Data Streams Scheduling Approach for WiMAX Networks

Khaled Abdalgader and Dinesh Kumar Saini
Faculty of Computing and Information Technology, Sohar University, Oman
Email: komar@su.edu.om; dinesh@su.edu.om (dkssohar@gmail.com)

Abstract**—**This study aims to enhance the capacity of the mesh mode WiMAX network by designing an efficient and fair scheduling algorithm, which resolves the bandwidth contention problem and determines the transmission order for end-users. It is also analyzed the algorithmic complexity of different existing WiMAX network scheduling approaches and demonstrated their lack of fairness. In this paper, therefore, we proposed new variation of a centralized scheduling algorithm for IEEE 802.16 mesh mode standard that is able to schedule the nodes efficiently while ensuring fairness. The algorithm is also providing an equal amount of bandwidth and handling extensible traffic without interference to the users in the same range of a networking coverage area. We then provided a complexity analysis for our proposed algorithm followed by a comparison with the complexity of the existing approaches. We also evaluated the proposed algorithm on a well-known networking simulation to ensure that it satisfies the fair scheduling issue requirements. The empirical results indicate that the proposed algorithm achieves a significant improvement in performance over existing scheduling algorithms.

Index Terms**—**Fair Scheduling, WiMAX Networks, Data Streams, Bandwidth, IEEE 802.16 standard

I. INTRODUCTION

Due to the growing demand for Internet access requirements in different locations, the Worldwide Interoperability for Microwave Access (WiMAX) technology was introduced to enhance access for both stable and moveable operators on the superiority of wired communication networks [1]. Existing wired communication networks based on cable and Digital Subscriber Line (DSL) infrastructure are considered to have suboptimal performance in terms of pervasive and high data rates, wide network coverage, and quality of service capabilities [1], [2]. WiMAX, a technology based on the IEEE 802.16 standard, has been developed to provide cost-effective deployment and delivery of last-mile wireless broadband access (i.e., Internet) as an alternative to cable and DSL infrastructure [2], [3]. Starting with only line-of-sight support, IEEE 802.16 standards have evolved into a technology that is now capable of handling and processing the traffic of mobile mesh networks (i.e., multipoint-to-multipoint). In the mesh mode of WiMAX, which is defined by the IEEE networks standard, all subscriber stations (SSs) operate under one base station (BS). SS nodes forward their packets with the help of other SS nodes, and the SS node that is connected to the backbone network is referred to as the mesh BS.

Communication between SS nodes requires two independent channels: the downlink channel (from the BS to the SS) and the uplink channel (from the SS to the BS) [4]. All SS nodes share the uplink channel, while the downlink channel is used only by the BS node. With mesh support, an IEEE 802.16 network is required to handle extensible and scalable traffic while avoiding interference. To achieve this, the IEEE 802.16 standard requires scheduling that assigns time slots for user data transmission without collision. Many scheduling algorithms have been proposed [4]-[7]; however, to the best of our knowledge, none of these algorithms provides fairness when allocating time slots to contending nodes. The notion of fairness refers to the mechanism of providing every node with a value that reflects the importance of the data it pushes to the BS. In this study, we modify a centralized scheduling algorithm that is able to resolve the fairness problem in $O(n2)$ time, while the lower bound for MAC layer scheduling is $\Omega(n)$ [6], [7]. The proposed algorithm provides fairness by considering the depth of SS nodes away from the BS as well as the load of each node (including its own and that of its children or subnetwork). This signifies that even if a SS node is far from the BS, its packet is not more delayed in reaching the BS than the packets of SS nodes closest to the BS.

Wireless mesh networks are a recent development. Despite their numerous advantages, such as robustness and simple deployment, a number of challenges remain in terms of speed and efficiency. Wireless communication is highly susceptible to interference and collision. The traditional strategy for this type of transmission is to follow collision avoidance protocols, such as IEEE 802.16 networks. The solution to these challenges involves the use of a scheduling algorithm that controls exchanged messages for both the sender and receiver. The main goal of the majority of wireless mechanisms is to increase the number of nodes transmitting in one-time slot as much as possible (greedy approach) when the nodes transmit their packets concurrently and without interference, thus maximizing the throughput, as achieved by time-division multiple access (TDMA). In this study, we examine and analyze existing solutions in the literature to illustrate that existing algorithms do not provide an equal opportunity for all nodes to transmit

Manuscript received December 20, 2019; revised April 28, 2020.
Corresponding author email: komar@su.edu.om.
doi:10.12720/jcm.15.6.469-479

their data, as some nodes always remain at the end of the transmission queue. We develop an algorithm to ensure that all SS nodes receive an equal chance to transfer their packets and are not always scheduled last. We also develop a new wireless greedy scheduling algorithm for WiMAX/IEEE802.16 that provides fair allocation of transmission opportunities to participating nodes.

Bandwidth is the current standard for measuring network effectiveness, and it is one of the most important metrics for users because it represents the bit rate of available information that can be measured in bits per second. In this paper, we modified an algorithm for the scheduling problem in WiMAX networks. WiMAX is a new popular technology based on IEEE 802.16 standards, and is also known as wireless broadband access. Currently, the standard provides two network architectural modes: the point-to-multipoint (P2MP) mode and multipoint-to-multipoint (mesh) mode. In addition, there are two types of communicating nodes in WiMAX: SSs and BSs. Our aim and scope in this paper is to enhance the capacity of a mesh mode WiMAX network by designing an efficient and fair scheduling algorithm that resolves the bandwidth contention problem and determines the transmission order for users. We analyzed the algorithmic complexity of different existing solutions and illustrated their lack of fairness. We then modified a centralized scheduling algorithm for the IEEE 802.16 mesh mode standard that is able to schedule nodes efficiently while ensuring fairness. Furthermore, we performed a complexity analysis for our algorithm followed by a comparison with existing algorithms.

The remainder of the paper is structured as follows. Section II starts with a brief background and related work. Section III presents the proposed approach to fair scheduling algorithm. Section IV shows experimental results and Section V concludes the paper.

II. Background and Related Work

A wired network can be defined as a collection of devices such as servers, workstations, printers, routers, switches, and other devices connected through Ethernet interfaces and cables. A wired (Ethernet) network is considered one of the fastest networks with data transfer rates (i.e., speeds) of 10 megabits per second (Mbps) to 100 megabits per second (Mbps). To connect a workstation or any other networked device to a network through an Ethernet cable, the workstation must have an Ethernet interface card (i.e., network adapter). Most networked devices have a built-in (internal) Ethernet adapter port, which eliminates the need for an external adapter. There are three common wired network topologies that are widely used today: *the start*, *bus*, and *ring* topologies [7], [8].

A wireless network is another networking technology suitable for home and small-to-medium enterprises, and is based on using high-frequency radio waves rather than wires to establish communication between networked devices. Homes and enterprises can use this technology to upgrade their existing wired networks or switch to wireless solutions entirely. Wireless technology enables connected devices to exchange data without using cables, but with a limited distance range (area). There are two main types of wireless networking: *point-to-point* and *multipoint-to-multipoint* [8]-[11].

