Decentralized Throughput Maximization in Cognitive Radio Wireless Mesh Networks

Amr A. El-Sherif, *Member, IEEE* and Amr Mohamed, *Member, IEEE*

Abstract—Scheduling and spectrum allocation are tasks affecting the performance of cognitive radio wireless networks, where heterogeneity in channel availability limits the performance and poses a great challenge on protocol design. In this paper, we present a distributed algorithm for scheduling and spectrum allocation with the objective of maximizing the network's throughout subject to a delay constraint. During each time slot, the scheduling and spectrum allocation problems involve selecting a subset of links to be activated, and based on spectrum sensing outcomes, allocate the available resources to these links. This problem is addressed as an aggregate utility maximization problem. Since the throughput of any data flow is limited by the throughput of the weakest link along its end-to-end path, the utility of each flow is chosen as a function of this weakest link's throughput. The throughput and delay performance of the network are characterized using a queueing theoretic analysis, and throughput is maximized via the application of Lagrangian duality theory. The dual decomposition framework decouples the problem into a set of subproblems that can be solved locally, hence, it allows us to develop a scalable distributed algorithm. Numerical results demonstrate the fast convergence rates of the proposed algorithm, as well as significant performance gains compared to conventional design methods.

Index Terms—Resource allocation, distributed algorithms, cognitive radios, mesh networks

1 Introduction

Cognitive radio is a communication paradigm in which wireless users are classified into two categories based on whether they are licensed to use a particular spectrum band (primary users (PUs)) or are unlicensed (secondary users (SUs)) [1]. SUs are allowed to opportunistically use the spectrum as long as they do not cause harmful interference to active PUs. This is achievable if PU receivers are far enough from the SU transmitter (spatial channel availability), or no PU receivers are receiving while the SU transmitter is transmitting (temporal channel availability). This opportunistic and dynamic communication paradigm leads to higher spectrum utilization, and provides SUs with good service availability and reliability. One of the biggest challenges in cognitive radio networks is spectrum sharing, which defines the set of rules and strategies that regulate the behavior of SUs regarding spectrum mobility, allocation, and access. In general, spectrum sharing architectures are classified into two categories: centralized and distributed [2]. For the centralized case, a spectrum management entity controls both spectrum allocation and spectrum access [3]–[5]. In a distributed architecture, on the other hand, each SU is responsible for the channel allocation and access decisions. The SU may make its decisions based on its local observation of the network and spectrum status or by cooperating with other SUs to have a more global observation [6]–[8].

A cognitive radio mesh network is a wireless mesh network (WMN) that deploys cognitive radios for its nodes, and relies on opportunistic and dynamic spectrum access for its operation [9]–[11]. In addition to increasing spectrum utilization and overcoming spectrum scarcity, cognitive radio mesh networks were motivated by a number of potential applications. For instance, cognitive mesh networks could alleviate congestion in traditional WMNs by searching for available channels in the primary band so they can reduce the congestion on the operational band of the WMN by moving some of the mesh clients to those available channels [10], [12]. In some situations, mesh nodes need to restrict their transmission power levels so that the interference they cause at the location of other mesh nodes in neighboring cells stays within a pre-calculated threshold that insures the required QoS. However, restricting the transmission power means restricting the network coverage. Exploiting cognitive radios allows mesh nodes to heal this problem by extending their coverage on any available channels in the primary band [13]. Moreover, recent research initiatives suggest the integration of different heterogeneous wireless access networks into one cognitive radio mesh network using the ability of cognitive radios to adapt to different transmission/reception parameters like power, frequency, modulation, and medium access [14]. Therefore, a mobile cognitive mesh client can observe the performance (in terms of throughput, service availability, and reliability) of different coexisting wireless access networks and select the network that best fits its requirements [15].

- *A. A. El-Sherif is with the Department of Electrical Engineering, Alexandria University, Alexandria 21544, Egypt. E-mail: amr.elsherif@ieee.org.*
- *A. Mohamed is with the Computer Science and Engineering Department, Qatar University, Doha, 2713, Qatar. E-mail: amrm@ieee.org.*

Manuscript received 30 Sep. 2012; revised 18 June 2013; accepted 20 June 2013. Date of publication 19 Feb. 2014; date of current version 22 July 2014. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below. Digital Object Identifier 10.1109/TMC.2013.82

The problem of resource allocation in cognitive radio mesh networks has been addressed in a number of studies recently. In [16], three different frequency assignment problems were studied: common broadcast frequencies, non-interfering frequencies for simultaneous transmissions, and frequencies for direct source-destination communications. Each is viewed as a graph-coloring problem, and both centralized and distributed algorithms were presented. However, these algorithms only guarantee non-interfering direct communication between pairs of nodes without considering any end-to-end performance measure. In [17], a cluster-based approach was proposed such that the network is clustered into 1-hop clusters based on channel availability. Nodes that belong to the same cluster use a common control channel to negotiate their data channels. However, no inter-cluster interference guarantees were obtained, and the ability to adapt to varying spectrum availability is lacking. The main objective in [18] was to select a channel that a node can transmit on such that the interference within the transmission range of the node does not exceed a predefined threshold. Fixed and adaptive power control strategies were proposed for this purpose. In [19], the authors defined two bandwidth allocation problems based on max-min fairness models, and presented linear programming solutions as well as heuristic algorithms for those problems. The presented algorithms are centralized and require global information about the network to be collected at a central point. Moreover, no performance measures, such as throughput or delay, are considered in the bandwidth allocation process. In [20], the authors studied how to assign frequency bands at each node to form a topology such that a certain performance metric can be optimized. A layered graph was proposed to model frequency bands available at each node and to facilitate topology formation and achieve optimization objective. The authors considered the so-called fixed channel approach whereby the radio is assumed to operate on only one channel at a specific time. The main limitation of the works mentioned above is that they don't accommodate any end-to-end performance measures in their channel allocation decisions. Therefore, no QoS guarantees can be obtained using such designs. The authors in [21] tried to overcome this shortcoming by proposing a cross-layer routing and channel allocation algorithm that minimizes the end-to-end packet dropping probability. However, the design presented was a centralized one and of limited scalability.

This paper studies the resource allocation problem in cognitive radio mesh networks. The objective of the resource allocation problem is to maximize the aggregate end-to-end throughput of the different traffic streams in the network. The problem is cast as an aggregate utility maximization problem, where the utility function takes a logarithmic form and is a function of the minimum throughput along a stream's end-to-end path. This choice of utility guarantees fairness among the different streams [22]. The cognitive mesh network's dynamics and its dependence on primary users' activities are captured through the development of a queueing theory model describing the network operation. This model enables us to study the throughput as well as the delay at each node in the network. Characterization of the delay at each node makes it possible to introduce an end-to-end delay constraint for each traffic stream, where the constraints can be different for different streams according to their QoS requirements. The utility maximization problem is formulated as a non-linear integer programming (NIP) problem having combinatorial complexity. By relaxing the integer valued constraints imposed on the decision variables, an efficient solution to the relaxed problem is found based on the dual decomposition of the problem. This decomposition enables us to subdivide the network wide resource allocation problem into a set of local subproblems, one subproblem per traffic stream. Such decomposition makes it easy to implement the solution algorithm in large networks. Since the target is an integer valued solution, a simple algorithm that reconstructs an integer valued solution from the relaxed one is developed and presented. Performance gains in terms of end-to-end throughput and delay are demonstrated in comparison to a trivial uniform allocation scheme and the max-min bandwidth allocation scheme presented in [19].

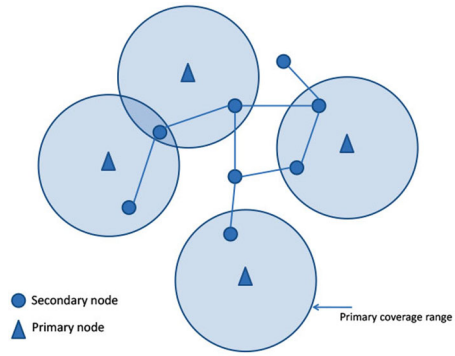


Fig. 1. Network model.

The rest of the paper is organized as follows. The different aspects of the system model are presented in Section 2. In Section 3 the resource allocation problem is formulated as an integer programming optimization problem and suboptimal approximate solution is provided. Section 4 presents an efficient distributed solution algorithm for the resource allocation problem. Numerical results are presented and discussed in 5. Finally, the paper is concluded in Section 6.

2 System Model

2.1 Network Model

The cognitive radio based wireless mesh network model used is depicted in Fig. 1. The cognitive mesh network consists of M nodes that opportunistically share the spectrum resources with a primary network composed of N transmitter-receiver pairs. Each primary transmitter-receiver pair operates over a unique channel that does not overlap with other users' channels. Therefore, there will be N non-overlapping channels. Furthermore, all primary channels have the same bandwidth. 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.

