[MHSS07] *Parag Mogre, Matthias Hollick, Christian Schwingenschlögl, Ralf Steinmetz*; **QoS Architecture for Efficient Bandwidth Management in the IEEE 802.16 MESH Mode**. In: Yang Xiao: WiMax/MobileFi: Advanced Research and Technology, Auerbach Publications, CRC Press, December 2007. S.

Chapter 1

QoS Architecture for Efficient Bandwidth Management in the IEEE 802.16 Mesh Mode

Parag S. Mogre* and Matthias Hollick* and Ralf Steinmetz*
and Christian Schwingenschlögl†

contact author: parag.mogre@KOM.tu-darmstadt.de

The IEEE 802.16 standard series represents the state-of-the-art in technology for metropolitan area broadband wireless access networks. The *point-to-multipoint* (PMP) mode of IEEE 802.16 has been designed to enable quality of service (QoS) in operator-controlled networks and, thus, is foreseen to complement existing third generation cellular networks. In contrast, the optional *mesh* (MESH) mode of operation in IEEE 802.16 enables the setup of self-organizing wireless multi-hop mesh networks. A distinguishing characteristic of the IEEE 802.16 standard series is its support for QoS at the MAC layer. However, the QoS specifications and mechanisms for the PMP and the MESH mode are not consistent. This article presents a novel QoS architecture as a key enhancement to the IEEE 802.16 MESH mode of operation. The architecture is based on the QoS mechanisms outlined for the PMP mode and, thus, enables a seamless coexistence of the PMP and the MESH mode. In particular, we look at the various options the standard provides and the trade offs involved when implementing QoS support in the 802.16 MESH mode, with a focus on the efficient management of the available bandwidth resources. This article is meant to provide researchers and implementers crucial anchor points for further research.

1.1. Introduction and motivation

The demand for ubiquitous connectivity is the driving force for innovation in the field of wireless networks. To satisfy the differing demands of users a huge variety of wireless network platforms has developed over the years. Fig. 1.1 (a) shows some of the contemporary wireless access technologies.

From the figure, one can see that the IEEE 802.16 standard as published in [1] intends to support metropolitan area networks, rural networks, or enterprise wide networks. Initially these networks are expected to support only static nodes (subscriber stations). The standard IEEE 802.16-2004 [1] specifies two modes of op-

*Technische Universität Darmstadt, Multimedia Communications Lab (KOM), Merckstr. 25, 64283 Darmstadt, Germany.
†Siemens AG, Corporate Technology (CT IC 2), Dept. of Information and Communications, Otto-Hahn-Ring 6, 81730 Munich, Germany.

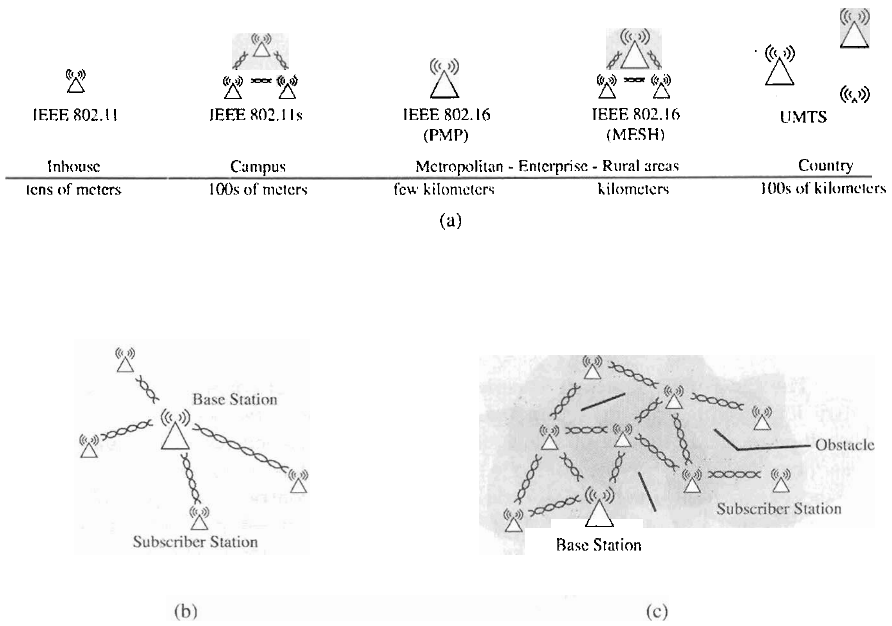

Fig. 1.1. (a) Overview of contemporary wireless access network technologies showing the geographical scale of the wireless technologies. (b) Point-to-multipoint (PMP) mode of operation in 802.16. (c) Mesh mode (MESH) of operation in 802.16.

eration as shown in Fig. 1.1 (b), (c). In the *Point-to-multipoint* mode (PMP) all the subscriber stations (SS) are required to be in direct range of the base station (BS). The SSs can directly communicate only with the BS. Direct communication between two SSs is not supported in the PMP mode. On the other hand, when operating in the MESH mode, the subscriber stations are allowed to establish communication links with neighbouring nodes and are able to communicate with each other directly. In addition, they are also able to send traffic to and receive traffic from the BS (a MESH BS is a SS, which provides backhaul services to the mesh network). The MESH mode of 802.16 allows for flexible growth in the coverage of the mesh network and also increases the robustness of the network due to the provision of multiple alternate paths for communication between nodes. An overview of the 802.16 standard is provided in Ref. [2]. In addition, current efforts in the 802.16e task group have led to the publication of the IEEE 802.16e specification [3]. The latter mainly offers enhancements to the IEEE 802.16-2004 in order to support mobility.

A distinguishing feature of the IEEE 802.16 standard is the extensive support for QoS at the medium access control (MAC) layer. In addition, the IEEE 802.16

standard outlines a set of physical layer (PHY) specifications which can be used with a common MAC layer. This flexibility allows the network to operate in different frequency bands based on the users needs and the corresponding regulations. The QoS support and flexibility at the PHY layer in the 802.16 standard make it an optimal base to support multi-service networks. Thus, 802.16 networks are expected to play a significant role in next generation broadband wireless access (BWA) networks. Such networks cater to the demand for the so called "*Triple Play*" networks, i.e., a single network supporting broadband Internet access, telephony, and television services. They are thus expected to replace conventional DSL based access networks. The needs of each of the above application categories are however varying. For example, applications such as interactive video conferencing, telephony, etc. require predictable response time and a static amount of bandwidth continuously available for the life-time of the connection. On the other hand, traffic like variable rate compressed video streams (e.g. to support television services) relies on accurate timing between the traffic source and destination but does not require a static amount of bandwidth over the duration of the connection. Some other applications such as data transfer using FTP (file transfer protocol) have no inherent reliance on time synchronization between the traffic source and destination. However, these applications benefit when the network attempts to provide a guaranteed bandwidth or latency. Some other services may not be very important from the providers point of view, and traffic belonging to this class may be serviced on space-available basis. This type of traffic has usually no reliance on time synchronization between the traffic source and destination. An example of application generating the latter type of traffic is web surfing. The 802.16 standard defines different data scheduling services to support the above types of traffic; thus, providing tools for network operators to support multi-service networks.