The difference between wired and wireless networks is that the former use cables while the latter uses radio frequencies [3], [12]. A wired network provides high speed and a more secure connection and can be used for distances less than 6,000 m. A wireless network is less secure than a wired network, and the transmission rate and speed can be affected by outside noise (i.e., interference). Although wireless networking is more mobile than wired networking, the network range is usually 50–100 m indoors and up to 3,000 m outdoors based on the environment [10].

The aim of IEEE 802.16 was to develop standards to be applied for the design and operation of wireless metropolitan area networks (MANs) worldwide. Later, these standards were renamed as WiMAX technology, and problems such as interoperability, certification, and promotion of the system were managed by the WiMAX working group, which was formed in mid-2001. IEEE 802.16, however, was founded to operate in the 10–66 GHz spectrum, and it assigns the physical layer and medium access control layer of the air interface. It was then discovered that the physical layer is not appropriate for lower-frequency applications, in which non-line-of-sight procedures are necessary. As a result, the IEEE committee designed a new version of IEEE 802.16 called the IEEE 802.16a standard to accommodate the technology requirements. The new standard operates in both licensed and unlicensed frequencies that are between 2 GHz and 11 GHz, and it has been continuously developed up to now in several series. Although the IEEE 802.16 series of standards is widely known as Wireless MAN in IEEE, it has been commercialized under the name of WiMAX [3], [8].

A standard IEEE 802.16 network contains a BS and SS. A BS operates as the gateway between the external network and IEEE 802.16 network [13]-[15]. In contrast, the SS operates as a terminal (i.e., client) in which users with wireless devices can access the network through the air interface (radio frequency). The IEEE 802.16 standard defines two connection architectures (i.e., topologies) of the network: a point-to-multipoint network and a multipoint-to-multipoint network (mesh network). In a point-to-multipoint network, traffic only appears between BSs and SSs that are within direct connection of transmission range. In a multipoint-to-multipoint network, however, the SSs, which are multiple hops away from the BS, can exchange data with the BS through intermediate nodes. They can also exchange data with each other. Therefore, the node in a WiMAX multipoint-to-multipoint (mesh) network connected to the backhaul network is known as a mesh base station (MBS), and all other nodes in the mesh are known as mesh subscriber station (MSSs). WiMAX networks provide frequency ranges of 10 GHz to 66 GHz and 2 GHz to 11 GHz for network line-of-sight and non-line-of-sight, respectively. Fig. 1 presents an example of an IEEE 802.16 multipoint-to-multipoint (mesh) network [1], [4].

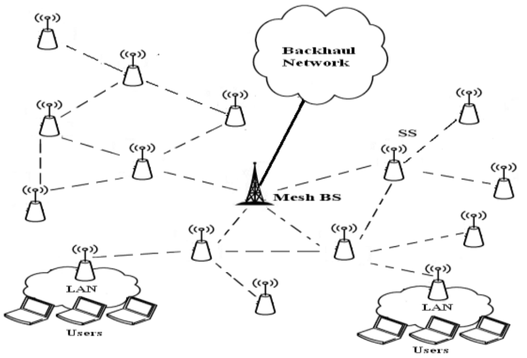

Fig. 1. IEEE 802.16 multipoint-to-multipoint (mesh) network [1], [4].

There are two-time slot allocation schemes (i.e., communication modes) in the IEEE 802.16 multipoint-to-multipoint network: centralized scheduling and distributed scheduling. In the distributed scheduling method, communications are scheduled in a completely distributed manner without requiring any interaction with the MBS. In this study, however, we consider a centralized scheduling method in which the MBS is responsible for generating the schedule of communications in the entire network [1]-[16].

In the centralized scheduling scheme, the MBS collects resource requests from all MSSs that belong to a certain hop range. It then determines the quantity of granted resources for the participating nodes in the network both in the downlink and uplink directions using a defined scheduling method, and disseminates these grant messages to all MSSs within the hop range. The length of scheduling is considered the key to measuring the performance of a scheduling method, namely, the time slots required to complete all data transmissions. To achieve this, the scheduling method should provide a fair distribution of time slots among the participating nodes so that any node has the opportunity to transmit its data packets as long as it does not interfere with other nodes in the network [4], [17].

Available data scheduling algorithms for mesh networks can be classified into two groups [1], [5]. The first group includes traditional methods that use techniques such as first-in first-out (FIFO), which simply queues processes in the order that they arrive in the ready queue. Another scheduling technique is Round Robin (RR), which serves an equal portion to each queue in a circular order, handling all processes with no consideration of priority [5], [18], [19]. This is what the current networks in the data transfer prioritize depending on the type of data, and this causes delays and collisions with other nodes. The second group of WiMAX scheduling algorithms has been developed under the standards of IEEE 802.16, and they consider different quality of service (QoS) classes.

The two types of scheduling methods perform transmission without considering data type. Therefore, to provide these new methods, we must develop a scheduling algorithm that takes into account the interference of each node to maximize the number of concurrent transmissions. To determine the scheduling order, the algorithm computes the optimal route from the BS to each node [1], [19].

Scheduling algorithms for mesh networks can be classified to two groups [20]: traditional methods that use techniques such as FIFO and RR, and methods developed for IEEE 802.16 that consider different QoS classes defined in the standard [21]-[28].

The interference aware scheduling algorithm [14], [15], [29]-[31] takes into consideration the interference of each node to maximize the number of concurrent transmissions. To determine the scheduling order, the algorithm computes the optimal route from the MBS to each node (i.e., subscriber). There has been some research based on the interference aware route construction. The objective is to determine the optimal route between each node and the BS in terms of least interference. To compute the interference, they denote the blocking metric B(k) as the number of blocked nodes by all intermediate nodes in the route from the BS to the node. Furthermore, the blocking value b(η) is defined as the number of blocked nodes when node η is transmitting. The blocking metric B(k) is thus the summation of all the nodes' blocking values b(η) in the route. Fig. 2 and Fig. 3 present an example of calculating the blocking metric for a particular route from the source to the destination. In both figures, red represents the sender node, green represents the destination node, and the shaded nodes represent the blocked nodes due to the transmission of node η.

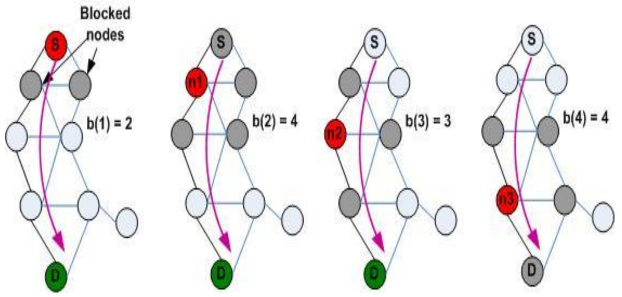

Fig. 2. Example of blocking metric B(k) = 2 + 4 + 3 + 4 = 13 [7], [14].

