



# A traffic-aware scheduling algorithm for IEEE 802.16 mesh mode

F. Dabiran<sup>a</sup>, H.R. Rabiee<sup>a,\*</sup>, M. Salehi<sup>b</sup> and A.A. Amidian<sup>c</sup>

a. Department of Computer Engineering, Sharif University of Technology, Tehran, Iran.

b. Faculty of New Sciences and Technologies, University of Tehran, Tehran, P.O. Box 14395-1374, Iran.

c. Department of Communication, Faculty of Applied Science of Post and Communication, Tehran, P.O. Box 44484005, Iran.

Received 18 February 2012; received in revised form 16 July 2013; accepted 9 September 2013

## KEYWORDS

IEEE 802.16 mesh mode;  
 Slot allocation;  
 Centralized scheduling;  
 Download traffic;  
 Concurrent transmission.

**Abstract.** Since there is no difference between uplink and downlink subframes in the IEEE 802.16 mesh mode; both downlink and uplink traffics are transmitted within a single time frame. Hence, in most scheduling methods, only one scheduling algorithm is used for both uplink and downlink traffic. However, because of the different characteristics of uplink and downlink traffic, different scheduling methods should be used for each of the traffic types. In this paper, we focus on the mesh centralized scheduling of downlink flow in the data subframe. After comprehensive analysis of the characteristics of downlink traffic, we propose a new algorithm, called Traffic-Aware Scheduling (TAS). In this algorithm, the downlink traffic distribution is tuned for maximum concurrent transmission rate. This goal is achieved by choosing different link selection criteria, such as maximum demandant sender, farthest receiver and least interfered path, based on the downlink traffic analysis results. The simulation results show that our algorithm outperforms existing methods in terms of scheduling length, link concurrency, and throughput (about 13.7% in average) for the downlink traffic. Moreover, the proposed algorithm is scalable. In particular, on average, 3% improvement is achieved in terms of scheduling length at higher traffic loads and in the number of nodes.

© 2014 Sharif University of Technology. All rights reserved.

## 1. Introduction

The IEEE 802.16 is a broadband wireless communication standard for the development and deployment of data transmission networks such as internet, VPN and VoIP in metropolitan areas. Features such as high bandwidth, easy installation and inexpensive maintenance charge have made IEEE 802.16 a suitable substitute for wired and cabled modem networks.

The initial IEEE 802.16-2004 [1] standard was developed to serve fixed Subscriber Stations (SSs)

through a central Base Station (BS) using a Point to Multi Point (PMP) mode, and the mesh mode was provided as an additional operating topology. The main difference between these two modes is that in the PMP mode, the BS acts as a coordinator and relays all the communications, and any SS that has a direct link with BS has to communicate with BS first before transmitting data to other SSs. But, in the mesh mode, any SS can send its data packets to BS through sponsor or relay nodes. Hence, the mesh topology not only increases wireless coverage and network throughput [2], but also provides features such as lower backhaul deployment cost, rapid deployability and re-configurability [3].

Employing appropriate scheduling algorithms is becoming a challenging task in improving the Quality of Service (QoS) in interactive applications over IEEE

\*. Corresponding author. Tel.: +98 21 66046719;

Fax: +98 21 66047662

E-mail addresses: dabiran@ce.sharif.edu (F. Dabiran);  
 rabiee@sharif.edu (H.R. Rabiee); mostafa\_salehi@ut.ac.ir  
 (M. Salehi); amidian@ictfaculty.ir (A.A. Amidian)

802.16 [4]. In the mesh mode, channel multiplexing is TDMA-based and each time frame is divided into mini-slots. An SS uses these time slots to send and receive its data packets. A scheduler computes the range and position of mini-slots and allocates them based on SS bandwidth requests in an interference-free manner. Since there is no difference between uplink and downlink subframes in the mesh mode, both down and uplink traffic is transmitted within a single time frame. Here, downlink (DL) refers to data transmission from the BS to the SS, and uplink (UL) to data transmission in the reverse direction. Hence, in most scheduling methods, only one scheduling algorithm is used for both up and downlink traffic. However, uplink and downlink flows have different characteristics, and, therefore, by considering each of the flow characteristics in the scheduling algorithm, more efficient networks can be expected.

In this paper, we propose a centralized scheduling algorithm for downlink traffic based on an analytical study of traffic distribution in the mesh mode. Consideration of the characteristics of downlink flow and the use of multiple criteria for choosing the sender node are among the main features of the proposed algorithm. To the best of our knowledge, this is the first centralized scheduling algorithm in mesh mode that considers multiple criteria in scheduling downlink traffic. The experimental results show that the proposed algorithm has high throughput, reduction in scheduling length, and more concurrent transmission, compared to the state of the art algorithms. The rest of the paper is organized as follows. Section 2 provides a discussion on related works. The scheduling mechanism in the IEEE 802.16 mesh mode is described in Section 3. Section 4 presents the proposed algorithm, along with a discussion on interference modeling and downlink flow analysis. Section 5 provides the performance evaluation, and the concluding remarks are provided in Section 6.

## 2. Related work

Early work on the IEEE 802.16 standard has primarily focused on the PMP mode [5–8]. Studies on mesh mode have been devoted to tree construction [3,9,10], packet scheduling [11–14], spatial reuse [15–17] and QoS support [18–21]. It has been proved that the problem of finding the minimum scheduling length in centralized scheduling is NP-complete [3,18,22]. Therefore, some heuristic algorithms have been developed to solve this problem.

The authors in [18] and [20] proposed a QoS algorithm to allocate bandwidths to various types of flow, separately. In [12], the authors introduce an Enhanced-Frame Registry Tree Scheduler (E-FRTS) that uses the frame registry tree; a data structure that

aims to prepare the time frame creation and to reduce processing needs at the beginning of each frame.

Several studies have addressed the issue of spectral reuse to solve the resource allocation problem in the context of concurrency. In [23], the scheduler uses four separate criteria (*random node*, *min interference node*, *nearest to BS* and *farthest to BS*) for selecting the scheduled link. The results show that the nearest criterion has the best performance in scheduling length for the uplink traffic. In [24], the authors have improved the scheduler in [23], and have used multiple measures in scheduling, such as *node hop count*, *amount of bandwidth request* and *interference*.

In [3], the authors proposed three different scheduling algorithms. The *Fair Queuing* algorithm schedules the maximum number of packets from the entire network in each time slot. The *Max Weight* algorithm schedules the layer closest to the BS first, before scheduling the other layers. The *Line Scheduling* algorithm considers the fairness of each node. This algorithm uses a greedy approach to schedule the network after scanning the entire network. In [25], a weighted time slot scheduler is designed for IEEE 802.16 multi-hop relay networks to improve network throughput and guarantee the desired frame length. It independently assigns time slots to uplinks and downlinks, in accordance with the specifications of IEEE 802.16j in which both links may become active in different sub-frames.