Each cognitive mesh node will opportunistically access idle primary channels to transmit its packets. Local channel availability can be detected using one of the different spectrum sensing techniques available [23]–[26]. The cognitive mesh network employs hybrid TDMA/FDMA 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 frequency channels to transmit over.

Since the primary network transmissions are also slotted, it is customary that the cognitive network adjusts the boundaries of its time slots to match those of the primary network [27]–[30]. In any practical system, PUs have pilots, preambles, synchronization words or spreading codes that are used by their receivers for coherent detection. For example: TV signal has narrow-band pilot for audio and video carriers; CDMA systems have dedicated spreading codes for pilot and synchronization channels; OFDM packets have preambles for packet acquisition and pilot tones for channel estimation and equalization. A cognitive radio that has prior knowledge about the primary system can benefit from these known structures to acquire the different parameters of the primary system [31]–[34]. In general, when SUs do not try to decode any part of the PUs signals for cooperation [35], [36] or for exploiting the PUs feedback [37], [38], coarse acquisition of the time slot boundaries in addition to the use of guard intervals can guarantee the protection of PUs from any interference.

A time slot and a channel pair (t, c), is considered as the minimum unit for resource allocation and we will call it resource element. As defined, a resource element is similar to the resource block concept in LTE systems [39]. 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 the packet at the head of its queue to the next node along the route to its 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. The case of imperfect sensing, where SUs can sometimes make false detections, can be easily incorporated into the problem formulation similar to the model in [40]. After any successful transmission, the receiving node acknowledges the reception of the 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 given 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 $c \in C_u \bigcap C_v$ 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. Since the graph is directed, then there will be two edges between nodes u and v in this case, the first has node u as a transmitter, while the second has node v as transmitter. For the ease of presentation, it is assumed that all cognitive nodes use the same fixed transmission power, therefore, all nodes have the same transmission range R. However, as it will be discussed later, each node collects information about its neighbors at the beginning of network's operation, and all calculations are based on these gathered information. Therefore, the presented protocol does not depend on this assumption, and can already accommodate nodes with different power levels and transmission ranges. Due to the primary nodes' activity, channel availability will vary with time, which poses a challenge to the resource allocation protocol design.

In this work, the effect of the wireless interference between different nodes is modeled based on the protocol model [41], 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

1) there is a shared node between e_1 and e_2 (because of half duplexing, unicast communications, or collisions) or
2) any node from e_1 is within interference range $I > R$ of any node that is part of e_2, and they are using the same channel.

Because of the ACK packet sent back to the transmitter by the receiver, both the link's transmitter and receiver need to be interference free. It is worth mentioning here that ACK packets are usually very small in size, which allows them to be protected by low rate error correction codes resulting in a high level of protection while still having a relatively small size. Therefore, the effect of lost feedback packets will not be considered in this paper. It is noted that, the use of error correction codes for information packets protection is not considered in this paper. While the use of error correction codes will enhance the performance of any network, its use will complicate the analysis without providing any new insights to the paper's main focus.

2.2 Channel Model

The wireless channel between a node and its destination is modeled as a Rayleigh flat fading channel with additive white Gaussian noise [42]. 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 [42] and [28].

2.3 Queuing Model

Each node in the cognitive mesh network has an infinite buffer for storing packets of fixed length. The finite buffers case could also be accommodated into our model with slight modifications to the optimization problem formulated in the next section. The packet transmission time equals to one time slot duration. Since at the start of each time slot secondary users spend time sensing the channel, they will have less time to transmit their packets compared to primary users. However, SUs can choose a modulation scheme and packet length such that the useful part of a time slot is sufficient to transmit one complete packet.

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 λ_u^f [43].

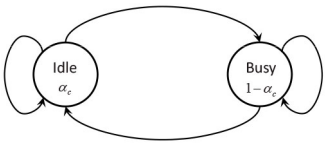


Fig. 2. Primary node's Markov chain model.

In other words, the probability that a new packet arrives at any given time slot t is λ_u^f. Moreover, the packet arrival processes are assumed to be independent from one data stream to another.

The state (idle or busy) of any of the N primary channels is modeled using a two state Markov chain as shown in Fig. 2. Using the stationary distribution of the Markov chain, at any given time slot channel c will be idle (Markov chain in the off state) with probability α_c. The evolution of any channel is independent from all other channels.

3 Problem Statement

In this work, we focus on allocating channel and time resources to maximize the aggregate utility for all the traffic streams in the network. We assume that the routes between each stream's source and its destination are already established.

Before presenting the optimization problem and the solution approach we need first to analyze the effect of the resource allocation decisions on the network's performance.

3.1 Queueing Analysis

In this work we rely on queueing theory to model the different aspects of the cognitive mesh network and to form a base for our resource allocation protocol design.

We start by calculating the average arrival and service rates at the different nodes in the cognitive mesh network. For this calculation we need to first introduce the decision variables that define the resources allocated to the different links in the network. We define the following set of decision variables:

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

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

To define the average arrival rate for any node v along the route of data stream f we start by identifying 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) 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 probability of a packet arrival 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 this probability can be written as

$$a_v^{f,t} = \sum_{e \in E_v^{in}} \sum_{c=1}^{N} y_{f,e}^{t,c} \frac{\lambda_{e(s)}^f}{\mu_{e(s)}^f} \left[I_e^c \alpha_c + (1 - I_e^c) \right] \left(1 - P_e^{out}\right), \quad (1)$$

where E_v^{in} is the set of incoming edges to node v, $e(s)$ is the source node for edge (link) e, $\lambda_{e(s)}^f$ is its arrival rate and $\mu_{e(s)}^f$ its service rate. By modeling each queue as discrete time Markov chain, it can be shown that the fraction $\lambda_{e_s}^f / \mu_{e_s}^f$ is the probability that the queue has at least one packet [44], and hence will transmit a packet to the following node on the route whenever it has a chance. α_c is the probability that channel c is idle during time slot t, P_e^{out} 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 $I_e^c = 0$. Note that for simplicity we assume perfect sensing at all cognitive mesh nodes. 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. Interference from other cognitive nodes is not reflected in (1), since the resource allocation scheme discussed below makes sure that no two interfering links can share the same resources.

It is clear that the packet arrival probability can vary from one time slot to another within the same frame since assigned channels can vary between time slots. This dependence is emphasized in (1) by the use of the superscript t (which is the index of the time slot in a TDMA frame). Since the same resource allocation pattern is repeated each TDMA frame, a given time slot within any frame will always have the same packet arrival probability. For the purpose of mathematical tractability, instead of using different packet arrival probabilities for the different time slots in a TDMA frame, we will use the average packet arrival probability over one complete frame and use it for all the time slots. The average packet arrival probability can then be calculated as,

$$\lambda_v^f = \frac{1}{T} \sum_{t=1}^{T} a_v^{f,t}, \quad (2)$$

which is interpreted as the probability that a packet from data stream f arrives at node v in any time slot. Given this definition, packet arrivals can be seen as Bernoulli trials at each time slot with success probability λ_v^f. Therefore, the packet arrival process can be approximated as a Bernoulli process with average arrival rate λ_v^f.

Similarly, to calculate the average service rate, we start by identifying the events necessary for a successful packet transmission. This will take place if in a given time slot t

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 probability of a packet being serviced from node v along the route of data stream f is defined as

the joint probability of these two events, which is given by

$$s_v^{f,t} = \sum_{e \in E_v^{out}} \sum_{c=1}^{N} y_{f,e}^{t,c} \left[I_e^c \alpha_c + (1 - I_e^c) \right] \left(1 - P_e^{out} \right), \tag{3}$$

where E_v^{out} is the set of outgoing edges to node v.

Similar to the arrival probabilities, the service probabilities vary between time slots. Here also we resort to using the average service probability per time slot, calculated as

$$\mu_v^f = \frac{1}{T} \sum_{t=1}^{T} s_v^{f,t}, \tag{4}$$

which is the probability that a packet from data stream f leaves the queue of node v in any time slot. Similar to the arrival events, the service events can be seen as Bernoulli trials at each time slot with success probability μ_v^f. Therefore, the packet service process can be approximated as a Bernoulli process with average service rate μ_v^f.

