January 2014

Delay aware joint routing and scheduling for multi-Tx-Rx Wireless Mesh Networks

Luyao Wang
University of Wollongong, lw233@uowmail.edu.au

Kwan-Wu Chin
University of Wollongong, kwanwu@uow.edu.au

Raad Raad
University of Wollongong, raad@uow.edu.au

Sieteng Soh
Curtin University of Technology, s.soh@curtin.edu.au

Delay aware joint routing and scheduling for multi-Tx-Rx Wireless Mesh Networks

Abstract

Recently, researchers have created Wireless Mesh Networks (WMNs) where routers have multiple transmit (Tx) or receive (Rx) capability. A fundamental problem in such WMNs is deriving a transmission schedule that yields minimal end-to-end delays. In this paper, we approach this problem via joint routing and link scheduling. Specifically, we consider two fundamental issues that influence end-to-end delays: superframe length and transmission slot order. We propose two algorithms: JRS-Multi-DEC and JRS-BIP, where the former uses a novel metric to minimize the load of each link whilst the latter uses a binary integer program solver. Both algorithms have the similar aim of minimizing overall delay and to re-order slots such that packets are forwarded quickly along their path. Numerical results show that our algorithms can reduce average delay by approximately 50% as compared to a non joint routing and scheduling algorithm.

Keywords

mesh, wireless, rx, networks, tx, delay, multi, scheduling, routing, joint, aware

Publication Details

L. Wang, K. Chin, R. Raad & S. Soh, "Delay aware joint routing and scheduling for multi-Tx-Rx Wireless Mesh Networks," in IEEE International Conference on Communications (ICC), 2014, pp. 2773-2778.

Delay Aware Joint Routing and Scheduling for Multi-Tx-Rx Wireless Mesh Networks

Luyao Wang, Kwan-Wu Chin, Raad Raad
School of Electrical, Computer and Telecommunications Engineering
University of Wollongong, NSW, Australia
Email: lw233@uowmail.edu.au, {kwanwu, raad}@uow.edu.au

Sieteng Soh
Department of Computing
Curtin University of Technology, WA, Australia
Email: s.soh@curtin.edu.au

***Abstract*—Recently, researchers have created Wireless Mesh Networks (WMNs) where routers have multiple transmit (Tx) or receive (Rx) capability. A fundamental problem in such WMNs is deriving a transmission schedule that yields minimal end-to-end delays. In this paper, we approach this problem via joint routing and link scheduling. Specifically, we consider two fundamental issues that influence end-to-end delays: superframe length and transmission slot order. We propose two algorithms: JRS-Multi-DEC and JRS-BIP, where the former uses a novel metric to minimize the load of each link whilst the latter uses a binary integer program solver. Both algorithms have the similar aim of minimizing overall delay and to re-order slots such that packets are forwarded quickly along their path. Numerical results show that our algorithms can reduce average delay by approximately 50% as compared to a non joint routing and scheduling algorithm.**

I. Introduction

A promising approach to improve the network capacity of Wireless Mesh Networks (WMNs) is to equip routers with multi transmit (Tx) or receive (Rx) capability. An example Multi-Tx-Rx (MTR) WMN is proposed in [14], where Raman et al. created a low cost, long distance WMN using off-the-shelf IEEE 802.11 hardware and parabolic antennas. Other examples include [18] and [6] where nodes have one or more beam-forming antennas. A key problem in MTR WMNs is link scheduling. For example, in [14], the authors proposed a Spatial reuse Time Division Multiple Access (STDMA) scheduling protocol, called 2P. Nodes operate in one of two phases: Synchronous Transmitting (SynTx) or Synchronous Receiving (SynRx). This means when a node is in the SynTx phase, it is transmitting on all links, and vice-versa if it is in the SynRx phase. A key limitation of 2P is that it requires a WMN to be bipartite. Consequently, the authors of [5] proposed a novel, fast, greedy link scheduling algorithm, called Algo-1, that operates on random topologies. One key limitation of 2P and Algo-1 is that nodes transmitting in slot i become a receiver in slot $i+1$. This feature leads to a longer superframe length. To this end, Loo et al. [13] outline an algorithm called Algo-2 that improves 2P and Algo-1 by maximizing link activations on a slot-by-slot basis.

End-to-end delay is of paramount concern for real-time applications such as audio and video. However, guaranteeing end-to-end delays in WMNs is a challenging problem. In particular, it is affected by the superframe length and slot order within the derived superframe [7]. In particular, at each intermediate node, a packet has to wait for its out-going link's time slot to occur [4]. When there is only one traffic demand, i.e., traffic from one source to one destination, routing the traffic via its shortest path gives the minimum end-to-end delay, with slots placed consecutively. However, if there are multiple traffic demands, routing them through their shortest path may lead to a bottleneck link.

