

# Scheduling links with air-time in multi transmit/receive wireless mesh networks

Yuanhuizi Xu<sup>1</sup> · Kwan-Wu Chin<sup>1</sup> · Sieteng Soh<sup>2</sup> · Raad Raad<sup>1</sup>

Published online: 1 October 2015  
© Springer Science+Business Media New York 2015

**Abstract** A key advance in enabling higher wireless mesh network capacity is allowing routers to transmit *or* receive (MTR) from multiple neighbors simultaneously over the same frequency. Achieving this capacity, however, is predicated on a link scheduler that is able to capitalize on the MTR capability of nodes to activate the maximum number of active links, and also to derive the shortest schedule that ensures all links are activated at least once. To date, existing schedulers do not consider the transmission or air-time of packet(s). Henceforth, this paper fills this gap and propose to derive the shortest superframe length, defined as the end time of the last transmitting link. Our scheduler, called A-TxRx, greedily adds links whenever a link finishes its transmission. As a result, unlike previous schedulers, links can start transmitting/receiving as soon as there is no conflict. We have evaluated the performance of A-TxRx in various network configurations, and compared it against two state-of-the-art approaches: 2P and JazzyMAC. The results show A-TxRx outperforming these algorithms significantly, especially

when the network becomes denser. Specifically, the superframe length of A-TxRx is typically less than half of 2P and JazzyMAC, with 60 % more concurrently transmitting links.

**Keywords** Wireless mesh networks · Multi-transmit/receive · Link scheduling · Directional antennas

## 1 Introduction

Wireless mesh networks (WMNs) have matured significantly over the past few years. In WMNs, nodes are connected to one another wirelessly and forward packets via multi-hop communications to each other or to gateways. WMNs can be deployed in both urban and rural areas to deliver video, voice and data in indoor and outdoor environments and have applications in home networking [1], enterprises [2], and metropolitan area networks [3]. In this respect, the Quality of Service (QoS) experienced by users is a critical consideration; see [4]. Recently, researchers have proposed Multi-Transmit-Receive (MTR) WMNs, whereby nodes are equipped with multiple Directional Antennas (DAs) or adaptive arrays in order to increase network capacity. This means all nodes can transmit or receive simultaneously from their respective neighbouring nodes; see Fig. 1(a, b). The resulting WMNs thus have a higher network capacity than those that use an omni-directional or a single directional antenna [5–7]. We note that the WMN under consideration is distinct from past works that assume nodes have multiple radios and multiple channels [8].

As illustrated in Fig. 1(c), a key constraint is no Mix-Tx-Rx, meaning a node operates in half duplex mode [9]. Consequently, any developed scheduler that aims to derive

---

✉ Kwan-Wu Chin  
kwanwu@uow.edu.au

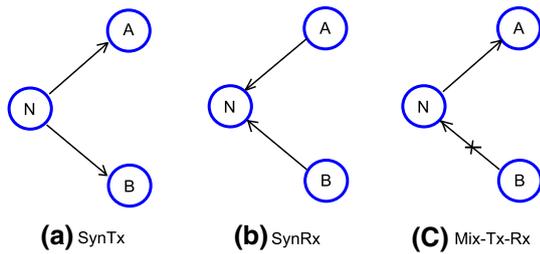
Yuanhuizi Xu  
yx879@uowmail.edu.au

Sieteng Soh  
s.soh@curtin.edu.au

Raad Raad  
raad@uow.edu.au

<sup>1</sup> The School of Electrical, Computer, and Telecommunications Engineering, University of Wollongong, Wollongong, NSW, Australia

<sup>2</sup> The Department of Computing, Curtin University of Technology, Perth, WA, Australia



**Fig. 1** Different phases of MTR transmission/receptions; **a** transmission, **b** reception, **c** illegal transmission; i.e., no transmit and receive in the same phase. Note that, all transmissions are over the *same* frequency

the shortest superframe length must adhere to this constraint. Note, a ‘superframe’ is defined as a sequence of transmission slots; hence, the *shortest* superframe denotes one that has the minimum number of slots. In [9], Raman et al. outlined a spatial time division multiple access (TDMA) medium access control (MAC), called 2 Phase (2P), that separates transmissions, called SynOp, at each node into two phases: SynTx and SynRx. However, this MAC can only be applied to MTR WMNs with a bipartite topology. This problem is solved by Chin et al. in [10] whereby they presented a novel scheduling algorithm called Algo-1 that schedules links in MTR WMNs with an arbitrary topology by recursively partitioning nodes into maximally connected bipartite sets. This algorithm is then improved in [11] to realize the maximal number of concurrent transmitting links.

The said prior works, however, do not consider links with different weights. This is necessary because of two reasons. Firstly, in practice it is likely that links will have different loads. For example, links on shortest paths or a router may be serving an area with a large number of subscribers. Secondly, different data rates may be used by links in order to counter the vagaries of the wireless channel. For example, IEEE 802.11a supports data rates up to 54 Mbps. This means, for a given packet size, the air-time required to transmit said packet will vary depending on the data rate or channel condition as well as the number of packets to be transmitted.

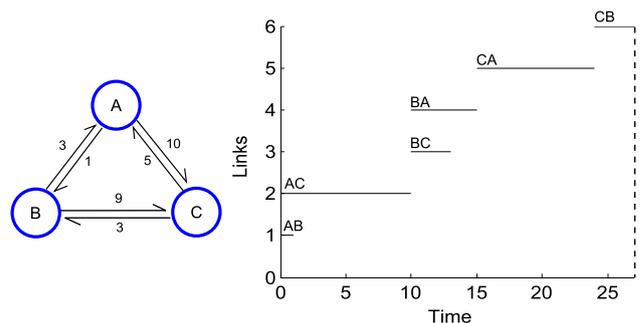
In [12, 13] and [14], the authors consider the weight of links. Dai et al. [13] propose two schedulers called Heavy-Weight-First (HWF) and Max-Degree-First (MDF). As their name implies, links are either scheduled depending on their weight or the number of neighbors. However, the authors of [12] and [13] did not consider links that require different transmission times. This problem is tackled in [14] where the authors present an adaptive MAC called JazzyMAC. A node first monitors the volume of traffic on its outgoing links. It then dimensions the transmission slot of each link as per the observed traffic volume. To schedule transmission, a token exchange mechanism is used

whereby a node only transmits whenever it has the token for all its links. This, however, precludes nodes from taking advantage of opportunistic links; see Sect. 3.1. Furthermore, it is sensitive to the initial token assignments. More importantly, although the transmission finishing time for all incident links to one node varies depending on actual traffic demand, the transmissions of the said links still start at the same time.

Given the above discussions, we now show a key limitation of past solutions. Consider Fig. 2. We see a MTR WMN with three nodes connected in a clique. The number next to each link represents its required transmission or air-time. Also shown in Fig. 2 is the schedule for each link derived using 2P [9]. In this example, at time = 0, link AB and AC start to transmit simultaneously in the first slot for 10 units of time. Then at time = 10, the corresponding opposite links BA and CA are activated in the second slot for the duration of five units only after both AB and AC finish. At time = 15, the activation of link BC and link CB takes place in the third and fourth slot for a duration of nine and three time units respectively. Thus, for this example, 2P derives superframe that has length  $T = 27$ , and schedules every link to be activated at least once for the period of its assigned air-time without conflict.

In the aforementioned example, we see that a group of links is to remain active in SynTx or SynRx in one slot for the same slot duration regardless of the actual link weight. This situation leads to the following problem. Consider link AB and AC, which are activated simultaneously in slot 1. The slot length needs to be the longer air-time, which is 10. Notice that link AB finishes in one slot, causing 9/10 of the link capacity to be wasted, leading to a lower throughput. This is because no other links are scheduled until link AC finishes. In fact, if node B wants to transmit to C, it should be able to initiate transmission at any time after the first 1/10 of slot. If link BC avoids having to wait until link AC finishes, throughput improves.

In light of the above observations, we propose A-TxRx, a scheduler that activates link with different transmission



**Fig. 2** An example network where links with different air-times are scheduled using 2P

or *air-time* in a MTR WMN. Unlike existing schedulers, A-TxRx aims to minimize the finishing time of the last transmitting link as well as maximize the number of links at any time instant. In a nutshell, our main contributions are:

- A-TxRx is the first centralized link scheduler for MTR WMNs that considers the assigned air-time of links in order to meet the underlying link demand. Our results show that A-TxRx has better performance in all considered scenarios, i.e., an average of 40 % shorter superframe length than 2P, especially when the network is fully connected. For example, as will be shown in Sect. 3, A-TxRx generates a superframe with length  $T = 16$  for the WMN of Fig. 2. In all experiments, the superframe length of A-TxRx is at most 70 % shorter than state-of-the-art approaches.
- We outline an improvement to A-TxRx, where we opportunistically add scheduled links to further increase network capacity. Our results show that the number of concurrent transmitting links when running A-TxRx to be 40 % more than JazzyMAC [14] on average.
- We analyze and prove that A-TxRx is a collision free scheduler for arbitrary topologies and has a running time of  $\mathcal{O}(|V|^5)$  for WMN with  $|V|$  nodes.