Table 1 provides a comparison of the current centralized scheduling schemes in the IEEE 802.16 mesh mode. According to this table, most previous work has focused on the design of scheduling algorithms for uplink traffic. Little work has been done on downlink scheduling by considering the characteristics of this kind of traffic, such as the source and destination of flow, interference model and concurrent transmission. Because of the different characteristics of uplink and downlink traffic, it is more reasonable to employ different scheduling methods for each of the traffic types.

In downlink scheduling, the BS is responsible for allocating the resources to the subscribing stations. In [26], a scheduler has been designed for downlink video traffic, but it does not efficiently incorporate an interference model. The algorithm proposed in [27] is the only scheme that considers downlink scheduling with concurrent transmission. However, this work does not provide any analytical analysis on the downlink flows. Moreover, [27] shows that the scheduling algorithm introduced in [23] for downlink traffic has a lower performance than for uplink traffic.

In this paper, we focus on the centralized scheduling of downlink traffic flow in the IEEE 802.16 mesh mode. We propose a new algorithm by considering the characteristics of this type of traffic flow, which

**Table 1.** Comparison of centralized scheduling studies in the IEEE 802.16 mesh mode (up: uplink and down: downlink).

Reference/ year	Concurrent transmission	Algorithm design direction	Downlink deployment	Simulation direction
[11]/2006	✓	up	×	up
[18]/2006	×	up/down	✓	up
[31,23]/2007	✓	up	✓	up
[3]/2007	✓	up	✓	up
[12]/2008	×	up	✓	up
[20]/2008	×	up/down	✓	up
[24]/2008	✓	up	✓	up
[26]/2008	×	down	✓	down
[27]/2009	✓	up/down	✓	up/down
[32]/2009	✓	up	✓	up/down
[13,21]/2009	×	up/down	✓	up
[25]/2010	✓	up/down	✓	up/down
[14]/2010	✓	up	✓	up

improves concurrent transmission rate and network throughput.

### 3. Overview of the IEEE 802.16 mesh mode

A mesh frame is partitioned into two subframes: control and data. The control subframe itself consists of schedule control and the network control subframe. The network control subframe is designed to manage the joining a new node to the mesh network. The schedule control subframe contains symbols that are introduced for centralized or distributed scheduling requests. The data subframe is used for data transmission and includes multiple mini-slots. This subframe is partitioned into centralized and distributed data.

There are two kinds of scheduling in the mesh mode: centralized and distributed. In centralized scheduling, the BS coordinates all the bandwidth requests within the network. For backhaul applications, such as Internet, centralized scheduling is preferable because all traffic is to or from the BS. Each node in a distributed scheduling scheme, including the BS, coordinates the specification of its transmission, such as its resources and bandwidth request with a two-hop neighborhood. Moreover, distributed scheduling can be established by directed uncoordinated requests between two nodes. Although distributed scheduling schemes are more scalable, they are inefficient in the QoS guarantee and more complicated than centralized ones. This is used only for intranet traffic [28]. Based on the scope of this paper, we will focus on centralized scheduling in the mesh mode. For distributed scheduling, we refer interested readers to [1,29,30].

As mentioned above, a frame includes control and data subframes, both of which need to schedule. A con-

trol and data scheduler is used to manage bandwidth request and data, respectively. In control scheduling, the BS sends the scheduling tree to the subscribing stations using a MSH-CSCF message. This process defines the order of the subscribing stations in sending their bandwidth requests. A Breadth-First Search (BFS) algorithm is used to create a scheduling tree. In data scheduling, the BS schedules the transmission of each subscribing station by allocating the mini-slots of data subframes. Data scheduling is not introduced by the IEEE 802.16 standard. Moreover, concurrent transmission is not considered by the standard and, in each mini-slot, only one node can transmit data. Since using concurrent transmission can significantly lower scheduling length and will not increase computational overhead for the scheduling algorithm notably, this method has, recently, been even more considered.

### 4. The proposed algorithm

As mentioned before, there are differences between uplink and downlink traffic, which necessarily should be considered in defining scheduling algorithms. Therefore, in this section, first, uplink and downlink traffic are going to be analyzed and compared. Then, based on the results, the proposed scheduling algorithm is introduced.

#### 4.1. Uplink and downlink traffic distribution

The distribution of traffic load in an 802.16 network is one of the important differences between uplink and downlink traffic. In uplink traffic, the load is distributed in the network, initially, and gradually concentrates on BS. However, in downlink, the load is

concentrated in BS and progressively distributes in the network. If simultaneous transmission is not possible, scheduling lengths for downlink and uplink will be the same, because the load is equal in both directions and gets, transmitted over the same path. In the case of simultaneous transmission, scheduling length varies based on the number of simultaneous transmissions. It is important to note that the concentration or dispersion of traffic in the network gives a measure of concurrent transmission rate and scheduling length. Most scheduling algorithms consider concurrency to decrease scheduling length, while these algorithms are used for uplink traffic.

In uplink scheduling, the load is initially distributed through the entire network. Therefore, at the beginning of scheduling, the possibility of simultaneous transmissions (the concurrency rate) is high. This rate decreases progressively as the load gets closer to the BS. Therefore, it is necessary to consider this issue to design an efficient uplink algorithm.

Since, in downlink scheduling, the network load is initially concentrated in the BS, the scheduling algorithm should distribute the load in such a way as to increase the possibility of simultaneous transmissions. In fact, downlink scheduling is more difficult than uplink. In uplink scheduling, the concurrency exists inherently. However, in downlink scheduling, it happens gradually as a result of load distribution in the network. Therefore, an efficient scheduling algorithm for downlink traffic must consider the distribution of load in the network to achieve the maximum possibility of simultaneous transmission.

#### 4.2. Interference model

Since multiple SSs may be allowed to transmit in the same mini-slot, a time slot can be reused by multiple SSs, as long as the SSs do not interfere with each other. This property is called spatial reuse, which can be used to increase the capacity of the wireless mesh network by decreasing the length of scheduling. Wireless links interfere with other links, if their packets collide in simultaneous transmissions. There are four types of transmission interference that need to be considered in TDMA networks. Figure 1 shows a complete interference model. The first three types of conflict are between the links that share a neighbor. In case 1, a single transmitter cannot separate packets intended for the two different receivers. In Case 2, the parallel transmissions garble each other at the common receiver. Case 3 cause conflict because the nodes cannot transmit and receive at the same time. In case 4, the two interfering links are shown with a solid line. Since the two transmitters share a neighbor hearing both transmissions, they cannot transmit at the same time, which is shown by a dashed line for the overheard transmission. The first three cases are called, *Primary*

*Interference*, and the fourth case is called, *Secondary Interference*.

The collision between two sender (or receiver) nodes in a downlink transmission can be shown in a matrix. The collision matrix is an  $n \times n$  binary matrix ( $n$  represents the number of nodes) that is obtained from the network topology and interference model, regardless of the status of bandwidth requests of the nodes. If nodes  $a$  and  $b$  interfere with each other, the entries  $(a, b)$  and  $(b, a)$  of this matrix are 1, otherwise they are 0. This matrix will be updated only when a node joins or leaves the network.