The optional MESH mode of operation specified by the standard allows for organic growth in coverage of the network, with low initial investment in infrastructure. In addition, a mesh inherently provides a robust network due to the possibility of multiple paths for communication between nodes. Thereby, a mesh can help to route data around obstacles or provide coverage to areas which may not be covered using the PMP setup with a similar position for the BS as in the MESH mode. A comprehensive description of the MESH mode can be found in Ref. [4]. A mesh also enables the support of local community networks as well as enterprise wide wireless backbone networks. The above application scenarios make the MESH mode very attractive to network providers, companies, and user communities. This article focuses on efficient management of bandwidth for realization of QoS in the MESH mode. In particular we look shortly at the nuts and bolts involved and the options provided by the standard. Although the IEEE 802.16-2004 standard specifies an extensive set of messages and mechanisms to realize QoS, the algorithms in order to realize QoS and manage bandwidth are left open to foster innovation and provide scope for vendor optimization. This article provides the readers with an overview of the scope for innovation and some critical challenges that need to be addressed in order to obtain a robust and efficient implementation of the IEEE 802.16 MESH mode.

In the next section we provide an overview of the QoS specification for the MESH mode as specified by the 802.16 standard. In particular, we introduce the mechanisms available in the PMP mode and the MESH mode. In future IEEE 802.16 based networks are expected to support seamless interworking of nodes operating in the PMP mode and the MESH mode. The QoS specifications for the two modes however are not consistent. We provide an overview of our proposed QoS architecture for management of the bandwidth in the MESH mode to enable support of the data scheduling services similar to those available in the PMP mode.

Finally, we provide an insight into the benefits and effects of deploying our proposed QoS architecture which we have been able to observe via an intensive simulation study. The simulation study was carried out using a MESH mode simulator we built into the JiST/SWANS [5] environment. We will here also highlight some promising areas for further investigation and outline areas for research which can build up on and extend the QoS architecture described by us.

1.2. Realizing QoS using the 802.16 standard

In this section we first provide an overview of the QoS support mechanisms specified in the standard for the PMP mode followed by an overview of those for the MESH mode. The focus of this article is on the bandwidth management mechanisms required to efficiently support the different classes of traffic. The admission control as well as queueing and priority mechanisms needed to support hard QoS requirements of individual connections are not in the focus of this article.

1.2.1. *QoS support in the 802.16 PMP mode*

The 802.16 MAC is connection oriented. Quality of service is provisioned in the PMP mode on a per-connection basis. All data, either from the SS to the BS or vice versa is transmitted within the context of a connection, identified by the connection identifier (CID) specified in the MAC protocol data unit (PDU). The CID is a 16-bit value that identifies a connection to equivalent peers in the MAC at both the BSs as well as the SSs. It also provides a mapping to a service flow identifier (SFID). The SFID defines the QoS parameters which are associated with a given connection (CID). The SFID is a 32-bit value and is one of the core concepts of the MAC protocol. It provides a mapping to the QoS parameters for a particular data entity.

Fig. 1.2 shows the core objects involved in the QoS architecture as specified in the standard for the PMP mode. As is seen from Fig. 1.2, each MAC PDU is transmitted using a particular CID, which is in turn associated with a single service flow identified by a SFID. Thus, many PDUs may be transmitted within the context of the same service flow but a single MAC PDU is associated with exactly

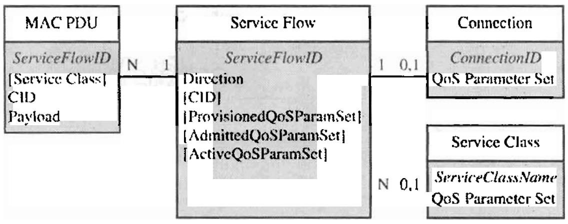

Fig. 1.2. QoS object model [1] for IEEE 802.16-2004 PMP mode.

one service flow. Fig. 1.2 also shows that there are different sets of QoS parameters associated with a given service flow. These are the "*ProvisionedQoSParamSet*", "*AdmittedQoSParamSet*", and "*ActiveQoSParamSet*". The provisioned parameter set is a set of parameters provisioned using means outside the scope of the 802.16 standard, such as with the help of a network management system. The admitted parameter set is a set of QoS parameters for which resources (bandwidth, memory, etc.) are being reserved by the BS (SS). The active parameter set is the set of QoS parameters defining the service actually being provided to the active flow. For example the BS transmits uplink and downlink maps specifying bandwidth allocation for the service flow's active parameter set. Only an active service flow is allowed to transmit packets. To enable the dynamic setup and configuration of service flows, the standard specifies a set of MAC management messages, the so-called dynamic service messages (DSx messages). These are the dynamic service addition (DSA), dynamic service change (DSC), and the dynamic service deletion (DSD) messages. The various QoS parameters associated with a service flow are negotiated using these messages.

Typical service parameters associated with a service flow are traffic priority, minimum reserved rate, tolerated jitter, maximum sustained rate, maximum traffic burst, maximum latency, and scheduling service. The BS may optionally create a service class as shown in Fig 1.2. A service class is a name given to a particular set of QoS parameters, and can be considered as a macro for specifying a set of QoS parameters typically used. The value for the scheduling service parameter in the QoS parameter set specifies the data scheduling service associated with a service flow. The 802.16 standard currently defines the following data scheduling services: *unsolicited grant service* (UGS), *real-time polling service* (rtPS), *non-real-time polling service* (nrtPS), and *best effort* (BE). The UGS is meant to support real-time data streams consisting of fixed-size data packets issued periodically. The rtPS is meant to support data streams having variable-sized data packets issued at periodic intervals. The nrtPS is designed to support delay-tolerant streams of variable-sized data packets for which a minimum data rate is expected. The BE traffic is serviced on a space-available basis. For service flow associated with the scheduling service UGS, the BS allocates a static amount of bandwidth to the SS in every frame. The amount of bandwidth granted by the BS for this type of scheduling

