Wireless Video Transmission using Feedback-Controlled Adaptive H.264 Source and Channel Coding

M. M. Ghandi, B. Barmada, E. V. Jones, and M. Ghanbari[*†]

Abstract— For wireless video, this paper demonstrates that combining adaptive source and channel coding can provide a robust service over a wide range of transmission channel conditions. Various feedback-controlled adaptive coding solutions are considered for orthogonal frequency division multiplexing (OFDM) based systems. At the receiver, the reliabilities of subchannels are estimated and decisions are fed back to the transmitter to adapt modulation modes and channel code-rates to provide reliable transmission and yet avoid over-protection. The resulting available transmission capacity (which can vary widely with channel conditions) can inform an online collaborating source coder which itself exploits the available feedback to reduce error propagation. To ensure further improvement for fading channels, an efficient low delay retransmission arrangement is also considered in which only the contents of unreliable subchannels within an erroneous packet are retransmitted. Simulation results demonstrate that such collaborative adaptive source and channel coding can provide efficient use of a continuously varying wireless channel.

Index Terms—Source/channel coding, OFDM transmission, H.264/AVC video coding, Turbo Coding

I. INTRODUCTION

There is currently an increasing demand for video services over wireless channels. To achieve the requisite high bit rates and acceptable bit error rate (BER), the limited resources of the channel have to be used in a highly efficient manner. Because of the multipath phenomenon in the wireless environment, broadband single carrier systems in particular suffer from inter-symbol interference (ISI) and so require complex adaptive equalizers. These can be avoided by using multicarrier

---

[*] This project is in part supported by the Engineering and Physical Sciences Research Council (EPSRC) of the UK. The authors are with Department of Computing and Electronic Systems, University of Essex, Colchester, CO4 3SQ, UK. Telephone: +44-1206-872434, Fax +44-1206-872900, e-mails {mahdighandi, bbarmada}@gmail.com, {ed, ghan}@essex.ac.uk.

[†] Corresponding author

modulation methods such as orthogonal frequency division multiplexing (OFDM) [1]. This approach has been adopted in a number of wireless standards including digital video broadcasting (DVB) [2] and IEEE 802.11 [3] and is a strong contender for the fourth generation mobile systems (4G) [4].

However, a broadband multicarrier system operating in a multipath dominant environment is likely to suffer from frequency selective fading whereby each subcarrier can experience a significant range of signal attenuation over time. Thus, some of the individual subchannels of a transmitted packet (coded video units) may suffer more errors than the others. On the other hand, video codecs such as the state of the art H.264 advanced video codec with its high compression efficiency [5] produce a bitstream that is very sensitive to error such that even errors in one subchannel may cause the whole coded video unit to be undecodable and so useless. Such errors can also propagate to future frames that may depend on the current data for their decoding [6]. Therefore, what is ideally required for a successful video service is to provide efficient and immune transmission for all subchannels and at the same time avoid error propagation in the source codec.

For an OFDM system with a fixed modulation mode (i.e. a given signal constellation), a conservative design would consider the worst subchannel and use a modulation mode that gives acceptable immunity for all subchannels. This sacrifices capacity on the subchannels with higher symbol signal-to-noise ratios (SNR). It follows that the overall capacity could be increased by adaptively selecting the modulation modes, such that less capacity is allocated to the degraded subchannels, and more capacity to the higher SNR subchannels. Determining the most appropriate modulation mode for each subchannel is at the heart of such an adaptive system. Several studies assume perfect knowledge of the channel quality at the receiver [7]-[9]. Others use long special symbols inserted in the transmitted packets to estimate the channel quality [10].