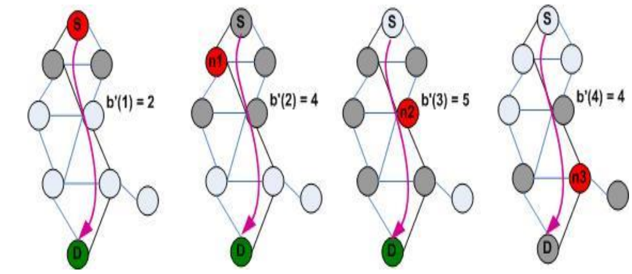

Fig. 3. Blocking metric of an alternative route B(k) = 2 + 4 + 5 + 4 = 15 [7], [14].

Initially, the algorithm computes the blocking metric B(k) for all available routes of each node, and then selects the route that has the lowest B(k) value. The algorithm assumes that starting from the BS, each node (subscriber) joins the network sequentially and selects the sponsoring node with the minimum block metric.

A widely known fair and efficient WiMAX scheduling algorithm has been reported in [27], [32]. This algorithm consists of two phases. In the first phase, the nodes (subscribers) are ordered according to a satisfaction index to give priority to one node over others. In the second phase, collision-free scheduling is determined for the simultaneous transmission of multiple nodes taking into consideration interference. In the following two paragraphs, we explain how to compute the satisfaction index and describe how this index helps perform fair and balanced scheduling for nodes.

To order the nodes, each node *i* is associated with a weight *Wi*. In a simple scenario, the weight can be considered the total number of child nodes including the node itself. Thus, the weight of a leaf node is 1, while the weight of the BS is the total number of nodes in the network. The satisfaction index is defined as the ratio of the average bandwidth allocated in a given number of frames to the node's weight. The average is taken over a number of frames called the satisfaction window *T*. Thus, the satisfaction index, Si, of a node in frame *x* is defined as follows:

$$S_i(x) = \frac{\frac{\sum_{y=x-T}^{x-1} B_i(y)}{T}}{w_i} \tag{1}$$

where *x* is the current frame number, *Bi*(*y*) is the link bandwidth allocated in frame *y*, and *T* is the satisfaction window. To schedule different time slots of a frame, the BS first computes the satisfaction index of all nodes and sorts them in increasing order of the satisfaction index. Therefore, a node that is closer to the BS is given high priority over nodes that are far from the BS. In other words, one-hop nodes have a higher priority than two-hop nodes.

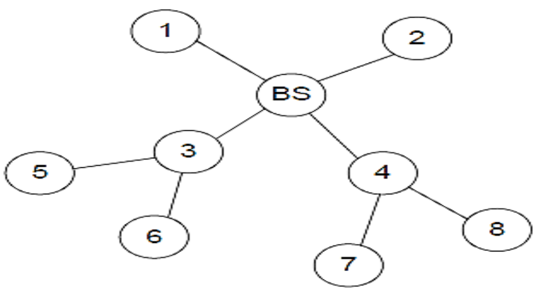

Fig. 4. Example of mesh network (MBS).

After the node ordering, the BS assigns the time slots based on the requested number of slots (data capacity request) of the nodes (subscribers). The node that is first on the list is assigned all the requested slots starting from the first slot. The algorithm then blocks all interfering nodes for those slots and assigns slots to the next non-interfering node from the list. This continues until all the capacity requests are fulfilled. Fig. 4 illustrates a typical example of a mesh network in which an ordered list contains the following nodes: {4; 3; 7}, where the capacity request includes 2, 3, and 1-time slot, respectively. Table I demonstrates that node 4 is assigned the firsts two slots because it has the highest priority and its request is 2. The nodes that interfere with node 4 are also listed in the table. Because nodes 3 and 7 are both interfering nodes, none of them are scheduled in time slots 1 and 2. Thus, node 7 is scheduled in time slot 3, and node 3 is scheduled in time slots 3, 4, and 5 [7], [27]-[32].

TABLE I: SCHEDULE AND COLLISIONS BETWEEN NODES

Time slots	Scheduled nodes	Interfering nodes
1	4	1, 2, 3, 7, 8
2	4	1, 2, 3, 7, 8
3	3, 7	1, 2, 4, 5, 6, 8
4	3	1, 2, 4, 5, 6
5	3	1, 2, 4, 5, 6

Resource allocation in WiMAX mesh networks is another WiMAX scheduling study reported in [7]. In their study, the authors provided simple joint routing and link scheduling algorithms that outperformed most existing algorithms in the literature. The authors of [7] also studied the problem of QoS provisioning in WiMAX mesh networks. In addition, they provided a comprehensive scheme consisting of routing, link scheduling, call admission control, and channel assignment that considered all classes of service. Their routing schemes used a metric that combined interference and traffic load to compute routes for requests, while their link scheduling ensured that the QoS requirements of admitted requests were strictly met. Their empirical results indicated that the modified routing and link scheduling schemes significantly improved network performance in the case of network congestion.

III. PROPOSED FAIR SCHEDULING ALGORITHM

The proposed scheduling algorithm for IEEE 802.16 provides a fair allocation of transmission opportunities to participating nodes in a WiMAX mesh network. The principle of our algorithm is that a time slot is allocated for all SSs whether they are near the BS or far away from it. The algorithm also takes the capacity request into consideration during scheduling (three classes of data types). In Sections A and B, we introduce the domain analysis, requirement specifications and fair scheduling assumptions, and in Section C we describe our proposed algorithm with a walk-through example.

A. Domain Analysis

Wireless communication is highly susceptible to interference and collision. The traditional strategy for this type of transmission involves following collision avoidance protocols, such as IEEE 802.16 networks. We believe that the solution to these problems is to use a scheduling algorithm that controls exchanged messages for both the sender and receiver. This satisfies the main goal of most wireless mechanisms, which is to utilize the maximum number of nodes transmitting in one-time slot (greedy approach), whereby the nodes transmit their packets concurrently and without interference, thus maximizing the throughput, as achieved by TDMA.

The majority of existing scheduling algorithms do not provide fairness (i.e., equal opportunity) for all SSs to transmit their data. Instead, some SSs always remain at the end of the transmission queue (low priority). In this study, we modified a fair scheduling algorithm to ensure that all SSs receive equal chances to transfer their packets and not always be scheduled last. In other words, this paper presents a new fair scheduling algorithm for WiMAX/IEEE802.16 that provides fair allocation of transmission opportunities to participating SSs. For example, for a tree consisting of a number of SSs, to determine how many SSs should send data to the BS at the same time, the following required assumptions and constraints should be considered to maintain a maximum capacity for the BS.