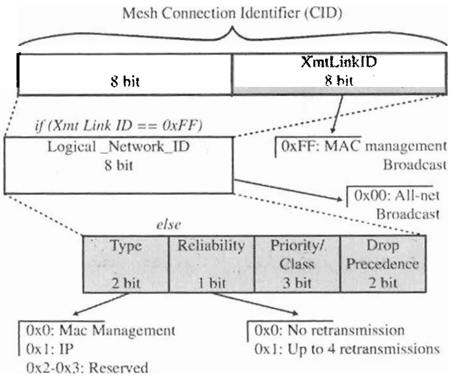

Fig. 1.3. Mesh connection identifier (CID).

service depends on the maximum sustained traffic rate of the service flow. For rtPS service flows, the BS offers real-time, periodic, unicast request opportunities meeting the flow's requirements and allowing the SS to request a grant of the desired size. For nrtPS the BS, similar to the case of a rtPS service flow, offers periodic request opportunities. However, these request opportunities are not real-time, and the SS can also use contention based request opportunities in addition to the unicast request opportunities for a nrtPS service flow as well as the unsolicited data grant types. For a BE service flow no periodic polling opportunities are granted. The SS uses contention request opportunities, unicast request opportunities and unsolicited data grant burst types. A brief overview and evaluation of the QoS support in the PMP mode can be found in Ref. [6].

To summarize, the PMP mode provides the BS with efficient means to manage the bandwidth optimally and at the same time satisfy the requirements of the individual admitted service flows.

1.2.2. *QoS support in the 802.16 MESH mode*

In stark contrast to the PMP mode, the QoS in MESH mode is provisioned on a packet-by-packet basis. Thus, the per-connection QoS provisioning using the DSx messages as introduced previously is not applicable. This design decision helps to reduce the complexity of implementing the MESH mode considerably. However, the MESH mode even with the above simplification is quite complex.

The connection identifier (CID) in the MESH mode is shown in Fig. 1.3. The mesh CID is used to differentiate the forwarding service a PDU should get at each individual node. As can be seen from Fig. 1.3 it is possible to assign a priority to each MAC PDU. Based on the *priority* the transmission scheduler at a node can decide if a particular PDU should be transmitted before another. The field *reliability* specifies the number of retransmissions for the particular MAC PDU (if needed). The *drop precedence* specifies the dropping likelihood for a PDU during congestion. Messages with a higher *drop precedence* are more likely to be dropped. In effect, QoS specification for the MESH mode is limited to specifying the priority of a MAC PDU, the reliability and its drop precedence. Given the same reliability and drop precedence and MAC PDU type (see Fig. 1.3), the MAC will attempt to provide a lower delay to PDUs with higher priority. The above QoS mechanism, however, does not allow the node to estimate the optimal bandwidth requirement for transmissions on a particular link. This is because, (just based on the above interpretation as presented in the 802.16 standard), the node is not able to identify the expected arrival characteristics of the traffic and classify it into the different categories as traffic requiring UGS, rtPS, nrtPS or BE service.

To summarize, QoS mechanisms in the MESH mode are not consistent with those provided for the PMP mode. In addition, the per-packet QoS specification for the MESH mode does not allow a node to optimally estimate the amount of bandwidth required for transmission on a link, as no information about the data scheduling service required for the traffic is included explicitly in the QoS specification in the mesh CID.

We next give an overview of the existing bandwidth request and grant mechanisms specified for the MESH mode of 802.16. This is followed by a description of our proposed QoS architecture, which enables efficient bandwidth management in the MESH mode and allows support of the data scheduling services consistent with those outlined for the PMP mode.

1.3. Frame structure and bandwidth management in the MESH mode

The 802.16 network supports only TDD in the MESH mode [1]. Fig. 1.4 shows the corresponding frame structure. The time axis is divided into frames of a specified length decided by the mesh BS. Each frame is in turn composed of a *control subframe* and a *data subframe*. There are two types of control subframes, namely the *network control subframe* and the *schedule control subframe*. Network control subframes are used to transmit network configuration information as well as to allow new nodes to register and join the network. The schedule control subframe is used by nodes to transmit scheduling information, and to request and grant bandwidth for transmission. All data transmissions take place in the data subframe using slots previously reserved by the node for transmission. The control subframe is divided into a number of *transmission opportunities* and the data subframe is divided into a number of *minislots*. The length of the control subframe depends on the mesh configuration in use. This decides the number of transmission opportunities in the control subframe and the number of minislots in the data subframe. The MESH mode supports coordinated centralized scheduling, and coordinated as well as uncoordinated distributed scheduling for allocating bandwidth for transmis-

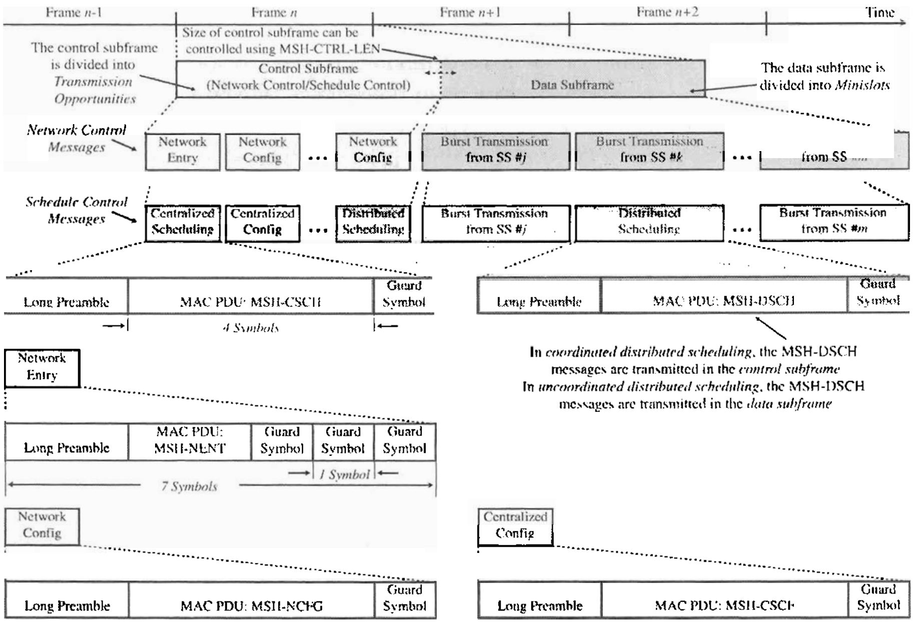

Fig. 1.4. Mesh frame structure.

sion on individual links in the MESH mode of operation. The mesh configuration specifies a maximum percentage of minislots in the data subframe allocated to centralized scheduling. The remainder of the data subframe as well as any minislots not occupied by the current centralized schedule can be used for distributed scheduling.

