INVITED PAPER

Centralized Resource Allocation for Multimedia Traffic in IEEE 802.16 Mesh Networks

Simulation results show that efficient scheduler algorithms for metropolitan wireless networks can be designed to handle quality voice and video distribution.

By Spyros Xergias, *Student Member IEEE*, Nikos Passas, *Member IEEE*, and Apostolis K. Salkintzis, *Senior Member IEEE*

ABSTRACT | The IEEE 802.16 standard provides a high degree of flexibility for setting up and operating wireless broadband networks in metropolitan environments. The standard supports numerous capabilities, including mesh topologies and multimedia communications. In this paper, we study these two features by investigating how efficiently an IEEE 802.16 mesh network can treat distributed multimedia traffic by providing differentiated quality of service (QoS). A key component of the system is the "enhanced frame registry tree scheduler" (E-FRTS) that provides QoS-aware resource allocation using a tree structure to prepare the creation of time frames and reduce processing requirements at the beginning of each frame. Simulation results show that distributed multimedia traffic can be efficiently supported in mesh 802.16 networks, provided efficient scheduling and a reasonable number of hops.

KEYWORDS | Algorithm; deadlines; mesh networks; scheduling; traffic scheduling; tree structure; 802.16

I. INTRODUCTION

IEEE 802.16 [1] is a standard that aims at filling the gap between local- and wide-area networks by introducing an advanced system for metropolitan environments. In such a system, both point-to-multipoint (cellular) and mesh mode configurations can be supported, while node mobility is also covered by the recent amendment 802.16e [2]. One of the main advantages of the standard is the large degree of flexibility it provides by supporting a wide range of traffic classes with different characteristics and quality of service (QoS) requirements. This is attained through a large set of parameters that allow users to describe in detail their traffic profiles and service needs. Being out of its scope, the standard does not describe a specific traffic scheduler to utilize these parameters, leaving that as an implementation differentiator.

This paper describes a scheduler for the centralized mesh mode of 802.16 referred to as the "enhanced frame registry tree scheduler" (E-FRTS). Its main characteristic is that it prepares each time frame in advance and avoids complex processing during the short time period between two consecutive frames. Its operation is based on a flexible tree structure (the "enhanced frame registry tree") that maps the scheduler's decisions. We describe the basic operation of this scheduler and show how it can be efficiently used in an IEEE 802.16 mesh network to handle multimedia traffic, including voice and multicast video. Although this paper focuses on stationary subscriber stations, E-FRTS can easily serve mobile stations as well, as it does not use any features that have been canceled in 802.16e [2].

This paper is organized as follows. Section II presents the main characteristics of IEEE 802.16, and more specifically its QoS capabilities and proposed scheduling algorithms. Section III describes the operation of E-FRTS, focusing on the procedures required in the enhanced frame registry tree. Section IV contains the description of the simulation model used in our experiments and the obtained results. Section V contains conclusions and plans for future work.

II. IEEE 802.16 BROADBAND ACCESS SYSTEMS

A. Basic Operation

The IEEE 802.16 standard covers the two lower layers of the protocol stack, i.e., the physical and the data link

Manuscript received October 23, 2006; revised July 5, 2007.
S. Xergias and **N. Passas** are with the Department of Informatics and Telecommunications, University of Athens, 15784 Athens, Greece (e-mail: xergias@di.uoa.gr; passas@di.uoa.gr).
A. K. Salkintzis is with Motorola, 15125 Athens, Greece (e-mail: salki@motorola.com).

Digital Object Identifier: 10.1109/JPROC.2007.909929

layer. The physical layer operates at 10–66 GHz (802.16) and below 11 GHz (amendment 802.16e [2]) with data rates between 32 and 130 Mbps, depending on the channel frequency width and modulation scheme. Three operation modes are defined: point-to-multipoint (PMP) and centralized and distributed mesh modes, but a combination of centralized and distributed mesh is also possible. The system architecture consists of i) base stations [(BSs) for PMP and mesh base stations (MBSs) for mesh mode] that have direct connections to backhaul services and are responsible for a specific area cell, ii) stationary subscriber stations (SSs), and iii) mobile subscriber stations (MSSs). As shown in Fig. 1, MBSs, SSs, and MSSs constitute neighborhoods operating in different frequencies, although schemes of frequency reuse are often used where possible.

In the PMP and centralized mesh modes, each BS/MBS regulates all communications in its cell. The main difference between PMP and centralized mesh is that, in PMP, all direct communications occur only between the BS and any SS of its cell, while centralized mesh allows MBS-controlled communications between different SSs of the same cell. MBS is informed about the network topology through the network control subframe during system initialization. On the contrary, in distributed mesh, there are only direct communications between SSs without the control of a MBS. An example topology of centralized mesh is shown in Fig. 1.

All link-layer communications are multiplexed with either time-division duplex (TDD) or frequency-division duplex (FDD). The scheduler described later in this paper applies in the TDD mode. A TDD frame has a fixed duration, which may take several values (0.5, 1, or 2 ms for PMP mode and 2.5, 4, 5, 8, 10, 12.5, or 20 ms for mesh mode). Various transmission parameters, including the modulation and coding schemes, may be adjusted individually for each SS on a frame-by-frame basis. Unlike the PMP mode, where the frame is divided into a downlink (DL) subframe and an uplink (UL) subframe, in the mesh mode each frame is divided into a control subframe and a data subframe. Each PMP subframe, as well as the data subframe in mesh mode, consists of an integer number of physical slots (PSs) or minislots (few PSs each), respectively, that represent the bandwidth allocation units.