The rest of the paper is structured as follows. Sect. 2 describes the network model of MTR WMNs and the formal definition of our problem. Section 3 outlines the details and analysis of A-TxRx. The research methodology is presented in Sect. 4. Section 5 presents our experiments and results. The paper concludes in Sect. 6.

## 2 Preliminaries

We model a MTR WMN as a directed graph  $G(V, E)$ . The set  $V$  is comprised of nodes that are equipped with  $b \geq 1$  directional antennas. Each node  $u$  has a transmission range of  $r$  and  $b_u \geq 1$  radios. There is a directional edge  $(u, v) \in E$  connecting node  $u$  and  $v$  if they are within each other's transmission range. We also consider there to be a directional link connecting node  $v$  to  $u$ ; i.e.,  $(v, u) \in E$ . Note, in practice, nodes may have a different transmission range due to the vagaries of the wireless channel. To this end, nodes are required to ensure incoming and outgoing link to each neighbor is functional. This can be achieved via HELLO messages in the neighbor discovery process whereby nodes include the neighbors they can hear in their HELLO messages. The function  $f_i : E \rightarrow \mathbb{R}$  returns the

allocated air-time of a given link. Hence,  $f_i$  models the required transmission time in order to meet a given traffic load. All nodes are able to concurrently transmit or receive on all links. Each link is supported by a radio and we assume  $b_u \geq |N(u)|$  for all nodes.

We remark that there are two main realizations of MTR WMNs. The first is outlined in [9], where each router is equipped with multiple radios connected to a parabolic antenna. All radios operate on the same frequency. Nodes disable their carrier sense to allow concurrent transmissions, and transmit power control is used to ensure incoming links have sufficient signal strength to ensure correct reception. The second realization is to employ multi-user multi-input multi-output (MU-MIMO); see [15] or [16] for details. Nodes have multiple antenna elements that they can use to transmit independent data or to null interfering transmissions. In addition, nodes have channel state information (CSI). This assumption is reasonable given that nodes are primarily static and pilot symbols can be transmitted periodically to learn the CSI.

Formally, we have the following *Mix-Tx-Rx constraint*: for a given node  $u$ , let  $IN(u, t)$  and  $OUT(u, t)$  be its set of receiving and transmitting links at time  $t$  respectively. This constraint is met if both  $|IN(u, t)|$  and  $|OUT(u, t)|$  are *not* greater than zero simultaneously at any time  $t$  for all node  $u \in V$ .

In our model, we make the following assumptions:

- All nodes are synchronized globally. This is reasonable given that nodes are static and can use GPS to synchronize their clock.
- Nodes are tuned to a single frequency. According to [9], the main reasons for employing a single channel are as follows: (a) it is convenient to use a single channel for the backbone whilst using other channels for local access, (b) the more channels it uses, the higher the operational cost because the IEEE 802.11b/a bands are licensed for outdoor use in some developing countries, and (c) to avoid RF pollution as there are many WiFi networks in existence. We also note that the MTR capability of nodes can also be achieved using MIMO [17] or by equipping nodes with 60 GHz radios [18].
- We assume the traffic load on each link is aggregated, i.e., see [14], and remains fixed for a non-negligible amount of time, e.g., every hour. This also means the routing for source destination pairs are fixed for this period of time. We leave the joint case of optimizing both routing and link scheduling to a future work.

- Each air-time unit can correspond to a specific amount of time, e.g., one second, or the time it takes to transmit one packet. It is important to note that once an air-time is assigned to a link, upon scheduled for transmission, it is guaranteed to transmit for said air-time without pre-emption.

Our problem is as follows: given a MTR WMN, whereby each link has a different air-time, design a centralized algorithm that derives the minimal superframe length whereby links are activated at least once and in accordance to their requested air-time. Note, we define a superframe as the sequence  $(\{e_1, \dots, e_x\}, t_1) \cup \dots \cup (\{e_y, \dots, e_{|E|}\}, t_{|E|})$ , whereby  $e_i \in E$  is the link to be scheduled at time  $t_i$ . For example, in Fig. 2, we have  $(\{AB, AC\}, 0) \cup (\{BA, CA\}, 10) \cup (\{BC\}, 15) \cup (\{CB\}, 24)$ . The problem is to derive a superframe with minimal  $t_{|E|} + f_i(e_{|E|})$  value subject to the Mix-Tx-Rx constraint (cf. Fig. 1), and each link  $e \in E$  receiving at least  $f_i(e)$  of air-time.

Note, if all links have the same air-time, then the problem is similar to prior works such as [11]. In particular, the authors of [11] show that deriving the maximum number of links in each slot is equivalent to solving the NP-complete, MAX CUT problem, meaning our problem is just as hard. To this end, in the next section, we propose a greedy heuristic that aims to determine the shortest possible superframe.

### 3 The solution: A-TxRx

The basic idea is to greedily schedule new non-interfering links whenever a link finishes transmission. In Sect. 3.1, we show how the resulting schedule can be improved by adding so called opportunistic links. In addition, in Sect. 3.2, we simplify a computationally expensive step used to calculate a MIS with a greedy step that adds non-conflicting links according to their transmission time.

Our algorithm has the following key steps. Firstly, it constructs a conflict graph  $G'(V', E')$  based on the network topology  $G(V, E)$ . In the conflict graph, each vertex  $v' \in V'$  denotes a link in  $E$ , and a conflict between two links in  $E$  is represented by an edge  $e' \in E'$  between the corresponding vertices; see [19]. It then greedily determines all Maximal Independent Sets (MISs) of  $G'$ . Recall that an MIS is the subset of all the links in  $G'$  that can be activated at the same time without interference. Note that a link may appear in different MISs. In the third step, we choose the MIS with the most links, which ensures high throughput. We activate all links in the selected MIS, and label them as *active links*.

We also record their air-time. Among all the *active links*, the ones that have the same minimum air-time are regarded as *finishing links*. We then record these finishing links' current air-time. At such time, we remove them from  $G'$ . If  $G'$  is not empty, we first determine if there are any active links. These links are denoted as *remaining links*. If there are no *remaining links*, then a new MIS is obtained directly from the current  $G'$ . This MIS contains new links that can be added into the superframe. However, if there are *remaining links*, we check to see whether we can add new non-interfering links. We first construct a new conflict graph  $G''$  from  $G'$ . Specifically, we remove the finishing links, all remaining links and their neighbors. Then a new MIS is obtained from  $G''$ , which contains the most new links that do not interfere with remaining links. The next set of *finishing links* are then determined and we repeat the process. A-TxRx terminates when  $G'$  is empty (Table 1).

We will now show how A-TxRx, see Algorithm 1, determines the schedule for the topology shown in Fig. 2. As depicted, it is an MTR WMN with three nodes A, B, and C connected with bidirectional links. The value next to each link indicates its allocated air-time.

**Table 1** Key notations

Symbol	Description
$G$	The directed graph
$V$	The set of nodes or vertices in $G$
$E$	The set of links or edges in $G$
$G'$	The conflict graph generated from $G$
$V'$	The set of nodes or vertices in $G'$
$E'$	The set of links or edges in $G'$
$G''$	The altered graph from $G'$
$e_{AB}$	A link with source node $A$ and destination node $B$ in $E$
$\delta(G)$	The maximum length of the superframe derived by A-TxRx for graph $G$
$f_i(e)$	The air-time of link $e$
$V_1 \setminus V_2$	The pair of subsets of a bipartite graph
$i_1 \setminus i_2$	The link emanating from $V_1$ or $V_2$ with the longest air-time
$\mathbb{A} \setminus \mathbb{B}$	Two MISs derived from the conflict graph of a bipartite graph
$I_t^{\mathbb{A}} \setminus I_t^{\mathbb{B}}$	The activation time of links in $\mathbb{A}$ or $\mathbb{B}$
$\mathcal{A}$	The set of active links
$\mathcal{F}$	The set of finishing links
$\mathcal{R}$	The set of remaining links
$\mathcal{N}$	The set of interfering nodes of links in $\mathcal{R}$
$t_{finish}$	The next finishing time of the currently active links

**Algorithm 1:** A-TxRx

---

**Input:** network graph  $G(V, E)$ , air-time of links  $f_t : E \rightarrow \mathbb{R}$   
**Output:**  $SF$  containing set of links  $\mathcal{A}$  and their activation time  $t$