In centralized scheduling, the bandwidth is managed in a more centralized manner than when using distributed scheduling. Thus, although the computation of the actual transmission schedule is done by the individual nodes independently (in a distributed manner), the grants for each individual node are controlled centrally by the BS in coordinated centralized scheduling (also called centralized scheduling). The BS uses centralized scheduling to manage and allocate bandwidth for transmissions up and down the routing tree (scheduling tree, see Fig. 1.5 for an example) from the BS to the SSs up to a specified maximum hop limit. The routing tree is advertised by the BS periodically using MSH-CSCF messages. The BS in the mesh network gathers resource requests from individual SSs within the maximum hop range. Each SS in the scheduling tree accumulates the requests from its children and adds to it its own requirement for uplink bandwidth before forwarding the request upwards along the scheduling tree (uplink here implies transmission along a link in the scheduling tree from a SS to another SS that is closer to the BS, downlink will be considered to be a transmission down the tree in the opposite direction).

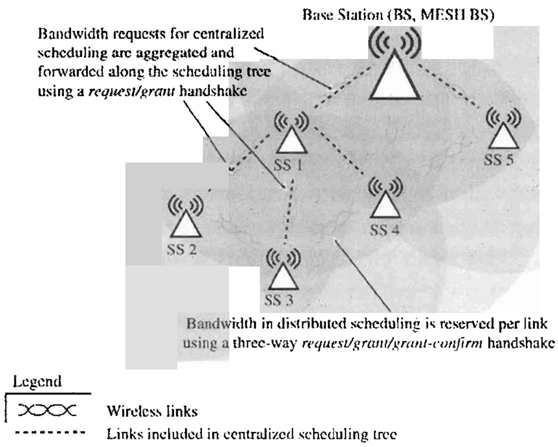

Fig. 1.5. Overview of scheduling in the MESH mode.

The BS collects all the requests and transmits the grants to its children. The grants for each individual SS are then propagated down the scheduling tree hop by hop. Nodes use MSH-CSCH messages to propagate requests and grants for centralized scheduling.

The grants propagated to the SSs in the scheduling tree do not contain the actual schedule. Each SS computes the schedule using a predetermined algorithm and the parameters obtained from the grant. Using centralized scheduling transmissions can be scheduled only along the links in the scheduling tree. To reserve bandwidth for transmission on links not in the scheduling tree distributed scheduling has to be used.

Distributed scheduling is used by a node to reserve bandwidth for transmission on a link to any other neighbouring node (also for links included in the centralized scheduling tree). Nodes use distributed scheduling to coordinate their transmissions in their two-hop neighbourhood. The nodes use a distributed election algorithm to compete for transmission opportunities in the schedule control subframe. A pseudo-random function (the mesh election algorithm specified in the 802.16 standard), with the node IDs of the competitors and the transmission opportunity number as input determines the winning node. The losing nodes compete for the next DSCH transmission opportunity until they win. The parameter *XmtHoldoffExponent* of each node determines the magnitude of transmission opportunities a node has to wait after sending a distributed scheduling message (MSH-DSCH) in a won transmission opportunity. The details as to computation of the hold off period can be found in Ref. [1]. The mean time a node has to wait between two won transmission opportunities for distributed scheduling messages depends on the number of nodes in the two-hop neighbourhood, the node's own *XmtHoldoffExponent*, and the network topology. A detailed analysis of the transmission characteristics of the MSH-DSCH messages in the schedule control subframe is provided in Ref. [7]. The authors in [7] show that the time a node has to wait between two distributed scheduling transmission opportunities it wins increases with an increase in the number of two-hop neighbours and moreover with an increase in the value of the *XmtHoldoffExponent*.

When using coordinated distributed scheduling, the nodes broadcast their individual schedules (available bandwidth resources, bandwidth requests, and bandwidth grants) using transmission opportunities won by the node in the schedule control subframe. The mesh election algorithm ensures that when a node wins a transmission opportunity in the schedule control subframe for transmission, no other node in its two-hop neighbourhood will simultaneously transmit. Thus, it is ensured that the scheduling information transmitted by a node in the schedule control subframe can be received by all of the nodes neighbours. To enable a conflict free schedule to be negotiated each node maintains the status of all individual minislots in the frame. A minislot at any point in time may be either in status *available* (node can receive or transmit data in minislot), *receive available* (node can only receive data in minislot), *transmit available* (node can only transmit data in the minislot), or *unavailable* (node may not transmit or receive data in the minislot).

The schedule negotiated using coordinated distributed scheduling is such that it does not lead to conflict with any of the existing data transmission schedules in the two-hop neighbourhood of the transmitter. On the other hand, nodes can also establish their transmission schedule by directed uncoordinated requests and grants between two nodes. In contrast to coordinated distributed scheduling requests and grants which are sent in the schedule control subframe, the uncoordinated requests and grants are sent in the data subframe. The latter scheduling mechanism is called uncoordinated scheduling. When a node *SS3* wants to reserve slots for transmission to a neighbour node *SS4*, they exchange scheduling information using slots in the data subframe reserved for transmissions between the two nodes (see Fig. 1.5). Nodes individually need to ensure that their scheduled transmissions do not cause collisions with the data as well as with control traffic scheduled by any other node in their two-hop neighbourhood. Transmissions in the data subframe using slots reserved for transmission to a particular neighbour may not be received by all the other neighbours due to other simultaneous transmissions. Thus, the schedule negotiated using the data subframe (uncoordinated scheduling) may not be known to all the neighbours of the nodes involved in the uncoordinated schedule. The neighbours of these nodes may then schedule conflicting transmissions due to lack of the above uncoordinated schedule information. Hence, uncoordinated scheduling may lead to collisions and is not suitable for long term bandwidth reservations. Nodes use MSH-DSCH messages to transmit the bandwidth requests grants and negotiate schedules when using distributed scheduling (both coordinated as well as uncoordinated distributed scheduling).

In contrast, centralized scheduling allows the setup of a transmission schedule for transmissions only along links in the scheduling tree, and hence, is not very suitable for enabling a wireless mesh network in the traditional sense [8]. We next outline our novel proposed QoS architecture for bandwidth management in the MESH mode. Without loss of generality and in order to avoid confusion in the following discussion we will assume that the nodes in the mesh network use only distributed scheduling.