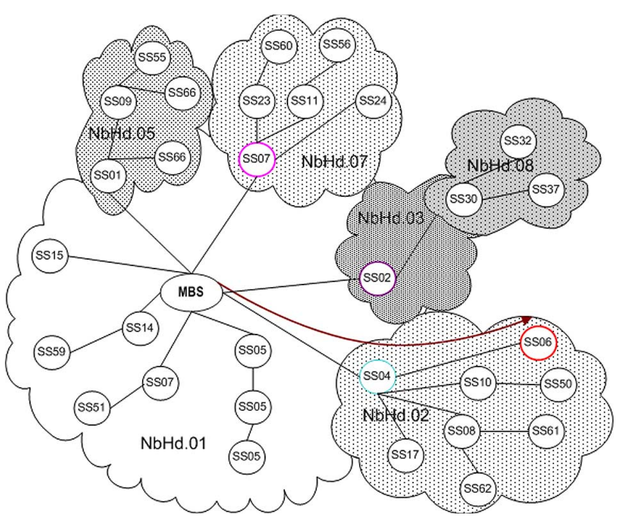

Fig. 1. ***Example topology of an IEEE 802.16 mesh network.***

In PMP mode, the schedule of each frame is carried through the DL-MAP and UL-MAP fields that state the PSs at which transmission bursts begin on the downlink and uplink, respectively. Both DL-MAP and UL-MAP are transmitted during one or more downlink bursts, since each one of them can refer to different modulation or coding types. Each SS receives and decodes the DL-MAP looking for MAC headers that indicate data for that particular SS in the remainder of the downlink subframe. Similarly, in the UL-MAP, each SS receives information on its transmission opportunities in the uplink subframe. In this way, an SS that decoded the corresponding control information contained in the DL-MAP/UL-MAP knows exactly during which PSs of the downlink/uplink subframe it is allowed to receive/transmit data together with the reception/transmission parameters (modulation, coding). Allocations in both directions should be compatible with the traffic characteristics and QoS requirements of each active connection and satisfy, where possible, time-varying transmission requests.

On the other hand, in mesh mode, both DL-MAP and UL-MAP are combined in one message that carries the frame's schedule and is transmitted in the control subframe. This message is called mesh centralized scheduling (MSH-CSCH) for the centralized mesh mode and mesh distributed scheduling (MSH-DSCH) for the distributed mesh mode. Each SS decodes the MSH-CSCH or MSH-DSCH message and obtains the transmitting schedule for all its downlink and uplink transmission opportunities. Again, it is the scheduler that decides on the specific allocations in both directions of all neighborhoods.

In all transmissions, data bits are randomized, forward error correction encoded, and mapped to one of the mandatory [spread binary phase-shift keying (BPSK), BPSK, quadrature phase-shift keying (QPSK), 16-quadrature amplitude modulation (QAM), 64-QAM] or optional (256-QAM) signal constellations.

B. Scheduling Services and QoS in IEEE 802.16

IEEE 802.16 can support multiple communication services (data, voice, video, etc.) with different QoS requirements. Unlike the mesh mode, a detailed QoS mechanism is described in the standard for the PMP mode where different data communications of an SS are organized into different connections. Each connection is associated with a single data service and specifies a set of traffic and QoS parameters that quantify its traffic behavior and QoS expectations (e.g., minimum reserved traffic rate, maximum sustained traffic rate, maximum latency, traffic priority, etc.). Every connection should belong to one of the four different services as defined by the standard: unsolicited grant service (UGS), real-time polling service (rtPS), non-real-time polling service (nrtPS), and best effort service (BE). Video multicasting, which is the most usual multicast traffic, is usually characterized as rtPS, in contrast to constant-bit-rate voice, which is set to UGS; and video on demand, which is set to nrtPS. In [2], a new service, referred to as enhanced rtPS (ertPS), was defined to better support real-time service flows that generate variable size data packets on a periodic basis, e.g., voice over IP with silence suppression.

The traffic scheduler located at the BS decides on the allocation of the PSs in each time frame. In addition to whatever other factors the scheduler may consider, the following items can be taken into account for each active connection:

- the scheduling service specified for the connection;
- the values assigned to the connection's QoS parameters;
- the availability of data for transmission (queue size);
- the capacity of the available bandwidth.

Uplink scheduling is performed by the BS with the aim of providing each SS with enough bandwidth for uplink transmissions or opportunities for extra transmission requests. When an SS needs additional bandwidth, it utilizes its transmission opportunities during contention periods or when it is polled by the BS, depending on its agreed QoS characteristics, to pass its transmission requests. Downlink scheduling, on the other hand, considers packets waiting for transmission at the BS as implicit requests for bandwidth allocation.

In mesh mode, all communications are considered connectionless and occur in the context of a link that is established between two SSs. QoS is actually provided over these links on a message-by-message basis. This scheme supports a basic QoS provision system and requires numerous control messages and requests.

C. Scheduling Algorithms for IEEE 802.16

A specific scheduling algorithm is not described in the IEEE 802.16 standard, neither for PMP nor for mesh modes, because it is not included among the mandatory modules required for the standardized system's operation. On the other hand, the operation of the scheduler is important for the performance of the whole system, and this is why it has attracted growing attention over the last couple of years. In the literature so far, a limited number of papers can be found proposing scheduling algorithms for 802.16. These proposals are based mostly on extensions and combinations of ideas already applied in systems prior to IEEE 802.16, such as the IEEE 802.11 wireless local-area network, and they focus mainly on the PMP mode.

A call admission control (CAC) mechanism together with a scheduler for PMP mode is described in [3], where "earliest deadline first" is used for rtPS connections and "weighted fair queuing" is used for nrtPS connections. In [4], a scheduler based on the CAC mechanism of [3] is proposed that uses token buckets to characterize the traffic flows. Reference [5] describes a scheduling algorithm for centralized mesh that provides end-to-end QoS to different flows in the network by handling user datagram protocol (UDP) and transmission control protocol (TCP) traffic separately at first and then considering them jointly. Based on a queuing analytical model, [6] proposes a queue-aware adaptive uplink bandwidth allocation and rate-control mechanisms for polling service in IEEE 802.16. In [7], an adaptive bandwidth reservation method based on data mining techniques is described. A two-layer service flow management architecture is proposed in [8], which is based on deficit fair priority queue. Reference [9] proposes an interference-aware framework for 802.16 mesh mode based on an interference-aware route construction algorithm. Lastly, in [10], an analytical model for the distributed scheduling algorithm in the IEEE 802.16 mesh mode is presented.

