i on the multicast tree;
13 Let ν be the channel allocated to link (i,k);
14 $\mathbf{ACP^{(i)}} \cdot cost(m) = \infty,\ \forall m \in \mathcal{L}_i / \{\nu\}$;
15 $\mathbf{ACP^{(i)}} \cdot cost(\nu) = \min_{n \in \mathcal{L}_i \cap \mathcal{L}_j} \left(C_{n,\nu}^{j,i} + \mathbf{ACP^{(j)}} \cdot cost(n) \right)$
16 **else** `/* Not on the multicast tree` $\mathcal{T}(g)$ `*/`
17 **upon** *receiving* $\mathbf{ACP^{(j)}}$ *from some* $j \in \mathcal{C}(i)$ **do**
18 Create a new ACP packet, $\mathbf{ACP^{(i)}}$;
19 **if** *link* (i,k) *is allocated a channel* ν **then**
20 $C_{n,m}^{j,i} = \infty, \forall m \in \mathcal{L}_i / \{\nu\}$;
21 $C_{n,\nu}^{j,i} = d_{sw}(n,\nu) + L_{\nu}^{i,j} + \Delta_i(\{n,\nu\})$;
22 **else**
23 $C_{n,m}^{j,i} = d_{sw}(n,m) + L_m^{i,j} + \Delta_i(\{n,m\}), \forall m \in \mathcal{L}_i$;
24 $\mathbf{ACP^{(i)}} \cdot cost(m) = \min_{n \in \mathcal{L}_i \cap \mathcal{L}_j} \left(C_{n,m}^{j,i} + \mathbf{ACP^{(j)}} \cdot cost(n) \right)$;
25 Send $\mathbf{ACP^{(i)}}$ to all MRs $k \in \mathcal{P}(i)$;
26

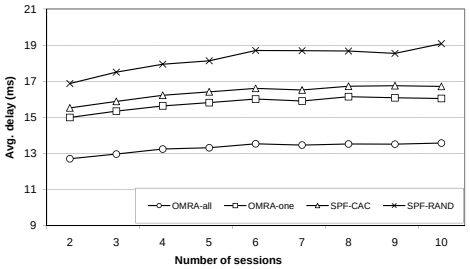

Fig. 3. Average end-to-end delay for multiple sessions with 4 MHz spacing.

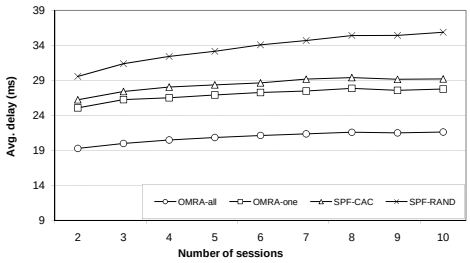

Fig. 4. Average end-to-end delay for multiple sessions with 10MHz spacing.

the communication range is $\frac{A}{\sqrt{N}}$). We have a total of K=10 licensed channels, each has an achievable rate of 10 Mbps. Channels are evenly spaced by either 4 or 10 MHz. For all experiments, N=49, A=$500m$, and α=$1ms/10MHz$ [6].

We compare the end-to-end delay between four different routing schemes: 1) OMRA routing where an MR investigates the possible paths through all of its parents (*OMRA-all*). 2) OMRA routing where an MR randomly picks one of its parents as the next hop on the path, and uses the dynamic program to allocate channels along that path (*OMRA-one*). 3) Shortest path routing where an MR randomly picks one of its parents as the next hop on the path. The closest available channel (CAC) to that allocated to the link between the MR and its child, is allocated to the link between the MR and its parent. The member MR randomly selects the first channel from the its set of available channels. We denote this approach as *SPF-CAC*. 4) The last scheme, denoted *SPF-RAND*, is similar to SPF-CAC except that the allocated channel is chosen randomly.

We obtain average results for the end-to-end delay for the cases of single multicast session and multiple multicast sessions. Each point on the curves in Figures 2-4 corresponds to the average over 1500 randomly generated instances. In each instance, we vary the channel availability at each MR. An MR has channel k available with probability p and unavailable with probability $1-p$. p is set to 0.393 for all experiments.