The proposed QoS architecture using distributed scheduling is easily extensible and can be adapted for use in centralized scheduling, too. The proposed architecture uses a combination of coordinated distributed scheduling and uncoordinated distributed scheduling to efficiently manage the bandwidth in the network.

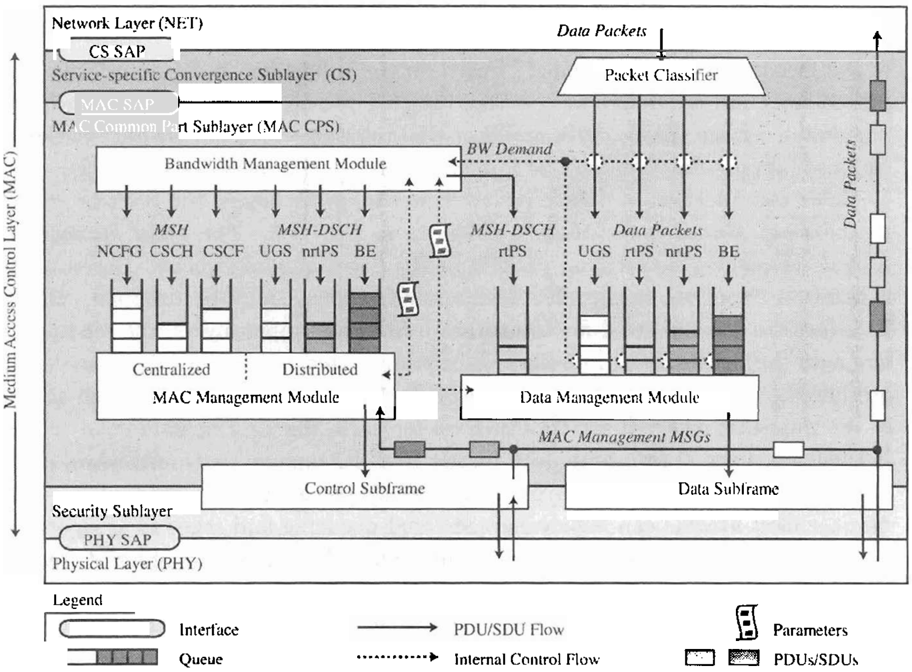

Fig. 1.6. Proposed QoS architecture.

1.4. Proposed QoS architecture for the 802.16 MESH mode

Fig. 1.6 shows our proposed QoS architecture for efficient management of bandwidth in the MESH mode. For the current discussion we assume IP as the network layer protocol. The module *Packet Classifer* shown in the figure provides the functionality of the service-specific convergence sublayer (see scope of the IEEE 802.16 standard [1]). Fig. 1.7 shows the mapping we used to classify traffic from the network layer using the IP TOS field and the corresponding values assigned to fields

Network Layer Priority (e.g., IP Type of Service)	802.16 Service Class	802.16 MSH CID Priority/Class	802.16 MSH CID Drop Precedence	802.16 MSH CID Reliability
0	Best Effort (BE)	0	3	0
1	BE	1	3	1
2	non-realtime Polling Service (nrtPS)	2	2	0
3	nrtPS	3	2	1
4	real-time Polling Service (rtPS)	4	1	0
5	rtPS	5	1	1
6	Unsolicited Grant Service (UGS)	6	0	0
7	UGS	7	0	1

Fig. 1.7. Table showing mapping from the IP type of service (TOS) to the appropriate mesh CID and data scheduling service.

of the mesh CID by our classifier. Based on the values for the fields priority, drop precedence, and reliability we use the mapping shown in Fig. 1.7 to identify the scheduling service (UGS, rtPS, nrtPS or BE) to be provided for the data packet. A similar mapping function may be implemented for other network protocols.

After classification of data received from the upper layers, the packets are sent to the *Data Management Module* as shown in Fig. 1.6. The *Data Management Module* enqueues the arriving packets in the corresponding queue. Based on the congestion situation, it can also decide which packets may be dropped. Besides handling the data received for transmission from the upper layers, the module also manages the MSH-DSCH messages to be transmitted in the data subframe (uncoordinated distributed scheduling). The *Data Management Module* keeps an account of the minislots reserved for transmission for each link to a neighbour at a node. It then sends the appropriate data packet from its queues for transmission on the wireless medium to the lower layer in a minislot reserved for transmission. The *Data Management Module* can deploy sophisticated queueing and scheduling algorithms internally in order to meet the QoS requirements of the different types of traffic in its queues. For the proof-of-concept evaluation of our QoS architecture we used a simple weighted fair queueing (WFQ) scheduler. Our simple scheduler services the MSH-DSCH queue (MAC management messages) with a higher priority than the data queues. Within the data queues the WFQ scheduler serves the UGS, rtPS, nrtPS and BE queues with weights in decreasing order. As previously mentioned, the focus of this article is to provide insights into tools for efficient bandwidth management in the MESH mode and not to verify the satisfaction of hard QoS requirements for each kind of traffic. The *Data Management Module* can use an admission control policy and a QoS scheduling scheme similar to the one outlined in [9] to meet hard per-hop QoS requirements for each kind of traffic. Thus, the *Data Management Module* is responsible for handling all transmissions during the data subframe. In addition this module keeps a running estimate of the incoming data rate in each queue and, based on the policy to be implemented, notifies the *Bandwidth Management Module* of the current bandwidth requirements for each class of traffic.

The *MAC Management Module* shown in Fig. 1.6 is responsible for handling all kinds of MAC management messages. It handles MAC management messages received from the lower layer. If the MAC management message corresponds to a bandwidth request or a grant or grant-confirmation, this module updates the respective internal tables and extracts the relevant parameters (information elements, IEs, contained in the message). These parameters are then sent to the *Bandwidth Management Module* for further processing when required. In addition, it is also responsible for processing MAC management messages received during the network control subframe. This module maintains information about the schedules of the neighbours, the node identifiers of the neighbours, details about the physical two-hop neighbourhood, the Link IDs assigned for transmission to and reception from each neighbouring node. The *MAC Management Module* is responsible for executing the mesh election algorithm specified in the standard to decide if management messages may be transmitted in a given transmission opportunity in the control subframe. We, for our QoS architecture, introduce the concept of traffic classified as belonging to various data scheduling services. We also provide similar means to allow nodes to distinguish the MSH-DSCH and find out the service class to which the requests contained in the MSH-DSCH message correspond. This enables the *Bandwidth Management Module* at the node receiving the MSH-DSCH request to give an appropriate grant based on the expected traffic behaviour. For example, when the requested bandwidth is to serve traffic of class UGS (constant bit rate traffic with time synchronization requirements between sender and receiver), it is better to grant a fixed number of minislots for a longer period of time as the data traffic can be expected to be sent at a constant bit rate for a longer period. The existing MSH-DSCH message structure is shown in Fig. 1.8. To enable a receiver of a MSH-DSCH message to find out which scheduling service the MSH-DSCH corresponds to we propose to use the two reserved bits (see Fig. 1.8) in the MSH-DSCH message to map the MSH-DSCH message to one of the four data scheduling services.

