Joint Routing and Resource Allocation in Cognitive Radio-Based Mesh Networks

Amr A. El-Sherif and Amr Mohamed
Computer Science and Engineering Department
Qatar University, P.O. Box 2713, Doha, Qatar.
{amr.elsherif2, amrm}@qu.edu.qa

Abstract—We consider wireless mesh networks in which the nodes are utilizing cognitive radios and try to opportunistically gain access to spectrum resources. In such networks, the timely delivery of the traffic is a challenging task due to the licensed (primary) users’ activities and their traffic characteristics. To overcome this challenge we propose an algorithm that minimizes the end-to-end delay through joint routing and spectrum resources allocation. The network is analyzed from a queueing theory perspective to capture the effects of dynamic spectrum availability on mesh network’s traffic. The joint routing and resource allocation problem is formulated as a non-linear integer programming problem, for which we propose a decentralized solution based on the Lagrangian dual problem. Results demonstrate the performance of our proposed algorithm, as well as the efficiency of the decentralized implementation.

I. INTRODUCTION

Cognitive radio is a promising technology aiming at better spectrum utilization by prescribing the coexistence of licensed (or primary) and unlicensed (secondary or cognitive) radio nodes on the same bandwidth [1]. One of the key challenges in cognitive radio networks is the design of dynamic spectrum allocation algorithms that enable opportunistic access to the wireless spectrum. In [2] and [3] the cognitive radio problem was investigated from an information theoretic standpoint, where the cognitive transmitter is assumed to transmit at the same time and on the same bandwidth of the primary link using complex precoding techniques. The concept of a time-spectrum block was introduced in [4] and protocols to allocate such blocks were proposed. In [5] the authors derived optimal and suboptimal distributed strategies for channel sensing and access under a Partially Observable Markov Decision Process (POMDP) framework.

The cognitive radio concept is desirable for a wireless mesh network (WMN) in which a large volume of traffic is expected to be delivered since it is able to utilize spectrum resources more efficiently. However, the dynamic nature of the radio spectrum calls for the development of novel spectrum-aware routing algorithms. In fact, spectrum occupancy is location-dependent, and therefore in a multi-hop path available spectrum bands may be different at each node. Hence, controlling the interaction between the routing and the spectrum management functionalities is of fundamental importance. In [6], each source node finds candidate paths based on Dynamic Source Routing (DSR). For each candidate route, the algorithm finds all feasible spectrum assignment combinations and selects the route and spectrum assignment with maximal throughput. A routing and spectrum selection algorithm for cognitive radio networks was proposed in [7]. The algorithm chooses the path that has the highest probability to satisfy the demands of secondary users in terms of capacity. However, it does not cover the issue of scheduling. For a network with cognitive radios, [8] formulates a cross-layer optimization problem to minimize the required network-wide radio spectrum resources needed to support traffic for a given set of user sessions. The problem is formulated as a mixed integer non-linear problem (MINLP), and a sequential fixing (SF) algorithm is developed which provides a near-optimal solution to the original problem.

Many popular multimedia applications, e.g., voice over IP, IPTV, and online gaming, have strict delay requirement. Therefore, unlike previous work, in this paper, we aim at designing a cross-layer joint routing and resource allocation protocol to minimize the end-to-end delay while opportunistically access spectrum which is not utilized by the primary users. To characterize the end-to-end delay for each data stream in the network, we resort to queueing theory analysis. Each cognitive mesh node is modeled as an infinite length queue. A queuing theoretic analysis of the network characterizes the arrival and service rates for each queue given the routes that pass through a node and the resources allocated to that node. The joint routing and resource allocation design problem is then formulated as an optimization problem having as objective the minimization of the end-to-end delay, and having integer valued decision variables. As formulated, the optimization problem is a non-linear integer programming (NIP) problem, which has combinatorial complexity. It is shown that by relaxing the integer constraints imposed on the decision variables, one can efficiently find a solution to the relaxed optimization problem. Finally, using the Lagrangian dual function, an efficient distributed solution to the optimization problem is presented.