Two of the major limitations of mesh mode are i) the lack of an advanced QoS scheme, similar to that of the PMP mode, and ii) the lack of multicasting support. The former is handled in [12], which proposes a solution referred to as service adaptive QoS (SAQoS). According to this, the MBS assigns five node identifiers (IDs) instead of one to each MSS. These five virtual nodes represent the five scheduling services of PMP mode, explained earlier (UGS, ertPS, rtPS, nrtPS, and BE), and correspond to different connection types within the node. Each one of these virtual nodes may request bandwidth individually, and the MBS handles these requests according to their scheduling services in a way very similar to the PMP mode. In this way, a PMP-like QoS provision mechanism can be supported. The latter limitation is handled in [13], where an easy-to-implement tree construction algorithm is presented based on the adjustment of the unicast centralized routing tree.

The scheduler described in the next section utilizes the ideas of [12] and [13] in order to propose an advanced QoS scheduling scheme for the centralized mesh mode that supports multicasting. For simplicity reasons, in the rest of the paper "mesh mode" stands for "centralized mesh mode" and "connection" stands for "virtual node" as defined in [12].

III. EXTENDED FRAME REGISTRY TREE SCHEDULER

Traditionally, most traffic schedulers for TDD systems operate right before the beginning of each time frame to decide on its structure and content. Considering the complexity of the QoS provision mechanism in IEEE 802.6 as the price to pay for flexibility, efficient traffic scheduling implies complex calculations, increasing the requirements for expensive hardware. The support of advanced characteristics such as mesh topologies, separation in neighborhoods, frequency reuse, use of sectorized antennas, etc., should all be considered in an efficient resource allocation process. This means that even a simple scheduling mechanism must include a minimum set of tasks, such as classification of the connections and selection of the packets to be sent per connection, based on a large set of traffic and QoS parameters. For the mesh mode, supplementary actions include:

- consideration of the frequency reuse scheme;
- classification of the connections, based on their neighborhood and the sector of their father SS or MBS;
- consideration of the path that a packet must follow to reach its destination;
- proper allocations to intermediate links;
- grant of supplementary slots to provide request opportunities to distant SSs.

The packet scheduler described here, referred to as the E-FRTS, comes as an extension for mesh mode to the previously proposed frame registry tree scheduler [11] that focused only on the PMP mode. Its main aim is to distribute the computational complexity, required for a time frame preparation, throughout the whole lifetime of data packets and lead to reduced computational requirements. Furthermore, it provides advanced organization of the transmitted data, facilitating transmission either from the MBS to neighboring and distant SSs or from individual SSs to the MBS and their parent SSs.

Besides basic operation, the scheduler has to deal with possible modifications of one or more transmission parameters of a connection, such as the modulation type and the service type. For example, if a distant SS, located two hops away from the MBS, changes its modulation from 64-QAM to 16-QAM, more modulated symbols will be required for the data transmission between that SS and its father SS. This means that future schedules in the whole transmission path need reconfiguration.

A. The Scheduler Goals

The basic idea of E-FRTS is to schedule the transmission of each piece of data in the last time frame before its deadline. More specifically, the main objectives of E-FRTS are the following.

1) Each frame should have its transmissions organized in modulation order from BPSK to 256-QAM, in both downlink and uplink subframes (as indicated in [1]).
2) A per QoS service treatment of the transmissions should be possible, based on a specific priority strategy (strict, weighted, portioned, etc.).
3) Data packet transmissions should be based on their deadline, as explained in more detail later on.
4) Transmissions should be organized per SS and per connection.
5) For distant nodes, proper allocations to intermediate links should be made.
6) The required operations, before the beginning of a time frame transmission, should be limited.
7) Changes in network topology, node modulation, or QoS services should be possible with minimum processing effort.

The above objectives are served well with the use of the tree structure of Fig. 2, referred to as the *enhanced frame registry tree*. In its full version, the tree consists of seven levels, although some of them can be omitted, depending on the specific network configuration, reducing memory requirements and computational complexity.

- *First Level:* Represents the different neighborhoods in the specific network topology.
- *Second Level:* Represents the time frames immediately following the present one, in a sequential order (i.e., TF_1 is immediately after the present frame, TF_2 the next one, etc.). Since the maximum latency parameter of the QoS profile can be 2^{32} ms (4 bytes), as indicated in [1], the maximum number of time frames existing in the enhanced frame registry tree can be easily calculated considering the frame length used in the system, which can vary from 2.5 to 20 ms [1]. Naturally, this is a theoretical upper limit, and some tens of time frames are sufficient in most of the cases.
- *Third Level:* Represents the direction (uplink or downlink).
- *Fourth Level:* Represents the different sectors in which the children SSs belong to.
- *Fifth Level:* Corresponds to the available modulation types. This level can consist of a maximum of 12 nodes, each one representing one of the six possible modulation types (spread BPSK, BPSK, QPSK, 16-QAM, 64-QAM, and 256-QAM) both for uplink and downlink.

