Project	**IEEE 802.16 Broadband Wireless Access Working Group <http://ieee802.org/16>**
Title	***ARQ with reduced overhead***
Date Submitted	**2009-12-29**
Source(s)	Zdenek Becvar, Pavel Mach, Robert Bestak **Czech Technical University in Prague** zdenek.becvar@fel.cvut.cz, machp2@fel.cvut.cz, bestar1@fel.cvut.cz
Re:	**IEEE 802.16m-09/0034r2: IEEE 802.16m System Description Document (SDD)**
Abstract	This design note proposes three variation of automatic repeat request mechanism for a reduction of the management overhead without the negative impact on the delay of packets.
Purpose	For review and adoption into 802.16m
Notice	
Release	
Patent Policy	

ARQ with reduced overhead

Zdenek Becvar, Pavel Mach, Robert Bestak

Czech Technical University in Prague, Faculty of Electrical Engineering,

Department of Telecommunication Engineering

1. Objective

Generally, an error can occur during transmission of data in wireless networks. The data with errors cannot be used for further processing and has to be repaired in some way. Wireless networks commonly use techniques based either on Automatic Repeat reQuest (ARQ) or Forward Error Correction (FEC). ARQ mechanism uses a feedback channel for confirmation of error-free packets delivery or request for packet retransmission. This method can increase a network throughput when the channel conditions are getting worse. Simultaneously, FEC methods allow the increase of throughput for impaired channel quality by the adding of redundant coding information at the transmitter side. ARQ also increases a delay of packets by a time spent for retransmission of error packets. Both methods can be combined to so called Hybrid ARQ (HARQ). All above mentioned methods have implemented a mechanism on MAC (Medium Access Control) layer allowing to control and manage these methods to achieve an optimal performance of a link or/and of a network. The management overhead of those methods reduces the user's throughput, therefore it should be minimized. This proposal is focused on the reduction of ARQ overhead.

2. Problem Statement

The ARQ method is used for retransmission of data received with errors that cannot be repaired by other error correction mechanism such as FEC. This technique assumes a separation of user data into blocks and its transmission to the receiver. The receiver checks the data for errors and reply to transmitter with request on retransmission of packets received with errors. The method of retransmission request is different for variety of wireless networks.

The conventional ARQ mechanism works as follows. Each burst from a user carried in a frame is segmented into PDUs. A PDU consists of several blocks N_{block}. The number of blocks is given by following equation:

$$N_{blocks} = \frac{S_{data}}{S_{ARQ\_block}}$$

where S_{data} is a total size of data in one frame by one user, parameter S_{ARQ_block} represents a block size defined by parameter denoted in the standard as ARQ_Block_Size. This parameter is carried in TLV (Type/Length/Value) section of registration messages (REG-REQ/RSP) exchanged between BS (Base Station) and MS (Mobile Station) during a process of a dynamic service addition or modification. The parameter ARQ_Block_Size can take values from the following range: 16, 32, 64, 128, 256, 512 and 1024 bytes. A sequence of consecutive blocks is transmitted in the MAC PDU. The receiver checks the received data and sends an acknowledgment (ACK) feedback message to the transmitter. The feedback is sent in the subsequent frame after the data transmission. The feedback message contains 8 bit field Message ID and field ARQ_Feedback_Payload. The ARQ payload can be carried either by using standalone ARQ feedback message or by piggybacking the ARQ payload to the user’s data block. The payload is always carried in a single PDU. The ARQ_Feedback_Payload consists of one or more ARQ_Feedback_IE (see Table 1) where IE stands for Information Element.