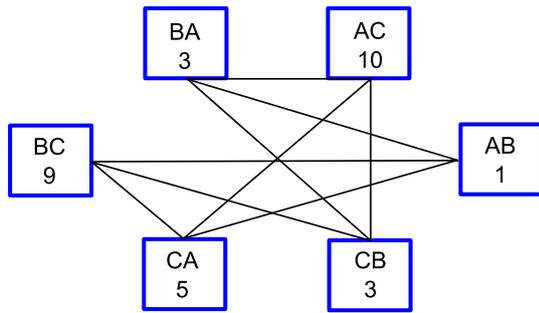
```

1 if  $|V| \leq 1$  then
2   return
3 else
4    $t \leftarrow 0$ 
5    $SF \leftarrow \emptyset$ 
6    $\mathcal{A} = \mathcal{F} = \mathcal{R} \leftarrow \emptyset$ 
7    $G' \leftarrow \text{ConflictG}(G(V, E))$ 
8   while  $(G' \neq \emptyset)$  do
9     if  $(\mathcal{R} = \emptyset)$  then
10       $\mathcal{A} \leftarrow \text{MaxMIS}(G')$ 
11       $SF \leftarrow SF \cup (\mathcal{A}, t)$ 
12       $G' \leftarrow G' - \mathcal{A}$ 
13       $t_{finish} \leftarrow \min f_t(\mathcal{A})$ 
14      for  $e \in \mathcal{A}$  do
15        if  $f_t(e) == t_{finish}$  then
16           $\mathcal{F} \leftarrow \mathcal{F} \cup e$ 
17        end
18         $f_t(e) \leftarrow f_t(e) - t_{finish}$ 
19      end
20       $t \leftarrow t + t_{finish}$ 
21       $\mathcal{R} \leftarrow \mathcal{A} \setminus \mathcal{F}$ 
22       $\mathcal{F} \leftarrow \emptyset$ 
23    else
24       $\mathcal{N} \leftarrow \text{Neighbor}(\mathcal{R}, G')$ 
25       $G'' \leftarrow G' - \mathcal{N}$ 
26       $\mathcal{A} \leftarrow \text{MaxMIS}(G'')$ 
27       $SF \leftarrow SF \cup (\mathcal{A}, t)$ 
28       $G' \leftarrow G' - \mathcal{A}$ 
29       $\mathcal{A} \leftarrow \mathcal{A} + \mathcal{R}$ 
30       $t_{finish} \leftarrow \min f_t(\mathcal{A})$ 
31      for  $e \in \mathcal{A}$  do
32        if  $f_t(e) == t_{finish}$  then
33           $\mathcal{F} \leftarrow \mathcal{F} \cup e$ 
34        end
35         $f_t(e) \leftarrow f_t(e) - t_{finish}$ 
36      end
37       $t \leftarrow t + t_{finish}$ 
38       $\mathcal{R} \leftarrow \mathcal{A} \setminus \mathcal{F}$ 
39       $\mathcal{F} \leftarrow \emptyset$ 
40    end
41  end
42 end

```

---

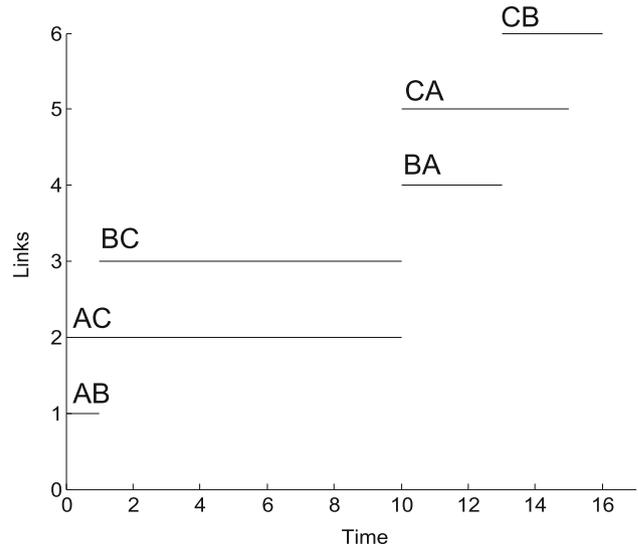
- A-TxRx takes as input the network graph  $G(V, E)$ , and the air-time  $f_t(e)$  assigned to each edge  $e \in E$ .
- A-TxRx produces  $SF$  as its output. The set  $SF$  contains tuples  $(\mathcal{A}, t)$ , where set  $\mathcal{A}$  represents a group of *active links* that start their transmission at time  $t$ .
- *Line 1–6:* Initially,  $t$  is set to 0. The set  $SF$  is empty. The sets  $\mathcal{F}$  and  $\mathcal{R}$  represent the group of finishing links and remaining links respectively. Sets  $\mathcal{A}$ ,  $\mathcal{F}$  and  $\mathcal{R}$  are initially empty.
- *Line 7:* Function  $\text{ConflictG}()$  computes the corresponding conflict graph  $G'$  from  $G(V, E)$ ; see Fig. 3.
- *Line 9–12:* When  $\mathcal{R}$  is an empty set, the function  $\text{MaxMIS}()$  performs graph coloring on  $G'$  to find the MISs:  $\{e_{AB}, e_{AC}\}$ ,  $\{e_{BA}, e_{BC}\}$  and  $\{e_{CA}, e_{CB}\}$ . We then choose the MIS containing the most links. Here, as all three MISs are of the same size, we randomly choose  $\{e_{AB}, e_{AC}\}$ . Thus,  $e_{AB}$  and  $e_{AC}$  are chosen as *active links* and included in the set  $\mathcal{A}$ , meaning all links in this MIS start to transmit at  $t = 0$ . Then  $\mathcal{A}$  and the corresponding  $t$  are recorded in  $SF = (\{e_{AB}, e_{AC}\}, 0)$ . The graph  $G'$  is updated by removing the links in  $\mathcal{A}$  from  $G'$ . In this case, we remove link



**Fig. 3** Conflict graph for the example topology shown in Fig. 2

$e_{AB}$  and  $e_{AC}$ , i.e., node AB, AC and its incident links in Fig. 3.

- **Line 13–19:** Among all air-time of links in  $\mathcal{A}$ , we set the shortest one as  $t_{finish}$ . The link(s) with the shortest air-time is (are) included in the set  $\mathcal{F}$ . The reason is that these links, amongst those in  $\mathcal{A}$ , will finish the earliest. In our example, we have  $t_{finish} = 1$ , which corresponds to the air-time of link  $e_{AB}$ . Hence, the set  $\mathcal{F}$  contains  $e_{AB}$ . Also, we update the air-time of the links in  $\mathcal{A}$  by subtracting  $t_{finish}$  from the original air-time  $f_i(e)$ . For example,  $f_i(AC) = 10 - 1 = 9$ , meaning that at this point in time, link  $e_{AC}$  has transmitted for one time unit, but it still needs another nine time units to complete its transmission.
- **Line 20:** The variable  $t$  becomes  $t = 0 + 1 = 1$  because one time unit has been used for links in  $\mathcal{A}$  to operate.
- **Line 21–22:** The set  $\mathcal{R}$  consists of links that are in  $\mathcal{A}$ , but not in  $\mathcal{F}$ . Afterwards, we clear set  $\mathcal{F}$ . In this example, the set  $\mathcal{R}$  contains link  $e_{AC}$  since it has not finished transmitting. At this point, since  $\mathcal{R}$  is not an empty set, A-TxRx restarts from Line 24.
- **Line 24–25:** the function Neighbor() returns all interfering nodes of links in  $\mathcal{R}$ . A-TxRx copies  $G'$  to a new graph  $G''$ , and removes  $\mathcal{N}$  from  $G''$ ; in our example, this means  $e_{BA}, e_{CA}$  and  $e_{CB}$  are removed from  $G''$ . This step is essential because it ensures that newly activated links do not interfere with the links in  $\mathcal{R}$ .
- **Line 26–28:** A-TxRx then calls MaxMIS() on  $G''$  to derive links that can be added into set  $\mathcal{A}$  without any conflict, which is link  $e_{BC}$  in this case. As a result, we have  $SF = (\{e_{AB}, e_{AC}\}, 0) \cup (\{e_{BC}\}, 1)$ . The graph  $G'$  is updated by removing active links  $e_{BC}$  from  $G'$ .
- **Line 29:** All the links that are transmitting at this point in time are  $e_{AC}$  and  $e_{BC}$ .
- **Line 30–36:** The shortest air-time of links in  $\mathcal{A}$  is  $f_i(AC) = f_i(BC) = 9$ . The variable  $t_{finish}$  is then set to 9. Thus, link  $e_{AC}$  and  $e_{BC}$  become finishing links in  $\mathcal{F}$ . As before, the air-time of links are updated by subtracting  $t_{finish}$  from the original air-time  $f_i(e)$ . In this case,  $f_i(BC) = f_i(AC) = 9 - 9 = 0$ , meaning that at this point in time, both links finish their transmission.



**Fig. 4** Schedule timeline for Fig. 2

- **Line 37–39:** The variable  $t$  becomes  $t = 1 + 9 = 10$ . The set  $\mathcal{R}$  becomes empty. Then clear set  $\mathcal{F}$ . At this stage, since there are no links in  $\mathcal{R}$ , A-TxRx repeats from Line 10.
- **Line 10:** A-TxRx executes MaxMIS() on  $G'$ . The MISs are  $\{e_{BA}, e_{CA}\}$  and  $\{e_{CA}, e_{CB}\}$ . We select  $\{e_{BA}, e_{CA}\}$  into  $\mathcal{A}$  arbitrarily. The process continues until  $t$  reaches time 13. At this time,  $G'$  becomes empty, meaning that all the links in the network have been scheduled. The program terminates and the output is  $SF = (\{e_{AB}, e_{AC}\}, 0) \cup (\{e_{BC}\}, 1) \cup (\{e_{BA}, e_{CA}\}, 10) \cup (\{e_{CB}\}, 13)$ . After another three time units, which is the air-time of the last activated link  $e_{CB}$ , all the links in the network have transmitted once. Thus, for Fig. 2, A-TxRx generates a superframe with length of 16, reducing the superframe length generated using 2P by almost 41 %. The time line of the schedule is shown in Fig. 4.

