Project	**IEEE 802.16 Broadband Wireless Access Working Group** <http://ieee802.org/16>	
Title	**HARQ Signaling to Reduce NACK-to-ACK Error Overhead**	
Date Submitted	**2009-01-05**	
Source(s)	Ron Murias, Eldad Zeira InterDigital Communications, LLC	Voice: +1 403 889 1200 E-mail:ron@murias.ca *<http://standards.ieee.org/faqs/affiliationFAQ.html>
Re:	802.16m-08/052 Call for Comments on Project 802.16m System Description Document (SDD).	
Abstract	An additional signal is added to reduce delays caused by NACK-to-ACK errors.	
Purpose	Discuss and adopt into the SDD.	
Notice	*This document does not represent the agreed views of the IEEE 802.16 Working Group or any of its subgroups*. It represents only the views of the participants listed in the “Source(s)” field above. It is offered as a basis for discussion. It is not binding on the contributor(s), who reserve(s) the right to add, amend or withdraw material contained herein.	
Release	The contributor grants a free, irrevocable license to the IEEE to incorporate material contained in this contribution, and any modifications thereof, in the creation of an IEEE Standards publication; to copyright in the IEEE’s name any IEEE Standards publication even though it may include portions of this contribution; and at the IEEE’s sole discretion to permit others to reproduce in whole or in part the resulting IEEE Standards publication. The contributor also acknowledges and accepts that this contribution may be made public by IEEE 802.16.	
Patent Policy	The contributor is familiar with the IEEE-SA Patent Policy and Procedures: <http://standards.ieee.org/guides/bylaws/sect6-7.html#6> and <http://standards.ieee.org/guides/opman/sect6.html#6.3>. Further information is located at <http://standards.ieee.org/board/pat/pat-material.html> and <http://standards.ieee.org/board/pat>.	

HARQ Signaling to Reduce NACK-to-ACK Error Overhead

Ron Murias, Eldad Zeira
InterDigital Communications, LLC

Introduction

As detailed in “HARQ classification and comparison of solutions” spreadsheets, both synchronous and asynchronous solutions are sensitive to ACK-to-NACK and NACK-to-ACK errors. These errors can result in unnecessary PHY level re-transmission or even a loss of state synchronization, ultimately leading to the need for re-transmissions at the MAC level. One way to compensate for this increased “risk” is to design the NACK signaling with a lower probability of error to ensure that no NACK-to-ACK error occurs. This adds unnecessary complexity to the signal design and increases the signaling overhead.

The solution proposed in this contribution manages the logical consequences of NACK-to-ACK errors at the physical layer, which allows the designer to relax the requirements on the probability of a NACK-to-ACK error. Recovery from a single NACK-to-ACK error requires no extra re-transmissions.

Background

Assuming a clean channel, normal HARQ processing begins with the transmission of a HARQ subpacket. Upon reception of the packet, the CRC is verified and an ACK is returned to indicate that the subpacket was successfully decoded. When the HARQ transmitter receives the ACK, a new subpacket is created and sent, along with an indication that the subpacket being transmitted is a new one.

In the event that the original subpacket is not successfully received (but the control signaling was successfully received), the HARQ receiver sends a NACK message back to the transmitter. This results in the HARQ transmitter re-sending the subpacket, along with an indication that the subpacket being transmitted is a re-transmission of the bad packet.

There is a chance that the receiver is unable to correctly decode the transmitted subpacket and indicates a NACK, but the NACK signal is also distorted and the HARQ transmitter falsely decodes the signal as an ACK. In this contribution, we refer to this condition as a NACK-to-ACK error. In this case, the HARQ transmitter creates and transmits a new subpacket, while the receiver is expecting a re-transmission of the previous subpacket. The consequences of this error depend on the ARQ mode and transmission parameters, i.e. the relative size of the subpacket and the ARQ block.