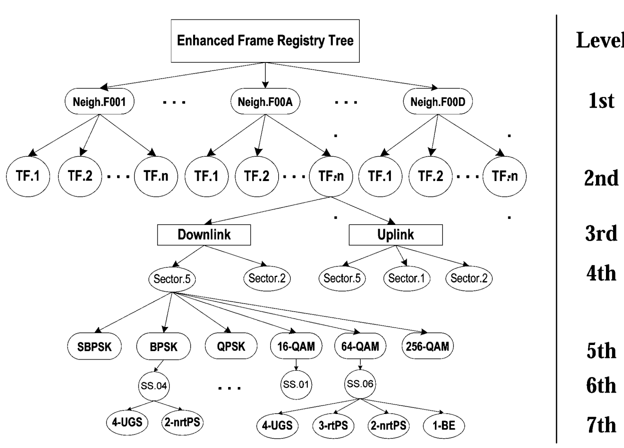

Fig. 2. *The enhanced frame registry tree.*

- *Sixth Level:* In this level, connections are organized per SS. Every SS can have a maximum of two nodes at this level, one for the uplink and one for the downlink, since every SS uses only one modulation each time.
- *Seventh Level:* Consists of one leaf for every active connection, i.e., virtual node, holding the number of data packets scheduled for transmission in that time frame for the specific connection. Optionally, the leaves could store the number of symbols required for the transmission of the packets, in order to save some calculations during the frame creation process.

B. The Scheduler Operation

The operation of the scheduler can be divided into two main procedures: 1) packet/request arrival and 2) creation of next time frame for transmission.

1) Packet/Request Arrival: Since the scheduler treats both new packets arriving for the downlink and new requests for packet transmissions from the uplink in the same way, only the downlink case is described here. It is assumed that a traffic policing mechanism is available at the MBS (e.g., leaky bucket) to ensure that the incoming traffic is consistent with each connection's declarations during setup. For every packet, we distinguish two cases, depending on whether the packet deadline can be calculated or not. In both cases, a number of leaves should be updated, equal to the number of hops between the packet's source (MBS in this case) and destination within the mesh network.

i) For UGS, ertPS, and rtPS services, the latency is given, and the deadline of a packet P_i can be calculated as

$$\mathrm{Deadline}(P_i) = \mathrm{ArrivalTime}(P_i) + Latency(C(P_i)) - \mathrm{FrameDuration} \times h(C(P_i))$$

where $C(P_i)$ is the connection that packet P_i belongs to and $h(C(P_i))$ the number of hops to the destination of $C(P_i)$. In this case, the leaves of the packet's connection in the last $h(C(P_i))$ time frame subtrees before the packet's deadline are updated (i.e., the number of packets scheduled for transmission in these leaves is incremented by one). If some of the corresponding leaves do not exist at that time (e.g., this is the first scheduled transmission for a time frame), the appropriate paths are created.

ii) For nrtPS and BE services, the latency is not given, so no specific deadline can be calculated. In this case, the leaves of the $C(P_i)$ in the last $h(C(P_i))$ *existing* time frames in the tree at the time of the packet arrival are updated.

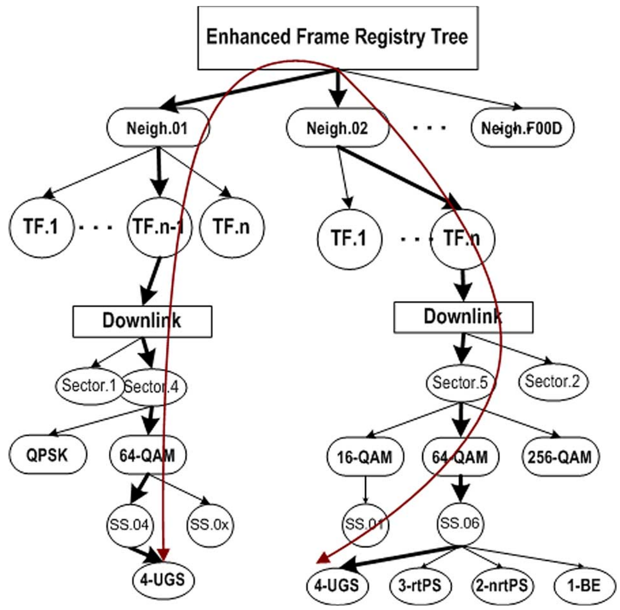

Fig. 3. *A packet insertion example.*

As an example, let us assume the topology of Fig. 1 and the case of a downlink packet arrival at the MBS for connection 4-UGS that should be sent to subscriber station 6 (SS06), which belongs to sector 05 of its father SS (SS04), and both belong to neighborhood 2 (NbHd. 02), using 64-QAM modulation. If the last frame that this packet can be received by SS06, based on its deadline, is TFn, the required path to be updated is NbHd.02 → TFn → DL → Sec.05 → 64QAM → SS06 → 4-UGS as is shown in Fig. 3 (right subtree). If the leaf of connection 4-UGS exists, its counter increases by one. In a different case, the required path is created and the leaf counter is set to one. Moreover, TF_{n-1} of MBS's neighborhood must be also updated, so SS04 can receive the particular packet in the previous frame. Thus, if the neighborhood of the MBS is NbHd.01, SS.04 belongs to sector 02 of the MBS, and modulation between MBS and SS.04 is 64QAM, then the path NbHd.01 → TFn – 1 → DL → Sec.02 → 64QAM → SS04 → 4-UGS must be also updated as is shown in Fig. 3 (left subtree).

The packet/request arrival procedure does not have to deal with the details of time frame sizes and structure. It simply places packet transmissions to the appropriate leaves, leaving the frame creation procedure to decide on the final transmitted frame structure and content.

2) Frame Creation: The frame creation procedure is responsible for deciding on the contents and structure of the next time frame of all neighborhoods, based on the information stored in the enhanced frame registry tree. Based on the topology of Fig. 1, let us assume that at the beginning of one time frame, there are $m1$ time frame subtrees for MBS's neighborhood ($\text{NbHd.01} \rightarrow \text{TF}_1, \ldots, \text{TF}_{m1}$) and m_2 time frame subtrees for neighborhood NbHd.02 ($\text{NbHd.02} \rightarrow \text{TF}_1, \ldots, \text{TF}_{m2}$). The next frames for each neighborhood will be generated mainly from the first time frame subtrees (TF_1s). In the case of empty slots remaining in some TF_1s, these can be filled with packets from the next time frames ($\text{TF}_2, \text{TF}_3, \ldots$). Three cases can be distinguished for each neighborhood's TF_1: i) the number of packets under subtree TF_1 fits exactly into one time frame, ii) the number of packets under subtree TF_1 is less than the capacity of a time frame, and iii) the number of packets under subtree TF_1 is more than the capacity of a time frame.