II. SYSTEM MODEL

A. Network Model

The cognitive mesh network depicted in Fig. 1 consists of \( M \) nodes sharing the spectrum with \( N \) primary transmitter-receiver pairs operating over \( N \) non-overlapping channels. We assume that the primary network follows a time-slotted transmission structure. Therefore, primary transmissions can only start at the beginning of a time slot. This assumption will
simplify the analysis of our cognitive network, however, our model and analysis could be extended to incorporate different primary transmission schemes. The cognitive mesh network employs hybrid TDMA/FDMA scheme for channel access. Therefore, time is divided into time slots of fixed duration, which are further grouped into frames of $T$ time slots each. In each time slot, a node selects one of the available non-overlapping frequency channels to transmit over. Since the primary network transmissions are slotted, then it is customary that the cognitive network adjusts the boundaries of its time slots to match those of the primary network [9], [10].

A time slot and a channel pair $(t, c)$, is considered as the minimum unit for resource allocation, we will call it resource element. A cognitive mesh node senses its assigned channel $c$ at the beginning of each time slot $t$. If the channel is detected as idle, the node transmits a packet to the next node along the route to the destination, otherwise it remains silent and keeps sensing the channel in subsequent time slots. For simplicity, we will assume that cognitive nodes have access to perfect spectrum sensing information. As it will be shown later, the case of imperfect sensing can be easily incorporated into the problem formulation. The receiving node acknowledges the successful reception of a packet by transmitting an ACK packet back to the transmitter.

The cognitive mesh network is modeled as a directed graph $G(V, E)$, where each vertex $v \in V$ corresponds to a cognitive mesh node. During any time slot, there will be a set $C_v$ of channels available to node $v$. An edge $e \in E$ exists between nodes $u$ and $v$ if there exists a channel $c \in C_u \cap C_v$ and the nodes are within transmission range of each other, i.e., $\|u - v\| \leq R$, where $\|u - v\|$ is the Euclidean distance between nodes $u$ and $v$, and $R$ is the transmission range. It is assumed that all cognitive nodes use the same fixed transmission power, therefore, all nodes have the same transmission range $R$. Due to the primary nodes’ activity, channel availability will vary with time. Therefore, network connectivity is time varying, which poses a challenge to the routing and resource allocation protocol design.

The effect of the wireless interference between different nodes is modeled based on the protocol model [11], i.e., simultaneous packet transmissions from interfering nodes results in the loss of all involved packets. We say that two links $e_1$ and $e_2$ interfere with each other if there is a shared node between $e_1$ and $e_2$ (because of half duplexing, unicast communications, or collisions) or any node from $e_1$ is within interference range $I > R$ from any node from $e_2$, and they are using the same channel. Because of the ACK packet sent back to the transmitter by the receiver, both transmitter and receiver need to be free of interference.

B. Channel Model

The wireless channel between a node and its destination is modeled as a Rayleigh flat fading channel with additive white Gaussian noise [12]. Success and failure of packet reception is characterized by outage events and outage probabilities. Details of the channel model and outage probability calculation can be found in [12] and [10].

C. Queuing Model

Each node in the cognitive mesh network has an infinite buffer for storing fixed length packets. The packet transmission time equals to one time slot duration. Multiple data connections or streams are present in the network. For data stream $f$ having node $u$ as source, packet arrivals at the source are modeled as a stationary Bernoulli process with i.i.d arrivals from slot to slot and mean $\lambda'_f$ [13]. In other words, the probability that a new packet arrives at any given time slot $t$ is $\lambda'_f$. Moreover, the packet arrival processes are assumed to be independent from one data stream to another.

The state of any of the $N$ primary channels is modeled using a two state Markov chain (idle and busy). Using the stationary distribution of the Markov chain, at any given time slot, channel $c$ will be idle (Markov chain in the idle state) with probability $\alpha_c$.