Table 1. Structure of ARQ_Feedback_IE

Syntax	**Size**	**Notes**
CID	16 bits	Connection ID
Last	1 bit	Identify the last IE in ARQ_Feedback
ACK Type	2 bits	0x0...Selective ACK 0x1...Cumulative ACK 0x2...Cumulative with Selective 0x3...Cumulative with Block Sequence
BSN	11 bits	Block Sequence Number (0...2047)
Number of ACK Map	2 bits	Number of Maps (**M**) = 1,2,3 or 4
Maps	**M** x 16 bits	Selective (16 blocks) or Cumulative maps (2 x 64 blocks / 3 x 16 blocks) Cumulative maps: 1 bit sequence format (2 or 3 blocks), 2/3bits Sequence ACK (ACK/NACK of sequence), (2x6) / (3x4) bits Sequence length

The size of an IE of each ARQ feedback message can be calculated according to equation:

$$Size_{ARQ\_FB\_IE} = 32 + (M \times 16)$$

where M represents a number of maps carried in one ARQ_Feedback_IE (see Table 1). The overall size of whole feedback message is given by the consequent formula:

$$Size_{ARQ\_FB} = 8 + \sum^{N_{IE}} Size_{ARQ\_FB\_IE_{N_{IE}}}$$

where N_{IE} is a number of information elements carried in one ARQ Feedback message and the number 8 (bits) represents the ARQ feedback message overhead (Message ID field). The overhead transmitted in all considered frames (N_{frame}) is equal to the sum of partial overheads over the N_{frame} :

$$OH_{ConvARQ} = \sum^{N_{frame}} Size_{ARQ\_FB_{Nframe}}$$

The principle of conventional ARQ method and the structure of user's information carried in the frame are depicted in Figure 1.

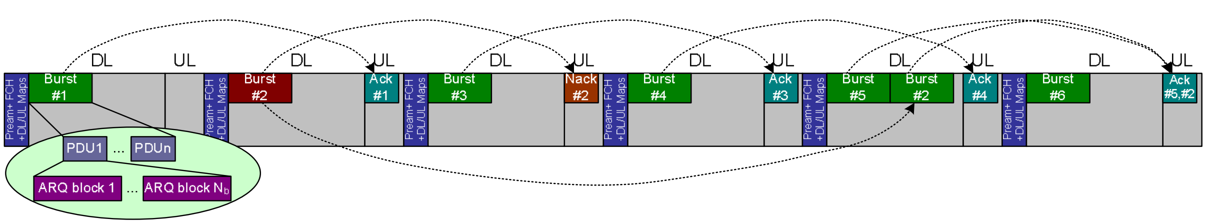

Figure 1. Principle of conventional ARQ

All transmitted blocks have to be confirmed by ACK or by negative ACK (NACK) even if all blocks are received without errors. Therefore, the size of overhead per frame and per user depends especially on the number of blocks transmitted in one frame by one user. The IEEE 802.16e standard defines four types of acknowledgments:

Selective ACK entry, Cumulative ACK entry, Cumulative with Selective ACK entry and Cumulative with Block Sequence ACK entry. The first type of acknowledgment uses selective maps to provide feedback to the transmitter. Each bit set to "1" in the selective map indicates error-free receiving of the corresponding ARQ block. The BSN corresponds to the most significant bit in the map. The second type, Cumulative ACK entry, is based on the utilization of sequence maps. A sequence map defines a group of consecutive blocks where each group includes a sequence of only erroneous blocks or sequence of only error free blocks. The sequence maps can contain two or three sequences with a length of 64 or 16 blocks respectively. The third type of ACK combines the previous two types. Finally, the last type combines the second type with ability to acknowledge ARQ blocks in the form of block sequences.
The retransmission of blocks with errors can be processed earliest in the third block after the first transmission since the transmitter receives NACK in the following frame after transmission (2^{nd} frame) and a request for additional resources can be done earliest at the next frame (3^{rd} frame). So the dedicated resources are available not before 4^{th} frame. The repeated data (#2 in Figure 1) can be transmitted either together with normally ordered data (#5 in Figure 1) or the following data (#5 in Figure 1) can be delayed by one. It causes a delay of retransmitted packets at least for 3 times frame duration.
Both, fresh and retransmitted data are sent in one frame only if the requested capacity can be provided. WiMAX use Stop-and-Wait mechanism which requests a confirmation of the previous block to allow send following blocks. The number of blocks that can be unconfirmed before transmission of the subsequent blocks is defined by parameter ARQ_Window_Size.