The first is the best possible case, where the packets for transmission exactly fit into the available frame slots. In this case, the frame can be transmitted without further processing, exactly as the leaves of TF1 indicate.

In the second case, there are two options for the empty slots: either to be filled with packets from the next time frames or to be left for contention. The exact policy to be followed depends on a number of factors and is out of the scope of this paper.

In the third case, nrtPS and BE packets can be moved to the next time frame. Using the tree structure, "moving" means a simple change of pointers and update of leaves. If, even after this operation, there are still excess packets for transmission (e.g. UGS, ertPS, rtPS), some of them should be rejected, since it is not possible to be transmitted in a later frame. In our simulations in the next section, we used a simple random rejection policy, but a number of other techniques may apply to ensure a fair selection process.

Other procedures of the scheduler to support modulation or QoS service changes can be easily performed through simple subtree repositioning.

The scheduler operation, as described above, reduces data fragmentation and physical overhead due to modulation changes. QoS classification is attained by transmitting high-priority before low-priority traffic and starting from low priority when packets have to be rejected. Additionally, the scheduler maintains low computational complexity by distributing the required processing in time, without significantly increasing memory requirements. A notable advantage of the trees structure is its scalability to support future changes of the standard. For example, it can easily support more QoS services, or more modulation types that may be added in the future.

IV. SIMULATIONS

In order to measure the performance and effects of distributed multimedia transmission in IEEE 802.16 mesh networks, a simulation program was constructed in C++. The program reflects the full functionality of IEEE 802.16 mesh mode, as well as the E-FRTS scheduler. We considered two simulation scenarios: one with multiple types of traffic per SS, including multimedia, and one with voice and multicast video, to evaluate different features of the standard and the scheduler.

A. Multiple Traffic Types Scenario

In this scenario, we used a simple topology with two hops, as in Fig. 4, and an increasing number of subscribers, each one with the following connections:

- one UGS with constant data rate of 64 Kbps (e.g., voice) and latency equal to 20 ms;
- one rtPS with mean data rate of 256 Kbps and latency equal to 40 ms;
- one nrtPS with mean data rate of 128 Kbps;
- one BE with mean data rate of 128 Kbps.

All the connections of a certain type were considered independent and statistically identical. The time frame length was set to 1 ms, the packet size to 54 bytes, and the modulation to 64-QAM for all SSs, leading to a transmission speed of 120 Mbps. To limit the simulation to reasonable numbers, we assume that only 50% of the bandwidth was available for the above traffic. E-FRTS was compared against a simpler scheduler that serves packets in a first-come first-served order, without caring about deadlines or other QoS characteristics. To provide minimum protection of UGS and rtPS traffic over nrtPS and BE, the simple scheduler sets the maximum number of BE packets in the output queue to 2000 and the maximum number of nrtPS packets in the output queue to 4000. Our intention was to show that the per service type differentiated treatment provided by E-FRTS, together with its deadline-based scheduling, can attain much better performance, especially for delay-sensitive traffic.

Fig. 5 shows the percentage of packet losses per service type for different numbers of subscribers. As expected, using the simple scheduler, UGS connections are the first that start losing packets due to expiration because the system cannot protect them from non delay sensitive nrtPS and BE traffic. E-FRTS on the other hand, manages to provide a differentiated service to packets according to their service type, leading to considerably better performance for high-priority types. With its deadline-based logic, it manages to transmit delay-sensitive packets in time and reduce the number of losses. This is depicted in the figure, where UGS connections

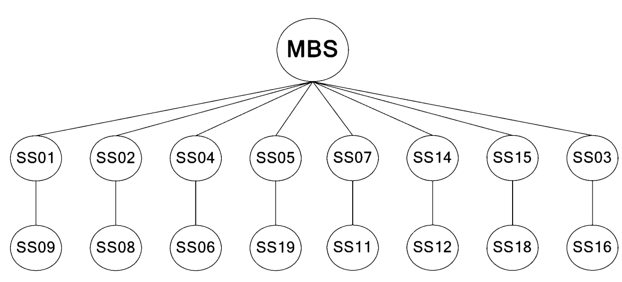

Fig. 4. ***The simulation topology.***

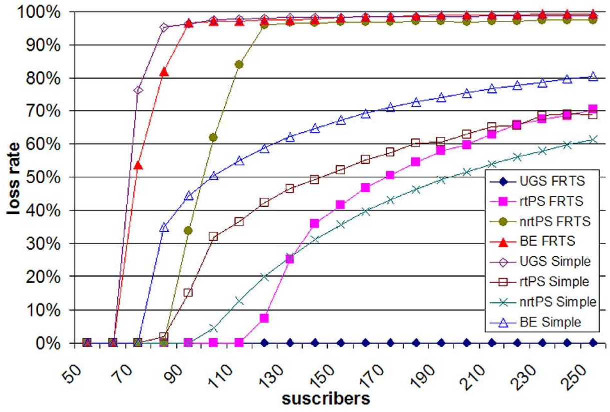

Fig. 5. ***Packet losses per service type.***

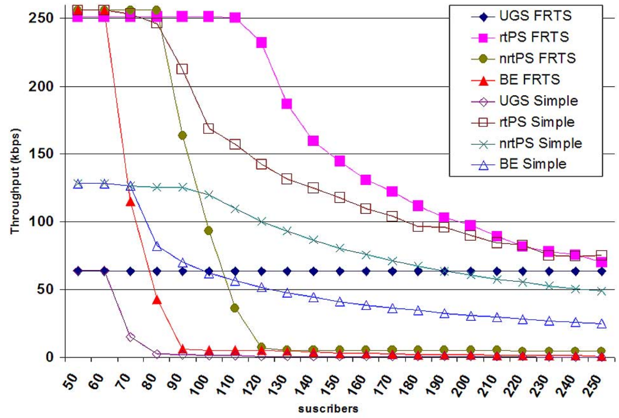

Fig. 7. ***Throughput per service type.***

have no losses even for a large number of SSs, while rtPS starts losing packets after 110 SSs.