To date, as we outlined in Section II, past works have either focused on maximizing network throughput in MTR WMNs [9], [12], [17], or only considered minimizing end-to-end delay in WMNs with omni-directional antenna [1], [3], [4], [11], [16]. However, no one has proposed solutions that minimize end-to-end delay through joint routing and scheduling in MTR WMNs. To this end, we propose two joint routing and scheduling algorithms: JRS-Multi-DEC and JRS-BIP. We show that these algorithms are able to reduce the average end-to-end delay as compared to a non joint routing algorithm.

This paper has the following structure. Section II discusses previous works. Our network model is described in Section III followed by a description of the problem at hand in Section IV. Our solutions are outlined in Section V. In Section VI, we present our experiment results. Lastly, our conclusions are presented in Section VII.

II. Related Work

This section focuses on works that address the joint routing and scheduling (JRS) problem in WMNs. Kodialam et al. [12] aim to compute the maximum flow rate of a WMN, where nodes have multiple radios, each tuned to a different channel. They derive necessary and sufficient conditions for the achievability of a link flow vector. In a different work, based on the work of [12] and [14], Dutta et al. consider the maximum concurrent flow problem [9] in MTR WMNs, and propose a joint routing and scheduling scheme. They first use a Linear Program (LP) to obtain the maximum concurrent flow of each demand. As the LP is intractable due to the exponential number of concurrent transmission groups, they propose a novel scheduling algorithm, called Multi-DEC. They first derive a multi-graph by duplicating links as per their weight. Next, they split the resulting multi-graph into several simple sub-graphs, which are then colored using Directed Edge Coloring (DEC) algorithm [8]. In a different work, the problem of maximizing network throughput in 60 GHz WMNs subject to half-duplex and radio constraints is addressed in [17] through joint routing, link scheduling and channel assignment. None of these works, however, address the problem of minimizing end-to-end delay through joint routing and scheduling.

To date, there has only been a handful of works that consider end-to-end delay. They include [1]–[4], [7], [11], [16] and [10]. The authors of [10] aim to construct an interference aware routing tree that leads to the minimum schedule length. In [7], the authors showed that scheduling delay occurs when an outbound link is scheduled before an inbound link. As a result, scheduling delay accumulates at every hop on a path, which leads to large end-to-end delays. They propose a polynomial time algorithm that uses the Bellman-Ford algorithm to find the activation time of each link and ensures two conflicting links do not have overlapping activation time. The problem of computing a conflict-free schedule that guarantees all end-to-end delays are within a given bound is addressed in [2]. The authors propose an iterative approach that separately considers scheduling and queuing delays. In summary, these works have several limitations. The authors of [2] and [7] only focus on the MAC layer. The authors did not propose a routing algorithm to establish paths. Moreover, they assume a tree topology. In [10], the authors propose an interference aware routing algorithm. However, their focus is on the network layer.

We note that there are a number of works that consider joint routing and scheduling in order to minimize end-to-end delays, but they are targeted at WMNs with omni-directional antennas. For example, the authors of [4] consider joint routing and scheduling in order to minimize end-to-end delays. They propose two cross-layer design schemes to minimize the number of hops and interference level along a path: (i) a loosely coupled cross-layer formulation, and (ii) a tightly coupled formulation. The approach in [3] aims to yield a low interference solution by determining a routing path for all demands, slot position and transmission order of each link within a superframe such that each route's worst-case end-to-end delay and flows' deadline are minimized. Similar to [3], the problem of finding a link activation pattern, which delivers packets to the gateway within a deadline through joint routing and scheduling, is addressed in [1]. The problem of guaranteeing end-to-end QoS is studied in [11] and [16]. In addition, the problem of maximizing network throughput is jointly considered in [11]. The authors of [11] aim to schedule the maximum number of voice calls while satisfying the calls delay constraint. The problem at hand involves choosing an appropriate path (routing), channel and slot assignment with the constraint that the resulting end-to-end delay is within a given bound. Shetiya et al. [16] consider the problem of routing and scheduling of real and non real time traffic. Their work is essentially a slot reservation protocol, which is similar to [11].

Our work is similar to [9], but we aim to address the problem of minimizing end-to-end delay, whereas they only consider the maximum flow problem. Moreover, we consider integral demands. Apart from that, we consider the transmission order of links; similar to [7] but in the context of MTR WMNs. To the best of our knowledge, this is the first paper that addresses the problem of minimizing end-to-end delay in MTR WMNs through joint routing and link scheduling.

III. Preliminaries

In this paper, we consider a single channel, TDMA-based MTR WMNs. We assume time is divided into time slots, and each slot is sufficient to transmit a single packet. A superframe is comprised of a number of slots, whereby each link is assigned one or more slots.