3. Proposal of novel ARQ technique

Generally, a number of blocks received with errors increases as decreases the link quality between the transmitter and receiver. Thus, if the BLER increases, the amount of NACK blocks also increases. We can assume that the major part of links with enabled ARQ have satisfactory channel quality to transfer most of blocks without errors (confirmed by ACK) than the number of blocks with errors (confirmed by NACK). In such case, the transmission of ACK blocks appears more often than NACK blocks. If we only consider the transmission of the request for retransmission (NACK), the ARQ overhead can be significantly reduced. The above mentioned assumption is a basis for all following proposals.

3.1 ARQ Scheme I – Only Negative ACK

This scheme assumes the same structure of ARQ_Feedback message and ARQ_Feedback_IEs such as conventional ARQ. However the feedback is sent only if at least one block from PDU is with errors (see Figure 2). If whole PDU is received without errors, no feedback is send. The block is assumed as error free if the transmitter receives no feedback in the following **W** frames (new ARQ parameter ACK_Window is required) after the transmission (W=1 in Figure 2). If the packet is received with errors, ARQ_Feedback message is transmitted by the same way and with the same content (ACK/NACK) as in the conventional ARQ. Hence, each error-free block is confirmed after **W** frames at the latest (if no NACK is received, the packet is considered as it would be confirmed by ACK). Each block with one or more errors is confirmed by the following ARQ_Feedback message not later than after **W** frames. If the feedback with NACK is lost, the data can be retransmitted on the higher layer. Since the probability of packet loss together with belonging NACK feedback is very low the overhead increases due to TCP/IP retransmission instead, not due to MAC layer retransmission which is negligible.

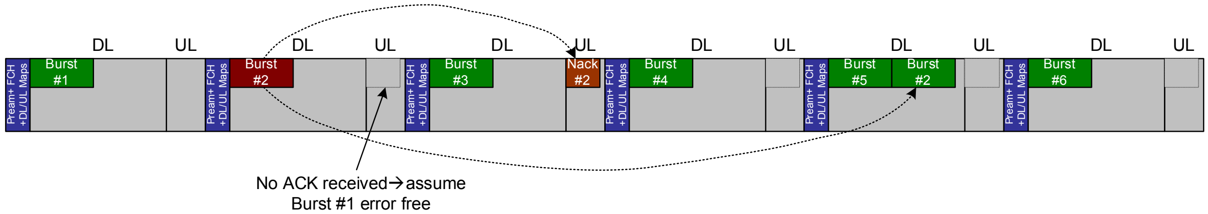

Figure 2. Principle of proposed ARQ scheme I

This proposal needs no modifications in ARQ MAC management messages besides addition of the ARQ TLV parameter ACK_Window. The new parameter is shown in Table 2. This parameter is carried in registration messages (REG-REQ/RSP) and in Dynamic Service messages (DSx-REQ/RSP). These messages are transmit very rarely (only during registration of MS (Mobile Station) to the network and during changes of dynamic service) therefore the increase of overhead is negligible when compared to overhead generated by ARQ acknowledgement.

Table 2. Structure of Acknowledgement Window parameter

Length	**Value**	**Notes**	**Scope**
1 byte	0...64	Number of frames after which is the transmission considered as error free	DSx-REQ/RSP REG-REQ/RSP

The size of ACK_Feedback message can be calculated in the same way as in the conventional ARQ case. The overhead saving is achieved since not all transmissions have to be confirmed by ARQ_Feedback message (at least one block has to be with errors to send an ARQ_Feedback). The overall overhead of proposed ARQ scheme I is a sum of ARQ overhead created in each frame over number of frames. It can be calculated according to the next equation:

$$OH_{SchemeI} = \sum^{N_{frame}} Size_{ARQ\_FB_{Nframe}}$$

3.2 ARQ Scheme II – BSN of blocks with errors

The second version of proposal is based on the same assumptions as the proposed scheme I. The ACK_Feedback is likewise transmitted only if at least one block is received with errors. The block is assumed to be error-free if no feedback is received in **W** frames after transmission of appropriate frame. Both proposals differ among each other in the structure of request for a block retransmission. The proposed scheme II does not use the same structure of ARQ_Feedback_IE. The feedback message carries the set of ACK maps in conventional ARQ. Instead of these maps, the proposed ARQ scheme II carries the set of BSN that corresponds to the blocks received with errors. Therefore, the ARQ_Feedback message contains only one IE and the field “Last” and “ACK Type” can be omitted since the resolution of ACK Type is useless. The format of modified ARQ_Feedback message is shown in Table 3.