3.2 Optimization Problem Formulation

First, we assign a utility function $U_f(\min_{v \in V_f} \mu_v^f)$ for each traffic stream f to measure the degree of service satisfaction based on the minimum service rate over the end-to-end path of that traffic stream. Let $\bar{\mathbf{y}} = \left\{ y_{f,e}^{t,c}, f \in F, e \in E, t \in [1,T], c \in [1,N] \right\}$ be the vector of all decision variables. The optimization problem is to find a resource allocation solution that maximizes the aggregate utility function of all traffic streams. This can be formulated as follows:

$$\max_{\bar{\mathbf{y}}} \sum_{f \in F} U_f \left(\min_{v \in V_f} \mu_v^f \right) \tag{5}$$

subject to:

$$\sum_{f \in F} \sum_{e \in E_v^{out}} \sum_{c=1}^{N} y_{f,e}^{t,c} \leq 1, \qquad \forall v \in V, \forall t \in [1,T]; \tag{6}$$

$$\sum_{f \in F} \sum_{e \in E_v^{in}} \sum_{c=1}^{N} y_{f,e}^{t,c} + \sum_{f \in F} \sum_{e \in E_v^{out}} \sum_{c=1}^{N} y_{f,e}^{t,c} \leq 1, \qquad \forall t \in [1,T], \forall v \in V; \tag{7}$$

$$\sum_{f \in F} \sum_{e' \in S_e} y_{f,e'}^{t,c} \leq 1, \qquad \forall e \in E, \forall t \in [1,T], \forall c \in [1,N]; \tag{8}$$

$$\lambda_v^f < \mu_v^f, \qquad \forall v \in V, \forall f \in F; \tag{9}$$

$$y_{f,e}^{t,c} \in \{0,1\}, \quad \forall e \in E, \forall f \in F, \forall t \in [1,T], \forall c \in [1,N]. \tag{10}$$

The objective function in (5) is set to maximize the aggregate utility function for all the traffic streams, where F is set of all traffic streams, and V_f is the set of nodes along the end-to-end path for traffic stream f.

Constraint (6) 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 slot. This constraint can be modified to accommodate multi-channel radios, as well as different capabilities for different nodes. Constraint (7) is a half duplex constraint, making sure a node cannot transmit and receive simultaneously. In (8), S_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, which avoids interference between mesh nodes during packet transmissions. Constraint (9) guarantees the stability of all the queues in the network. Unstable queues result in unbounded delays. Finally, constraint (10) ensures that the decision variable can only take a value of 0 or 1.

Assumption 1: There exists at least one vector $\bar{\mathbf{y}}$ of decision variables, such that constraints (6) through (9) are satisfied.

Assumption 2: The utility functions U_f are increasing, twice continuously differentiable and strictly concave in the interval $[0,1]$ (by definition, $\mu \in [0,1]$).

It is also noted that a delay constraint can be imposed at each node. Given that the arrival and services processes to and from any node are Bernoulli processes with average rates λ_v^f and μ_v^f, respectively. Each node's queue can be modeled as a discrete time M/M/1 queue [45]. Therefore, the average delay incurred by data stream f's packets when passing through cognitive node v is given by [45],

$$D_v^f = \frac{1 - \lambda_v^f}{\mu_v^f - \lambda_v^f}. \tag{11}$$

If constraint (9) is modified to take the form

$$\lambda_v^f < \frac{\mu_v^f}{d}, \qquad \forall v \in V, \forall f \in F, \tag{12}$$

where $d > 1$ is a constant parameter. It can be easily shown that (12) can be used to impose a per link delay constraint of the form,

$$D_v^f < \frac{1 - \lambda_v^f}{\lambda_v^f (d-1)}, \tag{13}$$

where the constraint can be adjusted through the constant parameter d. As the value of d increases, the constraint becomes more stringent. It is worth mentioning here that when the system is stable (i.e., all queues are stable), the arrival rates at any node along the path of a given traffic stream f will be constant and equal to the arrival rate at the source node [45]. In this way, we are able to impose a constraint on the delay without using the explicit nonlinear form of (11). It is also noted that, different traffic streams can impose different delay constraints reflecting their required QoS levels.

It is noted that the average arrival and service rates at each node are linear in the decision variables as given by (2) and (4). Therefore, all the constraints are linear in the decision variables.

To validate our analytical model, we simulate a 4 node network with two traffic streams. Fig. 3 shows a comparison between the average delay based on our analytical model and that based on Monte Carlo simulation. The results show that the delay calculated based on the analytical model is within 10% to 15% from that of the simulation model.

3.3 Alternative Formulation

The main target while trying to solve the optimization problem in (5) is to find a solution that is decentralized and

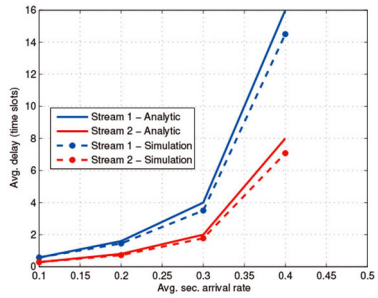


Fig. 3. Comparison between analytical and simulation models.

scalable. Presence of the *min* term inside the utility function makes the problem significantly more difficult. To simplify the problem, we propose to transform the *min* term into a set of linear inequality constraints. This transformation simplifies the objective function and allows us to use the duality theory to find a decentralized solution as will be shown in the next section.

To reformulate the optimization problem in (5), we introduce a new decision variable z_f for each one of the traffic streams in the network. Let $\bar{\mathbf{z}} = \{z_f, f \in F\}$ be the vector of such decision variables. Then the equivalent optimization problem can be written as,

$$\max_{\bar{\mathbf{y}},\bar{\mathbf{z}}} \sum_f U_f(z_f), \tag{14}$$

subject to constraints (6), (7), (8), (9), (10), and

$$z_f < \mu_v^f, \quad \forall f \in F, v \in V_f. \tag{15}$$

The maximization is now over the two sets of decision variables, $y_{f,e}^{t,c}$ and z_f. Each *min* term in the original optimization problem is now replaced by $|V_f|$ linear inequality constraints.

Note that since the objective function (14) is strictly concave with respect to the variables $(z_f, f \in F)$, these variables are unique in any optimal solution. However, the variables $(y_{f,e}^{t,c})$ in an optimal solution may not be unique.

3.4 Suboptimal Relaxation

Despite the concavity of the objective and the linearity of all the constraints, the integer-valued constraints on the decision variables renders the problem extremely complex to solve. To reduce the high complexity of the nonlinear integer programming problem presented in the previous section we propose to relax the binary value constraint (10) and allow the decision variables to take real values in the interval $[0, 1]$. The resulting optimization problem has the same form as in (14), and (6) to (9), with (10) rewritten as follows

$$y_{f,e}^{t,c} \in [0, 1], \quad \forall e \in E, \forall f \in F, \forall t \in [1, T], \forall c \in [1, N]. \tag{16}$$

This relaxation transforms the nonlinear integer program into a convex optimization problem with real valued variables for which efficient solution algorithms exist [46]. 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 resource allocation variables $y_{f,e}^{t,c}$ take values in the range $[0, 1]$. Solving this optimization problem results in fractional allocation of resource elements to the different links in the network. In other words, a resource element can no longer be seen as the minimum unit for resource allocation as discussed earlier. Because of the real-valued nature of $y_{f,e}^{t,c}$, fractions of a given resource element could be allocated to different links in the network.

A stochastic interpretation can be provided for the fractional values of the resource allocation variables $y_{f,e}^{t,c}$. The value of $y_{f,e}^{t,c}$ can be interpreted as the probability with which a given resource element (t, c) 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. Moreover, a given link might have several resource elements assigned to it, where it stochastically chooses which resource element to use for packet forwarding according to the value of the variable $y_{f,e}^{t,c}$.

Since the initial design goals were for deterministic resource allocation, we have to find an algorithm to transform the resulting stochastic resource allocation solutions into the required deterministic one. For this transformation to take place, the decision variables $y_{f,e}^{t,c}$ 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 real-valued to binary-valued (RV-BV) solution conversion while obeying the constraints imposed by the optimization problem. The algorithm is implemented at each node and does not need any centralized control.

The algorithm starts by defining the set X of all resource elements that are assigned to any of the network links. Then the highest assignment probability is found and the corresponding decision variable is set to $y_{f^*,e^*}^{t^*,c^*} = 1$. Given the constraints (6), (7), and (8) a set Q of all the resource elements with assignments that conflict with the above assignment is defined. All the conflicting assignments are then released, and the set X is updated by removing the element (f^*, e^*, t^*, c^*) and all elements in Q from X. These steps are repeated till all the elements are removed from the set X.