B. Required Assumptions and Constraints

The following assumptions should be considered to represent the mesh mode of a WiMAX network in graph theory:

- Every wireless device (subscriber) is a node of a graph.
- The BS is the root of a minimum spanning tree (MST) pre-calculated by an available algorithm, and all other nodes are called SSs.
- The weight of each edge is the cost of transmitting data between two nodes, and it measures the distance between the two nodes.
- A graph is a snapshot of the locations of nodes at a specified time **t**. Thus, edges have a fixed cost, and the MST does not change.
- Every node is also associated with a weight.
- Traversal from one node to another is limited by the possibility of interference/collision with other nodes, where interference is defined as follows. Interference occurs when two nodes of the same parent (except the root) send or receive at the same time.
- All SSs with packets in their buffer can only send their packets when the BS allocates a time slot for them.
- The BS assigns link bandwidth for some SSs according to the following:
  - Interference among nodes: the BS allows all SSs to send concurrently if there is no interference between them to maximize the throughput.
  - Interference is computed according to the power transmission for the neighborhood nodes.
  - The destination for each sender is considered to avoid collision. In other words, if any node has more than one child, it would not schedule more than one of them in the same time slot.
- The BS assigns bandwidth for some SSs in the first round and the remaining SSs in the next time slot. This provides a fair allocation of transmission opportunities to participating nodes.

C. Fair Scheduling Algorithm

Our proposed algorithm is based on the concept that a time slot is allocated for all SSs whether they are near or far from the BS. The algorithm also takes the capacity request into consideration during scheduling (three classes of data types).

Network Architecture: The scheduling of TDMA slots offers two variants: node scheduling and link scheduling. In node scheduling, the entities of the TDMA time slots are the nodes themselves, whereas in link scheduling, the links between the nodes are scheduled. In our experiments, we focus only on node scheduling.

Uplink Scheduling: There is a systematic mechanism for data flow in different networks. For data traffic configuration in WiMAX, there are two types: nodes, and between the mesh BS and nodes. However, for a IEEE 802.16 network, there must be an uplink and downlink. In addition, each node must be scheduled separately. Because of this, it just a planned algorithm which it is configured for the uplink, through that the downlink can make no additional effort.

Interference Model: WiMAX uses frames of several time slots of equal length providing transmission opportunities to non-interfering nodes. This connection scheme maximizes the spatial reuse of the existing bandwidth while simultaneously eliminating any possibility of collision in the same frame.

Tree Construction and Routing: For scheduling processes in WiMAX networks, a child node's traffic usually follows a path toward the mesh BS through its parent, not any other nodes in the mesh. In this study, we assume that there is always a link from a node to the mesh BS via its parents. For example, a child's bandwidth request reaches its parent, which in turn sends the request via its parent, thus finally reaching the mesh BS.

Definitions and Notations: Broadband traffic (i.e., capacity request) involves various types of data, such as internet and TV, that can be categorized as traffic types C, B, and A depending on the typical bandwidth requirements. For example, traffic with 5–7 packets is of type C, and its proportional load (prop load) is 2. Traffic with 8–11 packets is of type B, and its proportional load is 3. Traffic with 12–15 packets is of type A, and its proportional load is 4. The total proportional load, *totalPropload,* is defined as the sum of a node's own proportional load and all of its children's proportional load. We construct a priority queue from the total proportional load, which is discussed later. In this construction, we start from the outermost level and move toward the innermost level that is the closest to the mesh BS. In each level, we give priority to the node with the highest totalPropLoad.

Scheduling Algorithm: IEEE 802.16 provides a fair allocation of transmission opportunities to participating nodes. The principle of our algorithm is that a time slot is allocated for all non-interfering nodes regardless of whether they are near or far from the BS. The algorithm also considers the capacity request during scheduling. For clarity, a detailed description of our modified fair scheduling method is provided in Fig. 5. This figure presents a network mesh topology of 16 subscribers (nodes). The capacity request of each subscriber (N) is also illustrated. For example, the capacity requests (TPL) of N1 and N2 are 6 and 5, respectively.

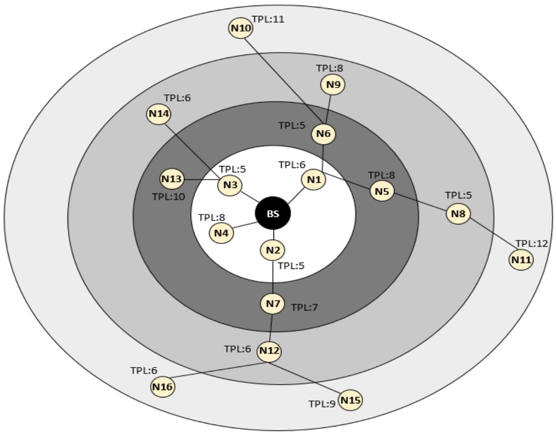

Fig. 5. Mesh network of 16 subscribers (nodes).

From Fig. 5, we can construct Table II, which contains the number of packets, the traffic type, the proportional load, and the total proportional load for each node. We also can determine the priority queue of the network nodes presented in Fig. 5. The priority queue for the network given the capacity request (TPL) is as follows: (N11, N10, N15, N16, N9, N14, N12, N8, N13, N5, N7, N6, N4, N1, N2, N3). Once the priority queue has been determined, our modified algorithm determines the scheduling order of the nodes. The full procedure for the modified fair scheduling of data traffic in WiMAX networks is described in Algorithm 1.

Table III depicts the resulting scheduling order of the nodes. Once the remaining total proportional load of all nodes reaches zero, we return to the initial step and again construct the priority queue depending on the node's capacity request. The mechanism of the algorithm is best understood with an example. Here, we briefly discuss how the time slots in Table III are determined by Algorithm 1. After constructing the priority queue as {N11, N10, N15, N16, N9, N14, N12, N8, N13, N5, N7, N6, N4, N1, N2, N3}, the time slots are assigned starting from the first node in the queue (i.e., N11). Once a time slot is assigned, we decrement *TRemPropLoad* by 1. Accordingly, we first assign the slot to [N11, N10, N15, N16, N9, N13, N4] nodes. However, N8 and N12 cannot be assigned the time slot due to their interference with N11 and N15, N16 respectively. Similarly, N5 is N8's parent, N6 is N9's and N10's parent, N7 is N12's parent, N3 is N13's and N14's parent, N1 is N6's and N5's parent, and N2 is N7's parent. Accordingly, using the same interference rule, the second time slot assignment is [N11, N10, N15, N16, N9, N13, N4]. For the seventh time slot, N12 is assigned the time slot due to its priority in the queue and for satisfying the interference condition. In the same time slot, although it has a *TRemPropLoad* of 0, N16 is assigned because it satisfies the chance condition of the algorithm. For the fourth time slot, the same condition holds, and the time slot is [N11, N10, N15, N12, N9, N13, N4]. Thus, the algorithm proceeds and constructs the remaining time slots. For simplicity, the procedure in Algorithm 1 (our modified fair scheduling algorithm) is illustrated in a flow chart presented in Fig. 6. The algorithm starts with packets of a node's buffer in ascending order upon generation time. Then, the priority queue on levels and *totalPropLoad* is updated in descending order, each node's copy *totalPropLoad* to *TRremPropLoad*. Next, packets are selected from the node buffer and placed in current-slot. Then, *TRremPropLoad* of the node is decremented, packets are picked from the other node's buffer and placed.