#### 4.3. Downlink flow analysis

The number of simultaneous transmissions in downlink scheduling varies, based on network topologies. In Figure 2, various possible structures for a 5-node network

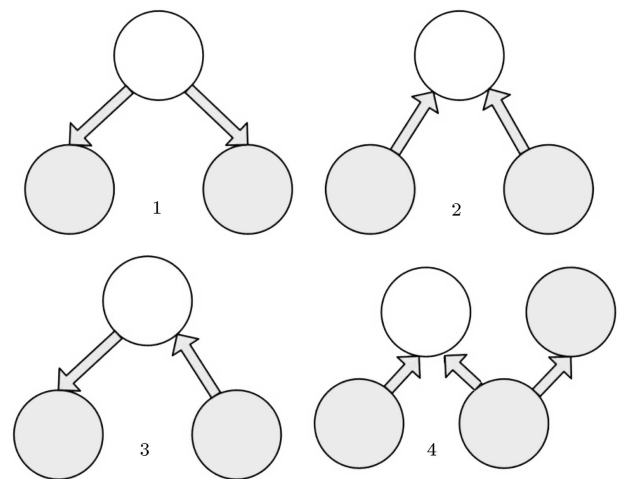


Figure 1. Interference model in the wireless mesh network.

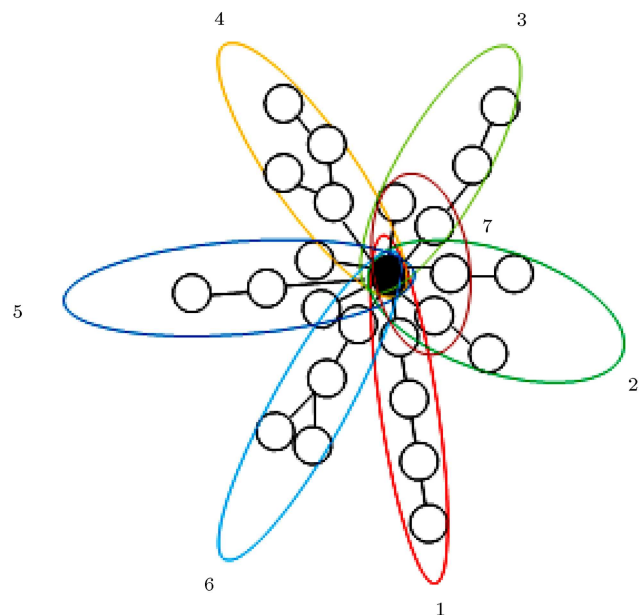


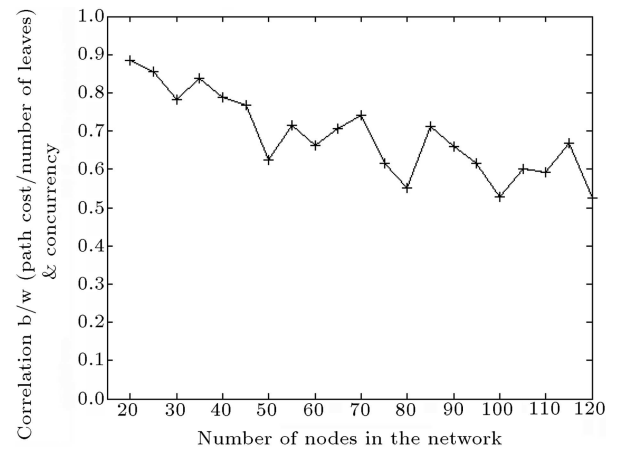
Figure 2. Various structures of a 5-node network.

**Table 2.** Concurrency in 5-node structures.

Structure number	Downlink concurrency opportunity	Number of leaves	Path cost
1	1	1	10
2	1	2	6
3	1	2	6
4	0	2	8
5	0	3	5
6	0	2	9
7	0	4	4

are illustrated. In these structures, in order to compute the number of possible concurrent transmissions, the nodes have been presumed in the best possible situation. It means that there are no links between the nodes other than the connection links designated in the figure. Any extra link will result in a need to lower the number of simultaneous transmissions, and since, in this section, the goal is to determine the maximum number of those transmissions, there is no need to consider the other links. Table 2 shows that there are more simultaneous transmissions in a *chain* (structure 1) and *semi-chain* structures (structures 2 and 3) than in other structures in the network. Semi-chain structures are similar to a chain with more leaves, and have rather saved the shape of a chain. For example, both structures 2 and 4 have two leaves, but structure 2 has two chains with the most connections to the BS. Therefore, it is called a semi-chain structure.

In order to obtain a better understanding, the relation between the number of leaves, path cost (the sum of distance of all nodes to BS) and the number of possible concurrent transmissions in downlink has been studied through simulation. In this simulation, for every randomly generated network with a determined number of nodes, a network tree is derived and the number of leaves, path cost and collision matrix in downlink are computed. Let  $X$  represent the path cost of a network,  $Y$  denotes the number of leaves and  $Z$  is the number of zeros in the collision matrix. Due to Figure 2 and Table 2, the greater node distance to the BS (higher  $X$  values) and the fewer network leaves (less  $Y$  values) makes more likely the network chain or semi-chain structure. So, the higher value of  $(X/Y)$  (more  $X$  and less  $Y$  values) shows that the network topology is closer to a chain pattern. By computing the correlation between  $(X/Y)$  and  $Z$  (Figure 3), we can study the relation between simultaneous transmissions and the network topology in downlink scheduling. The correlation ratio will be a value within a  $[-1,1]$  interval. The closer this value is to 1, the correlation between

**Figure 3.** Correlation between path cost/number of leaves and concurrency.

parameters is more and in direct relation. Figure 3 shows that there is a direct relation between chain or semi-chain structures and the number of concurrent transmissions in a network. For small size networks, this correlation is high (close to 1), but decreases by increasing the number of nodes. It is because of the high interference in large and accumulated networks.

According to Figure 3, if the structure of a network is chain or semi-chain, there is a higher probability of finding more concurrent transmissions. Despite the opportunity of having a chain structure is low, the chain or semi-chain structures can be extracted from the network topology. These paths can increase concurrency in the network. In the proposed algorithm, network structure is considered and the load is distributed through semi-chain paths to increase the chance of simultaneous transmission.

#### 4.4. Downlink scheduling formulation

In [3], an Integer Linear Programming (ILP) model is proposed for uplink scheduling in the mesh mode. Here, we use this model and modify it for downlink scheduling, i.e. by allocating mini-slots of the data subframe to the sender nodes in an interference-free manner. The mesh network is composed of one BS and several SSs. Let node  $v_0$  denote BS and node  $v_i$  ( $1 \leq i \leq n$ ) denote each SS, where  $n$  is the number of SSs. We assume that in adjacency matrix,  $E_{n \times n}$ , we have  $E_{ij} = E_{ji} = 1$  (i.e., links are bidirectional) if and only if nodes  $v_i$  and  $v_j$  are connected, and 0 otherwise.