III. JOINT ROUTING AND RESOURCE ALLOCATION STRATEGY

Because of the primary nodes activity, the spectrum resources available to the cognitive mesh nodes are varying in both time and space. This creates a strong interdependence between the routing and resource allocation problems. To guarantee acceptable network performance, we propose to deal with the routing and resource allocation strategies in a joint fashion rather than separating the two problems.

A. Queueing Analysis

Before presenting our joint design strategy we need to analyze the effect of the routing and resource allocation decisions on the network’s performance. Queueing theory is used to model the different aspects of the cognitive mesh network and to form a base for our routing and resource allocation protocol design.

To calculate the average arrival and service rates at the different nodes in the cognitive mesh network, we introduce the decision variables that define the different routes in the network as well as the resources allocated to the different network links. Two sets of decision variables are defined:
• $x_{f,e}$: $x_{f,e} = 1$ if link (edge) $e$ is selected for routing packets for data stream $f$, otherwise $x_{f,e} = 0$.

• $y^{I,e}_{f,e}$: $y^{I,e}_{f,e} = 1$ if the resource element $(t,c)$ is allocated to data stream $f$ over link (edge) $e$; otherwise $y^{I,e}_{f,e} = 0$.

It is worth mentioning here that the same resource allocation pattern is repeated every TDMA frame, and that the system is assumed to be stationary.

To define the average arrival rate for any node $v$ along the route of data stream $f$, we identify the events necessary for a packet arrival to take place. A packet from data stream $f$ enters the queue of node $v$ in a given TDMA frame if:

1) in the given frame there is a resource element $(t,c)$ allocated to one of $v$’s incoming edges,

2) the primary node owning channel $c$ is either idle during that time slot or the cognitive node $v$ is out of the primary node’s interference range,

3) the preceding cognitive node in the route has at least one packet in its queue to transmit to node $v$.

Therefore, the average arrival rate at node $v$ along the route of the data stream $f$ is the joint probability of these three events. Since these events are independent, then the average arrival rate can be written as

$$\lambda^f_v = \frac{1}{T} \sum_{e \in E^{in}^v} \sum_{t=1}^{T} \sum_{c=1}^{N} y^{I,e}_{f,e} \frac{\lambda^f_{e}(s)}{\mu^f_{e}(s)} \left[ I^e_c \alpha_c + (1 - I^e_c) \right] (1 - P^{out}_{e}),$$

(1)

where $E^{in}^v$ is the set of incoming edges to node $v$, $e(s)$ is the source node for edge (link) $e$, $\lambda^f_{e}(s)$ is its arrival rate and $\mu^f_{e}(s)$ its service rate. By modeling each queue as discrete time Markov chain, it can be shown that the fraction $\lambda^f_{e}/\mu^f_{e}$ is the probability that the queue has at least one packet [14], and hence will transmit a packet to the following node on the route whenever it has a chance. $\alpha_c$ is the probability that channel $c$ is idle during time slot $t$, $P^{out}_{e}$ is the outage probability between the transmitter and receiver of link $e$. Finally, $I^e_c = 1$ if the primary node owning channel $c$ interferes with transmissions over link $e$, otherwise $I^e_c = 0$.

Because of the summation with respect to time slots $t$ over one complete TDMA frame of $T$ time slots, the term $1/T$ is added such that the resulting $\lambda^f_v$ represents the average arrival rate per time slot as defined in the previous section.

Similarly, we identify the events necessary for a packet belonging to the $f$’s data stream to depart from node $v$’s queue. This will take place if in a given time frame

1) there is a resource element $(t,c)$ assigned to one of $v$’s outgoing edges,

2) the primary node owning channel $c$ is either idle during that time slot or cognitive node $v$ is out of its interference range.