Algorithm 1 Real-Valued to Binary-Valued (RV-BV) conversion

1: Define the set $X = \left\{(f, e, t, c): y_{f,e}^{t,c} > 0\right\}$
2: $(f^*, e^*, t^*, c^*) = \arg\max_{(f,e,t,c)\in X} y_{f,e}^{t,c}$.
3: Set $y_{f^*,e^*}^{t^*,c^*} = 1$
4: Define the set $Q(f^*, e^*, t^*, c^*) = \{(f, e, t, c) \in X$: given $y_{f^*,e^*}^{t^*,c^*} = 1$, any of the constraints (6), (7), or (8) is violated.$\}$
5: Set $y_{f,e}^{t,c} = 0$, $\forall (f, e, t, c) \in Q$
6: Update $X = X \backslash \{Q \bigcup (f^*, e^*, t^*, c^*)\}$
7: **if** $X \neq \Phi$ **then**
8: Goto step 2
9: **else**
10: End
11: **end if**

Since constraints (6) and (7) are local to each node and each node has knowledge of its interfering neighbors and their decisions (as discussed in the next section), entries in the set Q at each node can be easily obtained. If we consider an arbitrary node v, constraint (6) forces the node not to use more than one channel in any given time slot. Therefore, given (f^*, e^*, t^*, c^*), the entries in Q corresponding to constraint (6) are $\{(f, e, t^*, c) \in X{:}c \neq c^*\}$. The algorithm needs to simply go through these entries in memory and set them to 0, therefore the complexity of this operation is $\mathcal{O}(|F_v| \times |E_v^{out}| \times (N-1))$, where F_v is the number of data streams passing through node v, E_v^{out} is the set of outgoing links from v and N is the number of available channels. Constraint (7) imposes half duplex communications, therefore, if $e^* \in E_v^{in}$ (i.e., resources are allocated for receiving data into node v), then the corresponding entries in Q are $\{(f, e, t^*, c) \in X{:}e \in E_v^{out}\}$ (i.e., no resources are allocated to any outgoing link). The complexity of going through these elements and setting them to 0 is then $\mathcal{O}(|F_v| \times |E_v^{out}| \times N)$. Finally, constraint (8) makes sure that no two interfering links are active at the same time over the same channel. Since each node has knowledge of its neighbors solutions, if e^* belongs to a node interfering with node v, then the entries in Q at node v are $\{(f, e, t^*, c^*) \in X{:}e \in E_v^{out}\}$, and setting those entries to 0 is of complexity $\mathcal{O}(|F_v| \times |E_v^{out}|)$. It can be seen from this discussion that the three constraints may have overlapped entries in Q, thus, careful design of the algorithm should result in an overall complexity that is less than the sum of the individual complexities.

4 Distributed Solution Approach

Solving the resource allocation problem in a centralized fashion 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, each node has local information about its environment. These local information need to be 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.

4.1 Dual Decomposition

The distributed solution approach is based on dual decomposition. The first step is to define the Lagrangian function for the optimization problem in (14) as follows:

$$\begin{aligned} L = & \sum_{f\in F} U_f(z_f) - \sum_{f\in F}\sum_{v\in V_f} p^1_{f,v}\left(z_f - \mu^f_v\right) \\ & - \sum_{v\in V}\sum_{t=1}^{T} p^2_{v,t}\left[\sum_{f\in F}\sum_{e\in E_v^{out}}\sum_{c=1}^{N} y^{t,c}_{f,e} - 1\right] \\ & - \sum_{v\in V}\sum_{t=1}^{T} p^3_{v,t}\left[\sum_{f\in F}\sum_{c=1}^{N}\left(\sum_{e\in E_v^{in}} y^{t,c}_{f,e} + \sum_{e\in E_v^{out}} y^{t,c}_{f,e}\right) - 1\right] \\ & - \sum_{e\in E}\sum_{t=1}^{T}\sum_{c=1}^{N} p^4_{e,t,c}\left[\sum_{f\in F}\sum_{e'\in S_e} y^{t,c}_{f,e'} - 1\right] \\ & - \sum_{f\in F}\sum_{v\in V_f} p^5_{f,v}\left[\lambda^f_v - \mu^f_v\right] \end{aligned} \tag{17}$$

where $p^1_{f,v}$, $p^2_{v,t}$, $p^3_{v,t}$, $p^4_{e,t,c}$, and $p^5_{f,v}$ are the Lagrange multipliers associated with the problem's constraints. Rearranging the order of the summations, the Lagrangian function could be rewritten as,

$$\begin{aligned} L = \sum_{f\in F}\Bigg[& U_f(z_f) - \sum_{v\in V_f} p^1_{f,v}\left(z_f - \mu^f_v\right) \\ & - \sum_{v\in V}\sum_{e\in E_v^{out}}\sum_{t=1}^{T}\sum_{c=1}^{N} p^2_{v,t} y^{t,c}_{f,e} - \sum_{v\in V}\sum_{t=1}^{T} p^2_{v,t} \\ & - \sum_{v\in V}\sum_{t=1}^{T}\sum_{c=1}^{N} p^3_{v,t}\left(\sum_{e\in E_v^{in}} y^{t,c}_{f,e} + \sum_{e\in E_v^{out}} y^{t,c}_{f,e}\right) \\ & - \sum_{v\in V}\sum_{t=1}^{T} p^3_{v,t} - \sum_{e\in E_f}\sum_{t=1}^{T}\sum_{c=1}^{N} p^4_{e,t,c}\left(\sum_{e'\in S_e} y^{t,c}_{f,e'}\right) \\ & - \sum_{e\in E_f}\sum_{t=1}^{T}\sum_{c=1}^{N} p^4_{e,t,c} - \sum_{v\in V_f} p^5_{f,v}\left(\lambda^f_v - \mu^f_v\right)\Bigg], \end{aligned} \tag{18}$$

where E_f is the set of edges forming the end-to-end path for traffic stream f. From (18) it is concluded that this Lagrangian can be divided into $|F|$ separate subproblems, one for each of the traffic streams in the network. Each subproblem for stream f can be solved locally if the values of the Lagrange multipliers $p^1_{f,v}$, $p^2_{v,t}$, $p^3_{v,t}$, $p^4_{e,t,c}$, and $p^5_{f,v}$ at each node or link taking part in the routing path for stream f are known.

The dual problem can then be written as,

$$\min_{\bar{\mathbf{y}},\bar{\mathbf{z}}} \max L, \tag{19}$$

subject to

$$p^1_{f,v} \geq 0, \qquad \forall f \in F, \forall v \in V; \tag{20}$$

$$p^2_{v,t} \geq 0, \qquad \forall v \in V, \forall t \in [1, T]; \tag{21}$$

$$p^3_{v,t} \geq 0, \qquad \forall v \in V, \forall t \in [1, T]; \tag{22}$$

$$p^4_{e,t,c} \geq 0, \qquad \forall e \in E, \forall t \in [1, T], \forall c \in [1, N]; \tag{23}$$

$$p^5_{f,v} \geq 0, \qquad \forall f \in F, \forall v \in V. \tag{24}$$

4.2 Proximal Minimization Algorithm

Before getting into the minimization of the dual problem of (19), we first note that the dual objective function is *non-differentiable*, and therefore, its gradient may not always exist. This is because in general, differentiability of the dual requires a unique primal optimizer (Ch. 6 [47]), whereas in our case, the optimal values of the variables $y^{t,c}_{f,e}$ can be non-unique. Therefore, the well-known gradient-based algorithms do not apply in this case. The reason behind the non-differentiability of the dual objective function is the lack of strict concavity of the primal objective function (the primal objective function is strictly concave with respect to

Algorithm 2 Proximal Minimization Algorithm

1: Assume $\bar{\mathbf{y}}(1)$ is any feasible point.
2: Set $\bar{\mathbf{x}}(1) = \bar{\mathbf{y}}(1)$.
3: **for** i=1,2,...,n_{iter} **do**
4: Solve (25) to obtain new optimal solution $\bar{\mathbf{y}}(i+1)$ and $\bar{\mathbf{z}}(i+1)$.
5: Set $\bar{\mathbf{x}}(i+1) = \bar{\mathbf{y}}(i+1)$.
6: **end for**

the variables z_f, but not so with respect to the variables $y_{f,e}^{t,c}$).

The solution approach we present here is based on the *proximal minimization algorithm* proposed in (Section 3.4.3 [48]). To make the primal objective function strictly concave, a strictly concave term is added for each of the variables $y_{f,e}^{t,c}$, therefore, making the dual function differentiable with respect to all decision variables. For each variable $y_{f,e}^{t,c}$, we introduce an additional variable $x_{f,e}^{t,c}$, and define $\bar{\mathbf{x}}$ as the vector containing these variables. The approximate primal objective function is now written as,

$$\max_{\bar{\mathbf{y}},\bar{\mathbf{z}},\bar{\mathbf{x}}} \sum_f U_f(z_f) - \sum_{f\in F}\sum_{e\in E}\sum_{t=1}^{T}\sum_{c=1}^{N} \frac{1}{2\kappa}\left(y_{f,e}^{t,c} - x_{f,e}^{t,c}\right)^2, \quad (25)$$

subject to constraints (6) - (10). In (25), $\kappa > 0$ is any constant. The proximal minimization algorithm is an iterative procedure as stated in algorithm 2. It is also noted that the objective function in (25) is separable into $|F|$ subproblems, which will be solved locally as discussed in the next section.