The *Bandwidth Management Module* shown in Fig. 1.6 is responsible for generating bandwidth requests when more bandwidth is required, or generating cancel requests to free bandwidth when it is no longer required. It is also responsible for processing bandwidth requests received from the neighbouring nodes and taking appropriate action when a grant or grant-confirmation is received. All the above request, grants and grant-confirmations are sent as information elements within a MSH-DSCH message as shown in Fig. 1.8. The *Bandwidth Management Module* receives information about instantaneous bandwidth demand from the *Data Management Module.* The *Bandwidth Management Module* maintains internally a set of *MSH-DSCH_Availability_IEs* (see Fig. 1.8). The complete set of *MSH-DSCH_Availability_IEs* describes the local status of individual minislots over all frames in the future. When generating a MSH-DSCH message to request bandwidth for transmission, the *Bandwidth Management Module* creates a *MSH-DSCH_Request_IE* (see Fig. 1.8) describing the amount of minislots required (spec-

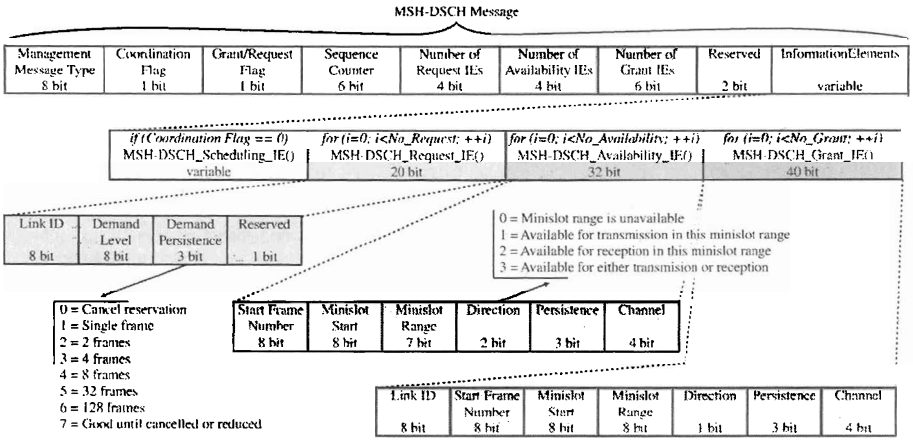

Fig. 1.8. Structure of the MSH-DSCH message and information elements contained in the MSH-DSCH message.

ified by the *demand level* field in the *MSH-DSCH_Request_IE*) in a frame and the number of frames over which the bandwidth is required (denoted by the *demand persistence* field in the *MSH-DSCH_Request_IE*). Due to the classification of traffic into the different scheduling services for the MESH mode as proposed by us, the *Bandwidth Management Module* is able to estimate the arrival characteristics of traffic and make an intelligent choice for the *persistence* value to be sent with the request. As an example, in our proof of concept implementation, the *Bandwidth Management Module* requests minislots with *persistence* 7 (good until cancelled or reduced, see Fig. 1.8) only when the data scheduling service associated with the traffic is UGS. This maps the UGS service provided in the PMP mode where a node receives a constant amount of bandwidth for the lifetime of the connection.

In the PMP mode the rtPS scheduling service is meant to support real-time data streams consisting of variable sized data packets arriving periodically. To support such a service in the MESH mode one requires opportunities for requesting bandwidth in real-time. Using coordinated distributed scheduling a node, however, has to compete with other nodes in its two-hop neighbourhood for transmission opportunities in which a bandwidth request can be sent. Nodes using distributed scheduling need to complete the three-way request/grant/grant-confirm handshake procedure before data can be transmitted using the reserved bandwidth. It is thus not possible to complete the handshake in real-time if we use only coordinated distributed scheduling and the topology is highly connected. To ensure an upper bound on the handshake delay, our QoS architecture proposes to reserve at least a single slot on each link to a neighbour with *persistence* 7 (i.e., the slot is available for transmission all the time). This slot can then be used for transmitting MSH-DSCH messages containing requests and grants for the rtPS service class. This ensures that the handshake completes in the next few frames irrespective of the topology or the value of *XmtHoldoffExponent* (in the best possible case within 4 frames). More details about the dependence of the handshake duration on the topology and the *XmtHoldoffExponent* parameter at the node can be found in [7]. Hence, as can be seen from Fig. 1.6 the *Bandwidth Management Module* sends all MSH-DSCH messages for the rtPS to the *Data Management Module* for transmission. In addition, internally, to ensure a minimum delay, the traffic from the rtPS class can borrow (be transmitted in) bandwidth reserved for UGS traffic. UGS traffic can then borrow bandwidth back from the reserved bandwidth for the rtPS class as soon as the uncoordinated scheduling handshake is over. A characteristic of rtPS is that it has a variable bit rate. Thus, it is highly inefficient to request a fixed amount of slots for transmission for rtPS with *persistence* 7. This may lead to many of these slots being unused in many frames. As a solution, in our proof-of-concept implementation, we used an estimation of the number of slots required per frame to send the arriving rtPS data, and request those slots with a *persistence* 5 (reservation is valid for 32 frames). Using uncoordinated scheduling to reserve bandwidth for a long term is not recommended as it may lead to collisions as explained earlier in this article.

For the nrtPS class we require periodic request opportunities, which need not be in real-time. nrtPS traffic is moreover delay tolerant. Thus we can use an estimator to find out the amount of minislots required per frame and send requests with a *persistence* smaller than 7. As a result, we can periodically (using transmission opportunities in the schedule control subframe) reserve the exact amount of bandwidth required for transmitting nrtPS data. The BE service is very similar to the nrtPS service with the difference that it is served on a space-available basis. Thus, for BE the estimated number of minislots is reserved with a *persistence* less than 7. The difference to nrtPS is that traffic belonging to UGS and rtPS are allowed to borrow bandwidth reserved for BE traffic.