Table 3. Structure of modified ARQ_Feedback_IE according to proposal II

Syntax	Size	Notes
CID	16 bits	Connection ID
Number of BSNs	10 bits	Number of BSNs (**B**) = 1...1024
Set of BSN	**B** x 11 bits	Set of Block Sequence Numbers (0...2047)
reserve	0-8	Align a message length to bytes

The maximum number of BSNs can be 1024 since ARQ_Window_Size is maximally a half of maximum value of BSN (2^{11}=2048).
Size of the overhead produced by confirmation of ARQ blocks in the case of ARQ scheme II can be defined by the following formula:

$$\mathrm{Size}_{\mathrm{ARQ\_FB\_II}} = 8 + 26 + (\mathrm{B} \times 11) + \mathrm{res}$$

where *B* is the number of BSNs included in the message and *res* is the number of bits used for an alignment of the message length to bytes. Only one IE filed is always carried in an ARQ feedback message since IE can carry BSNs of all erroneous blocks.
The total overhead due to ARQ scheme II is a sum over the overhead in all frames within the transmission:

$$\mathrm{OH}_{\mathrm{SchemeII}} = \sum^{\mathrm{N_{frame}}} \mathrm{Size}_{\mathrm{ARQ\_FB\_II}_{\mathrm{N_{frame}}}}$$

The ARQ scheme II reduces the overhead especially for low values of BLER. For high value of BLER can be assumed opposite idea – to transmit only confirmation of error free packets (ACK). It will be profitable only for very high level of BLER (over approx. 80%) and it is almost impossible to reach this state in the real networks since the network would be overloaded by retransmitted packets.

3.3 ARQ Scheme III – Combination of ARQ Scheme I & II

The third proposed ARQ scheme combines both previous proposals. This scheme selects the best one from the previous two proposals and conventional ARQ scheme based on the per frame calculation of minimum overhead generated in each frame. The result of this scheme might be the minimal overhead generated by ARQ from conventional ARQ and all proposals.
To achieve the minimal overhead, a new field in ARQ_Feedback_IE has to be introduced. The new field is utilized to selection of the type of ARQ scheme that will be used. The modified structure of message is presented in Table 4.

Table 4. Structure of ARQ_Feedback_IE according to proposal III

Syntax	Size	Notes
CID	16 bits	Connection ID
ARQ Scheme selection	2 bits	0x0…Conventional ARQ 0x1…ARQ Scheme I 0x2…ARQ Scheme II 0x3…Reserve
if ARQ Scheme selection=0x0 or 0x1 {		
Last	1 bit	Identify the last IE in ARQ_Feedback
ACK Type	2 bits	0x0...Selective ACK 0x1...Cumulative ACK 0x2...Cumulative with Selective 0x3...Cumulative with Block Sequence
BSN	11 bits	Block Sequence Number (0...2047)
Number of ACK Map	2 bits	Number of Maps (**M**) = 1,2,3 or4
Maps	**M** x 16 bits	Selective (16 blocks) or Cumulative maps (2 x 64 blocks / 3 x 16 blocks) Cumulative maps: 1 bit sequence format (2 or 3 blocks), 2/3bits Sequence ACK(ACK/NACK of sequence), (2x6) / (3x4) bits Sequence length
}		
if ARQ Scheme selection=0x2 {		
Number of BSNs	10 bits	Number of BSNs (**B**) = 1...1024
Set of BSN	**B** x 11 bits	Set of Block Sequence Numbers (0...2047)
}		
reserve	0-7	Align a message length to bytes