In addition to adaptive modulation, channel coding can also be employed [8]-[10] to give an extra protection and extend the performance to the poor channel SNR range that even very low modulation modes cannot serve effectively. In this work we are considering binary turbo coding as a channel coding technique. Turbo coding is an error correction method which can provide performance close to the Shannon limit, outperforming most other error correction methods and is already used in many commercial applications [13][14][15]. As a turbo decoder requires soft decision information, we introduce a simple technique to estimate the amplitude of each received bit relative to the nearest point in the constellation diagram. The code-rate of the channel coder can also be adapted according to channel conditions to provide more flexible overall adaptation. However, the effective capacity of the channel will vary as a result of adapting the channel code-

rate and modulation modes, and so the available rate for the coded source video will be dependent upon the channel conditions. Therefore, an online source coder must track capacity changes with its rate control algorithm. The source coder can also employ feedback information to improve the decoded video quality by avoiding the propagation of errors [11]. Furthermore, many wireless systems use an automatic repeat request (ARQ) mechanism employing feedback between receiver and transmitter to achieve low error reception [12].

In this paper, through simulations of a monocast video over mobile application, a complete video transmission system is analyzed that employs feedback to optimize modulation modes, channel code-rate, and source coding. To determine an appropriate modulation mode for each subchannel and an overall code-rate, the average reliability of the received symbols is estimated using a soft demapping process at the receiver; decisions are then fed back to the transmitter. The decision mechanism provides an estimation of channel quality without imposing an extra overhead on the transmitted packets. To further improve performance, only unreliable subchannels are retransmitted, rather than the whole of an erroneous unit. Since in a fading environment errors tend to occur in bursts, this retransmission method generates less repeated data than the conventional arrangements. However, since on-line video communication is assumed, retransmission is confined within the transmission period of each frame. Thus, the system endeavours to minimize complexity and yet provide comprehensive adaptation according to channel conditions, by adapting: the modulation mode for each subchannel, the channel code-rate for each packet and the bit rate for each source frame. Simulation results show that for both Gaussian and fading channels, such an adaptive solution can provide a graceful degradation of video quality as the channel SNR deteriorates.

The remainder of the paper is organized as follows. Section II outlines some properties of the turbo code and assumptions made for its usage in a combined OFDM system. The proposed adaptive channel coding, modulation and source coding methods are given in Section III. The proposed data rearrangement and retransmission strategy is described in Section IV. Finally, Section V presents simulation results followed by the concluding remarks in Section VI.

II. COMBINING OFDM AND TURBO CODING

A turbo encoder comprises two parallel concatenated recursive systematic convolutional (RSC) encoders and an interleaver between them. The basic coding rate of a turbo encoder is R=1/3, as the output of the encoder for every input bit is a systematic bit and two parity bits. The output of the encoder *is* $(N+M)/R$, where M is the input message length and N is the encoder memory size, where

N tail bits are added to the end of the input message and are used to drive the encoder to its initial state. As the encoder starts at state zero, the tails bits will be all zeros.

Higher coding rates are achieved by puncturing the parity bits. The puncturing is distributed equally and alternately between the parity bits, which means the two encoders are trying to send the same amount of parity. The adapted puncturing patterns are shown in Table 1 for different coding rates.

The design of the interleaver plays an important role in determining the turbo code performance. Besides scattering the error bursts caused by the channel into the whole data, it is also used to break the low weight input sequence, which improves the performance at high SNRs. The interleaver can be a block interleaver, which is the simplest but gives the worst performance. Random and non-uniform interleavers give better performance. Other interleavers such as the s-random interleaver and the code matched interleaver outperform the random one but are more complex and care must be taken in their design in order to avoid low weight sequences [16]. The size of the interleaver plays an important role in improving the performance. The larger the interleaver, the better the performance will be, but at the expense of increasing the latency [17]. In this work, as we compare the performance of different systems with the same turbo code, we do not focus on the interleaver design. Instead for simplicity a random interleaver is used for all the systems (the encoder and the decoder can agree on the same seed for their random generators to produce the same random positions, and hence there is no need to send these positions through the network).

The decoder is comprised of two decoders that are serially concatenated with interleavers between them. Each decoder accepts soft values as input (e.g. for QAM the bit reliability is a real value between -1 and +1 representing binary 0 and binary 1 respectively), and produces soft values as output exchanging information with the other decoder over a number of iterations. The larger the number of iterations, the better the decision will be [18], but at the expense of increasing the latency.