Every request has to be accompanied by a set of *MSH-DSCH_Availability_IEs* as shown in Fig. 1.8. A maximum of 16 *MSH-DSCH_Availability_IEs* may be transmitted with the request. This set of *MSH-DSCH_Availability_IEs* notifies the receiver of the request of the minislot range within which the bandwidth is to be granted. Thus, a poor choice of the set of *MSH-DSCH_Availability_IEs* to transmit with the request will lead to a failure of the request. In our proof-of-concept implementation outlined in this article we first select a subset of *MSH-DSCH_Availability_IEs* at the node which are just able to satisfy the request. Then a set of 16 of the above *MSH-DSCH_Availability_IEs* is selected randomly to be sent with the request. To understand what we mean by a *MSH-DSCH_Availability_IE* just satisfying the request consider the following example. Let us assume that we need a single slot for all future frames, then all availability information elements with *persistence* less than 7 are not able to satisfy this request. Now consider *MSH-DSCH_Availability_IEs* (see Fig. 1.8), all having one minislot and *persistence* 7, however a different value for the *direction* field (see Fig. 1.8). It should be clear that transmission is not possible in minislots with *direction* 0 (unavailable) or 2 (available for reception only). Thus, *MSH-DSCH_Availability_IEs* having *direction* 0 or 2 will not be able to satisfy the request at the sender and should not be sent along with the request. The *MSH-DSCH_Availability_IEs* with *direction* 1 and 3 from the above will be able to satisfy the request and may be sent along with the request. A poor choice may not only lead to a failure of the handshake but also result in less slots with status 3 (available for both transmission and reception) and 1 (transmit available) remaining at the nodes in the network.

On receiving a request, the *Bandwidth Management Module* is also responsible for processing the request to find a mutually suitable set of slots for a grant which is able to satisfy the request. The internal structure of a grant information element (MSH-DSCH_Grant_IE) can be seen in Fig. 1.8. A poor choice for the grant would be for example a grant starting at a frame before the three-way handshake can be completed, this means that the slots in that range will remain unused (data transmission using the granted slots may not start till the three-way handshake is complete as required by the standard). On the other hand, if the grant starts from a frame much in the future after completion of the three-way handshake it leads to additional delay before transmission can start. We, in our proof-of-concept implementation selected grants which would start at least 4 frames in the future after reception of the request.

A three-way handshake (request/grant/grant-confirmation) may fail after the grant has been sent. Nodes in the neighbourhood of the node sending the grant update the status of the minislot range being granted as being in use. Thus, these slots are no longer available for transmission at the nodes receiving the grant. If the grant was sent with *persistence* 7 (good until cancelled) these slots will not be available for transmission for all frames in the future at the nodes which received the grant. When the handshake now fails, the grant-confirmation will not be sent, and hence the slots will never be used for data transmission. Despite the fact that the slots will not be used, the IEEE 802.16 standard currently lacks a mechanism to indicate that the grant sent previously has become invalid (due to failure of the handshake). Thus, these slots are "lost" forever. To avoid the above phenomenon one can either use a soft-state reservation mechanism or introduce an explicit revoke of the grant. We, for our architecture propose to modify the *MSH-DSCH_Grant_IE* to include a revoke bit. When a grant-confirmation (for a grant with *persistence* 7) is not received within a specified time-out, the node which sent the grant sends a copy of the grant with the revoke bit set (we call it the grant-revoke message). This enables the *Bandwidth Management Module* at nodes receiving the grant-revoke to take appropriate action and update the status of *MSH-DSCH_Availability_IEs* stored locally. No grant-revoke-confirmation is sent as the grant-confirmation was not sent either.

The *Bandwidth Management Module* is also responsible for maintaining an up to date status of the *MSH-DSCH_Availability_IEs* stored locally at a node. This involves updating the status when receiving or transmitting either a grant or grant-confirmation. The exact details about each of the above algorithms are out of scope of the current article and hence have not been presented in favour of keeping the article easily accessible and understandable.

Thus, as seen from Fig. 1.6, the *Bandwidth Management Module*, *Data Management Module*, and the *MAC Management Module* comprise the MAC common part sublayer (see Ref. [1]) in our QoS architecture. In Fig. 1.6, arrows passing through the boxes labeled "Control Subframe" and "Data Subframe" represent transmissions/receptions in the control and data subframes respectively.

1.5. Conclusion and directions for future research

To test our proposed QoS architecture we implemented a standard-conform version of the distributed scheduler of the IEEE 802.16-2004 MESH mode using the JiST/SWANS [5] simulation environment and integrated our QoS architecture in the above simulation environment. In the current section we will highlight the key findings of the extensive simulation study we carried out using the implemented 802.16 MESH simulator. One of the key features of our QoS architecture is that it adapts the bandwidth requests and grants keeping in mind the traffic class for the bandwidth requests. In addition to the per hop differentiated handling (QoS) that can be provided to each packet (as specified in the standard for the MESH mode), our QoS architecture allows the network to tailor the bandwidth available at a node per QoS class and link. We expected that this would lead to an optimized usage of bandwidth. At the same time, the QoS model requests bandwidth sufficient to satisfy the QoS requirements (throughput and delay requirements) of the different traffic service classes supported. The QoS model enables the network to support scheduling services similar to those outlined for the PMP mode, namely, UGS, rtPS, nrtPS and BE.

Through our simulation study we observed the following advantages of the proposed QoS architecture. Bandwidth for UGS flows (mainly constant bit rate type of traffic) is reserved with *persistence* 7 (good until cancelled). This reservation profile is highly suitable for CBR type of traffic which maintains a constant throughput over a period of time. The reservation with *persistence* 7 avoids the need for periodic requests (grants and grant-confirmations as well) for the same constant amount of bandwidth. This leads to more free bandwidth in the scheduling control subframe which can then be used for other purposes. When sufficient bandwidth has been requested for UGS (with *persistence* 7), the QoS architecture is able to guarantee steady delay and jitter characteristics for UGS traffic over each hop. For rtPS, and nrtPS the bandwidth requests are expected to be highly varying over time, so the proposed QoS architecture avoids reserving the estimated bandwidth required for traffic flows belonging to these scheduling services for a longer duration (i.e.