The primal problem of (25) will be solved using the dual approach. From (18) and (25), the corresponding dual problem can be written as,

$$\min_{\bar{\mathbf{y}},\bar{\mathbf{z}},\bar{\mathbf{x}}} \max L' = \min_{\bar{\mathbf{y}},\bar{\mathbf{z}},\bar{\mathbf{x}}} \max L - \sum_{f\in F}\sum_{e\in E}\sum_{t=1}^{T}\sum_{c=1}^{N} \frac{1}{2\kappa}\left(y_{f,e}^{t,c} - x_{f,e}^{t,c}\right)^2, \quad (26)$$

subject to constraints (20) - (24).

Since the primal objective function is now strictly concave, the dual is differentiable, and the gradient of L' with respect to the different Lagrange multipliers are obtained as

$$\frac{\partial L'}{\partial p_{f,v}^1} = z_f - \mu_v^f, \quad (27)$$

$$\frac{\partial L'}{\partial p_{v,t}^2} = \sum_{f\in F}\sum_{e\in E_v^{out}} y_{f,e}^{t,c} - 1, \quad (28)$$

$$\frac{\partial L'}{\partial p_{v,t}^3} = \sum_{f\in F}\sum_{c=1}^{N}\left(\sum_{e\in E_v^{in}} y_{f,e}^{t,c} + \sum_{e\in E_v^{out}} y_{f,e}^{t,c}\right) - 1, \quad (29)$$

$$\frac{\partial L'}{\partial p_{e,t,c}^4} = \sum_{e'\in S_e} y_{f,e'}^{t,c} - 1, \quad (30)$$

$$\frac{\partial L'}{\partial p_{f,v}^5} = \lambda_v^f - \mu_v^f. \quad (31)$$

Applying the gradient projection method [48], the Lagrange multipliers are calculated iteratively as follows:

$$p_{f,v}^1(i+1) = \left[p_{f,v}^1(i) + \nu\frac{\partial L'}{\partial p_{f,v}^1}\right]^+, \quad (32)$$

where $\nu > 0$ is the gradient step size, and $[\cdot]^+$ denotes $\max(0,\cdot)$. The remaining Lagrange multipliers are obtained iteratively using similar equations.

Equating the gradient of L' with respect to $\bar{\mathbf{z}}$ to zero, results in

$$z_f(i+1) = U_f^{'-1}\left(\sum_{v\in V_f} p_{f,v}^1(i)\right), \quad (33)$$

similarly, equating the gradient of L' with respect to $\bar{\mathbf{y}}$ to zero, results in

$$y_{f,e}^{t,c}(i+1) = x_{f,e}^{t,c}(i) + \kappa\left[p_{f,v_1}^1(i)\beta_e^c - p_{v_1,t}^2(i) - p_{v_1,t}^3(i) - p_{v_2,t}^3(i) - \sum_{e'\in S_e} p_{e',t,c}^4(i) + p_{f,v_1}^5(i)\beta_e^c\right], \quad (34)$$

where v_1 is the source node of edge e, and v_2 is its destination node. And $\beta_e^c = \left[I_e^c\alpha_c + (1-I_e^c)\right](1-P_e^{out})$

4.3 Distributed Implementation

Here we describe how the proximal approximation algorithm can be implemented in a real network in a distributed way. For each traffic stream in the network, we assume that single path routes from source to destination is already established through any routing protocol. Therefore, at the start of the resource allocation protocol, each mesh node has the knowledge of

1) the routes it is part of,
2) the previous node in each of those routes, and
3) the next node along each of those routes.

In addition, each node will need to identify all the nodes within its interference range. This can be achieved through the use of a simple HELLO protocol to construct the contention domains [49].

At the i^{th} iteration, any node v updates the Lagrange multipliers associated with itself and with all outgoing links emanating from it according to (32) and using the gradients in (27) to (31). Then, using the updated multipliers, node v calculates the resources allocated to each of its outgoing links according to (34). Investigating the Lagrange multipliers updates (27) through (31), and the resource allocation solution (34), it can be noted that, node v needs to acquire the following information in order to perform its calculations;

1) resource allocation solutions ($y_{f,e}^{t,c}$) at all nodes within interference range (used in the update of $p_{e,t,c}^4$, as seen from (31)),
2) the value $p_{v_2,t}^3$ from the next node along each route node v is part of (used for the calculation of $y_{f,e}^{t,c}$ in (34)),
3) the value z_f for all streams passing through node v (used to update $p_{f,v}^1$).

From (33), it is seen that the calculation of z_f for each data stream requires the knowledge of $p^1_{f,v}$ from all nodes along stream's f route. Therefore, this calculation will take place at the source node, where the values $p^1_{f,v}$ from all nodes along the route are gathered at each iteration.

In order for the calculations and updates discussed above to take place, the required information need to be communicated between nodes. At each iteration, information exchange between nodes can be classified into 4 categories as follows;

1) Each node broadcasts its resource allocation solutions to all nodes within interference range. Depending on the transmission parameters, the interference range may contain nodes that are several hops away from the originating node.
2) The last node in any given route forwards its $p^1_{f,v}$ value to the next node on the backward path towards the source node. Subsequent nodes on that backward path add their own $p^1_{f,v}$ value to the value received from preceding nodes and then forward the sum to the next node towards the source. Therefore, the value received at the source will be $\sum_{v\in V_f} p^1_{f,v}(i)$.
3) Each node forwards its $p^3_{v,t}$ value to the next node along the backward path towards the source node. Transmission of $p^3_{v,t}$ and accumulated $p^1_{f,v}$ values can be combined into a single packet transmission at each node, since both values are propagated in the backward path.
4) Once the source calculates z_f, this value is transmitted propagated along the forward path from the source towards the destination.

Finally, each node sets $x^{t,c}_{f,e} = y^{t,c}_{f,e}$ at regular intervals (each n^{th} iteration for instance).

It should be mentioned that the information exchange process between nodes can be affected by errors in the wireless channel. These errors may result in the loss of some of these information. Different methods could be used to deal with data loss during the update procedure. For instance, a node that does not receive the expected updated data may simply choose not perform any updates or calculations during the current iteration. Since other nodes are expecting updates from that affected node, it cannot simply remain silent as this will result in the whole process to stall. Instead, this node can resend the updates it sent during the previous iteration. Another approach can be to use the last correctly received data or a historical moving average of the last k correct values to do the updates and broadcast the resulting solution. In all these cases, we argue that the convergence is still attained as discussed in chapter 7 of [48]. In both cases, the system is expected to require more iterations to converge. Due to space limitations, we will not further investigate this issue, but it should be the subject of a future work.

4.4 Complexity

Calculations at each node can be summarized in the following steps,

Equ.	Complexity
(27)	$\mathcal{O}\left(\lvert E^{out}_v\rvert \times T \times N\right)$
(28)	$\mathcal{O}\left(\lvert F\rvert \times \lvert E^{out}_v\rvert\right)$
(29)	$\mathcal{O}\left(\lvert F\rvert \times N \times \left(\lvert E^{in}_v\rvert + \lvert E^{out}_v\rvert\right)\right)$
(30)	$\mathcal{O}\left(\lvert S_e\rvert\right)$
(31)	$\mathcal{O}\left(\lvert E^{out}_v\rvert \times T \times N\right)$
(32)	$\mathcal{O}\left(1\right)$
(33)	$\mathcal{O}\left(\lvert V_f\rvert\right)$
(34)	$\mathcal{O}\left(\lvert S_e\rvert\right)$

1) Calculating the gradient of the Lagrangian w.r.t each Lagrange multiplier, as per (27) to (31),
2) Updating the Lagrange multipliers as in (32),
3) Getting resource allocation variables using (33) and (34).

The gradient calculation in (27) and (31) involves the calculation of μ^f_v using (3) and (4). This calculation can be seen to have a complexity of $\mathcal{O}(|E^{out}_v| \times T \times N)$ as it involves the summation over all the nodes outgoing edges and all possible resource elements (t, c) that each edge can use.

Getting the value of z_f in (33) involves the inverse of the gradient utility function. If a logarithmic utility function on the form $U_f(z) = \log(z)$ is used, then (33) becomes,

$$z_f(i+1) = \frac{1}{\left(\sum_{v\in V_f} p^1_{f,v}(i)\right)}, \tag{35}$$

which can be seen to have a complexity of $\mathcal{O}(|V_f|)$.

The remaining calculations and updates are all on the form of summation and their complexities are summarized in the following table.