### 3.1 Opportunistic links

We can improve capacity by adding “opportunistic links”. These links are defined as those that can be allocated additional transmit opportunities without interfering existing links. In the aforementioned example, note that in  $Time = [15, 16]$ , only link  $e_{CB}$  is transmitting. No other links are activated because all links have transmitted once. In fact, we can also activate link  $e_{AB}$  or  $e_{CA}$  as an opportunistic link as they do not interfere currently active links. Hence, we can select  $e_{AB}$  because it has the air-time of one unit and more importantly, adding it does not change the superframe length. On the other hand, if we activate  $e_{CA}$  at  $Time = 15$  with air-time of five time units, it will expand the superframe length from 16 to 20.

To add opportunistic links, we make the following improvement to A-TxRx. After constructing a new MIS consisting of new links to be added into the superframe in line 10 or line 26 of Algorithm 1, we delete all the links in the new MIS and their neighbors from the original conflict graph to obtain a graph where the previously scheduled links are included. Next, we color this new graph to obtain the largest MIS. This MIS contains all the links that can be activated without causing interference. We then choose only the links with an air-time that does not extend the superframe length. Hence, the derived schedule is not only a feasible one, but also has higher network capacity.

A question that arises is why only links that have been activated previously can be opportunistic links. In line 12 and 28 of Algorithm 1, as soon as a link is included in the chosen MIS and selected to transmit, it is deleted from  $G'$ . Thus, when we call MaxMIS() on graph  $G'$  in line 10 and 26, the resulting  $\mathcal{A}$  already contains the most non-interfering links that have yet to be activated. None of the links in  $G'$  can be scheduled because it interferes with at least one link in  $\mathcal{A}$  as  $\mathcal{A}$  is an MIS. Therefore, only by searching the links that are not in  $G'$ , i.e., links that has been activated at least once, can we find some other non-interfering links and use them as opportunistic links.

### 3.2 Greedy A-TxRx

In A-TxRx, the function MaxMIS() in Line 10 and 26 first performs graph coloring on the conflict graph  $G'$  in order to find the MISs. Then the MIS with the maximum cardinality is chosen as the set of *active links*  $\mathcal{A}$ . However, graph coloring is a time-consuming operation; see Sect. 3.3. Thus, we present a modified A-TxRx which is denoted as A-TxRx<sub>Greedy</sub>, where we replace the function MaxMIS() in Line 10 and 26 with Greedy(). This function Greedy() constructs  $\mathcal{A}$  by greedily iterating through all the links in  $E$  starting from the one with the longest transmission time. Link  $e$  is added to the set  $\mathcal{A}$  if  $e$  is not conflicting with any link that is already included in  $\mathcal{A}$ . Otherwise, link  $e$  is not selected to join  $\mathcal{A}$ .

### 3.3 Analysis

We now compute the running time complexity of A-TxRx for an arbitrary graph  $G$  with  $|V|$  nodes, whereby the number of edges  $|E|$  of  $G$  is upper bounded by  $|V|(|V| - 1)$ , i.e.,  $G$  is fully connected. In particular, we have the following result.

**Theorem 1** *The running time complexity of A-TxRx is  $\mathcal{O}(|V|^5)$ .*

*Proof* All lines of A-TxRx take  $\mathcal{O}(1)$  except for lines 7, 8, 10, 13, 14, 24, 26, 40 and 31. In line 7, the function

ConflictG() takes  $\mathcal{O}(|V|^2)$  times to convert the original graph  $G$  into a conflict graph  $G'$ . The reason is that every edge in  $G$  is considered as a vertex in  $G'$  and is connected to its interfering links. Consequently, the number of vertices in  $G'$  is equal to the number of edges in  $G$ , which is bounded by  $|V|(|V| - 1)$ . Similar for line 8, the time complexity is also  $\mathcal{O}(|V|^2)$  because it considers the number of vertices in  $G'$ . In line 10 and 26, the process of graph coloring is performed. According to [20], the smallest-last graph coloring algorithm has a time complexity of  $\mathcal{O}(|V| + |E'|)$  for  $G'(V', E')$ . To calculate the total number of edges in  $G'$ , we assume the worst case, whereby  $G$  is fully connected. For any link, say  $e_{AB}$ , there are  $2|V| - 3$  number of links originating from node  $B$  and directing to node  $A$  that are in conflict with  $e_{AB}$ . Hence, the total number of edges in  $G'$  is calculated as  $(2|V| - 3) \times |V|(|V| - 1)$ . As a result, the time complexity for line 10 and 26 is  $\mathcal{O}(|V|^2 + |V|^3) = \mathcal{O}(|V|^3)$ . For line 13, 30, 14 and 31, in the worst case, every vertex  $v' \in V'$  in  $G'$  is linked with only one vertex which represents its corresponding opposite direction link, and thus the size of the largest MIS, used as  $\mathcal{A}$ , is equal to  $\frac{1}{2}|V|(|V| - 1)$ . Similarly, in the worst case, the size of  $\mathcal{R}$  is equal to  $\frac{1}{2}|V|(|V| - 1)$ . Therefore, line 12, 30, 14, 31 and 24 take at most  $\mathcal{O}(|V|^2)$  time. Based on the above calculation, A-TxRx has a time complexity of  $\mathcal{O}(|V|^2(|V|^3)) = \mathcal{O}(|V|^5)$ .  $\square$

**Theorem 2** *A-TxRx produces a collision free schedule.*

*Proof* We use prove by contradiction. Suppose we have two links transmitting *and* receiving at the same time; specifically,  $e_{AB}$  and  $e_{xA}$  or  $e_{AB}$  and  $e_{By}$  are transmitting concurrently, where  $x$  can be any node but  $A$ ,  $y$  can be any node but  $B$ . So we have two cases:

Case 1: These two interfering links start to transmit at the same point in time. In this case, the two links are activated in line 10. After the graph coloring process, only links in one independent set are activated. Recall that, in an independent set of a graph, no two vertices of the graph are adjacent. This contradicts that the two links are interfering with each other. This leaves the second case.

Case 2: One of these two conflicting links, say  $e_{xA}$  (or  $e_{By}$ ) starts to transmit while the other link, say  $e_{AB}$  has not finished its transmission. It indicates that the newly added link is activated in line 26, which applied graph coloring to a new graph  $G''$ . Note that in line 25, all the remaining links including  $e_{AB}$  and their neighboring links are deleted from  $G''$ . Thus, any link that conflicts with link  $e_{AB}$  does not exist in graph  $G''$ . Hence, it contradicts that  $e_{xA}$  (or  $e_{By}$ ) are derived from line 25. As a result, it is impossible for A-TxRx to derive a schedule with interference.  $\square$

**Theorem 3** *A-TxRx generates a link schedule that is no longer than  $\delta(G) = l_t^A + l_t^B$  for a bipartite graph topology  $G(V, E)$ .*

*Proof* Let  $V_1$  and  $V_2$  be the two subsets of a bipartite graph  $G(V, E)$  such that nodes in  $V_1$  ( $V_2$ ) have links only to nodes in  $V_2$  ( $V_1$ ). Line 7 of A-TxRx generates the conflict graph  $G'(V', E')$  from  $G$ . Notice that one can generate two MISs, i.e.,  $\mathbb{A}$  and  $\mathbb{B}$ , from  $G'$ ;  $\mathbb{A}$  ( $\mathbb{B}$ ) contains links emanating from nodes in  $V_1$  ( $V_2$ ). Let  $i_1$  ( $i_2$ ) be a link emanating from  $V_1$  ( $V_2$ ) with the longest air-time, i.e.,  $f_i(i_1) = l_t^A$ ; similarly,  $f_i(i_2) = l_t^B$ . Note that there can be multiple number of links  $i_1$  and  $i_2$ . Consider three possible cases: (i) all links in  $\mathbb{A}$  ( $\mathbb{B}$ ) have the same air-time, i.e., each link in  $\mathbb{A}$  ( $\mathbb{B}$ ) has an air-time  $l_t^A$  ( $l_t^B$ ), (ii) links  $i_1$  and  $i_2$  are interfering with each other, and (iii) links  $i_1$  and  $i_2$  are not interfering with each other.