The number of packets to be delivered to node  $v_i$  through downlink is denoted by  $w_i$ . We assume that the network topology is fixed over scheduling time, and the upper bound of scheduling,  $U$ , is known. With these assumptions, the modeling problem can be stated as follows:

**Variables.** We introduce the following variables to represent the routing tree:

The routing matrix is presented by  $R$ . In this matrix, we have  $R_{ij} = 1$  if  $v_j$  is the parent of  $v_i$  in the routing tree, and 0, otherwise. As the routing tree is a subgraph of the network graph, we have:

$$R_{ij} \leq E_{ij} \quad \forall i \in \{1 \dots n\}, \forall j \in \{0 \dots n\}. \quad (1)$$

Since each node can have only one parent, we have:

$$\sum_{j=0}^n R_{ij} = 1 \quad \forall i \in \{1 \dots n\}. \quad (2)$$

We define binary variable  $X_{ijt}$  by the following constraints:  $1 \leq i \leq n$ ,  $0 \leq j \leq n$  and  $1 \leq t \leq U$ . Each binary variable,  $X_{ijt}$ , takes values defined as follows.  $X_{ijt}$  is 1 if  $v_i$  receives a data packet from  $v_j$  in time slot  $t$ , and 0, otherwise. Since each node has only one parent and can receive packets in the same way, we have:

$$X_{ijt} \leq R_{ij}, \quad (3)$$

$$\forall i \in \{1 \dots n\}, \forall j \in \{0 \dots n\}, \forall t \in \{1 \dots U\}.$$

The interference constraints can be mathematically captured for all pairs of edges  $(i_1, j_1)$  and  $(i_2, j_2)$  in time slot  $t$ , as follows:

$$X_{i_1 j_1 t} + X_{i_2 j_2 t} \leq 1. \quad (4)$$

Let  $w_{it}$  be the number of packets that have not reached  $v_i$  at the end of time slot  $t$ . If node  $v_i$  receives one packet from node  $v_j$  in time slot  $t$ , the number of received packets at  $v_j$  increases by 1. Thus, for each node,  $v_i$ , the packet flow constraints can be specified as:

$$w_{it} = w_{i(t-1)} + X_{ijt} - \sum_k X_{kit}. \quad (5)$$

Let  $A_t$  be the total number of packets that have not yet reached their destination at the end of time slot  $t$ . Therefore,  $A_t$  can be presented as follows:

$$A_t = \sum_{i=1}^n w_{it}. \quad (6)$$

Then, the problem is to find a scheduling, such that  $t$  is minimized, where  $A_t = 0$ . We introduce  $U$  binary variables,  $Y_t$ , where  $1 \leq t \leq U$ , to simplify our scheduling problem, and add the following constraints:

$$\sum_{t=1}^U Y_t = 1, \quad A_t \leq A_0(1 - Y_t) \quad \forall t \in \{1 \dots U\}. \quad (7)$$

These two equations together imply that there is exactly one  $Y_t = 1$ , and that must happen for some timeslot for which  $A_t = 0$ .

**Objective function.** The objective function is expressed as follows:

$$\text{Minimize } \sum_{t=1}^U tY_t. \quad (8)$$

$$\text{Subject to: } \sum_{t=1}^U Y_t = 1,$$

$$A_t \leq A_0(1 - Y_t) \quad \forall t \in \{1 \dots U\}.$$

**Complexity analysis.** The complete problem consists of  $O(n^2U)$  variables and  $O(n^4U)$  constraints. It is a NP-complete problem [3]. This means that the time required to solve this problem is significant. Therefore, in the following section, we propose a heuristic algorithm to solve the downlink scheduling problem.

#### 4.5. The proposed algorithm

The proposed algorithm called *Traffic-Aware Scheduling (TAS)* can be compared with the introduced algorithm in [27] to schedule downlink traffic. The greedy algorithm of [27] is based on the distance of a node from the BS. However, our proposed algorithm is based on downlink flow analysis and uses multiple measures in scheduling. To this end, a special matrix, called the *Request Matrix*, is introduced. In every timeslot, this matrix can be defined by the following equation:

$$D_{ijt} = k,$$

if node  $i$  has  $k$  data packet for node  $j$  then:

$$\text{s.t. } \forall j \in \{1, \dots, n\}, \text{ time slot } t, \forall i \in \{0, \dots, n\}. \quad (9)$$

In the first time slot, the whole load is concentrated in the BS. Therefore, in this slot, only the first line of the request matrix has non-zero value. For each transmission in each time slot, one unit will be deducted from the entry related to the specified (sender, receiver) pair in the request matrix, and one unit will be added to the (relay, receiver) pair. Therefore, the value of the entries in the first line of this matrix decreases progressively and the traffic load is distributed over the other lines.

The proposed algorithm consists of two phases (refer to Algorithm 1): selection of sender node, and selection of simultaneous sender nodes. In the first phase, in each time slot,  $t$ , the criterion for selection of sender node  $v_i$  is as follows:

1. The sender with the maximum number of transmission requests is selected in the current time slot.
2. If there exists more than one node in this situation, the sender with request from the farthest receiver is selected.

```

Input: Request Matrix( $D$ ), Routing Matrix( $R$ ), and Interference Matrix( $I$ )
Output: Sender Node(sender-node), Receiver Node(receiver-node), and Relay
Node(relay-node)
//Step 1: Select the sender with maximum number of transmission requests.
 $[v_i, v_j, v_k, mx] = \max(D, R)$  //  $v_i$  is sender node,  $v_j$  is receiver node,  $v_k$  is relay
node, and  $mx$  is Max
if there is only one  $v_i$ , then;
    sender-node =  $v_i$ ; receiver-node =  $v_j$ ; relay-node =  $v_k$ 
else;
    //Step 2: If there exist more than one sender in Step 1, select the sender with
    request from the farthest receiver.
     $[v'_i] = \max\text{-distance}(v_j, R)$ ;  $v'_i = \text{sender-of-}v'_j$ ;  $v'_k = \text{relay-of-}v'_i$ 
    if there is only one  $v'_i$  then
        sender-node =  $v'_i$ ; receiver-node =  $v'_j$ ; relay-node =  $v'_k$ 
    else;
        //Step 3: If there exists more than one sender in Step 2, select the sender
        with least possible interference.
         $[v''_i, v''_j, v''_k] = \min\text{-interference}(v'_i, v'_j, I, R)$ 
        if there is only one  $v''_i$ , then;
            sender-node =  $v''_i$ ; receiver-node =  $v''_j$ ; relay-node =  $v''_k$ 
        else;
            //Step 4: If there exists more than one sender in Step 3, select the sender
            with least node number.
            sender-node =  $\min\text{-number}(v''_i)$ ; receiver-node = receiver-of- $v''_i$ ; relay-
            node = relay-of- $v''_i$ 
        end if
    end if
end if

```

**Algorithm 1.** The pseudo code of the proposed Traffic-Aware Scheduling (TAS) algorithm.

3. If there is more than one node at the previous stage, the node is selected that transmits with the least possible interference with the other nodes.
4. If there is more than one node with minimum interference, the sender with the least node number is selected.