TABLE II: TOTAL REMAINING PROPORTIONAL LOAD

No	Transmitting nodes	N1	N2	N3	N4	N5	N6	N7	N8	N9	N10	N11	N12	N13	N14	N15	N16
1	11, 10, 15, 16, 9, 13, 4	6	5	5	8	5	5	7	5	8	11	12	6	10	6	9	6
2	11, 10, 15, 16, 9, 13, 4	6	5	5	7	5	5	7	5	7	10	11	6	9	6	8	5
3	11, 10, 15, 16, 9, 13, 4	6	5	5	6	5	5	7	5	6	9	10	6	8	6	7	4
4	11, 10, 15, 16, 9, 13, 4	6	5	5	5	5	5	7	5	5	8	9	6	7	6	6	3
5	11, 10, 15, 16, 9, 13, 4	6	5	5	4	5	5	7	5	4	7	8	6	6	6	5	2
6	11, 10, 15, 16, 9, 13, 4	6	5	5	3	5	5	7	5	3	6	7	6	5	6	4	1
7	11, 10, 15, 12, 9, 13, 4	6	5	5	2	5	5	7	5	2	5	6	5	4	6	3	0
8	11, 10, 15, 12, 9, 13, 4	6	5	5	1	5	5	7	5	1	4	5	4	3	6	2	0
9	11, 10, 15, 12, 8, 13, 4	6	5	5	0	5	5	7	5	0	3	4	3	2	6	1	0
10	11, 10, 7, 12, 8, 13, 2	6	4	5	0	5	5	7	4	0	2	3	2	1	6	0	0
11	11, 10, 7, 12, 8, 13, 2	6	3	5	0	5	5	6	3	0	1	2	1	0	6	0	0
12	11, 10, 7, 12, 8, 2	6	2	5	0	5	5	5	2	0	0	1	0	0	6	0	0
13	11, 1, 7, 6, 14, 8, 2	5	1	5	0	5	5	4	1	0	0	0	0	0	5	0	0
14	3, 1, 7, 6, 14, 8, 2	4	0	4	0	5	4	3	0	0	0	0	0	0	4	0	0

15	3, 1, 7, 6, 14, 5	3	0	3	0	4	3	2	0	0	0	0	0	0	3	0	0
16	3, 1, 7, 6, 14, 5	2	0	2	0	3	2	1	0	0	0	0	0	0	2	0	0
17	3, 1, 7, 6, 14, 5	1	0	1	0	2	1	0	0	0	0	0	0	0	1	0	0
18	3, 1, , 6, 14, 5	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
19	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

TABLE III: NODE SCHEDULING ORDER RESULTING FROM APPLYING ALGORITHM 1

Nodes	N1	N2	N3	N4	N5	N6	N7	N8	N9	N10	N11	N12	N13	N14	N15	N16
No. of packets of each node	6	5	5	8	5	5	7	5	8	11	12	6	10	6	9	6
Traffic type	C	C	C	B	B	B	C	C	B	B	A	C	B	C	B	C
Proportional load (Propload)	2	2	2	3	3	3	2	2	3	3	4	2	3	2	3	2
Total proportional load (total Propload)	20	11	7	3	9	9	9	6	3	3	4	7	3	2	3	2

Algorithm 1: Fair scheduling algorithm for WiMAX network

Input: Priority queue
Output: Time slot assignment to each node

```
1: packets of a node's buffer in ascending order upon generation
   time;
2: while true do
3:    update priority queue on levels and totalPropLoad in
      descending order;
4:    copy each node's totalPropLoad to TRremPropLoad;
5:    for node =1 to length(queue) do
6:       for counter = TRremPropLoad to 1 do
7:       pick packets from node buffer and put in current-slot;
8:       decrement TRremPropLoad of node;
9:       for chance = 1 to 2 do
10:         for otherNode = node+1 to node-1 do
11:         if (otherNode. TRremPropLoad > 0 OR chance=2)
               AND (other Node has no interference in the current
               slot)
            then
12               Pick packets from other Node's buffer and put in
                 current-slot
13:              Decrement TRremPropLoad of node
14:         end if
15:         end for
16:      end for
17:      end for
18:   end for
19: end while
```

D. Computational Complexity

Here, we present the algorithm (algorithm 2) with the time cost of each statement (i.e., line of the algorithm) and the number of times each statement is executed. Because there is a nested for loop in line 6, and the maximum value of *TRremPropLoad* is a factor of the number of nodes, we obtain a total running time of $T(\boldsymbol{n}) = O(\boldsymbol{n}^2)$ in the worst case. All other statements in the parent for loop have a constant running time.

Although the worst case is $O(\boldsymbol{n}^2)$, this is not likely to occur in practice due to the random nature of the WiMAX network topology in which the network shrinksand expands continuously. In addition, the farthest node from the BS would not have as many as (n-1) hops. The lower bound, however, for any network scheduling algorithm (working on the MAC layer as in the case of WiMAX) is actually $\Omega(\boldsymbol{n})$ even for the simplest communication requests. Some researchers have attempted to achieve lower bounds by applying scheduling algorithms only on strongly connected components (nodes) of a network graph.

Algorithm 2: Fair scheduling algorithm for WiMAX network

Input: Priority queue
Output: Time slot assignment to each node

```
1: packets of a node's buffer in ascending order upon
   generation time;
2: while true do                                                C1
3:    update priority queue on levels and totalPropLoad in      C2
      descending order;
4:    copy each node's totalPropLoad to TRremPropLoad;          C3
5:    for node =1 to length(queue) do                           C4,n
6:       for counter = TRremPropLoad to 1 do                    C5,n
7:       pick packets from node buffer and put in current-      C6
slot;                                                           C7
                                                                C8
8:       decrement TRremPropLoad of node;                       C9
9:       for chance = 1 to 2 do
10:         for otherNode = node+1 to node-1 do
11:         if (otherNode. TRremPropLoad > 0 OR
               chance=2) AND (other Node has no
               interference in the current slot)                C10
            then                                                C11
12               Pick packets from other Node's buffer and
                 put in current-slot                            C12
13:              Decrement TRremPropLoad of node
14:         end if
15:         end for
16:      end for
17:      end for
18:   end for
19: end while
```

IV. EXPERIMENTAL RESULTS

In this section, we have implemented our proposed WiMAX scheduling algorithm and the other well-known WiMAX scheduling algorithms *FIFO*, *Priority*, and *Round Robin* (RR) [1-5][15-19] for performance evaluation and comparison. We compare four parameters in the experimental results, namely, *transmission distance*, *packets transmitted*, *data rate*, and *throughput*. The results indicate that our algorithm provides fair scheduling while the other algorithms do not.