With regards to the above mentioned structure of ARQ_Feedback_IE, the overhead generated by ARQ scheme III by a user in one frame can be calculated according to the following equation:

$$\mathrm{Size}_{\mathrm{ARQ\_FB\_III}} = 8 + 18 + \min\left\{ \sum^{N_{IE}} 16 + 16 \times M_{N_{IE}}, 10 + B \times 11 \right\} + \mathrm{res}$$

where N_{IE} is the number of IEs carried in one ARQ feedback message, M_{NIE} corresponds to the number of ACK maps in ARQ_Feedback_IE, B is the number of BSNs included in one message and *res* is the number of bits used for an alignment of the feedback message length to bytes. The overhead generated by ARQ scheme III is given by the following equation:

$$\mathrm{OH}_{\mathrm{SchemeIII}} = \sum^{N_{frame}} \mathrm{Size}_{\mathrm{ARQ\_FB\_III}_{N_{frame}}}$$

All ARQ schemes are proposed with respect to need no additional hardware modifications of WiMAX equipments (MSs, BSs…) currently available at the market. All changes only implicate the MAC layer software modification.

The results of the simulation of proposed ARQ scheme are presented in [1].

4. Proposed Text

On the IEEE 802.16m-09/0034r2, page 48, append the following text into section 10.4.2

---Start of the Text---

The structure of ARQ_Feedback_IE is presented in Table A. The feedback IE contains the field utilized for selection of the type of ARQ scheme used in particular feedback.

The ARQ Scheme I confirms only not successfully transmitted blocks. The feedback is sent only if at least one block from PDU is with errors. If whole PDU is received without errors, no feedback is sent. The block is assumed as error free if the transmitter receives no feedback in the following ACK_Window frames after the transmission. If the packet is received with errors, ARQ_Feedback message is transmitted by the same way and with the same content (ACK/NACK) as in the conventional ARQ. Hence, each error-free block is confirmed at the latest after ACK_Window frames (if no NACK is received, the packet is considered as it would be confirmed by ACK). Each block with one or more errors is confirmed by the following ARQ_Feedback message not later than after ACK_Window frames. If the feedback with NACK is lost, the data can be retransmitted on the higher layer.

In case of the ARQ Scheme II, the receiver transmits the ACK_Feedback only if at least one block is with errors. The block is assumed as error-free if no feedback is received in ACK_WINDOW frames after transmission of appropriate frame. It differs from Scheme I in the structure of request for a block retransmission. The feedback message carries the set of ACK maps in conventional ARQ. Instead of these maps, the ARQ scheme II carries the set of BSN that corresponds to the blocks received with errors. Therefore, the ARQ_Feedback message contains only one IE.

Table A. Structure of ARQ_Feedback_IE

Syntax	Size	Notes
CID	16 bits	Connection ID
ARQ Scheme selection	2 bits	0x0…Conventional ARQ 0x1…ARQ Scheme I - only negative ACK 0x2…ARQ Scheme II - Block Sequence Number 0x3…Reserve
if ARQ Scheme selection=0x0 or 0x1 {		
Last	1 bit	Identify the last IE in ARQ_Feedback
ACK Type	2 bits	0x0...Selective ACK 0x1...Cumulative ACK 0x2...Cumulative with Selective 0x3...Cumulative with Block Sequence
BSN	11 bits	Block Sequence Number (0...2047)
Number of ACK Map	2 bits	Number of Maps (**M**) = 1,2,3 or4

Maps	**M** x 16 bits	Selective (16 blocks) or Cumulative maps (2 x 64 blocks / 3 x 16 blocks) Cumulative maps: 1 bit sequence format (2 or 3 blocks), 2/3bits Sequence ACK(ACK/NACK of sequence), (2x6) / (3x4) bits Sequence length
}		
if ARQ Scheme selection=0x2 {		
Number of BSNs	10 bits	Number of BSNs (**B**) = 1...1024
Set of BSN	**B** x 11 bits	Set of Block Sequence Numbers (0...2047)
}		
reserve	0-7	Align a message length to bytes

---End of the Text---

On the IEEE 802.16m-09/0034r2, page 48, insert new line to the end of section 10.4.3

--Start of the Text---

ACK_Window: the block is assumed as error free if the transmitter receives no feedback in the following ACK_Window frames after the transmission

---End of the Text---

References

[1] Becvar, Z., Bestak, R., "Overhead of ARQ mechanism in IEEE 802.16 Networks." Submitted to *Telecommunication Systems*. ISSN: 1018-4864.