The selected sender node in the first phase will enter into a list, called the *Transmission List*. The nodes in the *Transmission List* will transmit in time slot  $t$ . In the second phase of algorithm, the nodes that can transmit simultaneously to the nodes in the transmission list will be determined. The steps of this phase are as below:

1. Considering the collision matrix, the nodes with a request for transmission that can transmit concurrently with nodes in the *Transmission List*, are extracted. These nodes are added to the *Concurrent Transmission List*.
2. The *Concurrent Transmission List* is provided to the first phase to select the simultaneous sender nodes. The selected node is removed from the *Concurrent Transmission List* and added to the *Transmission List*.

The second phase continues until all the requests are granted or there is no possibility for simultaneous transmission. At the end of the second phase, the scheduling algorithm is repeated for time slot  $t+1$ . This

process continues until all data packets are received by their destinations. At the start of the scheduling algorithm, the fact that the load is mainly concentrated at the BS makes the BS selected as the first sender. This circumstance continues for the first few time slots. Selection of the sender with maximum transmission request in the first phase, results in fast distribution of load over the network. Although the possibility of simultaneous transmission is not feasible during the first few time slots, it can be achieved after distributing the load over the network. We mentioned before that in downlink scheduling, the network load is initially concentrated in the BS, and the possibility of simultaneous transmission is low. If load distribution is not considered in the scheduling algorithm, we cannot easily use the concurrent transmission opportunity. The first stage selects the most demanding sender, so the traffic is separated from the BS in the first time slots and is distributed over the network. After some time slots, load distribution in the network is the same as uplink flow, and concurrent transmission possibility is high. In the second stage, the distance of the packet receiver is considered. As shown in Section 4.3, there are more opportunities for simultaneous transmissions in chain structures. In the second stage, the chain manner is used to distribute the load in the network to increase the concurrency rate. The network structures are random and are not in chain form necessarily, but it is more probable that the father node to the BS is on a chain or semi-chain structure. So, the

farthest receiver selection transfers the load to the farthest points of the network and simulates motion on the chain or semi-chain configuration. Specifically, this is the difference between the TAS algorithm and the proposed algorithm in [27]. If there are multiple nodes that satisfy the second stage condition, the node with minimum interference with other nodes is selected. This selection increases the possibility of simultaneous transmission. Finally, at the last stage, a random criterion is considered for sending the package.

## 5. Experimental results

### 5.1. Performance metrics

Three metrics are set up for performance evaluation. They are scheduling length, network throughput,  $nthr$ , and link concurrency ratio,  $lcr$  [27].

Scheduling length is the number of time slots needed to transfer all downlink traffic from BS to SSs. It is the most important measure of the performance of a scheduling algorithm, and it is considered in most of the previous literature. The scheduling length has upper and lower bounds. The scheduling length gets closer to the upper bound when there is no concurrent transmission and only one link transmits at each time slot. Let  $h_i$  be the hop count of  $SS_i$  to the BS and  $d_i$  be the slot request of  $SS_i$ . The upper bound can then be calculated by Eq. (10).

$$\text{upper bound} = \sum_{i \in N} h_i \times d_i. \quad (10)$$

The lower bound is the sum of all requests without considering the nodes distance from BS. Then, the lower bound can be computed by:

$$\text{lower bound} = \sum_{i \in N} d_i. \quad (11)$$

The network throughput is used to evaluate the throughput performance. According to [27], we define network throughput as the ratio of the scheduling lower bound to the actual scheduling length:

$$nthr = \frac{\text{lower bound}}{\text{scheduling length}} = \frac{\sum_{i \in N} d_i}{\text{scheduling length}}. \quad (12)$$

The link concurrency ratio is used to evaluate the spatial reuse efficiency, which means how many links can transmit concurrently at a time slot [27]. Due to Eq. (13), the link concurrency ratio can be expressed as the ratio of the upper bound scheduling length to the actual scheduling length. Its minimum value is 1 when there is no concurrency. By using spatial reuse, the link concurrency ratio gets higher values.

$$lcr = \frac{\text{upper bound}}{\text{scheduling length}} = \frac{\sum_{i \in N} h_i \times d_i}{\text{scheduling length}}. \quad (13)$$

**Table 3.** Simulation parameters.

Parameters	Values
Frame duration	10 ms
Modulation scheme	QPSK1/2
Channel bandwidth	20 MHz
MSH-CTRL-LEN	10 minislot
MSH-CSCH-DATA-FRACTION	100%

### 5.2. Simulation setup

The MATLAB simulation tool was used to evaluate the performance of the proposed algorithm. We assume that the SSs are randomly and uniformly distributed in a square area of size  $100 \times 100$  units. The BS is located at the center of this area. All nodes have the same fixed transmission range of 20 units. Two nodes are neighbors if the distance between them is less than the transmission range. We assume a single channel in the network without any bit errors. In our simulation, the number of SSs is varied from 20 to 100, and the slot demands of all SSs within the allocation cycle time are varied from 0 to 10. We assume the traffic is non-real-time and Variable Bit Rate (VBR). The simulation results presented in this section are averaged over 300 independent simulation experiments. Other simulation parameters are provided in Table 3.

### 5.3. Main results

In this section, we compare our algorithm with three other algorithms: *Farthest Sender*, *Max Weight* [3] and *Farthest Receiver* [27]. The *Max Weight* algorithm is proposed for uplink traffic. Because of its similarity to the first and second steps of our algorithm, we optimize this algorithm for downlink traffic. In the *Max Weight* algorithm, for each time slot, across the farthest senders, the scheduler selects the sender with maximum data packets. The *Farthest Receiver* is actually the second step of our algorithm, which is implemented to study the impact of the second step of our algorithm on global efficiency. The difference between *Farthest Sender* and *Farthest Receiver* is that the first one selects the farthest sender from the BS in each time slot, but the other algorithm chooses the sender who has the data packet for the farthest receiver from the BS.

Figures 4, 5 and 6 show the simulation results in downlink scheduling of *TAS* (the proposed algorithm), *Farthest Sender*, *Max Weight* and *Farthest Receiver*. From Figure 4, we can see that the scheduling length increases with the number of nodes, and this is because the traffic demand increases with the number of nodes. The scheduling length of *TAS* and *Max Weight* is lower than the two other algorithms. The proposed algorithm has the best results in scheduling length and the performance of the *Farthest Sender* is the worst.