In Fig. 6, the mean packet delay per service type is shown. For the simple scheduler, the mean delay is increasing with the number of SSs for all types, as the scheduler does not protect delay-sensitive data. E-FRTS, on the other hand, schedules delay-sensitive packets close to their deadline and attains a much smoother behavior, leading to the support of more SSs. The price to pay is the delays for BE and nrtPS, which increase quickly up to 65 and 118 ms for 85 and 115 SSs, respectively, before falling down to zero, as no packets are served after these points.

Fig. 7 shows the throughput per service type. Again, the differentiated treatment of E-FRTS leads to a throughput close to the generated traffic of delay-sensitive connections for more number of SSs compared to the simple scheduler.

B. Multicast Scenario

In this scenario, we considered a balanced mesh topology similar to that of Fig. 8, with one MBS and a variable number of SSs. Again, we used a time frame length of 1 ms, ideal channels, and the same modulation for all users (64-QAM), resulting in an overall available bandwidth of 120 Mbps. Each SS had two connections: one voice connection of type adaptive multirate (AMR) [14] and one real-time compressed video connection. Additionally, half of the SSs belonged to a multicast group that received high-bit-rate video. Again, 50% of the bandwidth was considered available for this traffic.

Two multicast cases were tested for different numbers of SSs in order to measure the effects of multicasting on the overall network performance. In the first case (good case), the multicast SSs were all grouped together in the left part of the network [Fig. 8(a)], while in the second (bad) case, the multicast SSs were uniformly spread to the entire network [Fig. 8(b)]. In both cases, simulations were

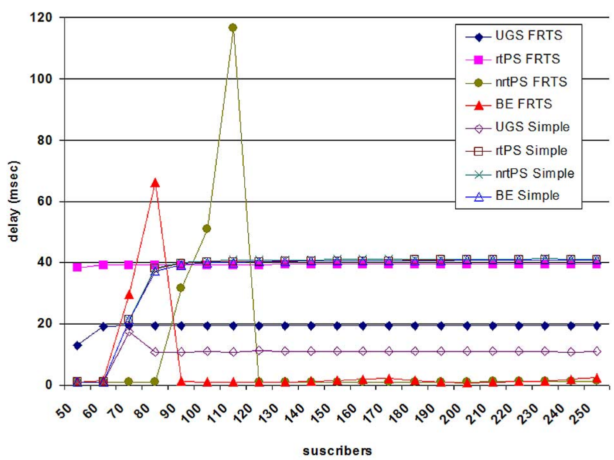

Fig. 6. ***Mean delay per service type.***

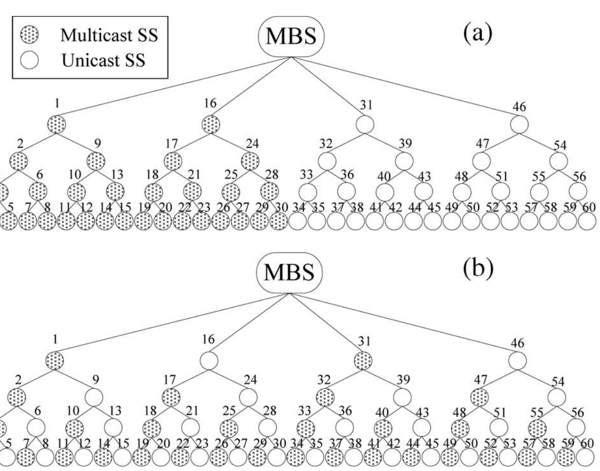

Fig. 8. ***The simulation example.***

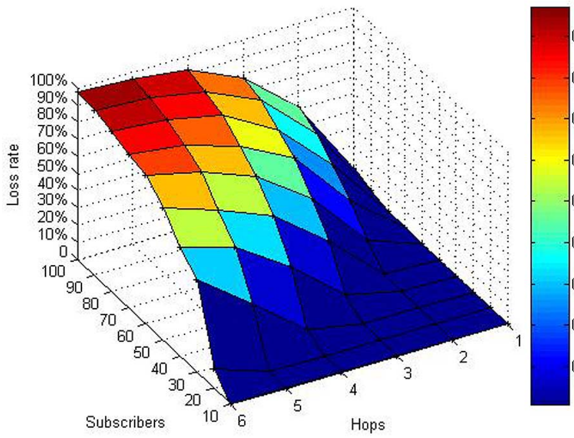

Fig. 9. *Voice packet loss in the grouped case.*

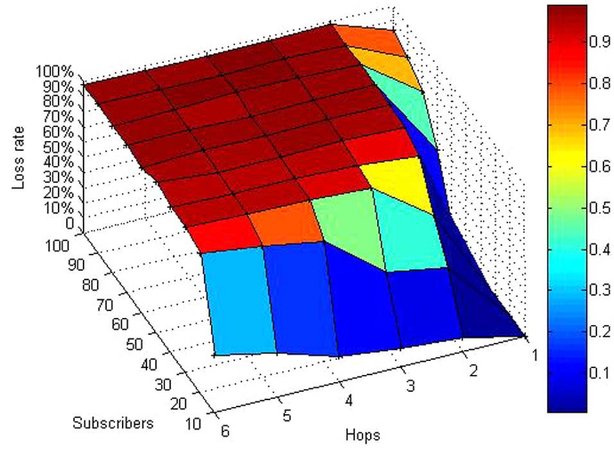

Fig. 11. *Multicast video packet loss in the grouped case.*

performed for different numbers of hops to investigate how this can affect the system's performance. Again, all the connections of a certain type were considered independent and statistically identical.