Consider an arbitrary graph $G(V, E)$, where V denotes the set of nodes and E represents the set of directed links between nodes. Each router $i \in V$ is equipped with $\mathcal{A}_i \geq 1$ antennas. Let e_{ij} or $(i, j) \in E$ denote a directional link from router i to j. We define $\mathcal{D}$ as the set of demands d between source r and destination s, each of which has a weight R_d (in slots). The set of available paths that can be used to route demand d is denoted as P_d. Each path $p_d \in P_d$ is comprised of a set of links; i.e., $p_d \subseteq E$. We will refer to each path, indexed by k, in P_d as p_d^k, where $k \in [1, |P_d|]$. Also, H_d^k denotes the number of hops of the k^{th} path in P_d. Let S indicate the resulting superframe with length $|S|$.

According to the Directed Edge Coloring (DEC) algorithm proposed in [8], if an undirected simple graph can be colored with x colors, the corresponding directed graph can be colored with $\xi(x)$ colors, where $\xi(x)$ is the minimum n that satisfies $\binom{n}{\lfloor \frac{n}{2} \rfloor} \geq x$. The coloring result is a feasible link schedule and the links with the same color are assigned to the same slot. In order to schedule weighted links, the authors of [9] propose Multi-DEC, which transforms the weight of a link into multiple parallel links to create a multi-graph. Then, the idea is to split the multi-graph into several simple sub-graphs and color each of them using DEC. To compute the chromatic index of the multi-graph, it first finds the heaviest loaded node that has the highest weight, denoted as W_m, which is the sum of transmitting and receiving link weight. Therefore, the value $\lfloor \frac{\xi(x) \times W_m}{2} \rfloor$ is the minimum number of colors used to color the multi-graph.

IV. Problem Definition

Given a MTR WMN, and a set of demands, we aim to derive the minimal superframe length that results in the smallest end-to-end delays for all demands. The end-to-end delay is affected by the superframe length and ordering of allocated slots. The superframe length is determined by the topology induced by paths and the total link load. Consider the example topology in Figure 1. We assume there is only one traffic demand $d = 1$ from node a to g and requires $R_1 = 1$ slot. We can construct two possible paths, $p_1^1 = \{e_{ah}, e_{hg}\}$ and $p_1^2 = \{e_{ah}, e_{he}, e_{eg}\}$. We observe that p_1^2 has one more hop than p_1^1. We assume the links along each path can be scheduled in a round robin fashion as shown in Table I. Then, the superframe length and end-to-end delay of the said paths are two and three slots respectively. Moreover, if there are multiple demands, always choosing the shortest path for each demand may lead to bottleneck links. Consequently, these links need to be assigned multiple time slots, which increases end-to-end delay. Using Figure 1, we consider the following traffic demands: $d = 1$ from node a to g, $d = 2$ from node b to g and $d = 3$ that from node c to g. If these three flows respectively select $p_1^1 = \{e_{ah}, e_{hg}\}$, $p_2^1 = \{e_{bh}, e_{hg}\}$ and $p_3^1 = \{e_{ch}, e_{hg}\}$, observe that link e_{hg} is used three times. In other words, the said link requires three time slots. The schedule resulting from the traffic flow order is shown in Table II. Links e_{ah}, e_{bh} and e_{ch} can use one slot because of MTR. The resulting superframe length is four slots. The resulting end-to-end delay of the three demands are two, three and four slots respectively.

p_1^1:

Slot 1	Slot 2
e_{ah}	e_{hg}

p_1^2:

Slot 1	Slot 2	Slot 3
e_{ah}	e_{he}	e_{eg}

TABLE I: Link schedule for p_1^1 and p_1^2

Another important aspect of the schedule is the order of time slots. For an intermediate node, if outgoing links are activated before incoming links, queued packets have to wait for the active time slot of outgoing links. We assume that demand $d = 1$ from node a to g selects path $p_1^1 = \{e_{ah}, e_{hg}\}$. In this case, demand $d = 1$ is not the only traffic demand and the superframe length $|S|$ is four slots. If the slot containing link e_{hg} is scheduled before the slot containing link e_{ah}, as shown in Table III, the end-to-end delay is equal to the superframe length. In other words, after receiving a packet from node a in slot 3, node h has to wait for slot 2 in the next superframe to transmit the packet to node g. The resulting end-to-end delay is four slots. In summary, end-to-end delay is affected by routing policy, link load, and transmission order of links.