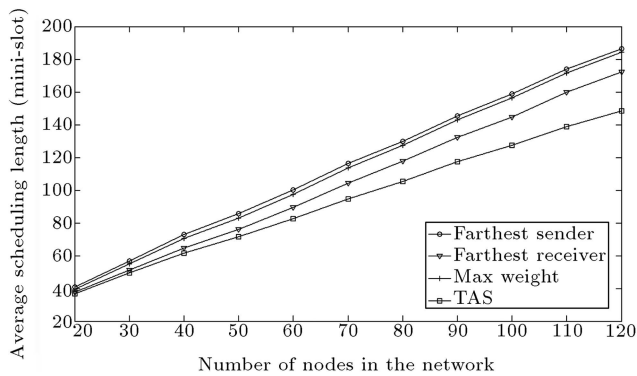


Figure 4. Scheduling length vs. number of nodes.

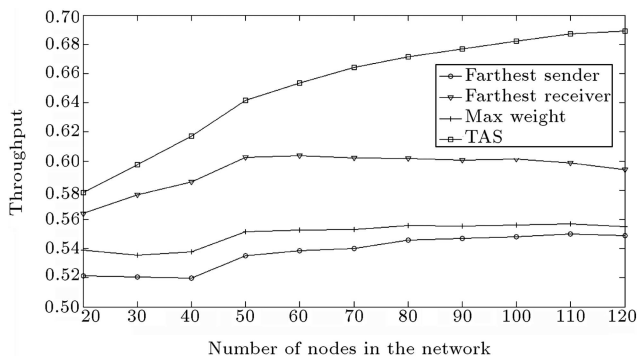


Figure 5. Network throughput vs. number of nodes.

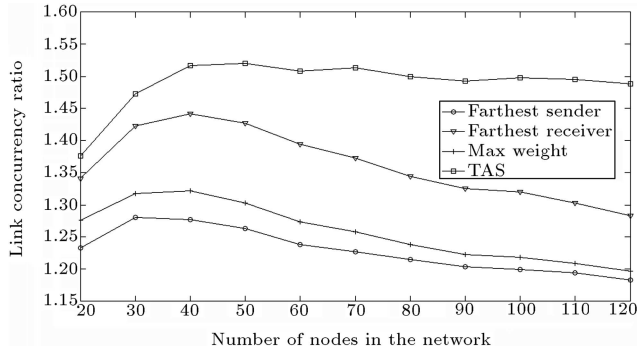


Figure 6. Concurrency ratio vs. number of nodes.

On average, the scheduling length of *TAS* is 13.7% less than the other algorithms.

In Figure 4, the *Max Weight* algorithm has better results than the *Farthest Sender* and *Farthest Receiver*, but *TAS* has the best performance. In *TAS* and *Max Weight*, we select the sender with the maximum number of waiting packets. In the first time slots, the BS has the most data packets; then our selection helps the BS to distribute the traffic load which leads to more concurrency. Moreover, Figure 4 shows that choosing the node with the farthest receiver in the *Farthest Receiver* and the *TAS* algorithms is better than selecting the farthest sender. Therefore, the download traffic load distributes in the network faster, i.e. in a chain-like manner, and more nodes can take advantage of the concurrent transmission. Since the

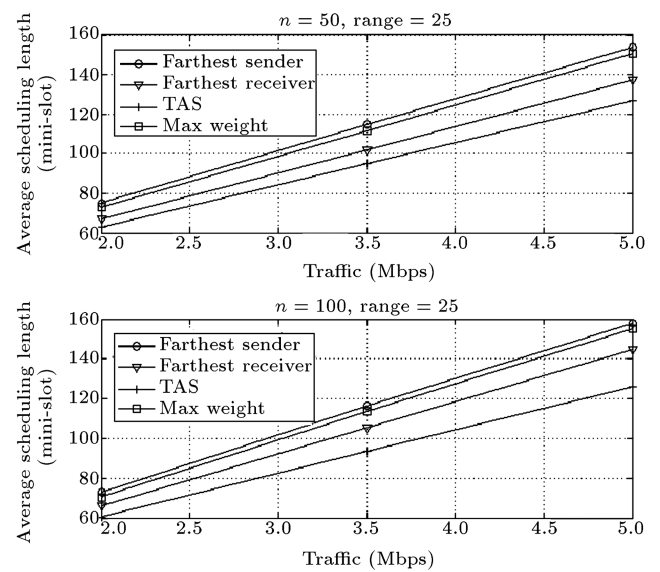


Figure 7. Scheduling length vs. network load.

load is concentrated in a few nodes (the BS and its neighbours) in the first time slots, choosing the farthest sender leads to provide chances for fewer nodes to send their packets.

Figure 5 shows that only in the *TAS* algorithm does the network throughput increase vs. the number of nodes, i.e. the proposed algorithm is scalable. When the number of nodes increases, although the interference may be more, we can compensate for its effect and improve the network throughput using spatial reuse. Moreover, this figure demonstrates that choosing the farthest sender is a more effective measure than choosing the node with the maximum load, in terms of network throughput.

The result of studying the concurrency ratio is shown in Figure 6. According to this result, the concurrency ratio of the *TAS* algorithm is more than other algorithms and does not decrease after reaching its peak. Although interference increases with the number of nodes, the *TAS* algorithm uses concurrency opportunities better than the other algorithms and maintains the concurrency ratio constant.

In Figures 7-9, the effect of increasing network load on the performance of the *TAS* algorithm is studied for two sizes of network. As seen in Figure 7, by increasing the network load, scheduling length increases for all four methods, and the proposed algorithm has a minimum value. Scheduling length in all methods are close to each other for low traffic load. However, with increasing the load, the difference in scheduling length becomes more. Therefore, the proposed algorithm is more efficient for high load. Like Figure 4, the *Farthest Sender* has the longest length and the performance of the *Farthest Receiver* is close to the proposed method. Moreover, changing the number of nodes has little impact on the scheduling length. If

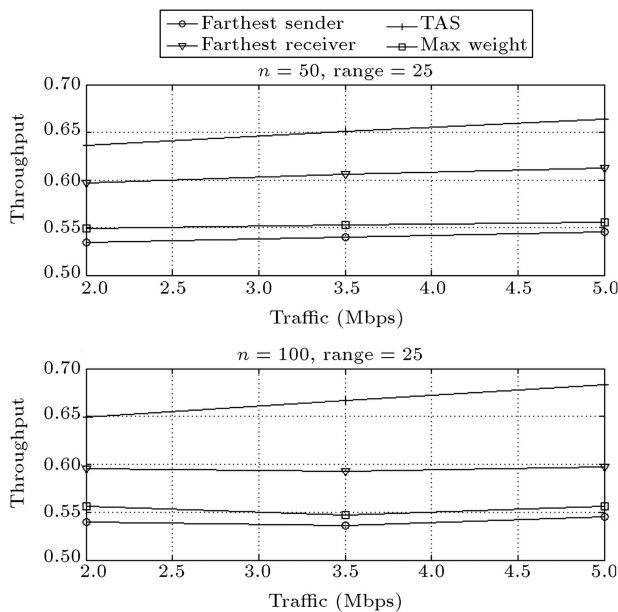


Figure 8. Network throughput vs. network load.