Each AMR voice connection was assigned to the ertPS type and modeled as an ON–OFF traffic source with 50% duty cycle and exponentially distributed ON and OFF periods with mean value of 3 s each [14]. At the beginning of an ON period (voice burst), the AMR transmission rate was randomly selected from within three possible values: 4.75, 10.2, and 12.2 kbps (as considered in [15]). During the OFF period (silence), the rate was constant and equal to 1.95 kbps [16]. Each AMR speech frame was encapsulated in one RTP/UDP/IP packet as per RFC 3267. The latency was set to 30 ms.

On the other hand, each real-time video connection was assigned to the rtPS type. Its transmission rate was variable with a minimum bit rate varying from 64 to 128 Kbps and a maximum bit rate varying correspondingly from 256 to 512 Kbps, while the latency was set to 40 ms. Similarly, multicast video connections were also characterized as rtPS, with a minimum bit rate varying from 1 to 2 Mbps and a maximum bit rate varying from 2 to 4.5 Mbps. The latency of this traffic was set to 50 ms.

Experiments for all the combinations of the number of SSs (ranging from 10 to 100), hops (ranging from 1 to 6), and multicast cases (grouped or spread) we performed.

Figs. 9–11 show the packet losses for the three kinds of traffic in the grouped case. As can be observed, the voice traffic has the least lost packets, as a result of the high-priority treatment it enjoys by the scheduler. What is common in the three figures is that the system's performance drops rapidly as the number of hops increases above two hops for more than 30 subscribers. This is mainly due to the deadline limits of these types that cannot be satisfied by a network of this

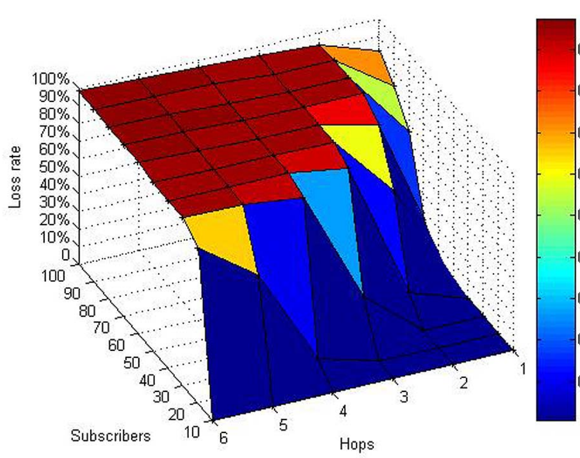

Fig. 10. *Unicast video packet loss in the grouped case.*

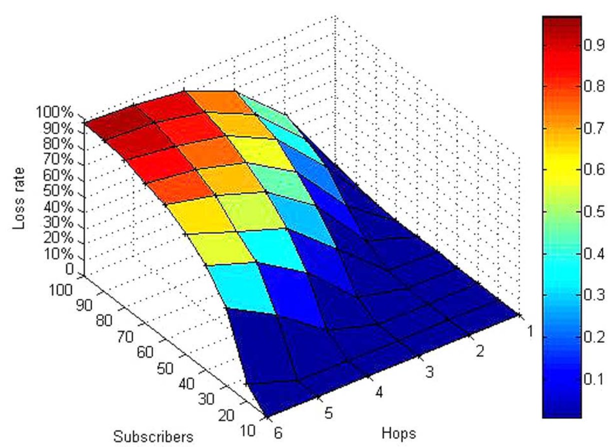

Fig. 12. *Voice packet loss in the spread case.*

configuration. This leads to the conclusion that network designers should be very careful when adding more hops in an IEEE 802.16 mesh network because this might lead to considerable degradation of the quality provided to real-time traffic. Finally, we noticed that the spread multicast case did not add considerably more complexity to the system, as the results were slightly worse than the grouped case. To show this, we present in Fig. 12 the packet losses for the voice traffic in the spread case, which are slightly worse than that in the grouped case (Fig. 9).

V. CONCLUSIONS

Multicasting and mesh mode are two of the advanced features provided by IEEE 802.16. In this paper, we investigated how distributed multimedia can be supported in an IEEE 802.16 mesh network and made extensive simulations experimenting with different topologies. Considering the traffic scheduler as a vital performance differentiator, we described the enhanced frame registry tree scheduler. E-FRTS uses the frame registry tree, a data structure that aims at preparing time frame creation and reducing processing needs at the beginning of each frame. Using this structure, the algorithm schedules each packet at the last time frame before its deadline, in order to allow more packets to be transmitted and increase throughput. Additionally, the scheduler takes into account the peculiarities of the mesh topology to maximize performance.

Simulation results show that distributed multimedia traffic can be efficiently served in IEEE 802.16 mesh networks with a reasonable number of hops, provided the use of an efficient scheduler. Service differentiation and a deadline-based logic are considered essential for supporting real-time traffic. Future plans include the adaptation of the algorithm in orthogonal frequency division multiple access (OFDMA) and scalable-orthogonal frequency division multiple access (S-OFDMA) as well as a theoretical approach of its computation complexity. ■

REFERENCES

[1] *IEEE Standard for Local and Metropolitan Area Networks—Part 16: Air Interface for Fixed Broadband Access Systems*, IEEE Standard 802.16-2004, Oct. 2004.

[2] *Amendment to IEEE Standard for Local and Metropolitan Area Networks—Part 16: Air Interface for Fixed Broadband Wireless Access Systems—Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands*, IEEE Standard 802.16e-2005, Feb. 2006.

[3] K. Wongthavarawat and A. Ganz, "Packet scheduling for QoS support in IEEE 802.16 broadband wireless access systems," *Int. J. Commun. Syst.*, vol. 16, pp. 81–96, 2003.

[4] C.-H. Jiang and T.-C. Tsai, "Token bucket based CAC and packet scheduler for IEEE 802.16 broadband wireless access networks," in *Proc. IEEE 3rd Consum. Commun. Network. Conf. 2006 (CCNC '06)*, Las Vegas, NV, 2006.