For case (i), A-TxRx generates a schedule in which links in  $\mathbb{B}$  are activated after all links in  $\mathbb{A}$  have completed their air-times, resulting in  $\delta(G) = l_t^A + l_t^B$ . Specifically, Line 10–12 generate MIS  $\mathcal{A} = \mathbb{A}$  and schedule all links in  $\mathbb{A}$  starting at time  $t = 0$  for  $l_t^A$  time unit, and  $G' = \mathbb{B}$ . Line 14–19 generate  $\mathcal{F} = \mathbb{A}$  because all links have the same  $t_{finish} = l_t^A$ , and thus Line 20–21 obtain  $t = l_t^A$ , and  $\mathcal{R} = \emptyset$ . Therefore, in the second iteration of the loop in Line 8, Line 10–12 generate MIS  $\mathcal{A} = \mathbb{B}$  and schedule all links in  $\mathbb{B}$  starting at time  $t = l_t^A$  for  $l_t^B$  time unit, and  $G' = \emptyset$ . Similar to for  $\mathbb{A}$ , this iteration produces  $\mathcal{F} = \mathbb{B}$ , and  $\mathcal{R} = \emptyset$ . Thus the algorithm completes with a schedule that starts at  $t = 0$  for all links in  $\mathbb{A}$  for  $l_t^A$  time unit, and completes with all links in  $\mathbb{B}$  that start from  $t = l_t^A$  for  $l_t^B$  time unit. Since the schedule completes at time  $l_t^A + l_t^B$ , for this case, A-TxRx generates the schedule with a superframe length of  $\delta(G) = l_t^A + l_t^B$ .

For case (ii), some links other than  $i_2$  can be activated at the same time with links in  $\mathbb{A}$  as long as they do not interfere. However, since  $i_2$  and  $i_1$  are interfering links,  $i_2$  can be activated only after  $i_1$  has completed its  $l_t^A$  air-time;  $i_2$  completes its air-time at time  $l_t^A + l_t^B$ , meaning the superframe length for this case is  $\delta(G) = l_t^A + l_t^B$  because all links in  $\mathbb{A}$  ( $\mathbb{B}$ ) complete no later than time  $l_t^A$  ( $l_t^A + l_t^B$ ). Specifically, Line 10–12 generate MIS  $\mathcal{A} = \mathbb{A}$  and schedule all links in  $\mathbb{A}$  at time 0. Since  $\mathcal{R}$  contains at least  $i_1$ , and  $G' \neq \emptyset$ , in the second iteration, Line 24–27 schedule links in  $\mathbb{B}$  that do not interfere with the remaining links in  $\mathcal{R}$  of  $\mathbb{A}$  that have air-time larger than  $t_{finish}$ . For Line 30, we consider two possible cases: (ii.1) at least one additional link from  $\mathbb{B}$  has an air-time longer than the remaining air-time of  $i_1$ ; (ii.2) the remaining air-time of  $i_1$  is the longest among links in the new MIS. For case (ii.1), A-TxRx iterates the else in Line 23, and eventually will generate an

MIS that contains only links in  $\mathbb{B}$ , including  $i_2$ , at time  $l_t^A$ . At this stage, Line 24–26 generate  $\mathcal{N} = \emptyset$  since nodes in  $\mathcal{R} = G' \subseteq V_2$  and the nodes are not neighbors of the others,  $G'' = \emptyset$ , and  $\mathcal{A} = \emptyset$ ; thus Line 26 does not add more links to SF, links that start at time  $l_t^A$  will complete for at most  $l_t^B$  time unit, meaning the superframe length is  $\delta(G) = l_t^A + l_t^B$ . For case (ii.2), A-TxRx will continue iterating from Line 24 until reaching case (ii.1) or the remaining air-time of  $i_1$  is zero. In either case, at time  $l_t^A$ ,  $\mathcal{R}$  will eventually contain links only from  $\mathbb{B}$ , and as described before, the links will be activated no longer than  $l_t^B$  time unit, giving the superframe length of  $\delta(G) = l_t^A + l_t^B$ .

For case (iii), link  $i_2$  can be activated at the same time with link  $i_1$  as long as its interfering link has completed its air-time. Thus, for this case, A-TxRx generates schedule with superframe length  $\delta(G) < l_t^A + l_t^B$ . Specifically, when A-TxRx reaches Line 24,  $\mathcal{N}$  does not contain  $i_2$ , and thus the SF in Line 27 contains both  $i_1$  and  $i_2$ , which eventually reduces the superframe length of the schedule to less than  $l_t^A + l_t^B$  without interfering links.

Based on the above analysis, when  $i_2$  and  $i_1$  are interfering links,  $i_2$  can be activated only after  $i_1$  has completed its  $l_t^A$  air-time;  $i_2$  completes its air-time at time  $l_t^A + l_t^B$ , meaning the superframe length for this case is  $\delta(G) = l_t^A + l_t^B$  because all links in  $\mathbb{A}$  ( $\mathbb{B}$ ) complete no later than at time  $l_t^A + l_t^B$ . In general, an arbitrary bipartite graph falls into case (iii), and thus A-TxRx should generate schedule with superframe length  $\delta(G) < l_t^A + l_t^B$ .  $\square$

## 4 Research methodology

To evaluate the performance of A-TxRx, we use MatGraph [21], a Matlab toolkit to work with simple graphs. In our experiments, all nodes are stationary and randomly located on a square area. Note that channel error is not taken into account in our experiments. In practice, retransmissions due to channel errors can be accounted for by dimensioning the transmission time accordingly. Moreover, directional transmissions tend to have high gains. Otherwise, links with poor channel condition can be omitted from the topology.

We study the impact of the following parameters on the performance of A-TxRx: node density, transmission radius, node degree and selected MISs, as well as its running time. The number of nodes ranges from 5 to 40 with an interval of 5. The transmission radius ranges from 10 to 130 m with an interval of 20 m. The network area ranges from  $50m \times 50m$  to  $250m \times 250m$ . The degree of each node varies from 2 to 10, assuming a total of 11 nodes. Five experiments are conducted with one change to the network configuration

while others are fixed; this will be made specific in the result sections later. The results are an average of 20 simulation runs, each with a different topology.

We compare A-TxRx, which represents both A-TxRx<sub>GC</sub> and A-TxRx<sub>Greedy</sub>, against 2P-1slot, 2P-node and JazzyMAC [14], where A-TxRx<sub>GC</sub> is the algorithm adopts graph coloring, and A-TxRx<sub>Greedy</sub> uses greedy searching. Specifically, for both 2P-1slot and 2P-node, the scheduling is done on time slot bases. The slot size is set to the longest air-time among the active links in that slot. For 2P-1slot, we color the conflict graph of a topology to yield its MIS. Then, we activate all the links in the MIS to transmit in the first time slot. These links then starts to receive in the following time slot. For example, in slot  $i$ , where  $i$  is an odd number, if  $e_{AB}$  is activated, then link  $e_{BA}$  will transmit in slot  $i + 1$ . For 2P-node, we perform a graph coloring on the network graph instead of the conflict graph. After the MIS with the most nodes is determined, all nodes in the MIS transmit and become receiver in the next time slot. Then, we remove the chosen nodes from the network graph and repeat the above process until all links are activated at least once. JazzyMAC also exploits TDMA but its slot length is dynamic. It is initialized centrally and then works according to the following fundamental rules: each node holds one token for each of its links. When a node finishes its transmission on a link, it passes the corresponding token to the other end of the link. A node becomes a transmitter when it holds the token for all its incident link(s). Other details of JazzyMAC can be found in [14].

In each experiment, we collected the following metrics with error bars indicate 95 % confidence interval:

- *Superframe length.* This is the total time duration for each link to transmit at least once.
- *Number of concurrent active links.* It corresponds to the average number of links that are operating concurrently at each point in time. This also includes the number of opportunistic links enabled outlined in Sect. 3.1. This metric is significant because it reflects the capacity of the WMN.
- *Computation time.* This is the time required for each algorithm to calculate the schedule for a given topology on a computer with an Intel Core i7 with 6 GB RAM.

### 5 Results

In the following sections, we present the results from our experiments concerning node density, transmission radius, node degree, running time and impact of the selected MIS.

### 5.1 Node density

In the first experiment, we study the effect of node density on superframe length and number of concurrent links. The number of nodes ranges from 5 to 40. The transmission range of each node is set to 70.