Therefore, the average service rate of node $v$ along the route of data stream $f$ is defined as the joint probability of these two events, which is given by

$$\mu^f_v = \frac{1}{T} \sum_{e \in E^{out}^v} \sum_{t=1}^{T} \sum_{c=1}^{N} y^{I,e}_{f,e} \left[ I^e_c \alpha_c + (1 - I^e_c) \right] (1 - P^{out}_{e}),$$

(2)

where $E^{out}^v$ is the set of outgoing edges to node $v$. And the term $1/T$ is necessary for $\mu^f_v$ to represent the average service rate per time slot.

B. End-to-End Delay

As discussed above, the arrival and services processes to and from any node are Bernoulli processes with average rates $\lambda^f_v$ and $\mu^f_v$, respectively. A queuing system with such arrival and departure processes is Geo/Geo/1 queue [15]. Therefore, the average delay incurred by data stream $f$’s packets when passing through cognitive node $v$ is given by [15], $D_v^f = \frac{1}{\mu^f_v} - \frac{1}{\lambda^f_v}$.

The end-to-end delay is the accumulation of the incurred delay by a packet when it goes through the different nodes along its route from source to destination. Therefore, the average end-to-end delay for the packets of data stream $f$ is given by $D^f = \sum_{v \in V_i} D_v^f$, where $V_i$ is the set of nodes forming the route for the packets of data stream $f$.

C. Optimization Problem Formulation

The considered mesh network has multiple data connections where the average end-to-end delay for each connection needs to be minimized. To group these requirements into a single objective function, we choose to minimize the total (sum) average delay of all the connections in the network. The problem is cast into a non-linear integer programming formulation as follows,

$$\text{min} \sum_{f \in F} \sum_{v \in V_i} D^f_v \quad \text{subject to:}$$

$$\sum_{e \in E^{out}_{v}(f)} x_{f,e} = 1, \forall f \in F;$$

(4)

$$\sum_{e \in E^{out}^v} x_{f,e} = \sum_{e \in E^{out}^v} x_{f,e}, \forall v \in V \setminus \{f(s), f(d)\}, \forall f \in F;$$

(5)

$$\sum_{f \in F} \sum_{e \in E^{out}^v} y^{I,e}_{f,e} \leq 1, \forall v \in V, \forall t \in [1,T], \forall c \in [1,N];$$

(6)

$$\sum_{f \in F} \sum_{e \in E^{out}^v} y^{I,e}_{f,e} \leq 1, \forall v \in V, \forall t \in [1,T], \forall c \in [1,N];$$

(7)

$$\sum_{c=1}^{N} \sum_{f \in F} \sum_{e \in E^{out}^v} y^{I,e}_{f,e} \leq 1, \forall v \in V, \forall t \in [1,T], \forall f \in F;$$

(8)

$$\lambda^f_v < \mu^f_v, \forall v \in V, \forall f \in F;$$

(9)

$$x_{f,e} \in \{0,1\}, \forall e \in E, \forall f \in F;$$

(10)

$1$The case of imperfect sensing can be simply accommodated by multiplying (1) with the probability of idle channel detection in order to get the correct arrival rate.
\[ y_{f,e}^{c} \in \{0,1\}, \forall e \in E, \forall f \in F, \forall t \in [1,T], \forall c \in [1,N]. \] (12)

The objective function in (3) minimizes the total average end-to-end delay for all the data streams, where \( F \) is set of all data streams in the network. Constraints (4) and (5) guarantee single path routing for each data stream. Where \( f(s) \) and \( f(d) \) denotes the source and destinations nodes for data stream \( f \), respectively. Constraint (6) establishes the link between the two sets of decision variables, and restricts the allocation of resources to the links taking part in the selected routes.

In (7) \( L_e \) denotes the set of links interfering with link \( e \) as per the interference conditions discussed in the previous section. This constraint ensures that interfering links are allocated distinct resource elements to avoid interference. Constraint (8) ensures that a node is assigned no more than one channel per time slot. This is assuming the available radios can only access a single channel at any given time. This constraint can be modified to accommodate multi-channel radios, as well as different capabilities for different nodes. Constraint (9) is a half duplex constraint. Constraint (10) guarantees the stability of all the queues in the network. Unstable queues result in unbounded delays. Finally, constraints (11) and (12) are binary value constraints ensuring that the decision variable can only take a value of 0 or 1.