1) In ARQ transparent mode this error is never detected resulting in bad subpacket. If the subpacket is a fraction of a larger ARQ block then the whole ARQ block is corrupt.
2) In ARQ unacknowledged mode the information receiver knows there is an error, but otherwise the results are the same as above.
3) In ARQ acknowledged mode the validity of the information is guaranteed (up to CRC uncertainty) but retransmission is necessary. If the subpacket is all of the ARQ block then the penalty for the retransmission consists of the additional overhead due to higher layer packaging. If the subpacket is only a fraction of the ARQ block then there is an additional penalty in that all of the ARQ block (with many subpackets) must be retransmitted.

The designers of 802.16e have been aware of the above issues and have designed extra robustness into the HARQ feedback mechanism. This extra robustness costs extra overhead.

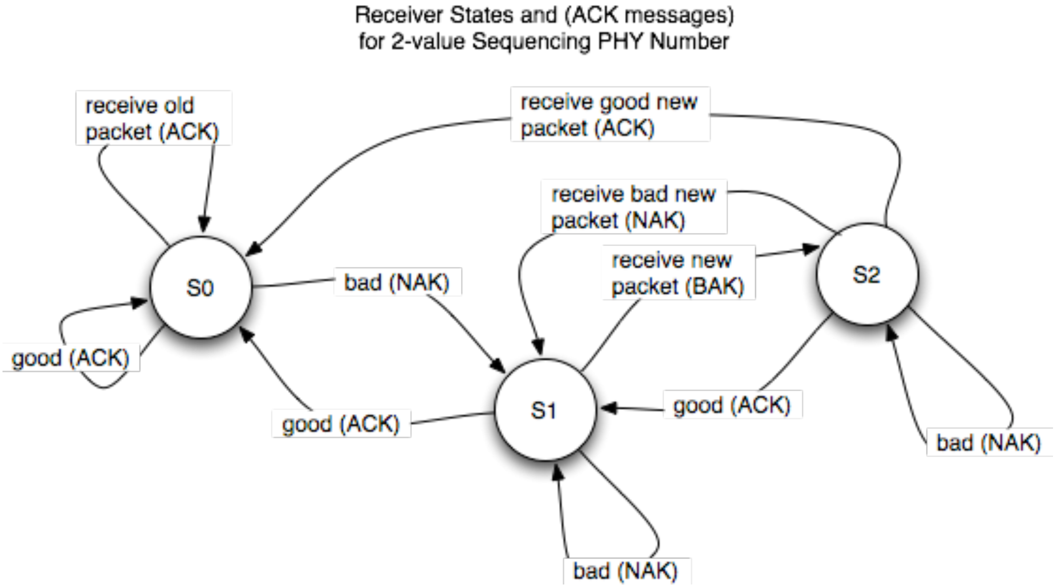
Proposal

One solution to the above case is to allow a third type of acknowledge message: the BACK (bad acknowledge). This signal, sent the same way as the ACK or NACK messages, indicates that the HARQ receiver has detected a transition to a new subpacket when it was expecting a re-transmission of the previous subpacket.

In the following figure, HARQ states are shown:

- S0 is "normal operation", where packets are successfully received.
- S1 is the state where a bad packet has been received and a re-transmission is required.
- S2 is the state where a NACK-to-ACK error has occured. This is indicated when the MS has sent a NACK yet received a new packet. In S2, the MS must re-synchronize its state machine with the BS and retrieve the previous (bad) packet. The MS may retain the unexpected new packet while a BACK is transmitted to re-synchronize the state machines and instruct the BS to re-send the previous packet.
- Note that in the event of a NACK-to-ACK error in the final HARQ retransmission attempt, there is no consequence to this event. Although the mechanism will not detect the NACK-to-ACK error, the ARQ process will detect an error and request re-transmission. Also, the probability of a NACK-to-ACK error on top of a final HARQ retransmission attempt is extremely low.