From Fig. 5a, we see that the superframe length increases when we add more nodes. The reason is that as the number of nodes increases from 5 to 40, the number of links grows from 15 to 1159. Thus, there are more interfering links at each point in time (or during every time slot). Additionally, in Fig. 5(a), we can also see that the superframe length of A-TxRx is significantly shorter than 2P-1slot and 2P-node, i.e., less than half as long. The reason is that, in 2P-1slot and 2P-node, as long as the MIS is determined, all links in the MIS are regarded as a group

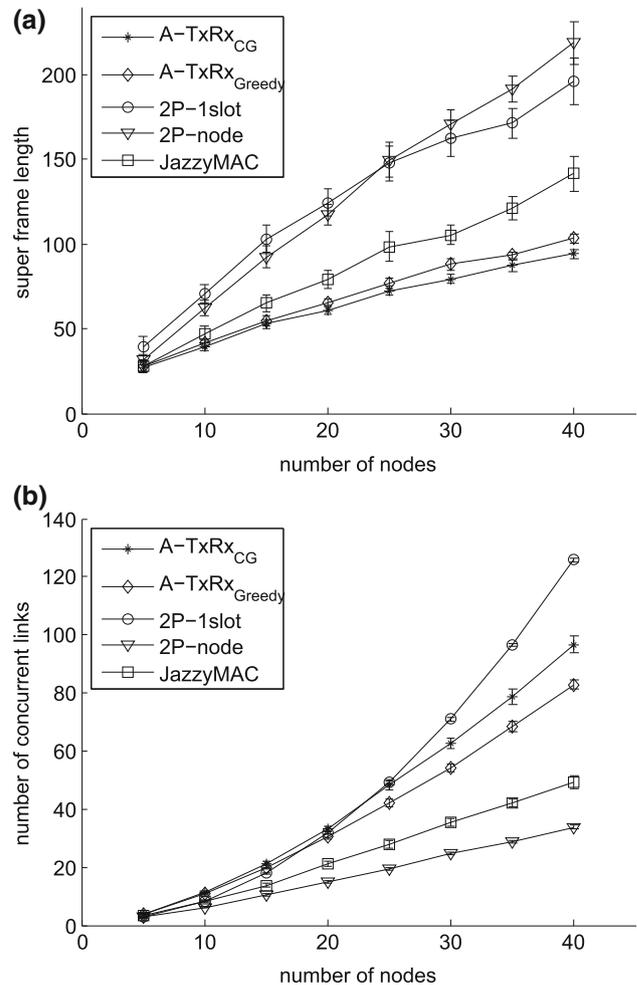


Fig. 5 Superframe length **a** and average number of active links at each time point **b** under different node densities

of links that are assigned with the transmission right in the following time slot for the same period of time to operate and finish their transmission. However, among this group of links, some links may end their transmission sooner, i.e., have a shorter air-time than the allocated slot duration. As a result, part of the channel will be idle. For A-TxRx, as long as one link finishes transmission, the largest set of new non-conflicting links are activated immediately to make maximum use of transmitting channel.

In Fig. 5(b), except for JazzyMAC, we extended 2P-1slot, 2P-node and A-TxRx by implementing opportunistic scheduling. Note that opportunistic links do not affect the superframe length. As shown, 2P-1slot produces about 25 % more concurrent links as compared to A-TxRx, especially for high density networks, i.e., when number of nodes reaches 40. The key reason is because 2P-1slot can add more opportunistic links due to the longer slot time. Both A-TxRx and 2P-1slot significantly outperform 2P-node and JazzyMAC. The reason 2P-node generates much less concurrent links is that 2P-node select transmitting links in a node bases. If one incident link of a node is determined to be an interfering link, then none of the other incident links of this node can be activated to transmit even if such links are non-interfering. For JazzyMAC, it also has less concurrent links because it is impossible to extend JazzyMAC with opportunistic scheduling.

To quantify the benefits of opportunistic scheduling, we repeat the experiments for A-TxRx. Fig. 6 compares the number of concurrent links with and without opportunistic scheduling for A-TxRx. We can see that when opportunistic scheduling is implemented, there are about 20 % more links as we add an increasing number of nodes whilst keeping the network area fixed.

Furthermore, by comparing A-TxRx<sub>GC</sub> and A-TxRx<sub>Greedy</sub>, we find that for small amount of nodes, i.e., nodes less than 15, these two algorithms have exactly the

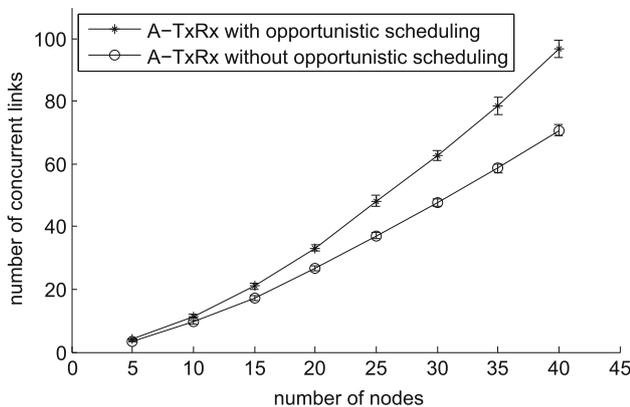


Fig. 6 Improvement in the number of concurrent links with opportunistic scheduling

same performance. However, as the number of nodes increases from 15 to 40, A-TxRx<sub>GC</sub> shows a slightly superior performance than A-TxRx<sub>Greedy</sub> of around 8 % shorter superframe length and around 10 % more concurrent links. This is because A-TxRx<sub>GC</sub> tends to activate the largest MIS among all MISs generated by graph coloring. While in A-TxRx<sub>Greedy</sub>, the set of activated links is simply a random MIS. Hence, the  $\mathcal{A}$  of A-TxRx<sub>GC</sub> is most likely to contain more links than that of A-TxRx<sub>Greedy</sub>. For this reason, A-TxRx<sub>GC</sub> activates more links in each iteration, and thus leads to a shorter superframe length.

### 5.2 Transmission radius

In this experiment, we evaluate the performance of all algorithms when we change the transmission range of nodes. There are 15 nodes located on a square area of  $100 \times 100m^2$ . The air-time range is from one to 10 time units.

Figure 7(a) shows that when the transmission range increases from 10 to 90 m, the superframe length of all four algorithms increases linearly. The superframe length of A-TxRx increases by 12 units after we increase the transmission range by a step size of 40 m.

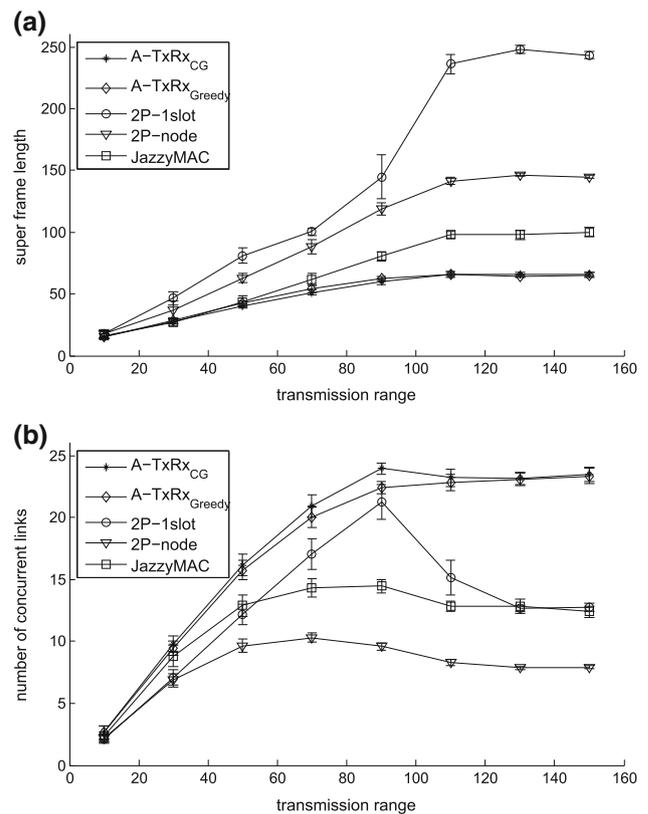


Fig. 7 Superframe length a and average number of active links at each time point b under different transmission radii

Correspondingly, 2P-1slot, 2P-node and JazzyMAC experienced an approximate increase of 30, 20, and 16 time units. For small transmission range, i.e., 10 m, the possibility of establishing a link between two nodes is small, and thus there are only an average of 6.8 links in total. On the other hand, increasing the transmission range to 110 m increases the possibility of establishing links, which also increases the superframe length since more links need to be scheduled. For the range, there are on average 208.7 links in the network, and thus the network becomes almost fully connected, whereby every node is connected with all other nodes. Thus, transmission ranges from 110 to 150 m produce almost the same conflict graph. As a result, the superframe length of A-TxRx, 2P-1slot, 2P-node and JazzyMAC becomes almost constant at values of 66, 243, 145 and 99 respectively. We can see that the difference between the superframe length of A-TxRx and that of the other three algorithms more than doubles at transmission range of 50 and 110 m. This shows that A-TxRx is particularly advantageous over the other three algorithms when the transmission range is large.

Figure 7(b) shows the average number of links that are active at each point in time under different transmission range. We can see that the number of concurrent links of A-TxRx increases and reaches its peak value of 24 when the transmission range is 90 m. With increased transmission radius, the graph becomes fully connected, resulting in the number of concurrent links fixes at around 23, regardless of the randomness in nodes distribution. However, for 2P-1slot, after it reaches a peak value of 21 at 90 m, the number of links notably decreases to 15, and then becomes steady around 12. This is because for a fully connected graph, excluding opportunistic links, when 2P-1slot first performs graph coloring, only 14 links can be activated. To be specific, these 14 links originated from 14 different nodes and are directed to the same destination node, say node A. After removing these 14 links, node A becomes disconnected from the network graph. As a result, during the second graph coloring process, there will only be 13 active links. Hence, the number of transmitting links decreases from 14 links (slot one) to one link (last slot). Thus, the average number of concurrent links is less than  $\sum_{i=1}^{i=14} i = 7.5$ . Although opportunistic links can be added to increase the total number of links in each slot, only the links with air-time that is less than the slot duration can be selected and activated in the current slot.