Henceforth, in this paper, we aim to consider the aforementioned issues via joint routing and link scheduling. Unfortunately, the problem at hand is NP-hard. In particular, deriving the minimal superframe length involves decomposing a MTR WMN into $|S|$ bipartite graphs such that each edge transmits in at least one of the $|S|$ slots. Unfortunately as shown in [13], this problem is equivalent to deriving a MAXCUT in each time slot – an NP-complete problem. In addition, for each router, there may be an exponential number of routes to a given destination. As a result, determining the combination of routes that yield the minimal superframe length becomes intractable quickly with an increasing network size.

V. The Solutions

We propose two joint routing and link scheduling solutions: JRS-Multi-DEC and JRS-BIP. In JRS-Multi-DEC, we select routing paths that result in the least colors, as per the Multi-DEC algorithm [9], and the smallest worst case delay (WCD). The latter metric ensures in the worst case each packet waits

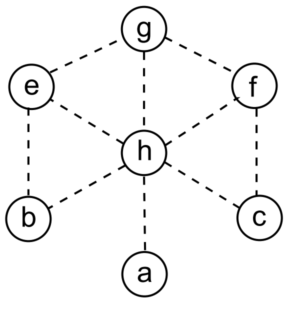


Fig. 1: An example WMN

Slot 1	Slot 2	Slot 3	Slot 4
e_{ah}	e_{hg}		
e_{bh}		e_{hg}	
e_{ch}			e_{hg}

TABLE II: An example schedule

Slot 1	Slot 2	Slot 3	Slot 4
	e_{hg}	e_{ah}	

TABLE III: One possible link schedule for path p_1^1

for one superframe length before it is transmitted to the next hop. In JRS-BIP, we formulate the problem as a Binary Integer Program (BIP) to minimize the maximal link load. The ultimate goal is to spread traffic demands widely to avoid overloading links, which helps reduce end-to-end delay. Both algorithms have two phases. In *Phase-1*, the problem is to select routing paths, and in *Phase-2*, the task is to schedule links and reorder the derived time slots to yield shorter end-to-end delays.

The key difference between these two algorithms is Phase-1. JRS-Multi-DEC starts by determining the shortest path for all demands $\mathcal{D}$ and sorts them according to hops in descending order, where $\mathcal{D}$ is the set of end-to-end demands; see *line 1-4* of Algorithm 1. It then establishes the first demand that has the shortest path. Using the topology shown in Figure 1, if the demand from a to g is the first, then JRS-Multi-DEC will establish the shortest path $p_1^1 = \{e_{ah}, e_{hg}\}$. We assume for each demand d, a set P_d containing k shortest paths is given, where the value k is big enough to find the best path. In the case of Figure 1, $k = 1$ for the first demand and $k = 2$ for the other demands. Then, for each path in P_d, it begins with the shortest one, establishes the path temporarily and uses Multi-DEC [9] to compute the minimum number of colors. Recall that this corresponds to the superframe length required to serve all established paths (or demands) and the temporarily added path; see *line 10-13* in Algorithm 1. For the demand from b to g of Figure 1, if we first construct $p_2^1 = \{e_{bh}, e_{hg}\}$, we need two colors. According to the DEC algorithm [8], we have $\xi(2) = 2$. Node h has a maximum node weight of three because its maximum incoming link weight is one (e_{ah} or e_{bh}) and maximum outgoing link weight is two (e_{hg}). Thus, the number of colors as computed by Multi-DEC in *line 13* is $(2\times3)/2 = 3$. To calculate WCD, we multiply the superframe length by the number of hops of a given path; see *line 14*. For the demand from b to g, WCD$= 3 \times 2 = 6$. After testing all k paths, we choose the path whose WCD is the smallest among the k paths; see *line 17-24*. If we select $p_2^2 = \{e_{be}, e_{eg}\}$, the chromatic index of the undirected graph is also two. Then, we have $\xi(2) = 2$. However, the maximum node weight is two; i.e., node e and h. Thus, the superframe length for path p_2^2 is $(2 \times 2)/2 = 2$. We can calculate the WCD value of p_2^2 to be $2 \times 2 = 4$. We then compare the WCD of p_2^1 and p_2^2, and select p_2^2 for the demand from b to g. For the demand from c to g, the WCD of $p_3^1 = \{e_{ch}, e_{hg}\}$ is $3 \times 2 = 6$ and the WCD of $p_3^2 = \{e_{cf}, e_{fg}\}$ is $2 \times 2 = 4$. We select p_3^2 for the demand from c to g. Upon completion of Phase-1, each demand is assigned at most one path.