Receiver States for ACK Sequence



The BS may receive one of three possible HARQ messages. The corresponding transmitter actions are shown in the following table:

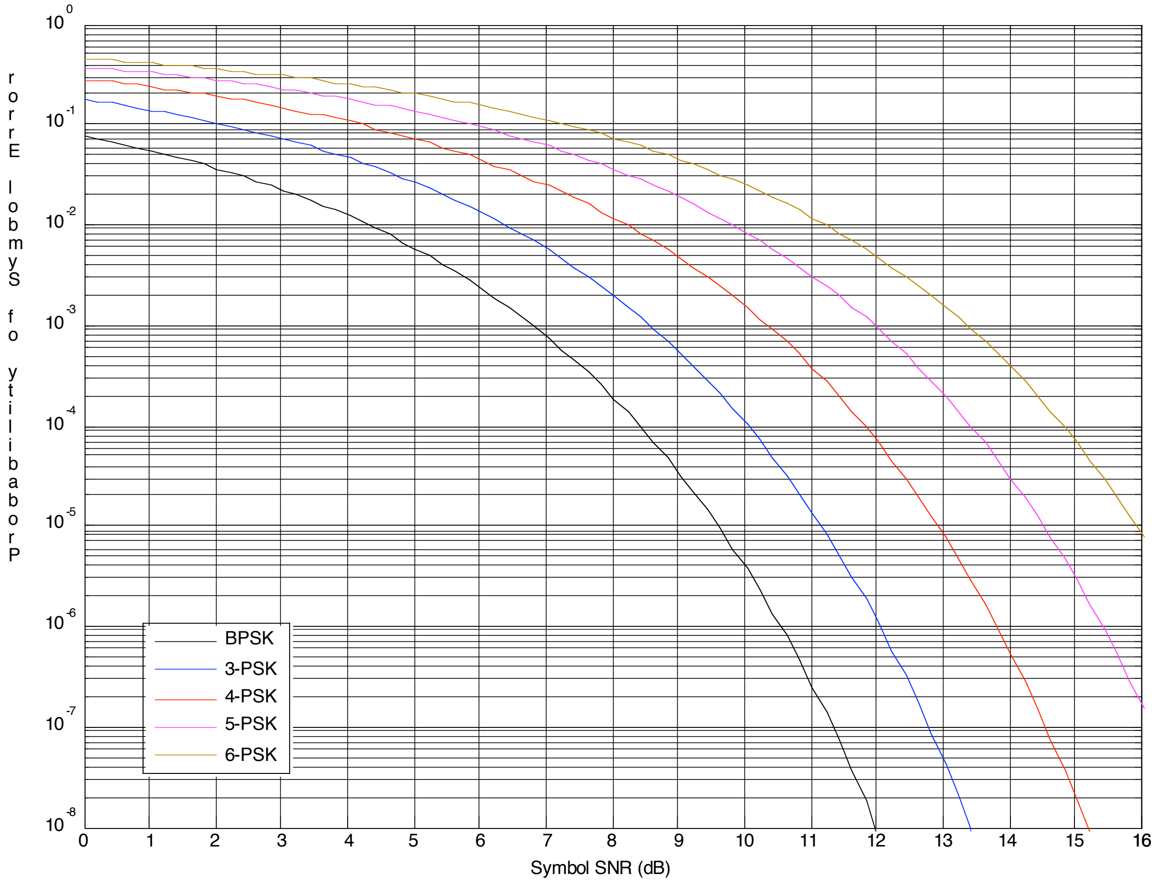
Signal Received	Next Transmission
ACK	n+1
NACK	n
BACK	n-1

The BACK signal may be sent on the uplink ACK channel using a different set of codewords. As the BACK transmission is synchronous, its presence uniquely identifies the error event and no other information needs to be sent.

In the UL, if orthogonal codes are used for HARQ signals, there are many patterns available for the purpose of an additional signal. The pattern may be selected such as to minimize probability of erroneously identifying it as either ACK or NAK. For that purpose it is useful to maximize the Euclidean distance between the BACK and ACK and BACK and NAK respectively. In the downlink, we may use some available bits to signal the additional information to the MS.