A. Simulation Environment

We used NetSim 10 (academic version) [25][26] to implement and simulate the proposed WiMAX scheduling algorithm. We also used the IEEE 802.16

protocol to set up and simulate a modified WiMAX mesh network. This WiMAX protocol is equipped with multiple channels and radios. It also supports different types of network topologies, such as chain, ring, and start. The supported and transmitted traffic types are CBR, Custom, Voice, Video, HTTP, E-mail, Database, and FTP. In our experimental simulation, SSs were designed in a start topology of space size (region) of 500 x 500 m. We set the number of SSs to eight (with four wired workstations and four wireless workstation), one BS, one main router, one switch, and one access point. All subscribers had different transmission ranges (distance) based on the scenarios described in following section. We measured the fairness of the algorithm by calculating the number of forwarded packets of each subscriber to its parent and BS.

B. Simulation Results

To validate the proposed WiMAX fair scheduling algorithm, we simulated a network of eight workstations or nodes (four wired nodes and four wireless nodes). We evaluated the performance (fairness) of the algorithm by calculating the number of forwarded packets of each node to its BS. Then, we compared the experimental results of the performance of our modified algorithm with that of three scheduling algorithms (FIFO, Priority, and RR*)* using the following four evaluation criteria (goals): transmission distance, packet transmitted, data rate, and throughput. The running time of all simulation results presented in this paper was 20 minutes. Fig. 6 presents the first network design scenario with which we simulated our modified WiMAX algorithm.

Table IV presents the experimental results of the modified WiMAX fair scheduling algorithm for the first network design scenario. The numbers in the table represent the transmission distance, packets transmitted, data rate, and throughput of each SS to its BS along with the data type (i.e., capacity request). The results demonstrate that our algorithm is able to consistently distribute the data rate (i.e., time slots). Thus, each subscriber node sends and receives its packet to its BS fairly.

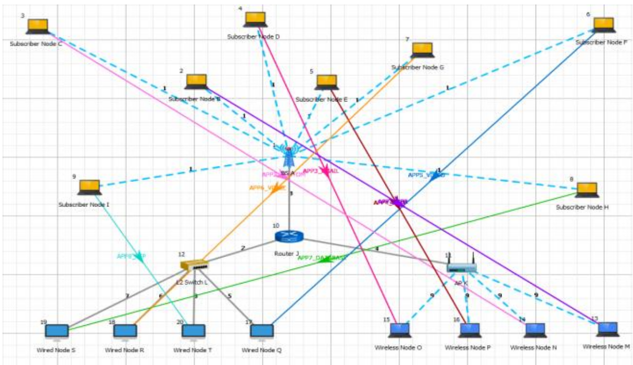

Fig. 6. First network design implementation scenario of modified WiMAX algorithm

TABLE IV: PERFORMANCE BASED ON THE USE OF PROPOSED WIMAX FAIR SCHEDULING ALGORITHM FOR THE FIRST IMPLEMENTATION SCENARIO

Subscriber # - data type	Transmission distance (meter)	Packets transmitted	Data rate	Throughput (Mbps)
Subscriber Node B - CBR	82.03	999	69.98	0.583
Subscriber Node C - Custom	204.47	999	62.20	0.583
Subscriber Node D - Email	131.52	266	69.98	0.152
Subscriber Node E - HTTP	40.80	252	69.98	0.144
Subscriber Node F - Video	254.53	199	69.98	0.005
Subscriber Node G - Voice	124.08	999	69.98	0.063
Subscriber Node H - Database	228.48	133	69.98	0.075
Subscriber Node I - FTP	160.46	207	69.20	0.120

TABLE V: PERFORMANCE BASED ON THE USE OF FIFO SCHEDULING ALGORITHM FOR THE FIRST IMPLEMENTATION SCENARIO

Subscriber # - data type	Transmission distance (m)	Packets transmitted	Data rate	Throughput (Mbps)
Subscriber Node B - CBR	82.03	999	34.99	0.583
Subscriber Node C - Custom	204.47	999	35.00	0.583
Subscriber Node D - Email	131.52	266	34.99	0.150
Subscriber Node E - HTTP	40.80	252	34.99	0.148
Subscriber Node F - Video	254.53	199	23.32	0.049
Subscriber Node G - Voice	124.08	999	34.99	0.063
Subscriber Node H - Database	228.48	133	31.10	0.076
Subscriber Node I - FTP	160.46	207	31.10	0.119

TABLE VI: PERFORMANCE BASED ON THE USE OF PRIORITY SCHEDULING ALGORITHM FOR THE FIRST IMPLEMENTATION SCENARIO

Subscriber # - data type	Transmission distance (m)	Packets Transmitted	Data rate	Throughput (Mbps)
Subscriber Node B - CBR	82.03	999	48.00	0.583
Subscriber Node C - Custom	204.47	999	48.00	0.583
Subscriber Node D - Email	131.52	266	48.00	0.152
Subscriber Node E - HTTP	40.80	252	48.00	0.152
Subscriber Node F - Video	254.53	199	42.66	0.050
Subscriber Node G - Voice	124.08	999	43.00	0.063
Subscriber Node H - Database	228.48	133	42.66	0.075
Subscriber Node I - FTP	160.46	207	43.00	0.113

TABLE VII: PERFORMANCE BASED ON THE USE OF RR SCHEDULING ALGORITHM FOR THE FIRST IMPLEMENTATION SCENARIO

Subscriber # - data type	Transmission distance (m)	Packets transmitted	Data rate	Throughput (Mbps)
Subscriber Node B - CBR	82.03	999	48.00	0.583
Subscriber Node C - Custom	204.47	999	48.00	0.583
Subscriber Node D - Email	131.52	266	48.00	0.152
Subscriber Node E - HTTP	40.80	252	48.00	0.152
Subscriber Node F - Video	254.53	199	42.66	0.050
Subscriber Node G - Voice	124.08	999	48.00	0.063
Subscriber Node H - Database	228.48	133	42.66	0.075
Subscriber Node I - FTP	160.46	207	48.00	0.113

TABLE VIII: PERFORMANCE BASED ON THE USE OF PROPOSED WIMAX FAIR SCHEDULING ALGORITHM FOR THE SECOND IMPLEMENTATION SCENARIO

Subscriber # - data type	Transmission distance (m)	Packets transmitted	Data rate	Throughput (Mbps)
Subscriber Node B - CBR	67.80	999	96.00	0.583
Subscriber Node C - Custom	196.02	999	64.00	0.583
Subscriber Node D - Email	224.39	266	69.00	0.152
Subscriber Node E - HTTP	238.17	252	69.00	0.143
Subscriber Node F - Video	50.08	199	69.00	0.050
Subscriber Node G - Voice	115.97	999	69.00	0.063
Subscriber Node H - Database	62.00	133	69.00	0.074
Subscriber Node I - FTP	84.53	207	69.00	0.120

For comparison, Tables V, VI and VII present results from applying FIFO, Priority, and RR employing the same evaluation criteria used in the first implementation scenario.