5 Results and Discussions

To evaluate the performance of the proposed algorithm, we consider a mesh network with nodes deployed uniformly on a square grid with a side length of 250m. The size of the mesh network varies between 5 and 35 nodes. Primary nodes are uniformly distributed in the same region. The number of primary transmitter-receiver pairs varies between 1 and 20. The transmission range of any node is set to $R = 100$m, and the interference range $I = 2R$. A TDMA frame has $T = 20$ time slots. 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 $N_0 = 10^{-11}$ W/Hz. The utility functions used are $U_f(z) = \log(z)$ which impose proportional fairness amongst the traffic streams. Performance of the proposed resource allocation algorithm is compared with a simple baseline algorithm in which the available resources are uniformly distributed over the different used links in the network. This uniform resource allocation scheme is subject to the same constraints as our optimization based scheme. Furthermore, the performance is compared with the max-min bandwidth allocation (MMBA) algorithm presented in [19]. The MMBA algorithm maximizes the sum of the throughput of all traffic streams in the network while achieving max-min fairness among them. The optimization variables in the MMBA algorithm are the bandwidth allocated to each link and the fraction of time a given link

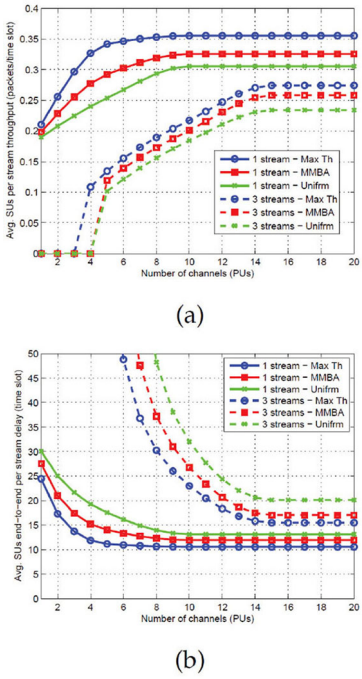


Fig. 4. Effect of the number of primary channels on (a) average secondary throughput, and (b) end-to-end delay.

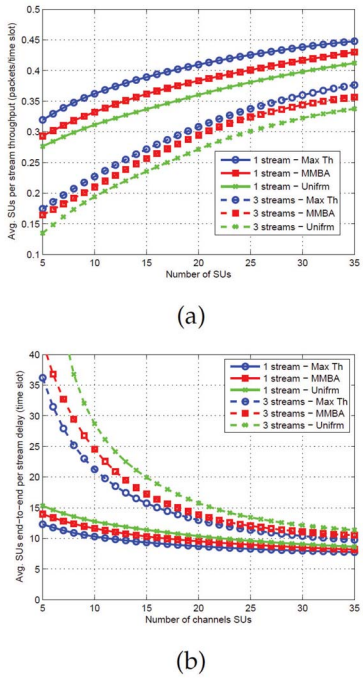


Fig. 5. Effect of the number of secondary nodes on average secondary (a) throughput, and (b) and-to-end delay.

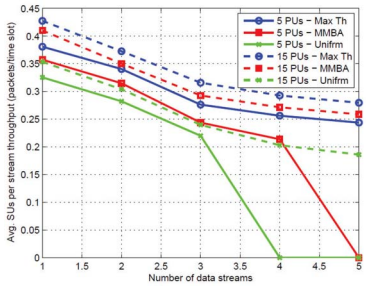
is active. The solution is obtained by solving two linear programs. It is noted that this algorithm is centralized. To average out the effect of random channels, random deployment of primary nodes, and random selection of the source and destination of the different traffic streams, each scenario is repeated 25 times and the results averaged over these runs.

In Fig. 4(a), we study the effect of the number of primary channels available on the average achievable secondary throughput. The network in this case has 10 secondary nodes and has either 1 or 3 traffic streams. It is noted that the proposed resource allocation algorithm outperforms both the MMBA and the uniform allocation schemes by 9% and 16%, respectively, in the case of 1 traffic stream, and 6% and 17% in the case of 3 traffic streams. Moreover, to accommodate the 3 traffic streams, the MMBA and uniform allocation schemes require 5 channels, while the proposed schemes requires only 4. Finally, it is noted that after a certain point, the system does not benefit from any additional primary channels available. For instance, with 1 stream all schemes cannot benefit from more than 15 channels, and with 3 streams they cannot benefit from more than 15 channels. That's because of the single channel per time slot constraint. Once all the links start using the maximum number of time slots possible, they cannot benefit from any additional channel resources.

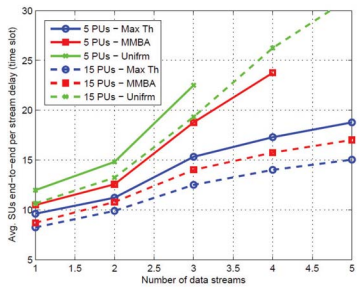
The end-to-end delay performance as a function of the number of primary channels is depicted in Fig. 4(b). The performance gain of the proposed resource allocation algorithm over the MMBA and uniform allocation algorithms is clear. For instance, there is a 20% decrease in the average delay for the proposed algorithm compared to uniform allocation, and 12% decrease compared to MMBA. Furthermore, the increase in delay due to the admission of additional traffic streams into the network is much less with our algorithm. For example, moving from a network with a single traffic stream and accepting two additional streams increases the per stream end-to-end delay by 40% with our algorithm compared to 47% with MMBA and 53% with uniform allocation. Finally, similar to the throughput performance, we note that after a given point the average delay per stream does not benefit from the increase in the number of available channels.

Fig. 5(a) and (b) depict the effect of the number of secondary mesh nodes on the per stream throughput and end-to-end delay, respectively. In this case, the number of available primary channels is fixed at 10. The important point to note here is that the network performance (in terms of throughput and delay) benefits from the increase in the number of mesh nodes. To interpret this observation, we note that, as the number of nodes increases, the node density increases, and hence nodes are closer together. Therefore, links tend to be shorter in length, and paths tend to have fewer hops. Where shorter links result in lower outage probability, and fewer hops result in more resources being allocated to each hop. On the other hand, the denser network means that each active link will interfere with more links. However, in our case, the performance increase outweigh the negative effect of this increased interference.

The effect of the number of traffic streams is shown in Fig. 6. Here the network has 25 secondary nodes and 5 or 15 primary channels. It is clear that the proposed scheme outperforms both the MMBA and uniform allocation schemes in all cases. As the number of traffic streams in the network increases, the share of each stream from the available resources decrease. Therefore, the achievable throughput per stream decreases and the end-to-end delay increases as shown in the figures. It is also noted that our proposed scheme can manage the available resources better than the

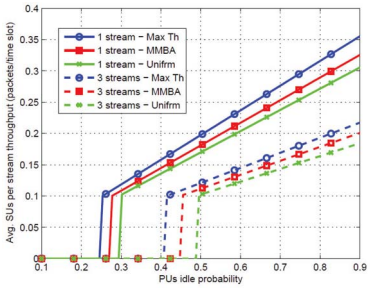


(a)

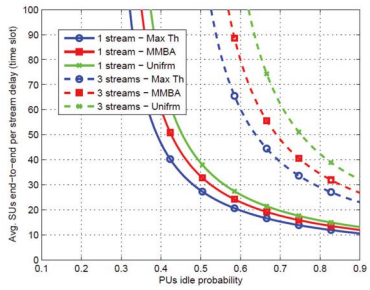


(b)

Fig. 6. Effect of the number of traffic streams on average secondary (a) throughput, and (b) and-to-end delay.



(a)

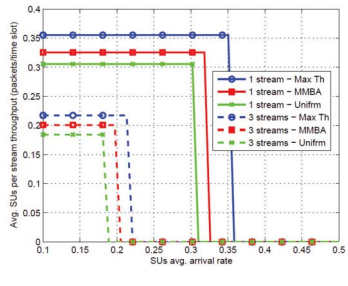


(b)

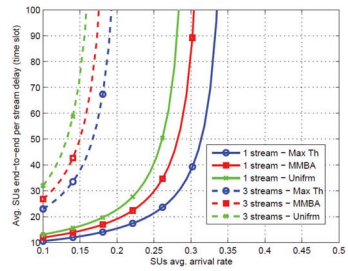
Fig. 7. Effect of the primary idle probability on average secondary (a) throughput, and (b) and-to-end delay.

other two. For instance, with 5 channels our scheme is able to support all the 5 traffic streams, while the MMBA scheme can support 4 streams and the uniform scheme supports only 3 streams.

Fig. 7 shows the effect of the primary channels idle probability on the secondary throughput and end-to-end delay, respectively. Here the network has 10 primary channels and 10 secondary nodes. It can be seen that there exists



(a)



(b)

Fig. 8. Effect of the secondary arrival rate on average secondary (a) throughput, and (b) and-to-end delay.

a lower bound on the values of the primary channels' probability below which the network cannot support the traffic demand of the secondary users. This lower bound depends on the number of available channels, with more channels available, the probability of finding an idle channel at any given time slot increases, therefore, the bound's value decreases. As shown in the figures, the bound's value increases with the number of traffic streams. Since as the number of streams increases, more resources are required to support them, which requires that the channels need to be available with higher probability. It is also observed that the lower idle probabilities result in higher end-to-end delay, since packets have to wait longer to find an idle channel. Finally, it is clear from the figures that our proposed scheme is able to support the secondary nodes traffic at lower primary channels idle probability compared to the two other schemes, and that as the number of streams increases, the difference between the required lower bounds for the different schemes also increases.