It is noted from (1) and (2) that the average arrival and service rates at each node are linear in the decision variables. The objective function is convex in \( \lambda_f^{\phi} \) and \( \mu_f^{\phi} \), thus convex in the decision variables as well. Moreover, all other constraints are linear in the decision variables. Despite the problem having a convex objective and linear constraints, it has combinatorial complexity because of the integer-valued constraints.

### D. Suboptimal Relaxation

To reduce the high complexity of the nonlinear integer programming problem presented in the previous section, we propose to relax the binary value constraints (11) and (12) and allow the decision variables to take any value in the interval \([0,1]\). The resulting optimization problem has the same form as in (3) to (10), with (11) and (12) rewritten as follows

\[
x_{f,e} \in [0,1], \forall e \in E, \forall f \in F; \quad (13)
\]

\[
y_{f,e}^{c} \in [0,1], \forall e \in E, \forall f \in F, \forall t \in [1,T], \forall c \in [1,N]. \quad (14)
\]

This relaxation transforms the nonlinear integer program into a convex optimization problem with real valued variables for which efficient solution algorithms exist [16]. However, the resulting optimal solution for the relaxed problem is not guaranteed to be optimal for the original integer-valued problem.

In the relaxed problem, the routing variables \( x_{f,e} \) take values in the range \([0,1]\). Therefore, the constraints in (4) and (5) can no longer guarantee single path routes between the source and destination of the different data streams. Multiple paths between the source and destination of a given data stream will be constructed, and resource elements will be assigned to the links along these paths.

One can interpret the resulting multi-path routes as if they are the solution to a stochastic routing problem. Given this interpretation, the fractional values of the decision variables \( x_{f,e} \) define the probability with which a packet belonging to data stream \( f \) is routed through link \( e \). Similarly, a stochastic interpretation can be provided for the fractional values of the resource allocation variables \( y_{f,e}^{c} \). The value of \( y_{f,e}^{c} \) can be interpreted as the probability with which a given resource element \((t,e)\) is used by link \( e \) for forwarding packets belonging to data stream \( f \). Fractions of the same resource element might be allocated to different links and/or different data streams.

Since the initial design goals were for single path routing and deterministic resource allocation, we have to transform the resulting stochastic routing and resource allocation solutions into the required deterministic ones. For this transformation to take place, the decision variables \( x_{f,e} \) and \( y_{f,e}^{c} \) have to be converted back into binary valued variables. In the course of this conversion process, it is crucial not to violate any of the original problem’s constraints.

Algorithm 1 describes our proposed scheme for converting the real-valued solution into a binary-valued one while obeying the constraints imposed by the optimization problem.

### Algorithm 1 Algorithm for transforming the real-valued solution into a binary-valued one

1. for \( f \in F \) do
2. Using the routing variables \( x_{f,e} \), construct all possible paths for data stream \( f \)
3. For path identified in step 2, calculate the probability of routing packets through that path
4. Set \( x_{f,e} = 1, \forall e \in K_e \), where is \( K_e \) the set of links in the path with highest probability
5. Set \( x_{f,e} = 0, \forall e \notin K_e \)
6. Set \( y_{f,e}^{c} = 0, \forall e \notin K_e, \forall t \in [1,T], \forall c \in [1,N] \)
7. end for
8. Define the set \( Z = \{(t,c,f,e) : y_{f,e}^{c} > 0\} \)
9. \( \left( f_{o}^{c},e^{c},t^{c},c^{c} = \arg \max_{(f,e,t,c) \in Z} y_{f,e}^{c} \right) \)
10. Set \( y_{f_{o}^{c},e^{c}} = 1 \)
11. Define the set \( Q(f^{c},e^{c},t^{c},c^{c}) = \{(f,e,t,c) \in Z : \) given \( y_{f_{o}^{c},e^{c}} = 1, \forall \) of the constraints (7), (8), (9) is violated \}
12. Set \( y_{f,e}^{c} = 0, \forall (t,c,f,e) \in Q \)
13. Update \( Z = Z \setminus \{Q(f^{c},e^{c},t^{c},c^{c})\} \)
14. if \( Z \neq \emptyset \) then
15. Goto step 9
16. else
17. End
18. end if