Our proposed WiMAX scheduling algorithm for data types *video* and *database* achieved a transmission distance of 254.53m and 228.48m with data rate percentages of 69.98% and 69.98%, respectively. These values are higher than those of any of the compared algorithms in Tables V, VI, or VII, and, to our knowledge, are superior to any simulation results achieved in the literature. This is because the proposed algorithm is able to resolve the fairness problem in O(n2) time, while the lower bound for any MAC layer scheduling is $\Omega(n)$. The algorithm provides fairness by considering the depth of SS nodes away from the BS as well as the load of each node.

Fig. 7 presents the second network design scenario with which we simulated our modified WiMAX algorithm. The simulation results in Table VIII indicate that in the second network design scenario, our WiMAX fair scheduling algorithm was able to schedule the subscriber nodes efficiently and fairly. Thus, the simulation results demonstrate that our proposed algorithm provides fairness in allocating data rates (time slots) for transmission.

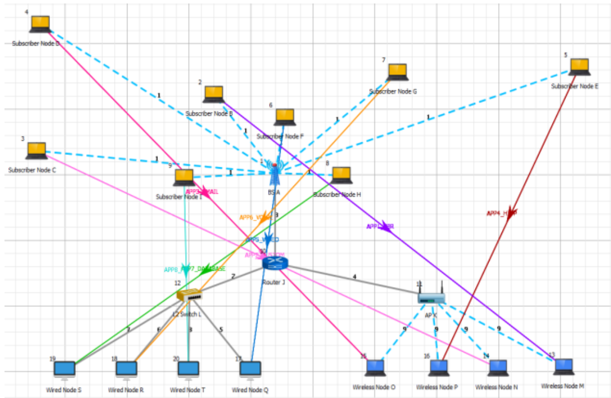

Fig. 7. Second network design implementation scenario of modified WiMAX algorithm

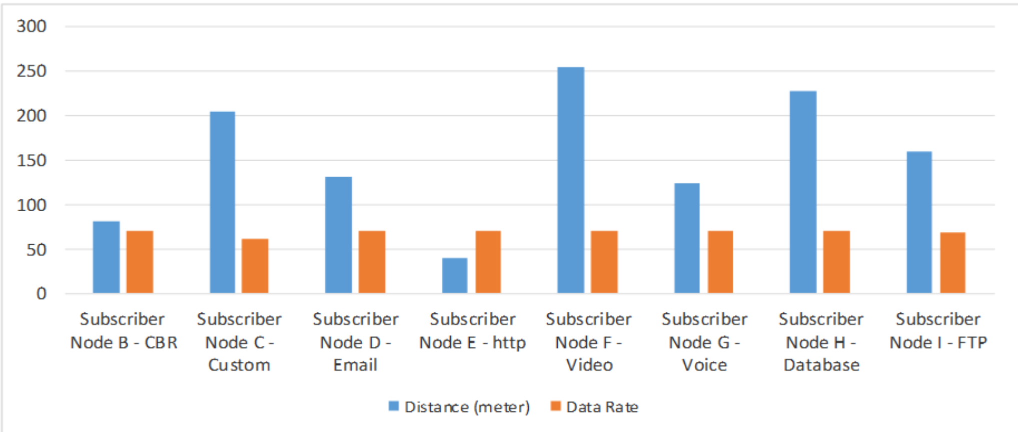

Fig. 8. First network design implementation scenario of modified WiMAX algorithm on distance and data rate (fairness).

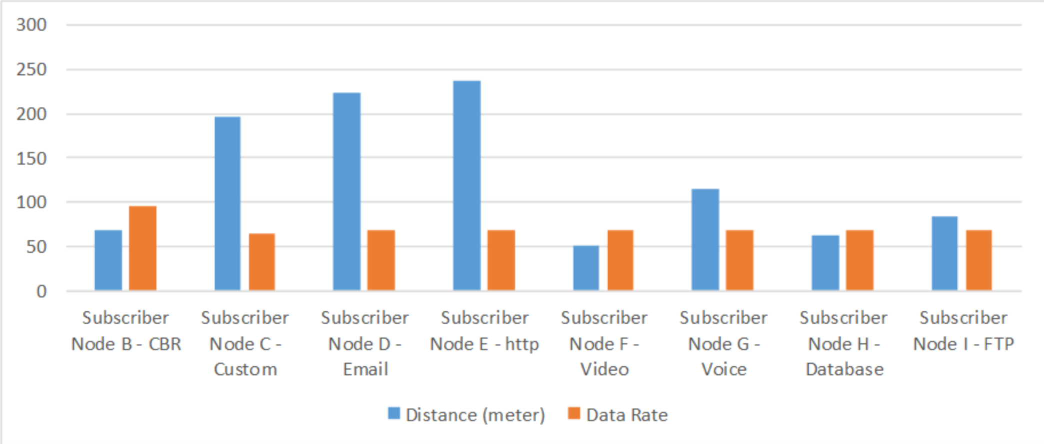

Fig. 9. Second network design implementation scenario of modified WiMAX algorithm on distance and data rate (fairness).

Fig. 8 and Fig. 9 indicate that the data rate percentages (time slots) of the two implementation scenarios are almost equal. When the transmission distance of the subscriber nodes increases or decreases, the data rate percentages of the two implementation scenarios are also almost identical (fairness). However, the experimental results demonstrate that our modified WiMAX fair scheduling algorithm achieves satisfactory performance against three common WiMAX scheduling methods, as evaluated on several metrics and parameters using two network design implementation scenarios.

V. Conclusions

Existing scheduling algorithms do not provide fairness when allocating data rates (time slots) to contending nodes. Thus, a scheduling algorithm must be used that controls exchanged messages for both the sender and receiver. The main goal of most wireless mechanisms is to increase the number of nodes transmitting in one-time slot (greedy approach) when the nodes transmit their packets concurrently and without interference, thus maximizing the throughput, as achieved by TDMA. Our proposed scheduling algorithm demonstrates high performance as well as fairness, providing a mechanism for giving every node a value that reflects the importance of the data it pushes to the BS.

The results presented in this paper are useful in the development of existing scheduling algorithms and can aid in enhancing the capacity of a mesh mode WiMAX network by designing an efficient and fair scheduling algorithm that resolves the bandwidth contention problem and determines the transmission order for users. In future work, we aim to develop an existing scheduling algorithm, which involves different fields. We aim to further develop a fair scheduling algorithm for WiMAX networks using appropriate and standard modeling tools, which will involve acquiring knowledge from experts in the field.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Khaled proposed the algorithm and wrote this paper. Khaled and Dinesh implemented the simulation scenarios. Khaled obtained the results and analyzed the simulation performance. Khaled and Dinesh had approved the final version.

References

[1] H. Kaur and G. Singh, "Implementation and evaluation of scheduling algorithms in point-to-Mutipoint mode in WiMAX networks," *Int. J. Comput. Sci. Inf. Technol.*, 2011.

[2] K. Sledziewski, B. Bordbar, and R. Anane, "A DSL-based approach to software development and deployment on cloud," in *Proc. International Conference on Advanced Information Networking and Applications*, 2010.