We now turn our attention to JRS-BIP. Phase-1 of JRS-BIP uses a BIP solver. Let $P_{d,k}$ be a binary variable that indicates whether path k of demand d is selected to route traffic. We define $P_{d,k}^{ij}$ to be a binary variable that is set to one if path k of demand d uses link (i, j). The BIP, which takes an integer parameter $\mathbf{f}$, is as follows, Minimize

$$\sum_{d \in \mathcal{D}} \sum_{k \in P_d} P_{d,k} H_d^k \quad (1)$$

subject to:

$$f_{ij} = \sum_{d \in \mathcal{D}} \sum_{p_d^k \in P_d} R_d P_{d,k}^{ij} \le \mathbf{f}, \quad \forall (i,j) \in E \quad (2)$$

$$\sum_{p_d^k \in P_d} P_{d,k} = 1, \quad \forall d \in \mathcal{D} \quad (3)$$

$$P_{d,k}^{ij} \ge P_{d,k}, \quad \forall (i,j) \in p_d^k, p_d^k \in P_d \quad (4)$$

The objective of the aforementioned BIP is to minimize the total path cost of all demands, where path cost means the end-to-end delay incurred by a demand. In addition, the load of each link must be no bigger than $\mathbf{f}$. Note, the value of $\mathbf{f}$ can be found via binary search. In constraint (2), we compute the load of each link. If we select paths p_1^1, p_2^1 and p_3^1 as we mentioned in Section IV, the link load for e_{ah}, e_{bh} and e_{ch} is one, and e_{hg} has a load of three. If we select paths $p_1^1, p_2^2 = \{e_{be}, e_{eg}\}$ and $p_3^2 = \{e_{cf}, e_{fg}\}$, the link load for $e_{ah}, e_{be}, e_{cf}, e_{eg}, e_{hg}, e_{fg}$ is one. For each demand, only one path can be selected, as specified by constraint (3). Lastly, constraint (4) ensures $P_{d,k}^{ij}$ is set to one only if path k is selected. In Figure 1, the maximum link load is one if we select p_1^1, p_2^2 and p_3^2. In this case, the total path cost is $2+2+2=6$ which is minimum.

Both JRS-Multi-DEC and JRS-BIP use the link scheduling algorithm in [13] in Phase-2 to derive the minimal superframe length by decomposing a MTR WMN into $|S|$ bipartite graphs. Each selected link transmits in at least one of the $|S|$ slots; see *line 26* of Algorithm 1. Consider Figure 1. After link scheduling, links e_{eg}, e_{hg}, e_{fg} are in slot-1 and links e_{be}, e_{ah}, e_{cf} are in slot-2. *Line 26* computes the minimal superframe length, but as mentioned in Section IV, a sub-optimal transmission order will lead to high end-to-end delay. To address this issue, we re-order the resulting slots according to how many first hops it contains; see *line 27-30*. This ensures all demands begin transmission promptly. Slot-2 contains the first hop of three demands. Thus, we place slot-2 first in the new schedule S' followed by slot-1.

Algorithm 1: JRS-Multi-DEC

input : $G(V,E)$, $\mathcal{D}$, P_d and H_d^k
output: $|\mathcal{D}|$ selected paths in G, modified schedule S'

`// Phase-1`
1 **for** $d \leftarrow 1$ **to** $|\mathcal{D}|$ **do**
2   $L[d] \leftarrow$ `Distance`(r,s,d)
3 **end**
4 $\mathcal{D}' \leftarrow$ `SortDesc`(L)
5 $G' \leftarrow$ `CopyNode`(G)
6 Get first demand in $\mathcal{D}'$ and select path p_1^1
7 Add path p_1^1 into G'
8 **for** $d \leftarrow 2$ **to** $|\mathcal{D}'|$ **do**
9   **for** $k \leftarrow 1$ **to** $|P_d|$ **do**
10   Add path p_d^k into G'
11   Color graph G' with x color
12     $W_m \leftarrow$ `MaxNodeWeight`(G')
13     $ncolor_k \leftarrow \lfloor \frac{\xi(x) \times W_m}{2} \rfloor$
14     $WCD[k] \leftarrow ncolor_k \times H_d^k$
15   Delete path p_d^k from G'
16   **end**
17   $minWCD \leftarrow$ `MIN`$(WCD[k])$
18   **for** $k \leftarrow 1$ **to** $|P_d|$ **do**
19   **if** $WCD(k) == minWCD$ **then**
20   Demand d selects path p_d^k
21   Add path p_d^k into G'
22   break
23   **end**
24   **end**
25 **end**

`// Phase-2`
26 $S \leftarrow$ `LinkSchedule` (G')
27 **for** $s \leftarrow 1$ **to** $|S|$ **do**
28   $nfp[s] \leftarrow$ `NumberofFirstHop`$(S[s])$
29 **end**
30 $S' \leftarrow$ `SortDesc`(nfp)