[5] H. Shetiya and V. Sharma, "Algorithms for routing and centralized scheduling in IEEE 802.16 mesh networks," in *Proc. IEEE Wireless Commun. Network. Conf. 2006 (WCNC '06)*, Las Vegas, NV, 2006.

[6] F. Niyato and E. Hossain, "Queue-aware uplink bandwidth allocation and rate control for polling service in IEEE 802.16 broadband wireless networks," *IEEE Trans. Mobile Comput.*, vol. 5, no. 6, pp. 668–679, Jun. 2006.

[7] M. Mojdeh, H. S. Alavi, N. Yazdani, and C. Lucas, "A quality of service architecture for IEEE 802.16 standard," in *Proc. Asia Pacific Conf. Commun. (APCC)*, Perth, Australia, Oct. 2005.

[8] J. Chen, W. Jiao, and H. Wang, "A service flow management strategy for IEEE 802.16 broadband wireless access systems in TDD mode," in *Proc. IEEE ICC'05*, Seoul, Korea, May 2005.

[9] H. Wei, S. Ganguly, R. Izmailov, and Z. J. Haas, "Interference-aware IEEE 802.16 WiMAX mesh networks," in *Proc. IEEE VTC'05*, Stockholm, Sweden, May 2005.

[10] J. Chen, W. Jiao, and Q. Guo, "An integrated QoS control architecture for IEEE 802.16 broadband wireless access systems," in *Proc. IEEE Globecom 05*, St. Louis, MO, Nov. 2005.

[11] S. Xergias, N. Passas, and L. Merakos, "Flexible resource allocation in IEEE 802.16 WMAN," in *Proc. 14th IEEE Workshop Local Metropolitan Area Netw. (LANMAN 2005)*, Chania, Greece, Sep. 2005.

[12] M. S. Kuran, B. Yilmaz, F. Alagoz, and T. Tugcu, "Quality of service in mesh mode IEEE 802.16 networks," in *Proc. IEEE Int. Conf. Software, Telecommun. Comput. Netw. (SoftCom 2006)*, Split, Croatia, Sep. 2006.

[13] J. Chen *et al.*, "A multicast mechanism in WiMax mesh network," in *Proc. Asia-Pacific Conf. Commun. (APCC)*, Busan, Korea, Aug. 2006.

[14] M. D. Yacoub, *Foundations of Mobile Radio Engineering*. Boca Raton, FL: CRC Press, 1993.

[15] Transparent end-to-end packet-switched streaming service (PSS), protocols and codecs (release 5), 3GPP TS 26.236 v5.7.0, Jun. 2005.

[16] AMR speech codec; General description (release 5), 3GPP TS 26.071 v5.0.0, Dec. 2002.

ABOUT THE AUTHORS

Spyros Xergias (Student Member, IEEE) received the bachelor's degree from the Computer Engineering Department, University of Patras, Greece, in 2001 and the M.Sc. degree from the Department of Informatics and Telecommunications from the University of Athens, Greece, in 2003, where he currently is pursuing the Ph.D. degree.

He is with the Communications Networks Laboratory, University of Athens. Among his research interests are wireless, mesh, and sensor networks as well as data structures and data bases.

Mr. Xergias is a Student Member of the Technical Chamber of Greece.

Nikos Passas (Member, IEEE) received the diploma degree (honors) from the Department of Computer Engineering, University of Patras, Greece, in 1992 and the Ph.D. degree from the Department of Informatics and Telecommunications, University of Athens, Greece, in 1997.

From 1992 to 1995, he was a Research Engineer with the Greek National Research Center "Demokritos." Since 1995, he has been with the Communication Networks Laboratory, University of Athens, as a sessional Lecturer and Senior Researcher on a number of national and European research projects. He has been a Guest Editor of *Wireless Communications and Mobile Computing Journal*. He has published more than 60 papers in peer-reviewed journals and international conferences and has also published one book and six book chapters. His research interests are in the area of mobile network architectures and protocols. He is particularly interested in QoS for wireless networks, medium access control, and mobility management.

Dr. Passas is a member of the Technical Chamber of Greece. He has been a Guest Editor of IEEE Wireless Communications Magazine and a Technical Program Committee Member of the IEEE Vehicular Technology Conference, IEEE PIMRC, and IEEE Globecom.

Apostolis K. Salkintzis (Senior Member, IEEE) received the diploma degree (honors) and the Ph.D. degree from the Department of Electrical and Computer Engineering, Democritus University of Thrace, Xanthi, Greece.

In 1999, he was a sessional Lecturer with the Department of Electrical and Computer Engineering, University of British Columbia, Canada. From October 1998 to December 1999, he was a Postdoctoral Fellow in the same department. During 1999, he was also a Visiting Fellow with the Advanced Systems Institute of British Columbia, Canada. During 2000, he was with the Institute of Space Applications and Remote Sensing, National Observatory of Athens, Greece. Since 1999, he has been with Motorola, Inc., working on the design and standardization of wireless communication networks, focusing in particular on IMS, GPRS, UMTS, WLANs, and TETRA. He has many pending and received patents and has published more than 60 papers in referred journals and conferences. He is a coauthor and editor of two books in the areas of mobile Internet and mobile multimedia technologies. He is an Editor of the *Journal of Advances in Multimedia*. His primary research activities lie in the areas of wireless communications and mobile networking, and particularly seamless mobility, IP multimedia over mobile networks, and mobile network architectures and protocols. He is an active participant and contributor to 3GPP.

Dr. Salkintzis is a member of the Technical Chamber of Greece. He is an Editor of IEEE Wireless Communications. He has been Lead Guest Editor for a number of special issues of IEEE Wireless Communications and IEEE Communications Magazine. He is Vice Chair of the Quality of Service Interest Group (QoSIG) of the IEEE Multimedia Communications Technical Committee.