The received bits are passed to the decoder in the form of reliabilities. For multi-state constellations, such as QAM, the bit reliability is calculated according to the positions of their received points in the constellation. Bit reliability will take a real value in the range [*max*, 0] where *max* is the highest reliability value which will be -1 (when the bit represents binary 0) or +1 (when the bit represents binary 1). This technique for obtaining the bit reliability facilitates the use of binary turbo coding with multi-state constellations. Otherwise a non-binary turbo code has to be used which introduces more complexity to the system, especially with the adaptation of the

modulation mode and the coding rate. Extracting bit reliability for square 64-QAM is explained in Appendix A.

A turbo code is likely to suffer from a high error floor, usually between a BER of 10^{-7} and 10^{-6}. The design of the interleaver plays an important role in lowering the error floor, as the performance at high SNRs is affected by the turbo code minimum free distance. If desired, a turbo code can be serially concatenated with other codes, such as Reed-Solomon, to eliminate the error floor completely [19][20]. Error-resilience techniques in the source coding/decoding can also conceal errors.

Fig. 1 shows the basic system diagram and Table 2 details the parameters and assumptions made for the purpose of analysis. Since the main aim of this paper is the adaptation of the system rather than the turbo code itself, more conventional turbo code parameters are chosen. It should be noted that theoretical performance analysis of such a complex system where modulation modes, channel code-rate, allocated capacity, source bit rate, channel condition, frequency selective fading, etc. vary and interact with each other, is not only cumbersome, but can only be achieved through extensive simplification of model parameters and their dependencies. Such simplifications are likely to cast doubts on the validity of any optimization method. Moreover, the complex error sensitivity of a sophisticated video codec such as H.264 is very difficult to deal with analytically. We conclude that a better end-to-end performance analysis can be achieved through simulations provided that the system parameters are realistically chosen. Nevertheless, a Shannon bound based quality comparison is carried out for the video at various bit rates.

In the system studied, the bandwidth can be shared between many users, but in this paper we need only to concentrate on one user that occupies all the subchannels for a certain time slot. In the studied system, 20 packets of size $15\times\sum_{i=1}^{100}M_i$ bits (M_i = 2, 4, or 6) are allocated to the user in each second. Hence, if all the subchannel modulation modes are 4-QAM, the available channel rate would be 60 kbps, 16-QAM can accommodate 120 kbps, and 64-QAM gives 180 kbps. The source data which is an H.264 coded video stream is protected with forward error correction codes before modulation and transmission. It should be noted that in this paper the turbo structures detailed in [21] are employed and the selection of modulation modes and code-rate define the source rate that must be selected by the rate controller in the online H.264 source encoder, as briefly explained in Section III.C.

Fig. 2 shows the average peak signal to noise ratio (PSNR) for a test video sequence as a function of channel symbol SNR for a variety of fixed system modes. The PSNR is defined as

$PSNR = 10\log\left(\frac{255^2}{\varepsilon^2}\right)$ in dB, where ε^2 is the mean squared error difference between the luminance of the non-compressed video frames and their received decoded versions, and averaged over the entire video sequence. We note that although PSNR is not a perfect measure for absolute video quality assessment, it is nevertheless a common objective measure for relative video quality, where improving PSNR also improves the subjective quality [22]. The bound shown in the figure is based on Shannon's equation for channel capacity: ($C/B = \log_2(SNR+1)$). As the maximum modulation mode used in this work is 64-QAM, this will form the upper bound for the system. The 6 bits/symbol of 64-QAM in this bound requires a channel SNR of 18dB, and the maximum PSNR of the video produced by the source encoder in this case is 41dB. When the channel SNR=0dB for example, the channel capacity will be 1 bit/symbol, and the source encoder will produce coded video with PSNR=31.7dB, and so on.

The three bold curves of Fig 2 correspond to the three fixed modulation modes without turbo coding; each representing an individual video bitstream with its appropriate source bit rate. For a given modulation mode, 4-QAM can tolerate a larger variation in SNR, but its maximum picture quality is lower on account of its lower source rate. In contrast, 64-QAM requires a relatively high SNR but yields a better maximum quality in terms of PSNR.