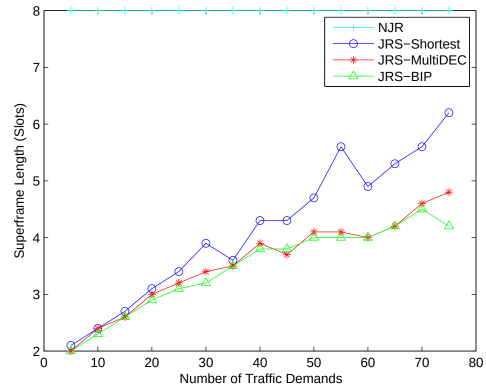
VI. EVALUATION

To evaluate the performance of JRS-Multi-DEC and JRS-BIP, we use Matlab with a toolkit named MatGraph [15]. We assume all nodes are stationary and are randomly connected to each other. We also assume each node is equipped with an antenna for each neighbor. We assume all packet transmissions take one time slot and queues are backlogged. We generate all demands randomly with a source and destination node. Note, we only consider scheduling delay. We defer the impact of queuing delay and retransmissions due to channel errors to a future work. In all experiments, we compute the average end-to-end delay.

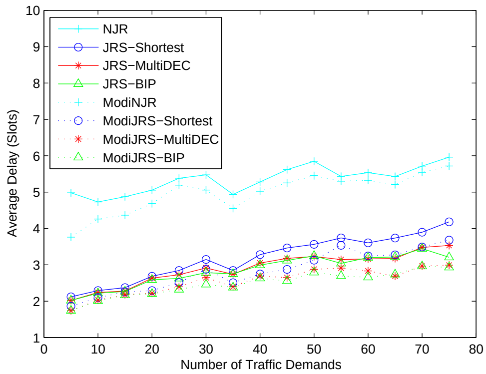
We compare the performance of JRS-Multi-DEC and JRS-BIP against JRS-Shortest and NJR. Specifically, JRS-Shortest is also a joint routing and scheduling algorithm. The difference is that in the routing phase, it always chooses the shortest path for each demand. NJR corresponds to a no joint routing algorithm where scheduling and routing are done independently. Specifically, NJR firstly applies Algo-2 [13] to schedule all links to yield a minimal schedule. Then all packets are transmitted along the shortest path of each demand. In each experiment, we collected the following metrics:

- *Superframe length*. The average total number of time slots required to satisfy all traffic demands.
- *Average delay*. This records the average, out of 10 different traffic demands, end-to-end delay of all traffic demands.

In the first experiment, we study the effect of the number of demands on superframe length and average end-to-end delay. The number of demands ranges from 5 to 75 on a random topology with 50 nodes. From Figure 2a, we see that for JRS-Shortest, JRS-Multi-DEC and JRS-BIP, the superframe length increases proportionally to traffic demands. However, for NJR, the superframe length remains fixed. Unlike other algorithms, NJR consider each link has a weight of 1, irrespective of the traffic demands. It schedules each link once in a superframe. For the other three algorithms, superframe length is dependent



(a)



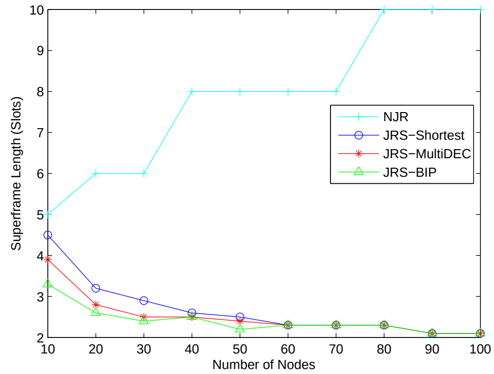
(b)

Fig. 2: Superframe length and average delay with different number of traffic demands

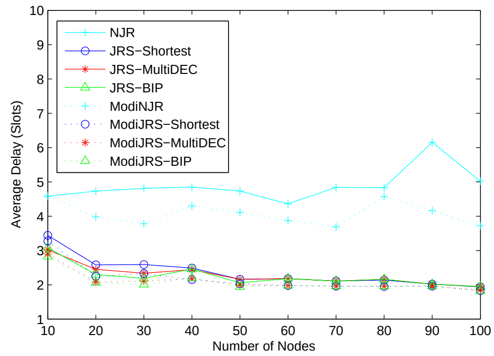
on link weights. JRS-Shortest only considers the number of hops, ignores link weights and number of bottleneck links. Therefore, JRS-Shortest, which has a higher link weight, will need more slots to schedule all links when traffic demand increases.