Fig. 8 shows the effect of the secondary packets arrival rate on the secondary throughput and end-to-end delay, respectively. As expected, the delay increases as the packets arrival rate increases. On the other hand, the maximum achievable throughput is independent from the secondary arrival rate, as it only depends on the primary idle probability and the physical channel conditions. It can be seen that when the secondary arrival rate exceeds the maximum achievable throughput, the network becomes unable to support the traffic load while maintaining the stability of the different queues and the problem becomes infeasible. That is why the throughput drops to zero after some point. It is clear that our proposed scheme results in a higher maximum achievable throughput, and therefore, is able to support higher traffic load compared to the MMBA and uniform allocation schemes.

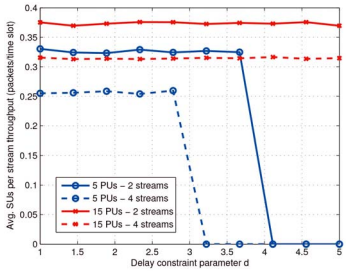
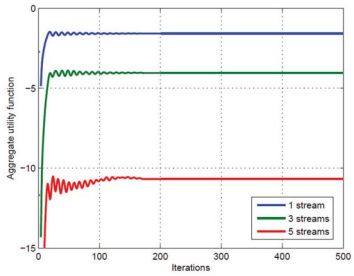
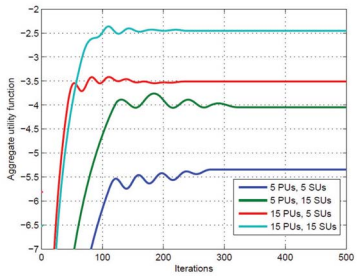


Fig. 9. Effect of the delay constraint parameter d on the achievable secondary throughput.



(a)



(b)

Fig. 10. Convergence behavior of the distributed algorithm, (a) with different number of streams, and (b) with different number of PUs and SUs.

The effect of the delay constraint parameter d on the network's performance is depicted in Fig. 9, where the parameter d takes values between 1 and 5. A value of $d = 1$ corresponds to no delay constraint, and a value of $d = 5$ corresponds to a delay constraint on the form $D < 2$ time slots. It is noted that the delay constraint is not an end-to-end constraint, but is applied on a link by link basis. The network here has 15 secondary nodes. It is noted that the achievable throughput by the secondary nodes is almost unaffected by the delay constraint as long as the available spectrum resources are enough to satisfy the delay constraint. However, when the delay constraint becomes more stringent there is a point where the available resources are not enough to satisfy it and the problem becomes infeasible and unable to converge to a useful solution. It is seen that by increasing the number of available channels, a more stringent delay constraint can be supported by the network.

Fig. 10 depicts the convergence behavior of our proposed resource allocation algorithm. The convergence behavior for the case of 1, 3, and 5 traffic streams is shown in 10(a). It can be seen that with a single stream, the algorithm converges in about 150 iterations. Here it is noted that, the number of streams has almost insignificant effect when it comes to convergence time. This is mainly because in our algorithm the resource allocation calculations for each stream are done in isolation from the remaining streams in the network. From Fig. 10(b), it can be noted that when the number of primary channels increases from 5 to 15 more degrees of freedom exists in the network, however, convergence is faster than the case with only 5 primary channels. This can be resorted to the fact that we are using the same step size in all the experiments. Adapting the step size to the system's parameters is expected to result in better convergence for the case of 5 PUs. The effect of the number of mesh nodes on the convergence was also studied. The results reveal that the number of nodes has no effect on the convergence speed, this is mainly because all the nodes perform their calculations in parallel.

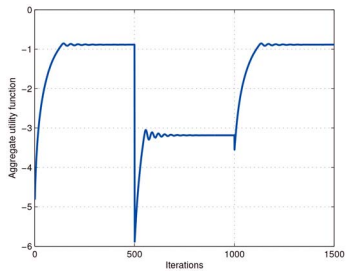


Fig. 11. Network behavior in the case of time varying number of admitted streams.

Finally, the impact of a dynamic load on the network is studied in Fig. 11. The network starts with a single stream and converges in around 250 iterations. After 500 iterations, a second stream is added. The aggregate utility initially drops since the utility of the newly added stream is very low (no resources assigned yet). After about 250 iterations the network converges to the optimal solution, distributing the resources between the two streams. At the 1000^{th} iteration, one of the two streams leaves the network. Here we see that the network adapts to the new situation and reassigns all of the freed resources to the remaining stream leading to an increase in its utility.

6 Conclusion

In this paper, the throughput maximization problem in cognitive radio based WMNs is formulated as a utility maximization problem. The utility function used is a function of the minimum service rate along that stream's end-to-end path, which provides a degree of fairness among different streams. Furthermore, the maximization problem formulation allows the incorporation of different end-to-end delay constraints. The centralized network wide resource allocation problem was decomposed into a set of subproblems that can be locally solved. An efficient and scalable decentralized solution protocol was proposed. Results demonstrate the efficiency of the proposed decentralized solution scheme, and its ability to adapt to varying network loads. Performance gains of the proposed protocol in comparison with uniform resource allocation and max-min bandwidth allocation are demonstrated. It was shown that, for a given amount of resources, the proposed protocol can

accommodate more traffic streams. Moreover, it can achieve up to 17% increase in throughput and 20% decrease in delay.

ACKNOWLEDGMENTS

This work was made possible by NPRP grant 08-374-2-144 and 4-1034-2-385 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors. This work was done while A.A. El-Sherif was with the Computer Science and Engineering Department, Qatar University, Doha, Qatar.

REFERENCES

[1] S. Haykin, "Cognitive radio: Brain-empowered wireless communications," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 2, pp. 201–220, Feb. 2005.

[2] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, "Next generation/dynamic spectrum access/cognitive radio wireless networks: A survey," *Comput. Netw.*, vol. 50, no. 13, pp. 2127–2159, Sep. 2006.

[3] R. D. Yates, R. S. Raman, and N. B. Mandayam, "Fair and efficient scheduling variable rate links via a spectrum server," in *Proc. IEEE ICC*, Istanbul, Turkey, 2006, pp. 5246–5251.

[4] O. Ileri, "Demand responsive pricing and competitive spectrum allocation via a spectrum server," in *Proc. IEEE DySPAN*, Baltimore, MD, USA, Nov. 2005, pp. 194–202.

[5] S. A. Zekavat and X. Li, "User-central wireless system: Ultimate dynamic channel allocation," in *Proc. IEEE DySPAN*, Baltimore, MD, USA, Nov. 2005, pp. 82–87.

[6] J. Zhao, H. Zheng, and G.-H. Yang, "Distributed coordination in dynamic spectrum allocation networks," in *Proc. IEEE DySPAN*, Baltimore, MD, USA, Nov. 2005, pp. 259–268.

[7] L. Cao and H. Zheng, "Distributed spectrum allocation via local bargaining," in *Proc. IEEE Conf. SECON*, Santa Clara, CA, USA, Sep. 2005, pp. 475–486.

[8] Y. Wu and D. H. K. Tsang, "Distributed power allocation algorithm for spectrum sharing cognitive radio networks with QoS guarantee," in *Proc. IEEE INFOCOM*, Rio de Janeiro, Brazil, Apr. 2009, pp. 981–989.

[9] R. Hincapie *et al.*, "Efficient recovery algorithms for wireless mesh networks with cognitive radios," in *Proc. IEEE ICC*, Dresden, Germany, Jun. 2009, pp. 1–5.

[10] K. R. Chowdhury and I. F. Akyildiz, "Cognitive wireless mesh networks with dynamic spectrum access," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 1, pp. 168–181, Jan. 2008.

[11] T. Chen, H. Zhang, G. M. Maggio, and I. Chlamtac, "Cogmesh: A cluster-based cognitive radio network," in *Proc. IEEE DySPAN*, Dublin, Ireland, Apr. 2007, pp. 168–178.

[12] R. C. Pereira, R. D. Souza, and M. E. Pellenz, "Using cognitive radio for improving the capacity of wireless mesh networks," in *Proc. IEEE 68th VTC*, Calgary, AB, Canada, Sep. 2008, pp. 1–5.

[13] H. M. Almasaeid and A. E. Kamal, "Receiver-based channel allocation for wireless cognitive radio mesh networks," in *Proc. IEEE DySPAN*, Singapore, Apr. 2010, pp. 1–10.

[14] R. J. Berger, "Open spectrum: A path to ubiquitous connectivity," *Queue*, vol. 1, no. 3, pp. 60–68, May 2003.