Analysis

To understand the effects of the mechanism, we need to equate the error probabilities for the 2 cases:
Without BACK, the HARQ feedback must be designed such that the ACK to NACK error probability is acceptable. For many services e.g. a 10^{-6} error probability is required. Thus we need to find the SNR that is required to signal binary information with an error probability not to exceed 10^{-6}.
With BACK, we need to in effect send ternary information (ACK, NACK or BACK) to correct a single error. Thus the probability of a single ACK to NACK error can be relaxed from e.g. 10^{-6} to e.g. 10^{-3}.



M-PSK Probability of Symbol Error

Based on the above plot, assuming a single error detection, there is a savings of approximately 1-1/2 dB.

Proposed Text

Add the following text to the SDD contribution:

11.9.1.3 HARQ feedback

HARQ feedback (ACK/NACK) is used to acknowledge DL transmissions. Multiple codewords in MIMO transmission can be acknowledged in a single ACK/NACK transmission.

HARQ feedback messages may include a message or signal to indicate that a NACK-to-ACK error has occurred.

Appendex A: Background

Most modern wireless systems today employ ARQ and/or Hybrid-ARQ protocols which requires transmission of acknowledgement from the receiver to the transmitter. The transmission involves the transmission of ACK if the message is received successfully or a NACK if it is not (in which case the transmitter may – but is not required to – retransmit). However, in most cases the ACK/NACK transmission is actually tri-valued: – the lack of transmission in a designated feedback slot is usually also important to the transmitter. Specifically, if the receiver completely misses the transmission, then no ACK/NACK signal is sent at all. We shall refer to this as the NOTX event. On the other hand, if the receiver processes the transmission but this results in errors (typically identified via a CRC check failure) a NACK is transmitted.

The ACK/NACK signaling is almost always structured as follows. Some basic communication resource is allocated for ACK/NACK. Depending on the nature of the air interface this may be a time slot, sub-carrier, code or a combination thereof. The ACK/ NACK is transmitted using some predefined signal (i.e. a constant or some signature), with the predefined signal being modulated with $\sqrt{P}\exp(j\phi_{ACK})$ for ACK and $\sqrt{P}\exp(j\phi_{NACK})$ for NACK. Here P is the power allocated for ACK/NACK and the locations the phase of ACK and NACK signals should be chosen antipodally (i.e. BPSK modulated) to minimize error probability.

Because the signal, when transmitted, is always at a constant power, a simple 2-step detection process suggests itself:

- Step 1: Determine the power of received signal and compare this to some threshold (*T*).
  - If below threshold, declare NOTX and Stop. I
  - If above threshold, proceed to Step 2.
- Step 2: Apply a typical BPSK detector/demodulator to decide whether and ACK or a NACK has been sent.

In analyzing the detection performance here and for the rest of this paper we assume additive Gaussian noise (mean 0, complex variance σ^2), and define $SNR = P/\sigma^2$.

Let $\Pr_{FA}$ and $\Pr_{M}$ denote the errors probabilities associated with Step 1. Specifically, $\Pr_{\mathrm{FA}}$ is the probability of proceeding to step 2 (and thus deciding that an ACK or NACK was sent) when to transmission occurred. $\Pr_{M}$ is the probability of declaring NOTX when a transmission occurred. We note that $\Pr_{FA}$ and $\Pr_{M}$ are completely determined by *SNR* and *T*. In particular, they do not depend on whether ACK or NACK is sent when a transmission does occur.
In addition to $\Pr_{FA}$ and $\Pr_{M}$ we also have the probability of ACK-to-NACK and NACK-to-ACK error, which we denote $\Pr_{NA}$ and $\Pr_{AN}$. Assuming antipodal selection for ACK and NACK phases, these are given by

$$\Pr_{NA} = \Pr_{AN} = \Pr_{E,BPSK}(SNR)\left(1 - \Pr_{M}\right) \qquad (1)$$

where $\Pr_{E,BPSK}$ is the BPSK error probability as a function of SNR.