In Figure 2b, the solid and dashed lines indicate the average end-to-end delay of NJR, JRS-Shortest, JRS-Multi-DEC and JRS-BIP before and after reordering of time slots, respectively. We see that reordering slots according to the number of links that constitute the first hop of paths reduces end-to-end delay by about 0.4 slots. NJR has the highest end-to-end delay as it generates the longest superframe length. Hence, a packet has to wait for several slots before being transmitted to its next hop. JRS-Shortest has a much lower delay than NJR because it schedules links according to their weight. JRS-Multi-DEC and JRS-BIP are about 0.3 slots quicker than JRS-Shortest. The reason is that both algorithms result in links having a lower weight. Consequently, they have shorter superframe lengths. The improvement is more obvious when the number of demands increases.

In the second experiment, we study the effect of node numbers on the superframe length and average end-to-end delay. The number of nodes varies from 10 to 100; the number of traffic demands is fixed at 10. From Figure 3a, we can see that for NJR, the superframe length increases with the number of nodes. However, for the other three algorithms, JRS-Shortest, JRS-Multi-DEC and JRS-BIP, the superframe length decreases with increasing number of nodes. The reason



(a)



(b)

Fig. 3: Superframe length and average delay with different number of nodes

is that when there are more nodes, there are more alternative paths between a source and destination pair. Moreover, the path length also reduces. Hence, link weights reduce significantly and the number of slots decreases.

From Figure 3b, we see that NJR has a much higher delay than the other three algorithms due to a longer superframe length. In particular, when the number of nodes is small, JRS-Multi-DEC and JRS-BIP recorded 0.4 fewer slots than JRS-Shortest. When the number of nodes increases to 50, JRS-Shortest, JRS-Multi-DEC and JRS-BIP experience the same delay. As we add more links, the probability of using the same link for different demands becomes smaller. Thus, for each demand, there are multiple shortest paths. When the number of nodes is significantly higher than the number of traffic demands, shortest path is the best option.

In the third experiment, we study the effect of node degree on the superframe length and average end-to-end delay. Node degree is the number of edges incident to the node. The node degree ranges from 2 to 7; the number of traffic demands is fixed at 10 on a topology with 30 nodes. From Figure 4a, for NJR, the superframe length increases with increasing node degree. The reason is that the number of slots is dependent on the number of links. However, for JRS-Shortest, JRS-Multi-DEC and JRS-BIP, the superframe length decreases because there are more alternative paths. JRS-Multi-DEC and JRS-BIP have shorter superframe lengths than JRS-Shortest because they choose paths that lead to minimum link weights.

From Figure 4b, we see that NJR has lower delays when nodes have a degree of two because it yields shorter superframe

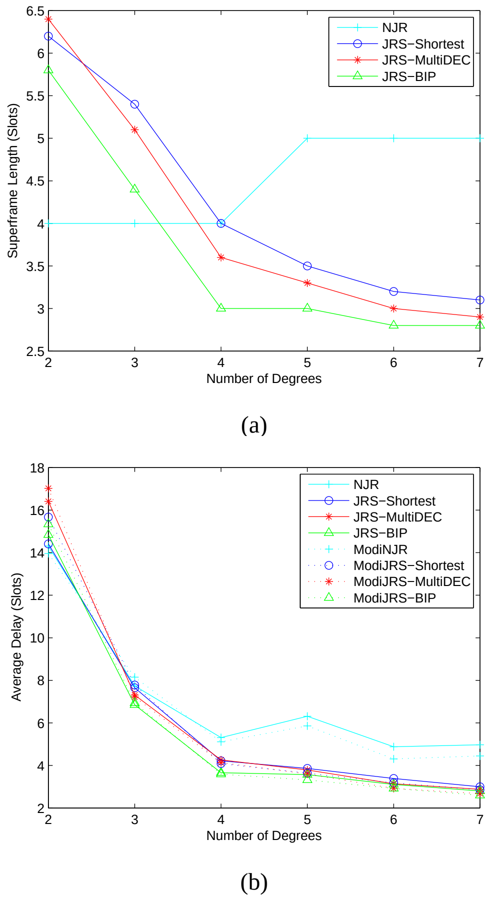


Fig. 4: Superframe length and average delay with different node degree

lengths than other algorithms. However, it has much higher delays when node degree increases. For JRS-Shortest, JRS-Multi-DEC and JRS-BIP, the delays are very close with different number of degrees. When nodes have a small degree, the number of alternative paths is also small. In addition, except for the shortest path, other paths may take many more hops. Thus, the shortest path is the best option. When node degree is large, there are more alternative paths for each demand and path length also reduces. Similarly with the second experiment, the shortest path is the best path for each demand.