### IV. DISTRIBUTED SOLUTION

The optimization problem described above is centralized in nature. It requires global information about the network to be present at a central point to be able to find a solution. In a wireless mesh network however, each node has local
information about its environment. This local information need to be wirelessly communicated to the central point from all the nodes in the network. In many cases such communication overhead is not practical, especially in networks with a large number of nodes. Therefore, a distributed and scalable solution scheme in which calculations are done locally at each node, or at local central points (or cluster heads) is desirable. In this section, we propose a decomposition of the original problem into smaller subproblems that can be efficiently solved in a distributed fashion.

The distributed solution approach is based on dual decomposition. The first step is to form the Lagrangian by relaxing the constraint in (6) which forms the link between the routing decision variables \( x_{f,e} \) and the resource allocation decision variables \( y_{f,e}^{t,c} \):

\[
L = \sum_{f \in F} \sum_{v \in V_f} D_v^f + \sum_{f \in F} \sum_{e \in E} \sum_{t=1}^{T} \sum_{c=1}^{N} p_{f,e}^{t,c} (y_{f,e}^{t,c} - x_{f,e}), \tag{15}
\]

where \( p_{f,e}^{t,c} \) are the Lagrange multipliers.

We note that in the Lagrangian (15), the routing \( x_{f,e} \) and resource allocation \( y_{f,e}^{t,c} \) decision variables are separable. Therefore, the Lagrangean minimization problem can be decoupled into two disjoint subproblems, each of which depends only on a single set of decision variables, \( x_{f,e} \) or \( y_{f,e}^{t,c} \). The two disjoint subproblems are:

1) The routing subproblem:

\[
\text{max} \quad \sum_{f \in F} \sum_{v \in V_f} D_v^f + \sum_{f \in F} \sum_{e \in E} \sum_{t=1}^{T} \sum_{c=1}^{N} p_{f,e}^{t,c} x_{f,e}, \tag{16}
\]

subject to constraints (4), (5), and (13).

2) The resource allocation subproblem:

\[
\text{min} \quad \sum_{f \in F} \sum_{v \in V_f} D_v^f + \sum_{f \in F} \sum_{e \in E} \sum_{t=1}^{T} \sum_{c=1}^{N} p_{f,e}^{t,c} y_{f,e}^{t,c}, \tag{17}
\]

subject to constraints (7), (8), (9), (10), and (14).

Thus, the global end-to-end delay minimization problem decomposes into a routing subproblem and a resource allocation subproblem. The resource allocation subproblem distributes the available channel resources among the different data streams and network links such that the end-to-end delay is minimized, and the stability of the queues is maintained. Solution of the resource allocation subproblem may result in multiple paths for packet routing between a source node and its destination. The routing subproblem then ensures that the optimal single path route is selected for packet routing between each source and its destination.

The dual variables \( p_{f,e}^{t,c} \) play a key role in coordinating between the solution of the routing and resource allocation subproblems. For instance, from the resource allocation point of view the value \( p_{f,e}^{t,c} \) can be interpreted as the price associated with using a given resource element \((t,c)\) to forward packets for traffic stream \( f \) over link \( e \). It is worth mentioning here that the same resource element can have different prices when assigned to different links, or when assigned for different traffic streams. Therefore, the resource allocation subproblem tries to minimize the end-to-end delay while utilizing a set of resources that has a low price. From the routing point of view, the value \( \sum_{t=1}^{T} \sum_{c=1}^{N} p_{f,e}^{t,c} \) which appears in (16) can be interpreted as routing metric associated with link \( e \) when it is used to forward packets for traffic stream \( f \). Similar to the resource allocation subproblem, each link may have different metrics when used for forwarding different traffic streams' packets. Therefore, the routing subproblem as formulated in (16) selects the single path with the largest overall metric.