Note also that within the context of this setup (i.e. if transmitting the same power for ACK or NACK), there is nothing that can be done to make $\Pr_{NA} \neq \Pr_{AN}$, if we do not choose the phases appropriately we can increase the error probabilities, but these will increase together.

1. Extended ACK/NACK transmission using M-PSK modulation.

We now consider transmission of more then 2 (specifically $M \geq 2$) different events in an ACK/NACK fashion. Specifically, we have $M+1$ events: NOTX and events 1 through M are indicated using a transmission of the signal $\sqrt{P}\exp(j\phi_i)$. The meanings assigned to these event (except the NOTX event which we restrict to being no transmission) are irrelevant – so we omit these for the rest of the paper (as an example, this could used for extension of ACK, NACK to allow for some additional signaling).

First we note that this change does not impact $\Pr_{FA}$ and $\Pr_M$ – these still depend only on *SNR* and T - and in exactly the same manner for all values of *M*. The reason for this is that Step 1 depends only on the power of received signal and all signal 1, …, *M*, have the same received signal power distribution.

Therefore, as we compare the performance of this signaling approach for various values of *M* it is sufficient to concentrate on the probability $\Pr_i$ – the probability that the i^{th} symbol is transmitted and is mistaken for some other symbol. This is given by

$$\Pr_i = \Pr_{i,PSK}(SNR)(1 - \Pr_M) \qquad (2)$$

where $\Pr_{i,PSK}$ is the error probability in the PSK modulation scheme selected (i.e. the error probability if the even NOTX never happens).

As we are interested in comparative analysis of schemes for different values of M , we note therefore that:

- It is sufficient to study the PSK scheme selected – i.e. to study $\Pr_{i,PSK}$.
- $\Pr_{i,PSK}$ depends only on SNR and not on T – therefore we ignore Step 1 in the detection process.
- For $M>2$, it is possible to make $\{\Pr_{i,PSK}\}$ unequal – by distributing the modulation points unequally in the angular circle.

We begin by considering the case when all error probabilities $\Pr_{i,PSK}$ are to be kept equal. In this case, we are simply interested in the probability of error for M-PSK modulation, which is given by [1, Sec. 5.2.7]

$$\Pr_M = 1 - \frac{1}{2\pi}\int_{-\pi/M}^{\pi/M} \exp(-\gamma\sin^2\Theta)\left(\int_0^\infty v\exp\left(-\frac{1}{2}\left(v-\sqrt{2\gamma}\cos\Theta\right)^2\right)dv\right)d\Theta \qquad (3)$$

where γ denote SNR.

Figure 1 shows probabilities of symbol error vs. symbol SNR for M-PSK with $M = 2, 3, 4, 5, 6$.

As an example, we provide some possible scenarios. Suppose we start with a M=2 and error probability of 10^{-6} and want to retain the same SNR (~ 10.5 dB), but add another feedback symbol (take it to M=3) while retaining the same error probability for all symbols. In that case the symbol error probability becomes about $4\text{x}10^{-5}$.

If going to 3 symbols allows us to relax the error probability to 10-3, we can save about 2 dB.

If we want to have unequal error probabilities for 3 symbols, we could, for example go to 4-PSK and combine 2 adjacent symbols for the better symbol. Let's say symbol 1 is the combined symbol while symbols 2 and 3 are degraded. Then, the error probability for symbol 1 is about (not quite) the same as that for the 2-PSK, while symbols 2 and 3 have error probabilities that follow the 4-PSK curve. Thus, if we wanted to fix the error probability for symbol 1 at 10^{-6} the degraded symbols would have error probabilities about $8\text{x}10^{-4}$.

References:

[1] J. G. Proakis, *Digital Communications*, 4th Ed., McGraw-Hill, New York, 2001.