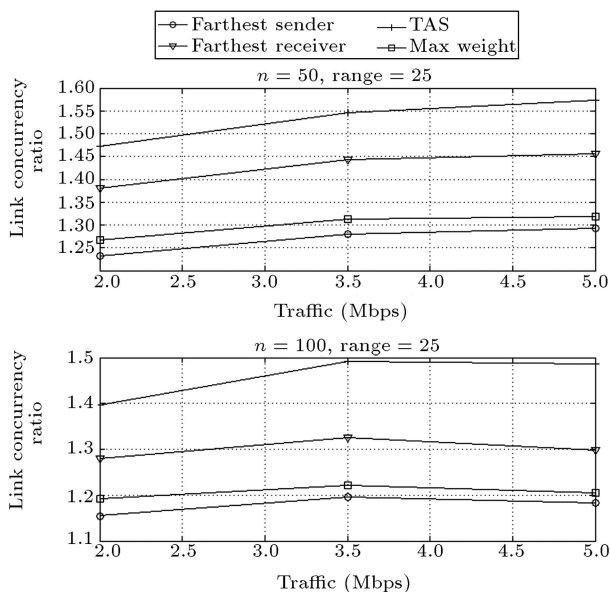


Figure 9. Concurrency ratio vs. network load.

the number of nodes in the network increases, but the network load remains constant, the scheduling length will be independent of the number of nodes.

According to Figure 8, TAS provides the best network throughput compared to other algorithms. By increasing the number of nodes from 50 to 100 nodes, only the throughput of the proposed algorithm has increased. Therefore, if the network load is low and the number of nodes is high, the throughput of the proposed algorithm, compared to the others, will be better.

Figure 9 shows the concurrency ratio of TAS compared to other methods. Although the proposed method leads to the best result, the growth rate of the

concurrency ratio in a 50-node network is more than in a 100-node one. Therefore, if the network load remains constant, increasing the number of nodes will have an undesirable effect on the concurrency ratio.

## 6. Conclusion and future work

In this paper, a novel algorithm was presented to improve centralized scheduling and optimal time slot allocation for downlink traffic in IEEE 802.16-based wireless mesh networks. The proposed algorithm considers the specifications of downlink traffic, spatial reuse, and a complete interference model. We have demonstrated the effectiveness of this algorithm across a range of scenarios by performing simulations. The experimental results showed that the proposed algorithm has high throughput, reduction in scheduling length, and more concurrent transmission, compared to the state of the art algorithms. Our future work is expanding the algorithm to consider QoS requirements in applying the algorithm to multimedia applications.

## Acknowledgement

The authors would like to thank Mr. Hadi Asheri of Sharif University of Technology for his help in preparing the manuscript.

## References

1. "IEEE. Standard 802.16-2004. Part 16: Air Interface for Fixed Broad-band Wireless Access Systems" (2008).
2. Mach, P. and Bestak, R. "Wimax throughput evaluation of conventional relaying", *Telecommunication Systems*, **38**(1-2), pp. 11-17 (2008).
3. Fanchun, J., Arora, A., Jinho, H. and Hyeong-Ah, C. "Routing and packet scheduling in WiMAX mesh networks", in *Fourth International Conference on Broadband Communications, Networks and Systems*, pp. 574-582 (2007).
4. Tarhini, C. and Chahed, T. "Qos-oriented resource allocation for streaming flows in ieee802.16e mobile wimax", *Telecommunication Systems*, **51**(1), pp. 65-71 (2012).
5. Cho, D.H., Song, J.H., Kim, M.S. and Han, K.J. "Performance analysis of the IEEE 802.16 wireless metropolitan area network", pp. 130-136 (2005).
6. Singh, V. and Sharma, V. "Efficient and fair scheduling of uplink and downlink in IEEE 802.16 OFDMA networks", in *IEEE Wireless Communications and Networking Conference*, pp. 984-990 (2006).
7. Chen, J., Jiao, W. and Wang, H. "A service flow management strategy for IEEE 802.16 broadband wireless access systems in TDD mode", *Proceedings of IEEE International Conference on Communications*, pp. 3422-3426 (2005).