We first study the case of routing a single multicast session. Members of a session join the multicast group sequentially. The size of the multicast session is varied from 1 to 25. Figure 2 shows the average delay over all the members of a session for 4 MHz spacing. As the figure implies, the proposed OMRA algorithm outperforms SPF-CAC and SPF-RAND. However, the gain is higher when all parents are explored (OMRA-all) instead of a single parent (OMRA-one).

To evaluate the performance of the OMRA algorithm under the existence of multiple multicast sessions, we vary the number of sessions from 2 to 10, where each session has a size that is drawn uniformly at random from the range $[2, 15]$. Figures 3 and 4 show the average end-to-end delay (over all sessions) for 4 MHz and 10 MHz spacings respectively. The results in these two figures confirm the superiority of the OMRA algorithm, in both of its variants, over the other approaches. Furthermore, the gain of using the OMRA algorithm (relative to the SPF-RAND approach) increases with the increase in the number of sessions in the network. For instance, the gain of OMRA-all increases by $\approx 4\%$ at 4 MHz spacing, and by $\approx 5\%$ at 10 MHz spacing by increasing the number of sessions from 2 to 10.

VII. Related Work

The current work in cognitive radio networks (CRNs) is mostly focused on efficient spectrum utilization, and few studies consider the effect of heterogeneity in channel availability on various network operations like multicast routing. In [7], a cross-layer optimization approach for video multicast on heterogeneous channels was proposed. The study aims at optimizing the overall received video quality, while achieving fairness among multicast users and avoiding interference with licensed users. The work in [8] proposes an online multicast scheduling scheme, including power control and channel allocation policies, based on stochastic Lyapunov optimization. In [9], the problem of constructing minimum energy multicast tree in CRNs is studied. The study considers the energy consumption at SUs due to spectrum sensing and data transmission. A routing and channel allocation algorithm based on a layered graph model was proposed in [10]. We consider the effect of channel heterogeneity and switching latency, which is an absent aspect in the previous studies.

VIII. Conclusions

In this paper, we have studied the multicast routing and channel assignment problem in cognitive radio mesh network, with the objective of minimizing the end-to-end delay. A distributed on-demand routing and channel allocation algorithm was proposed. Numerical results confirmed the superiority of the proposed algorithm over other baseline algorithms.

References

[1] P. Kyasanur and N. H. Vaidya. Routing in multi-channel multi-interface ad hoc wireless networks. Technical report, University of Illinois at Urbana-Champaign, 2004.

[2] B. Mercier, V. Fodor, et al. Sensor networks for cognitive radio: Theory and system design. *ICT Mobile Summit*, June 2008.

[3] T. Chen et al. Topology management in cogmesh: A cluster-based cognitive radio mesh network. *IEEE Int. Conf. on Comm. (ICC)*, 2007.

[4] Y. Kondareddy and P. Agrawal. Synchronized mac protocol for multi-hop cognitive radio networks. *IEEE Int. Conf. on Comm. (ICC)*, 2008.

[5] H. Almasaeid and A. Kamal. Assisted-multicast scheduling in wireless cognitive mesh networks. In *IEEE Int. Conf. on Comm. (ICC)*, 2010.

[6] TCI International. Tci 715 spectrum monitoring system data specification. *http://www.tcibr.com/ufiles/File/715Webp.pdf*, 2010.

[7] Donglin Hu, Shiwen Mao, and Jeffrey H Reed. On video multicast in cognitive radio networks. In *IEEE INFOCOM*, 2009.

[8] Jin Jin et al. Multicast scheduling with cooperation and network coding in cognitive radio networks. In *IEEE INFOCOM*, 2010.

[9] W. Ren et al. Minimum-energy multicast tree in cognitive radio networks. *IEEE Asilomar Conference*, 2009.

[10] C. Xin et al. A novel layered graph model for topology formation and routing in dynamic spectrum access networks. In *IEEE Int. Symp. on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, 2005.