Because of the different ways the two subproblems use to deal with the Lagrange multipliers, a mismatch between the selected path by the routing subproblem and where the resources are allocated occur. This mismatch necessitates an update in the value of the Lagrange multipliers as discussed in the next section. During this update, the price of the resources along the selected path will decrease, while the other resources will encounter a price increase. Given the new values for the prices/routing metrics, more resources may be allocated to the path selected by the routing subproblem, or a switch to a new path to which enough resources are allocated may take place. The dynamics of the Lagrange multipliers updates will be discussed in details in the results section.

### A. Primal-Dual Solution Framework

The convexity of the original optimization problem in (3) enables us to use the duality theory [17] to solve the problem via its dual. Here, we present a primal-dual algorithm to solve the optimization problem.

**Algorithm 2 Primal-Dual solution algorithm**

1: Set \( i = 0 \). Initialize \( p_{f,e}^{t,c}(i) \).
2: In the primal domain, solve subproblems (16) and (17).
3: In the dual domain, update the dual variables

\[
p_{f,e}^{t,c}(i) = \left[ p_{f,e}^{t,c}(i) + \nu(i)(y_{f,e}^{t,c} - x_{f,e}) \right]^+, \tag{18}
\]

where \([ \cdot ]^+\) denotes max(0,·), and \( \nu(i) \) is the step size.
4: Set \( i = i+1 \). Return to step 2 until convergence is attained.

Convexity of the constraints guarantees that the update in (18) is a subgradient for the dual variables. Thus, as long as the step sizes are chosen appropriately (e.g., as a square summable but not summable sequence), the dual update eventually converges.

### B. Routing Subproblem

In the routing subproblem, each edge in the graph modeling the mesh network has an associated metric or weight. A given edge may have different metrics for different data streams. As formulated in (16) the routing subproblems searches for a single path between each source and its destination such that the sum of the metrics of the edges along each path is maximized. Formulation in (16) and its associated constraints
is on the form of a linear program which have efficient solution methods.

In the form presented above, this problem is a classical problem in graph theory for which there exist several efficient centralized (e.g., Dijkstra’s and Bellman-Ford’s algorithms) and decentralized (e.g., Distributed Bellman-Ford’s algorithm) solution algorithms [14]. Given the weights $p_{f,c}$ associated with each link in the network any of these algorithms can be employed to solve the routing subproblem in a decentralized fashion. We note that to start constructing the routes, each node needs only to know the metrics $p_{f,c}$ associated with its outgoing links.

C. Resource Allocation Subproblem

As discussed above, the resource allocation subproblem (17) and its associated constraints do not impose any constraint on the number of paths a given data stream’s packets can be routed through. Therefore, the sum in (17) is now over all network nodes that are used to forward packets. Since the delay is additive, minimizing the end-to-end delay is equivalent to minimizing the delay at each one of the nodes used for packet forwarding. It is also noted that constraints (8), (9), and (10) are local constraints that are imposed at each individual node. Constraint (7), which prevents interference between different nodes involves nodes that are within interference range from each other. Therefore, for a distributed implementation of the resource allocation subproblem, each node need only to have information about the state of the nodes sharing the same contention domain [18] with it. A simple HELLO protocol can be used to help each node discover its neighboring nodes and construct the contention domains.