8. Fathi, M., Rashidi-Bajgan, S., Khalilzadeh, A. and Taheri, H. "A dynamic joint scheduling and call admission control scheme for IEEE 802.16 networks", *Telecommunication Systems*, **52**(1), pp. 195-202 (2013).
9. Ghiamatyou, A., Nekoui, M., Esfahani, S.N. and Soltan, M. "Efficient routing tree construction algorithms for multi-channel WiMax networks", in *International Conference on Computer Communications and Networks*, (Honolulu, HI), pp. 957-960 (2007).
10. Lu, Y. and Zhang, G. "Maintaining routing tree in IEEE 802.16 centralized scheduling mesh networks", in *Proceedings of 16th International Conference on Computer Communications and Networks*, pp. 240-245 (2007).
11. Cao, Y. and Liu, Z. "A centralized scheduling algorithm based on multi-path routing in WiMAX mesh network", in *International Conference on Wireless Communications, Networking and Mobile Computing*, pp. 1-4 (2006).
12. Xergias, S., Passas, N. and Salkintzis, A.K. "Centralized resource allocation for multimedia traffic in IEEE 802.16 mesh networks", *Proceedings of the IEEE*, **96**(1), pp. 54-63 (2008).
13. Mai, Y., Yang, C. and Wen, C. "A new cut through mechanism in IEEE 802.16 mesh networks", *Proceedings of World Academy of Science, Engineering and Technology*, **41**, pp. 98-104 (2009).
14. Liang, J.-M., Wu, H.-C., Chen, J.-J. and Tseng, Y.-C. "Mini-slot scheduling for IEEE 802.16d chain and grid mesh networks", *Computer Communications*, **33**(17), pp. 2048-2056 (2010).
15. Salehi, M. and Khorsandi, S. "A centralized scheduling with spatial reuse in IEEE 802.16 based wireless mesh network", in *Iranian Conference on Electrical Engineering* (2008).
16. Chen, L.-W., Tseng, Y.-C., Wang, Y.-C., Wang, D.-W. and Wu, J.-J. "Exploiting spectral reuse in routing, resource allocation and scheduling for IEEE 802.16 mesh networks", **58**(1), pp. 301-313 (2009).
17. Xiao, J., Zhang, W. and Wang, N. "Tsals - a centralized scheduling and channel assignment scheme in third generation router based WiMax mesh network", in *International Conference on Network Security, Wireless Communications and Trusted Computing (NSWCTC)*, **01**, pp. 262-265 (2009).
18. Shetiya, H. and Sharma, V. "Algorithms for routing and centralized scheduling to provide QoS in IEEE 802.16 mesh networks", in *ACM workshop on Wireless Multimedia Networking and Performance Modeling*, pp. 140-149 (2006).
19. Kuran, M., Yilmaz, B., Alagoz, F. and Tugcu, T. "Quality of service in mesh mode IEEE 802.16 networks", in *International Conference on Software in Telecommunications and Computer Networks*, pp. 107-111 (2006).
20. Sharma, V., Kumar, A.A., Sandeep, S.R. and Sankaran, M.S. "Providing QoS to real and data applications in WiMAX mesh networks", in *IEEE Wireless Communications and Networking Conference*, pp. 2645-2650 (2008).
21. Yang, C.-C., Mai, Y.-T. and Tsai, L.-C. "Design of the QoS framework for the IEEE 802.16 mesh networks", *International Journal of Communication Systems*, **22**, pp. 1543-1562 (2009).
22. Gandham, S., Dawande, M. and Prakash, R. "Link scheduling in sensor networks: Distributed edge coloring revisited", **4**, pp. 2492-2501 (2005).
23. Han, B., Jia, W. and Lin, L. "Performance evaluation of scheduling in IEEE 802.16 based wireless mesh networks", *Computer Communications*, **30**(4), pp. 782-792 (2007).
24. Al-Hemyari, A., Noordin, N.K., Ismail, A., Khatun, S., Tahir, Y.H. and Qassem, Y.A. "Centralized scheduling, routing tree in WiMAX mesh networks", in *International Conference on Innovations in Information Technology*, pp. 539-543 (2008).
25. Ishii, R., Hasegawa, G., Taniguchi, Y. and Nakano, H. "Time slot assignment algorithms in IEEE 802.16 multi-hop relay networks", pp. 265-270 (2010).
26. Xie, F., Hua, K.A. and Jiang, N. "A cross-layer framework for video-on-demand service in multi-hop WiMax mesh networks", *Computer Communications*, **31**(8), pp. 1615-1626 (2008).
27. Liu, S., Feng, S., Ye, W. and Zhuang, H. "Slot allocation algorithms in centralized scheduling scheme for IEEE 802.16 based wireless mesh networks", *Computer Communications*, **32**(5), pp. 943-953 (2009).
28. Ali, N.A.A., Taha, A.-E.M. and Hassanein, H.S. "IEEE 802.16 mesh schedulers: Issues and design challenges", *IEEE Network*, pp. 58-65 (2008).
29. Zhang, S., Yu, F.R. and Leung, V.C.M. "Joint connection admission control and routing in IEEE 802.16-based mesh networks", in *IEEE International Conference on Communications*, pp. 4938-4942 (2008).
30. Kas, M., Yargicoglu, B., Korpeoglu, I. and Karasan, E. "A survey on scheduling in IEEE 802.16 mesh mode", *IEEE Communications Surveys and Tutorials*, **12**(2), pp. 205-221 (2010).
31. Han, B., Tso, F.P., Ling, L. and Jia, W. "Performance evaluation of scheduling in IEEE 802.16 based wireless mesh networks", in *IEEE International Conference on Mobile Adhoc and Sensor Systems*, pp. 789-794 (2006).
32. Tang, Y., Cheng, R., Wu, T. and Yu, J. "On centralized scheduling and channel assignment scheme for WiMAX mesh networks", *Internet Technology*, **10**(3), pp. 207-215 (2009).

## Biographies

**Fatemeh Dabiran** received her BSc and MSc degrees in Information Technology Engineering from Sharif University of Technology, Tehran, Iran, in 2008, and 2011, respectively. Currently, she is Research Assistant

in the Rahavard Research Communications Company, Tehran, Iran. Her research interests include multimedia networks, communication networks, wireless communications and signal processing.

**Hamid Reza Rabiee** received his BS (1987) and MS (1989) degrees in Electrical Engineering from CSULB, his EEE degree in Electrical and Computer Engineering from USC, USA, and his PhD (1996) in Electrical and Computer Engineering from Purdue University, West Lafayette, USA. From 1993 to 1996 he was a Member of the Technical Staff at AT&T Bell Laboratories. From 1996 to 1999 he worked as a Senior Software Engineer at the Intel Corporation. He was also with PSU, OGI and OSU universities as an Adjunct Professor of Electrical and Computer Engineering from 1996-2000, and is currently Professor of Computer Engineering at Sharif University of Technology, Tehran, Iran, and the Director of AICT, DML, and VASL. He is the founder of the Sharif University Advanced Information and Communication Technology Research Center (AICT), Advanced Technologies Incubator (SATI), Digital Media Laboratory (DML), and Mobile Value Added Services Laboratory (VASL). He has been the initiator and director of international and national projects in the context of the UNDP International Open Source Network (IOSN), and the Iran National ICT Development Plan. He has received numerous awards and honors for his industrial, scientific and academic contributions. He has acted as Chairman in a number of national and international conferences, and holds three patents.

**Mostafa Salehi** received his BS (2005) degree in Computer Engineering from Iran University of Science and Technology, and his MS (2008) and PhD (2012) degrees in Computer Architecture from Amirkabir University of Technology and Sharif University of Technology, Tehran, Iran, respectively. He is currently Assistant Professor in the Faculty of New Sciences and

Technologies at the University of Tehran, Iran. He has also been involved in a number of large scale network design and consulting projects in the telecom industry. His research interests include multimedia networks, and complex networks.

**Ali Asghar Amidian** received his BS degree from K.N. Toosi University of Technology, Tehran, Iran, in 1986, his MS degree from Amirkabir University, Tehran, Iran, in 1993, and his PhD from Laval University, Quebec, Canada in 2002 in Telecommunication Engineering. He is currently Assistant Professor in the faculty of Applied Science of Communications, affiliated to the Ministry ICT and a Member of the Strategic Council of the Advanced Information and Communication Technology Research Center of Sharif University of Technology, Tehran, Iran. From October 2005 to January 2007 he acted as Consultant to the ICT Ministry and Director of the Iranian Telecommunication Research Center (ITRC) and also worked as Executive Manager of the National ICT Research Project. From 2004 to 2007 he worked as a Board Member and Managing Director of the Telecommunication Infrastructure Company of Iran (TIC). From 2002 to 2004 he acted as Chairman of the Board and Managing Director of the Iranian Remote Sensing Center (presently: Space Agency of Iran). From October 2009 to 2010, he worked as Deputy of Statistics and Informatics at the National Registration Organization and the Executive Director of the National Smart ID Card Project, and, from 2008 to 2009, he acted as Planning Deputy of Qom Governor-Generalship. Dr. Amidian has published a number of research papers, books, and translations, and has contributed academically to numerous Iranian Universities, such as Amirkabir University, and the council for socio cultural and economic studies and researches, at the ICT Ministry. He has also lectured at many national and international conferences.