[15] D. Niyato and E. Hossain, "Cognitive radio for next-generation wireless networks: An approach to opportunistic channel selection in IEEE 802.11-based wireless mesh," *IEEE Wireless Commun.*, vol. 16, no. 1, pp. 46–54, Feb. 2009.

[16] M. E. Steenstrup, "Opportunistic use of radio-frequency spectrum: A network perspective," in *Proc. IEEE DySPAN*, Baltimore, MD, USA, Nov. 2005, pp. 638–641.

[17] T. Chen, H. Zhang, G. M. Maggio, and I. Chlamtac, "Topology management in CogMesh: A cluster-based cognitive radio mesh network," in *Proc. IEEE ICC*, Glasgow, U.K., Jun. 2007, pp. 6516–6521.

[18] M. Sharma, A. Sahoo, and K. D. Nayak, "Channel selection under interference temperature model in multi-hop cognitive mesh networks," in *Proc. IEEE DySPAN*, Dublin, Ireland, Apr. 2007, pp. 133–136.

[19] J. Tang, R. Hincapie, G. Xue, W. Zhang, and R. Bustamante, "Fair bandwidth allocation in wireless mesh networks with cognitive radios," *IEEE Trans. Veh. Technol.*, vol. 59, no. 3, pp. 1487–1496, Mar. 2010.

[20] C. Xin, B. Xie, and C.-C. Shen, "A novel layered graph model for topology formation and routing in dynamic spectrum access networks," in *Proc. IEEE DySPAN*, Baltimore, MD, USA, Nov. 2005, pp. 308–317.

[21] A. A. El-Sherif, A. Mohamed, and Y. C. Hu, "Joint routing and resource allocation for delay sensitive traffic in cognitive mesh networks," in *Proc. IEEE Globecom Workshop RACCN*, Houston, TX, USA, Dec. 2011, pp. 947–952.

[22] F. P. Kelly, A. Maulloo, and D. Tan, "Rate control for communication networks: Shadow prices, proportional fairness and stability," *J. Operat. Res. Society*, vol. 49, no. 3, pp. 237–252, 1998.

[23] D. Cabric, S. M. Mishra, and R. W. Brodersen, "Implementation issues in spectrum ensing for cognitive radio," in *Proc. Asilomar Conf. Signals, System, Computer*, Pacific Grove, CA, USA, 2004, pp. 772–776.

[24] S. Enserink and D. Cochran, "A cyclostationary feature detector," in *Proc. 28th Asilomar Conf. Signals, Systems, and Computers*, Pacific Grove, CA, USA, Oct. 1994, pp. 806–810.

[25] A. Ghasemi and E. Sousa, "Collaborative spectrum sensing for opportunistic access in fading environments," in *Proc. 1st IEEE Symp. New Frontiers in Dynamic Spectrum Access Networks*, Baltimore, MD, USA, Nov. 2005, pp. 131–136.

[26] S. M. Mishra, A. Sahai, and R. W. Brodersen, "Cooperative sensing among cognitive radio," in *Proc. IEEE ICC*, Istanbul, Turkey, Jun. 2006, pp. 1658–1663.

[27] A. K. Sadek, K. J. R. Liu, and A. Epheremides, "Cognitive multiple access via cooperation: Protocol design and performance analysis," *IEEE Trans. Inf. Theory*, vol. 53, no. 10, pp. 3677–3696, Oct. 2007.

[28] A. A. El-Sherif, A. K. Sadek, and K. J. R. Liu, "Opportunistic multiple access for cognitive radio networks," *IEEE J. Select. Areas Commun.*, vol. 29, no. 4, pp. 704–715, Apr. 2011.

[29] H. Su and X. Zhang, "Cross-layer based opportunistic MAC protocols for QoS provisionings over cognitive radio wireless networks," *IEEE J. Select. Areas Commun.*, vol. 26, no. 1, pp. 118–129, Jan. 2008.

[30] K. Haghighi, E. G. Strom, and E. Agrell, "On optimum causal cognitive spectrum reutilization strategy," *IEEE J. Select. Areas Commun.*, vol. 30, no. 10, pp. 1911–1921, Nov. 2012.

[31] M. Shi, Y. Bar-Ness, and W. Su, "Blind OFDM systems parameters estimation for software defined radio," in *Proc. 2nd IEEE Int. Symp. New Frontiers DySPAN*, Dublin, Ireland, Apr. 2007, pp. 119–122.

[32] T. Yucek and H. Arslan, "Ofdm signal identification and transmission parameter estimation for cognitive radio applications," in *Proc. IEEE GLOBECOM*, Washington, DC, USA, Nov. 2007, pp. 4056–4060.

[33] J. G. Liu, X. Wang, and J.-Y. Chouinard, "Iterative blind OFDM parameter estimation and synchronization for cognitive radio systems," in *Proc. IEEE 75th VTC*, Yokohama, Japan, May 2012, pp. 1–5.

[34] P. D. Sutton, K. E. Nolan, and L. E. Doyle, "Cyclostationary signatures in practical cognitive radio applications," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 1, pp. 13–24, Jan. 2008.

[35] O. Simeone, Y. Bar-Ness, and U. Spagnolini, "Stable throughput of cognitive radios with and without relaying capability," *IEEE Trans. Commun.*, vol. 55, no. 12, pp. 2351–2360, Dec. 2007.

[36] S. Kompella, G. D. Nguyen, J. E. Wieselthier, and A. Ephremides, "Stable throughput tradeoffs in cognitive shared channels with cooperative relaying," in *Proc. IEEE INFOCOM*, Shanghai, China, Apr. 2011, pp. 1961–1969.

[37] R. A. Tannious and A. Nosratinia, "Cognitive radio protocols based on exploiting hybrid ARQ retransmissions," *IEEE Trans. Wireless Commun.*, vol. 9, no. 9, pp. 2833–2841, Sep. 2010.

[38] M. Levorato, U. Mitra, and M. Zorzi, "Cognitive interference management in retransmission-based wireless networks," *IEEE Trans. Inf. Theory*, vol. 58, no. 5, pp. 3023–3046, May 2012.

[39] 3GPP. (2012, Jul.). *Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol Specification*, TS 36.331 V11.0.0, Rel-11 [Online]. Available: http://www.3gpp.org/ftp/Specs/archive/36_series/36.331/36331-b00.zip

[40] A. A. El-Sherif and K. J. R. Liu, "Joint design of spectrum sensing and channel access in cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 10, no. 6, pp. 1743–1753, Jun. 2011.
[41] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 338–404, Mar. 2000.
[42] J. G. Proakis, *Digital Communications*, New York, USA: McGraw-Hill Inc., 1994, ISBN: 0072957166.
[43] G. R. Grimmett and D. R. Stirzaker, *Probability and Random Processes*. Oxford, NY, USA: Oxford University Press, 2001.
[44] D. Bertsekas and R. Gallager, *Data Networks*, 2nd ed. Englewood Cliffs, NJ, USA: Prentice Hall, 1992.
[45] R. W. Wolff, *Stochastic Modeling and The Theory of Queues*. Englewood Cliffs, NJ, USA: Prentice Hall, 1989.
[46] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge University Press, 2004.
[47] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA, USA: Athena Scientific, 2003.
[48] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Upper Saddle River, NJ, USA: Athena Scientific, 1989.
[49] A. Mohamed and H. Alnuweiri, "Utility-based optimal rate allocation for heterogeneous wireless multicast," in *Proc. IEEE ICC*, Glasgow, Scotland, Jun. 2007, pp. 3463–3470.

Amr A. El-Sherif (S'00, M'08) received his B.Sc. (with highest Honors) and M.Sc. degrees in electrical engineering form Alexandria University, Alexandria, Egypt, in 2002 and 2005, respectively. He received his Ph.D. degree in electrical engineering from the University of Maryland, College Park, in 2009. He is currently an Assistant Professor in the Electrical Engineering Department at Alexandria University, Egypt. His research interests include cooperative communications and networking, cross-layer design for wireless networks, multiple access technologies for wireless and sensor networks, and spectrum sharing and cognitive radio systems. He is a member of the IEEE.

Amr Mohamed (S'00, M'06) received his M.S. and Ph.D. in electrical and computer engineering from the University of British Columbia, Vancouver, Canada, in 2001, and 2006 respectively. He has worked as an advisory IT specialist in IBM Innovation Centre in Vancouver from 1998 to 2007, taking a leadership role in systems development for vertical industries. He is currently an assistant professor in the college of engineering at Qatar University and the director of the Cisco Regional Academy. He has over 20 years of experience in wireless networking research and industrial systems development. He holds 3 awards from IBM Canada for his achievements and leadership, and 3 best paper awards. His research interests include networking and MAC layer techniques mainly in wireless networks. Dr. Mohamed has authored or co-authored over 40 refereed journal and conference papers and one textbook. He is a member of the IEEE.