VII. Conclusion

In this paper, we have studied the problem of minimizing end-to-end delay in MTR WMNs through joint routing and scheduling. We have proposed JRS-Multi-DEC and JRS-BIP, both of which minimize superframe length and average end-to-end delay. Compared with JRS-Shortest and NJR, JRS-Multi-DEC and JRS-BIP reduce the end-to-end delay significantly when the number of demands increases. When the number of nodes or degree increases, JRS-Multi-DEC and JRS-BIP have better performance than NJR but do not have much improvement as compared to JRS-Shortest. As an immediate future work, we plan to consider joint routing, link scheduling and channel assignment to minimize end-to-end delay. We also plan to construct multiple paths for each demand to improve reliability.

References

[1] L. Badia, A. Botta, and L. Lenzini. A genetic approach to joint routing and link scheduling for wireless mesh networks. *Ad Hoc Networks*, 7(4):654–664, 2009.

[2] P. Cappanera, L. Lenzini, A. Lori, G. Stea, and G. Vaglini. Link scheduling with end-to-end delay constraints in wireless mesh networks. In *IEEE WoWMoM*, Kos Island, Greece, June 2009.

[3] P. Cappanera, L. Lenzini, A. Lori, G. Stea, and G. Vaglini. Optimal joint routing and link scheduling for real-time traffic in TDMA wireless mesh networks. *Computer Networks*, 57(11), Aug. 2013.

[4] M. X. Cheng, Q. Ye, and L. Cai. Cross-layer schemes for reducing delay in multihop wireless networks. *IEEE Transactions on Wireless Communications*, 12(2):928–937, 2013.

[5] K.-W. Chin, S. Soh, and C. Meng. Novel scheduling algorithms for concurrent transmit/receive wireless mesh networks. *Computer Networks*, 56(4):1200–1214, March 2012.

[6] J. Crichigno, M.-Y. Wu, S. K. Jayaweera, and W. Shu. Throughput optimization in multihop wireless networks with multipacket reception and directional antennas. *IEEE Transactions on Parallel and Distributed Systems*, 22(7):1206–1213, 2011.

[7] P. Djukic and S. Valaee. Delay aware link scheduling for multi-hop TDMA wireless networks. *IEEE/ACM Transactions on Networking (TON*, 17(3):870–883, 2009.

[8] P. Dutta, S. Jaiswal, D. Panigrahi, and R. Rastogi. A new channel assignment mechanism for rural wireless mesh networks. In *27th IEEE International Conference on Computer Communications (INFOCOM)*, pages 2261–2269, Phoenix, AZ, USA, April 2008.

[9] P. Dutta, V. Mhatre, D. Panigrahi, and R. Rastogi. Joint routing and scheduling in multi-hop wireless networks with directional antennas. In *29th IEEE International Conference on Computer Communications (INFOCOM)*, pages 1–5, San Diego, CA, USA, March 2010.

[10] V. Friderikos and K. Papadaki. Interference aware routing for minimum frame length schedules in wireless mesh networks. *EURASIP Journal on Wireless Communications and Networking*, 2008:1–13, 2008.

[11] V. Gabale, A. Chiplunkar, B. Raman, and P. Dutta. Delaycheck: Scheduling voice over multi-hop multi-channel wireless mesh networks. In *Third Intl. Conf. on Communication Systems and Networks (COMSNETS)*, Bangalore, India, January 2011.

[12] M. S. Kodialam and T. Nandagopal. Characterizing achievable rates in multi-hop wireless mesh networks with orthogonal channels. *IEEE/ACM Transactions on Networking*, 13(4):868–880, 2005.

[13] H. Loo, S. Soh, and K.-W. Chin. On improving capacity and delay in multi tx/rx wireless mesh networks with weighted links. In *IEEE APCC*, Bali, Indonesia, Aug. 2013.

[14] B. Raman and K. Chebrolu. Design and evaluation of a new mac protocol for long-distance 802.11 mesh networks. In *ACM MOBICOM*, pages 156–169, Cologne, Germany, August 2005.

[15] E. R. Scheinerman. *Matgraph: a MATLAB toolbox for graph theory*. Department of applied mathematics and statistics, the Johns Hopkins University, Baltimore, Maryland, USA, 2008.

[16] H. Shetiya and V. Sharma. Algorithms for routing and centralized scheduling to provide QoS in IEEE 802.16 mesh networks. In *Proceedings of the First ACM Workshop on Wireless Multimedia Networking and Performance Modeling (WMuNeP)*, pages 140–149, Montreal, Quebec, Canada, October 2005.

[17] H. Su and X. Zhang. Joint link scheduling and routing for directional-antenna based 60 GHz wireless mesh networks. In *Proceedings of the Global Communications Conference (GLOBECOM)*, pages 1–6, Honolulu, Hawaii, USA, November 2009.

[18] X. Wang and J. Garcia-Luna-Aceves. Embracing interference in ad hoc networks using joint routing and scheduling with multiple packet reception. *Ad Hoc Networks*, 7(2):460–471, 2009.