### 5.3 Node degree

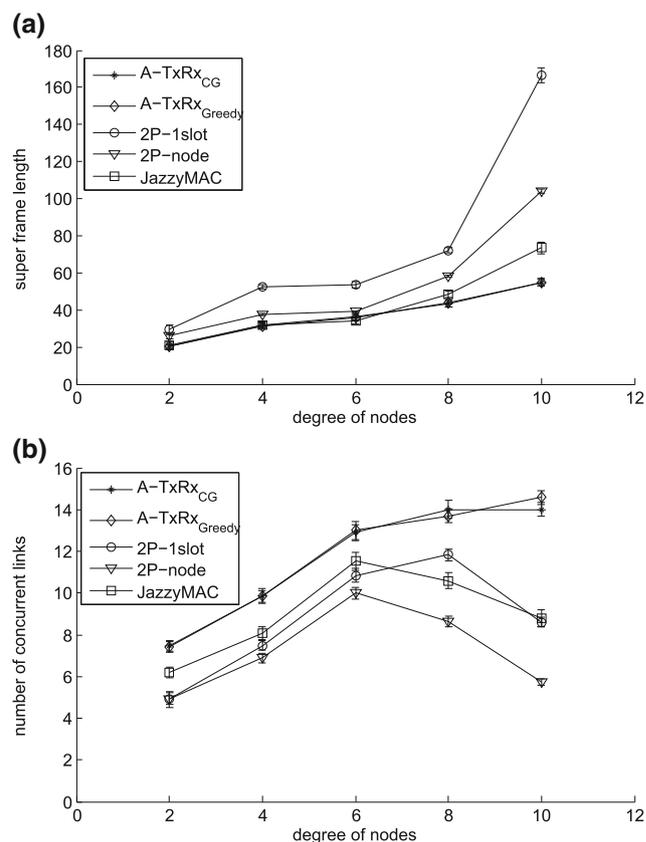
In this experiment, the number of nodes is set to 11. The degree of all nodes is the same and varies from two to 10. We study the relationship between node degree and

superframe length, and also the influence of superframe length and total number of links on delays.

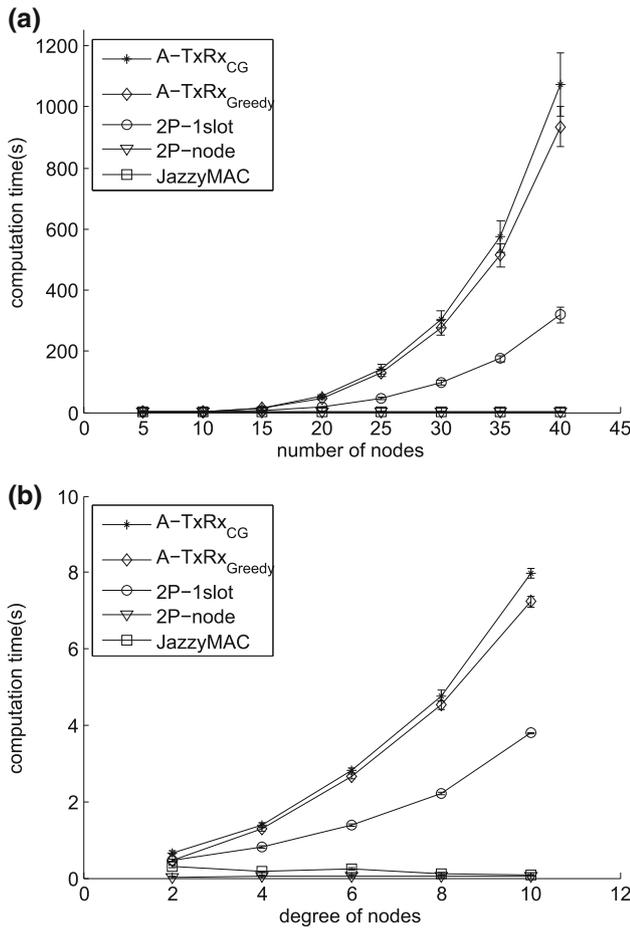
Figure 8(a) shows that the superframe length of A-TxRx increases linearly from 18 to 56 as the node degree rises. This is because the total number of links  $d \times n$  is a linear function, where  $d$  represents the degree of nodes that increases from two to 10, and  $n$  is the number of nodes that is set to 11. Interestingly, we can see that when the number of degrees of each node ranges from two to eight, the superframe lengths of JazzyMAC and A-TxRx are very close to each other. Then as the node degree increases from eight to 10, the superframe length of 2P-1slot, 2P-node and JazzyMAC rises by a dramatic 130, 80 and 46 % respectively. This can be explained using Fig. 8(b). We see that there are significantly fewer number of concurrent links for 2P-1slot, 2P-node and JazzyMAC when node degree increases from eight to 10.

### 5.4 Computation time

We now measure and compare the computation time required for each algorithm to compute the schedule for a given topology. Fig. 9(a, b) have the same network configuration as used in Sect. 5.1 and 5.3 respectively.



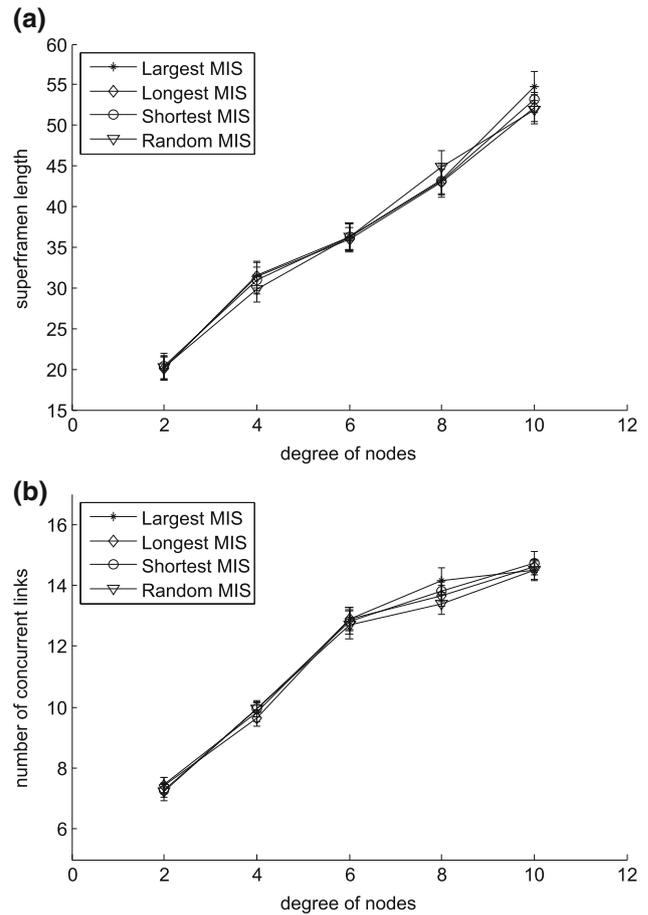
**Fig. 8** Superframe length **a** and average number of active links **b** under different node degrees



**Fig. 9** Computation time under different node densities **a** and different node degrees **b**

Figure 9(a) shows the computation time of each algorithm under different number of nodes. Note, the time required for 2P-node and JazzyMAC is less than 0.1 s, which is much smaller than A-TxRx. In particular, when there are 40 nodes, the computation time of 2P-node and JazzyMAC is only 0.02 % of that of A-TxRx. The main reason is because 2P-node and JazzyMAC runs a graph coloring algorithm on the network topology as opposed to its corresponding conflict graph, which contains up to  $|E|$  nodes and  $\frac{|E|(|E|+1)}{2}$  edges. In addition, 2P-1slot also has a faster running time than A-TxRx, i.e., up to three times faster. This is because 2P-1slot only runs every other slot. Specifically, after computing the transmitters of slot  $i$ , these transmitters become receivers in slot  $i + 1$ .

Figure 9(b) shows the algorithm running time with increasing node degrees. We see that the running time for all algorithms increases as the node degree rises. This is because the existence of more links. To be specific, 22 more links are established as node degree increases by two. Interestingly, the computation time for A-TxRx increases



**Fig. 10** Superframe length **a** and number of concurrent links **b** under different node degrees

from 1 to 3 s. This is due to the number of edges in the conflict graph, which increases by 30 times from 31 to 995. Hence, the graph coloring process requires more time, which incurs a longer computation time.

We find that A-TxRx<sub>Greedy</sub> is on average 10 % faster than A-TxRx<sub>CG</sub>. The reason is that the time complexity of the function Greedy() in line 10 and 26 of A-TxRx<sub>Greedy</sub> is  $O(|V|^2)$  as compared to MaxMIS(), which has a run time of  $O(|V|^3)$ .