Based on the knowledge about the different contention domains in the network and the prices $p_{f,c}^t$ for each resource element available at different points in the network, a source routing scheme could be used to allocate resources along the different paths between a source and its destination. Each source node sends a route request packet that propagates through the network till it reaches the destination through different possible paths. The destination then sends back route reply packets to the source node along those different paths, since no single path constraint is imposed. Each route reply packet contains information about the nodes through which it passed, the contention domain they belong to, and the free resources they have. The information provided by these packets to the source node enables the source node to solve the resource allocation subproblem (17).

V. Results and Discussions

We consider a cognitive mesh network consisting of $M = 6$ nodes uniformly distributed in a 500m x 500m square region. The transmission range of any node is set to $R = 100m$, and the interference range $I = 2R$. A TDMA frame has $T = 10$ time slots, and we assume all primary channels have idle probability $\alpha_i = 0.8$, $i \in [1,N]$, and all mesh network’s traffic streams have arrival rates $\lambda_f = 0.2$, $f \in [1,F]$. Channel parameters used are: transmission power $P = 100$ mW, SNR threshold $\zeta = 20$ dB, path loss exponent $\gamma = 3.7$, and noise power spectral density $N_0 = 1e^{-11}$.

First, we study the convergence behavior of primal-dual distributed solution framework. We consider the network with a single traffic stream and a single primary node, hence a single primary channel. Fig. 2 depicts the evolution of the average end-to-end delay with time. It is noted that the system converges to the same solution as the centralized framework in less than 50 iterations. We then study the impact of varying loads on the network’s performance in fig. 3. The network starts with a single data stream. It is noted that the distributed primal-dual algorithm converges to optimal delay value of 1.5 time slots in about 100 iterations. After 200 iterations a second data streams is admitted into the network. For this second stream the algorithm converges almost instantly since all the information about the network was already gathered in the first set of iterations. We note that the end-to-end delay per data stream has increased from 1.5 to 5 time slots, that is because the available resource are now shared between the two data streams. After another 200 iterations one of the data streams leaves the network. Again in this case we see almost immediate convergence, where the freed up resources are allocated to the remaining stream and its delay falls from 5 to 1.5 time slots.

Fig. 4 depicts the average end-to-end delay per data stream as a function of the number of primary channels for different number of data streams. The presented results are averaged
over multiple runs in order to average out the effects of node placements and channel fading. It is seen that the network cannot support 3 data streams when only a single channel is available (delay is infinite for one or more stream). As expected, increasing the number of available channels results in a decrease in the delay for all the data streams. But with one or two data streams increasing the number of channels beyond two has no effect on the delay. That’s because of the single channel per time slot constraint. Once all the links are using the maximum number of time slots possible, they cannot benefit from any additional channel resources. As the number of data streams increases, the maximum number of channels that can be used also increases as noticed from the case of 3 data streams.

The effect of the primary idle probability and the data streams’ arrival rate is shown in fig. 5 for the case of 2 primary channels and 2 data streams. As the primary idle probability increases, more spectrum opportunities are present, therefore, the delay decreases. On the other hand, increased arrival rates result in more contentions between mesh nodes, leading to increased delay for all the data streams. It can also be seen that for a given arrival rate, there exists a value for the idle probability below which the delay is infinite (queues are unstable) and service cannot be supported. For example, for \( \lambda_s = 0.3 \) for each of the two data streams a minimum idle probability of \( \alpha_c = 0.7 \) is required to support that traffic load.

VI. Conclusions

Routing and resource allocation in cognitive WMNs is considered. For this class of networks, it is shown that cross-layer design is crucial since separate design strategies lead to infeasible solutions in many cases. A cross-layer design minimizing average delay is presented, and centralized as well as decentralized protocols are proposed. Results reveal the efficiency of the decentralized scheme. Furthermore, it is concluded that for a given primary idle probability, there exist upper bounds on the number of admissible data streams in the mesh network, as well as the arrival rates for those streams.

ACKNOWLEDGMENT

This work is supported by Qatar National Research Fund (QNRF) No. 08-374-2-144.

REFERENCES