with higher *persistence*). Thus, the proposed QoS architecture sends bandwidth requests for these traffic classes and also grants and grant-confirmations with *persistence* less than 7. In our study we used *persistence* 5 for the above traffic classes. This helps to optimize the bandwidth usage as compared to the case when the bandwidth would be reserved only with *persistence* 7. In addition, an important parameter for rtPS traffic is the delay (for both the transmission of data, as well as that for completing the three-way handshake for bandwidth arbitration). The results we obtained through ours simulations tally with the analysis for the distributed scheduling handshake carried out by the authors in [7]. The time needed for completion of a three-way handshake increases with an increase in the number of competing nodes and with the holdoff time per node. This is criticial in the case of the rtPS handshake. A delay in reserving bandwidth for the rtPS traffic means that it may no longer be possible to satisfy the QoS requirements for that traffic class (in absence of long term, *persistence* 7, reservations which in turn wast bandwidth). To overcome this problem, the QoS architecture uses uncoordinated distributed scheduling to setup bandwidth for rtPs flows. Here, unlike coordinated distributed scheduling, the messages for the three-way handshake are transmitted in the data subframe. For reserving rtPS bandwidth for a link the MAC management messages (request, grant and grant-confirm) are transmitted in minislots already reserved for transmission of data on the links between the two neighbouring nodes connected by the above link. This leads to a guaranteed maximum delay for the three-way handshake when the MAC management messages in the data subframe have a higher priority as compared to the data messages. The short duration of validity of the reservation setup using uncoordinated distributed scheduling ensures that a very small (in most cases a negligible fraction) amount of rtPS messages could not be correctly received (due to a parallel transmission setup in the receiver's neighbourhood via coordinated/uncoordinated distributed scheduling). The bandwidth savings hold for the case of nrtPS data too. For nrtPS data the throughput is important and the handshake delay plays a relatively insignificant role. Hence, our QoS architecture uses the control subframe (coordinated distributed scheduling) for the nrtPS three-way handshake. For BE traffic, our architecture tries to use the remaining unused bandwidth reserved for the other three scheduling services. It also additionally requests a minimal possible number of minislots per frame for the BE traffic with a *persistence* less than 7. We observed via our simulations that this led to a starvation of the BE traffic when a strict priority mechanism was used for the three-way handshake and the scheduling of data. We therefore used a weighted fair queuing approach for scheduling the BE requests and data transmissions.

The additional grant revoke mechanism implemented by us helps to recover bandwidth when the three-way handshake fails. We observed a small amount of revokes being sent as compared to the total amount of grants. However, a single revoke message leads to the bandwidth being recovered at all nodes in the neighbourhood of the node transmitting the revoke. This in turn translates into significant bandwidth savings. This also helped to prove that the revoke mechanism functions as expected.

To summarize, good bandwidth management algorithms are crucial to the robust and efficient working of the MESH mode of IEEE 802.16. We presented a novel scheme for managing bandwidth in the 802.16 MESH mode of operation with an aim to support the data scheduling services similar to those currently supported by the PMP mode. In addition, we presented and introduced a bandwidth revocation mechanism which allows the recovery of bandwidth in case the three-way handshake fails. We also provided detailed insights into the working of the IEEE 802.16 MESH mode. The insights obtained should help researchers and implementors tackle the various challenges mentioned by us in this article. The presented QoS architecture provides a solid and extendable foundation for future work. In particular areas such as fair bandwidth distribution and fragmentation of bandwidth need to be looked into.

References

[1] IEEE Computer Society and IEEE Microwave Theory and Techniques Society. 802.16 IEEE Standard for Local and metropolitan area networks, Part 16: Air Interface for Fixed Broadband Wireless Access Systems. IEEE Std. 802.16-2004 (October, 2004).

[2] C. Eklund, R. Marks, K. Stanwood, and S. Wang, IEEE standard 802.16: a technical overview of the WirelessMAN air interface for broadband wireless access, *IEEE Communications Magazine*. **40**(6), 98–107 (June, 2002).

[3] IEEE Computer Society and IEEE Microwave Theory and Techniques Society. 802.16 IEEE Standard for Local and metropolitan area networks, Part 16: Air Interface for Fixed Broadband Wireless Access Systems, Amendment 2: Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands and Corrigendum 1. IEEE Std 802.16e-2005 (February, 2006).

[4] P. S. Mogre and M. Hollick. The IEEE 802.16-2004 MESH Mode Explained. Technical Report KOM-TR-2006-08, Multimedia Communications Lab, Department of Electrical Engineering and Information Technology, Technische Universität Darmstadt, Germany, (2006). ftp://ftp.kom.tu-darmstadt.de/pub/TR/KOM-TR-2006-08.pdf.

[5] R. Barr. JiST/SWANS user guides. http://jist.ece.cornell.edu/docs.html, (2004).

[6] C. Cicconetti, L. Lenzini, E. Mingozzi, and C. Eklund, Quality of service support in IEEE 802.16 networks, *IEEE Network*. **20**(2), 50–55 (Mar./April, 2006).

[7] M. Cao, W. Ma, Q. Zhang, X. Wang, and W. Zhu. Modelling and performance analysis of the distributed scheduler in IEEE 802.16 mesh mode. In *MobiHoc '05*, pp. 78–89, (2005).

[8] I. F. Akyildiz, X. Wang, and W. Wang, Wireless mesh networks: a survey, *Computer Networks*. **47**, 445–487, (2005).

[9] K. Wongthavarawat and A. Ganz, Packet scheduling for QoS support in IEEE 802.16 broadband wireless access systems, *International Journal of Communication Systems*. **16**, 81–96, (2003).

TECHNISCHE UNIVERSITÄT DARMSTADT

KOM – Multimedia Communications Lab

Home Contact § Legal note Sitemap Search Print page Login

Current location: > TU Darmstadt > ETIT > KOM > Publications

People

Publications

Jobs

Press

Enter search key

Publications (Details)

QoS Architecture for Efficient Bandwidth Management in the IEEE 802.16 MESH Mode

Key: MHSS07

Author: Parag Mogre, Matthias Hollick, Christian Schwingenschlögl, Ralf Steinmetz

Date: December 2007

Kind: In collection

Publisher: Auerbach Publications, CRC Press

Book title: WiMax/MobileFi: Advanced Research and Technology

Editor: Yang Xiao

Note: To appear

ISBN: 9781420043514

If the paper is not available from this page, you might contact the author(s) directly via the "People" section on our KOM Homepage.

[Export this entry to BibTeX]

Pisa

[back]

Leipzig

Verlag Auerbach, CRC Press

Seiten 19

ISBN 9781420043514

Anschläge 56 Seiten