### 5.5 Impact of choosing different MIS for A-TxRx<sub>CG</sub>

An interesting question that arises is whether the selection of different MISs of A-TxRx<sub>CG</sub> have any influence on the superframe length and the number of concurrent links. To answer this question, we compare the superframe length and number of concurrent links for the following cases: (i) largest MIS, which has the largest cardinality; (ii) longest MIS, which contains the link with the longest air-time; (iii) shortest MIS; and (iv) random MIS.

Figure 10(a, b) show the superframe length and number of concurrent links when the degree of each node increases from 2 to 10 for a network of 11 nodes. Observe that the choice of MIS does not have any significant impact on the performance of A-TxRx<sub>GC</sub>. The reason is that A-TxRx<sub>GC</sub> repeatedly performs graph coloring on the updated network graph. In this way, links are almost evenly assigned into different MISs. Thus, the disparity between the size or air-time length of the MISs is too slight to make a difference. Therefore, A-TxRx<sub>GC</sub> is insensitive to MIS selection.

## 6 Conclusion

This paper has studied link scheduling in MTR WMNs whereby nodes have the capability to form multiple links simultaneously. We propose a novel scheduling algorithm called A-TxRx that maximizes the number of concurrent transmissions at any point of time to boost network capacity as well as to minimize superframe length. A-TxRx is the first centralized algorithm that schedules links with different link weights on a general network topology. For MTR WMNs where the air-time for each link is given as a link weight, A-TxRx activates links whenever a link finishes transmission, whereby it adds links that are not in conflict with on-going transmissions/receptions. The results show that this rule enables A-TxRx to yield smaller superframe lengths, and hence higher network capacity, as compared to state-of-the-art approaches. Specifically, our results show A-TxRx to have superior performance with up to 70 % shorter superframe lengths and 60 % more concurrent links as compared to 2P and JazzyMAC. As a future work, we intend to investigate scenarios when there are insufficient directional antennas for a node to communicate with all neighbors. Another possible direction is to develop a distributed algorithm that matches the performance of A-TxRx. Lastly, it will be interesting to consider admission control; e.g., [22].

## References

- Shihab, E., Cai, L., Wan, F., Gulliver, A., & Tin, N. (2008). Wireless mesh networks for in-home iptv distribution. *IEEE Network*, 22, 52–57.
- Nandiraju, N., Nandiraju, D., Santhanam, L., He, B., Wang, J., & Agrawal, D. (2007). Wireless mesh networks: Current challenges and future directions of web-in-the-sky. *IEEE Wireless Communications*, 14, 79–89.
- Lee, M., Zheng, J., Ko, Y.-B., & Shrestha, D. (2006). Emerging standards for wireless mesh technology. *IEEE Wireless Communications*, 13, 56–63.
- Liu, C., Gkelias, A., Hou, Y., & Leung, K. (2009). Cross-layer design for QoS in wireless mesh networks. *Springer Wireless Personal Communications*, 51(3), 593–613.
- Chin, K. (2008). A new link scheduling algorithm for concurrent Tx/Rx wireless mesh networks. In *IEEE ICC* (pp. 3050–3054). Beijing, China.
- Patra, R., Nedeveschi, S., Surana, S., Sheth, A., Subramanian, L., & Brewer, E. (2007). WiLDNet: design and implementation of high performance WiFi based long distance networks. In *USENIX Symposium on Networked Systems Design & Implementation* (pp. 87–100). Cambridge, MA, USA.
- Dutta, P., Jaiswal, S., & Rastogi, R. (2007). Routing and channel allocation in rural wireless mesh networks. In *IEEE INFOCOM* (pp. 598–606) Washington, DC, USA.
- Wu, D., Yang, S., Bao, L., & Liu, C. (2013). Joint multi-radio multi-channel assignment, scheduling and routing in wireless mesh networks. *Springer Wireless Networks*, 20(1), 4–16.
- Raman, B., & Chebrolu, K. (Aug. 2005). Design and evaluation of a new MAC protocol for long-distance 802.11 mesh networks. In *ACM MOBICOM*. (pp. 156–169). Cologne, Germany.
- Chin, K., Soh, S., & Meng, C. (2010). A novel spatial TDMA scheduler for concurrent transmit/receive wireless mesh networks. In *IEEE AINA*, Perth, WA, Australia.
- Chin, K., Soh, S., & Meng, C. (2012). Novel scheduling algorithms for concurrent transmit/receive wireless mesh networks. *Computer Networks*, 56, 1200–1214.
- Chin, K., Soh, S., & Meng, C. (2012). A novel scheduler for concurrent Tx/Rx wireless mesh networks with weighted links. *IEEE Communications Letters*, 16(2), 246–248.
- Dai, H., Liew, S., & Fu, L. (Oct. 2011). Link scheduling in multi-transmit-receive wireless networks. In *IEEE Conference on Local Computer Networks* (pp. 199–202). Bonn, Germany.
- Nedeveschi, S., Patra, R., Surana, S., Ratnasamy, S., Subramanian, L., & Brewer, E. (2008). An adaptive, high performance mac for long-distance multihop wireless networks. In *ACM MOBICOM*. (pp. 259–270). New York, NY, USA.
- Sundaresan, K., Sivakumar, R., Ingram, M.-A., & Chang, T.-Y. (2004). Medium access control in ad hoc networks with mimo links: Optimization considerations and algorithms. *IEEE Transactions on Mobile Computing*, 3, 350–365.
- Shepard, C., Yu, H., an dE. Li, N. A., Marzetta, T., Yang, R., & Zhong, L. (2012). Argos: Practical many-antenna base stations. In *ACM MOBICOM Istanbul, Turkey*.
- Bleskei, H., Gesbert, D., Papadias, C. B., & v. d. Veen, A.-J. (2006). *Space-time wireless systems: From array processing to MIMO communications*. Cambridge: Cambridge University Press.
- Mudumbai, R., Singh, S., & Madhow, U. (Apr. 2009). Medium access control for 60 GHz outdoor mesh networks with highly directional links. In *28th IEEE International Conference on Computer Communications (INFOCOM) Rio de Janeiro, Brazil*.
- Jain, K., Padhye, J., Padmanabhan, V., & Qiu, L. (2003). Impact of interference on multi-hop wireless network performance. In *ACM MOBICOM* (pp. 66–80). New York, NY, USA.
- Matula, D. W., & Beck, L. L. (1983). Smallest-last ordering and clustering and graph coloring algorithms. *Journal of the ACM (JACM)*, 30, 417–427.
- Scheinerman, E. R. (2008). Matgraph: a MATLAB toolbox for graph theory. *Department of applied mathematics and statistics, the Johns Hopkins University, Baltimore, Maryland* (pp. 1–7).
- Liu, C., Leung, K., & Gkelias, A. (2014). A generic admission control methodology for packet networks. *IEEE Transactions on Wireless Communications*, 13, 604–617.



**Yuanhuizi Xu** received her Bachelor of Engineering in Telecommunications with first class Honours from the University of Wollongong in 2011. She is currently a Ph.D. candidate at the same university.



**Kwan-Wu Chin** is an Associate Professor at the University of Wollongong. He obtained his Bachelor of Science with First Class Honours from the Curtin University of Technology, Australia. He then pursued his Ph.D. at the same university, where he graduated with distinction and the vice-chancellors commendation. After obtaining his Ph.D., he joined Motorola Research Lab as a senior research engineer, where he developed zero configuration

home networking protocols and designed new medium access control protocols for wireless sensor networks and next generation bandwidth managers. In addition, he filed nine US patents and won a major grant from DARPA. In 2004 he joined the University of Wollongong as a Senior Lecturer before being promoted to the rank of Associate Professor in 2011. He is also the Head of Postgraduate Studies for the School of Electrical, Computer and Telecommunications Engineering, and the director of the Wireless Technologies Lab (WTL). His current research areas include medium access control protocols for wireless networks, resource allocation algorithms/policies for communications networks, routing protocols for delay tolerant networks

and mathematical programming. To date, he holds four US patents, and has published more than 100 conference and journal articles.



**Sieteng Soh** received the B.S. degree in electrical engineering from the University of Wisconsin-Madison and the M.S. and Ph.D. degrees in electrical engineering from Louisiana State University, Baton Rouge. From 1993 to 2000, he was a faculty member at Tarumanagara University, Indonesia, where he was the director of the Research Institute from 1998 to 2000. He is currently a lecturer with the Department of Computing at Curtin University of Technology, Perth, Western Australia. His research interests include network reliability, and parallel and distributed processing. He is a member of the IEEE and the IEEE Computer Society.



**Raad Raad** is a lecturer at the University of Wollongong where he has been there since 2004. Previously he held the position of Staff Research Engineer at Motorola Australia Research Labs, Sydney, Australia between 2001 and 2004. He has been awarded a number of patents by the US patent office for his work on wireless sensor networks and IEEE 802.11 WLAN. He has conducted research into a number of areas that include admission

control, Fuzzy Logic, Bandwidth Management, wireless sensor networks, and RFIDs.