[3] IEEE, Std 802 (Oct. 1, 2004) 16-2004, "IEEE standard for local and metropolitan area networks," Part 16: air Interface for Fixed Broadband Wireless Access Systems.

[4] Y. Xu, Z. Li, X. Zhong, and H. Xu, "A bandwidth scheduling algorithm based on minimum interference traffic in mesh mode," *Wirel. Commun. Netw. Mob. Comput.*, 2007.

[5] M. El-Shennawy, A. Youssef, N. El-Derini, and M. Fahmy, "A proposed real-time scheduling algorithm for WiMAX networks," in *Proc. Ninth International Conference on Wireless and Mobile Communications*, 2013.

[6] W. Wang, B. Liang, and B. Li, On Low Complexity Multi-Resource Packet Scheduling with Dominant Resource Fairness.

[7] I. Newton, Resource Allocation in WIMAX Mesh Networks, Bachelor of Science, Kwame Nkrumah University of Science and Technology, 2009.

[8] K. Chandrashekhar and P. Janes, *Optimal Design of Wireless local Area Networks (WLANs) Using Simulation*, *Military Communications Conference*, MILCOM, IEEE Publications, Boston MA, Oct 18-21, 2009.

[9] S. Zhou, M. Zhao, X. Xu, J. Wang, and Y. Yao, "Distributed wireless communication system: A new

architecture for future public wireless," *IEEE. Commun*., March 200.

[10] Mesh Networking Forum, Building the Business Case for Implementation of Wireless Mesh Networks, Mesh Networking Forum 2004, October, San Francisco, CA, 2004.

[11] P. Adya, J. Bahl, A. Padhye, Wolman, and L. Zhou, "A multi-radio unification protocol for IEEE 802.11 wireless networks," in *Proc. Int. Conf. Broadband Netw*., Broadnets, 2004.

[12] P. Barford, J. Kline, D. Plonka, and A. Ron, "A signal analysis of network traffic anomalies," in *Proc. ACM. SIGCOMM. Internet Measurement Workshop*, 2002.

[13] H. Wei, S. Ganguly, R. Izmailov, and Z. Haas, "Interference-aware IEEE 802.16 WiMax mesh networks," in *Proc. 61st IEEE Vehicular Technology Conference*, May 2005.

[14] X. Gupta, Lin, and R. Srikant, "low-complexity distributed scheduling algorithms for wireless networks," in *INFOCOM*, 2007.

[15] (July 2008). WiMAX Forum, WiMAX system evaluation methodology V2.1. [Online]. Available: www.wimaxforum.org/technology/documents/

[16] N. Bayer, X. Bangnan, V. Rakocevic, and J. Habermann, "Improving the performance of the distributed scheduler in IEEE 802.16 mesh networks," in *Proc. Vehicular Technology Conference 2007 Spring*, 2007.

[17] M. Cao, "A tractable algorithm for fair and efficient uplink scheduling of multi-hop WiMAX mesh networks," *IEEE WiMesh, Vi Raghunathan P Kumar*, 2006.

[18] S. Lu, V. Bharghavan, and R. Srikant, "Fair scheduling in wireless packet networks," *IEEE. ACM. Trans. Netw*., 1999.

[19] J. Lakkakorpi, A. Sayenko, and J. Moilanen, "Comparison of different scheduling algorithms for WiMAX base station: Deficit round-robin vs. Proportional fair vs. Weighted deficit round-robin," in *Proc. IEEE Wireless Communications and Networking Conference*, 2008.

[20] K. Vinay, N. Sreenivasulu, D. Jayaram, and D. Das, "Performance evaluation of end-to-end delay, by hybrid scheduling algorithm for QoS in IEEE 802.16 network," in *Proc. International Conference on Wireless and Optical Communication Networks*, April 5, 2006.

[21] H. Shetiya and V. Sharma, "Algorithms for routing and centralized scheduling to provide QoS in IEEE 802.16 mesh networks," in *Proc. 1st ACM Workshop on Wireless Multimedia Networking and Performance Modeling*, New York, NY, USA, 2005.

[22] H. Guesmi, S. Maaloul, and R. Tourki, "Design of scheduling algorithm for QoS management on WiMAX networks," in *COMPUTER*, 2012, pp. 43–50.

[23] H. Shetiya and V. Sharma, "Algorithms for routing and centralized scheduling to provide QoS in IEEE 802.16 mesh networks," in *Proc. ACM Workshop on Wireless Multimedia Networking and Performance Modelling*, October 2005.

[24] Q. Li and W. Ba, "A group priority earliest deadline first scheduling algorithm," *Front. Comput. Sci.*, October 2012.

[25] O. Votava, "A network simulation tool for task scheduling," Dept. of Computer Science, Czech Technical University, 2015.

[26] B. Lewis, "NetSim user's manual department of mathematics and computer science," University of Richmond, October, 1991.

[27] A. Mohammadi and S. G. Akl, "Scheduling algorithms for real-time systems," *Tech. Rep*., July 15, 2005.

[28] A. Al-Howaide, Doulat, and Y. Khamayseh, "Performance evaluation of different scheduling algorithms in WiMAX," *Int. J. Comput. Sci. Eng. Appl.*, 2011.

[29] Y. Zhang and S. Liew, "Link-adaptive largest-weighted throughput packet scheduling for real-time traffics in wireless OFDM networks," in *Proc. IEEE Global Telecommunications Conference*, 2005.

[30] S. Jain, V. Shanker, and S. Tiwari, "Bandwidth allocation based on traffic load and interference in IEEE 802.16 mesh networks," *Daniele Tarchi*, March 13, 2013.

[31] C. Wu, H. Ho, S. Lee, and L. Lung, "Fairness time-slot allocation and scheduling with QoS guarantees in multihop WiMAX mesh networks," *Teen-Hang Meen*., October 9, 2013.

[32] M. Grine, R. Messoussi, and A. Najid, "A new distributed scheduling algorithm for bandwidth allocation in IEEE 802.16 mesh networks," Laboratory of Telecommunications Systems and Decision Engineering, (LASTID), Ibn Tofaïl University, Faculty of sciences, Kenitra, Morocco, August 1, 2013.

Khaled Abdalgader received the BSc degree in computer science from the Sebha University, Libya, in 2001, the MSc degree from the University Utara Malaysia in 2004 and the PhD degree in the area of natural language processing from La Trobe University, Australia, in 2012. He was at the Sebha University, Libya, from 2004 to 2007 as a lecturer in the Department of Computer Science. Since 2013, he has been an Assistant Professor with the Faculty of Computing and Information Technology, Sohar University, Oman. His research focuses on natural language processing and understanding, particularly word sense disambiguation, semantic text similarity, text mining, sentiment analysis and knowledge discovery from textual collections.

Dinesh Kumar Saini is working as associate professor in the faculty of computing and information technology since 2008 till date. Prior to this assignment, worked in the King Saud University Saudi Arabia. Main area of research is security, Software Systems, Content Management systems, Learning Objects, Higher Education. Worked on mathematical modelling in cyber Systems, Malicious Object Propagation and Immune system design.