The addition of turbo coding will often yield a higher PSNR, this is apparent from the non-bold curves of Fig. 2 where turbo code-rates of R= 1/3, 1/2, 2/3 or 4/5, are shown for the resulting coded video rates listed in the figure inset. The outstanding performance of the turbo coding over the no code reference plots (bold curves) is very evident. For example, for R = 1/2 combined with 64-QAM (the solid curve with triangular identifier), it can be seen that the system gives its maximum PSNR when channel SNR exceeds about 15 dB compared to 28 dB for the uncoded 64-QAM. However, the price paid for such turbo coding is a 50% redundancy. This reduces the maximum PSNR when turbo coding is used (by nearly 4 dB in this example) but it may be a price worth paying. For example, comparing turbo-coded 64-QAM (R=2/3) with uncoded 16-QAM (both support the same source rate of 120 kbps), the turbo coded arrangement offers a 2 dB gain in channel SNR tolerance.

It follows from Fig. 2 that in order to obtain a useful video service over a wide range of channel SNRs and also improve the quality for larger channel SNR, a combination of adaptive modulation, adaptive code-rate and adaptive source coding rate should be considered. The 'ideal' performance would be obtained by following the best performance envelope of Fig. 2. However, compared with the estimated Shannon bound based upon the current H.264 codec, even this ideal performance falls

short. This is due to such practical constraints as low complexity, low-delay and discrete values of modulation modes and code-rates.

III. ADAPTIVE MODULATION, TURBO CODING, AND SOURCE CODING

A block diagram of the system studied is shown in Fig. 3. The receiver estimates the SNR quality for each subchannel, and so determines the appropriate turbo code-rate for the whole video packet and also the modulation modes for each subchannel. These adaptation control signals along with a report of erroneous video slices are fed back to the transmitter as well as being stored in the receiver for demapping and decoding of the next packet. An error free reception of the feedback control is assumed as the back channel low data rate can be efficiently protected using coding, immune modulation, etc. [10]. Also, rapid feedback reception is assumed as the propagation delay is negligible in the target mobile systems. An adaptation strategy that attempts to avoid exceeding a specified maximum BER bound is adopted, if necessary at the expense of the source data rate. Under especially poor conditions, it may even be necessary to exclude some subchannels until they improve. The source encoder must control the total number of bits in each frame according to the allocated source rate.

A. Adapting the modulation modes

The proposed decision strategy utilizes the perceived reliability of the received points in the constellation diagram by measuring their Euclidean distances from the nearest (noise free) reference constellation point. This information, which relates to the channel SNR is extracted from a soft demapping process without incurring an overhead on the transmitted packets.

In the presence of noise at the receiver a log likelihood function can be used to describe the distance between the received point and the constellation points. In detection using hard decision the reference point that has the maximum (or the one that has the shortest distance from the received point) is considered to be the point that gives the least probability of error and therefore will be selected. This minimum distance between the received point and the nearest reference point is a function of channel noise. The less the noise is, the shorter the distance will be and the more reliable the decision.

Each point in the constellation has an I and Q value, therefore its reliability can be expressed as two measures: one on each axis, given by:

$$R_i = 1 - |x_i - p_i|$$

$$Rq = 1 - |x_q - p_q| \qquad (1)$$

where (x_i, x_q) are the coordinates of the constellation point (point *X* in Fig. 4 and (p_i, p_q) are the coordinates of the received point in the signal constellation. The above equations assume that the maximum reliability is 1 and this is when the received point is placed on the top of its nearest reference point.

The overall reliability of a point is defined as the average reliability of its I and Q reliabilities, and is given by:

$$\mathrm{Rp} = 0.25(1 - \mathrm{Ri})(1 - \mathrm{Rq}) + \frac{\mathrm{Ri} + \mathrm{Rq}}{2} \qquad (2)$$

The first term in equation (2) is the correction factor. According to the equation, if the received point is placed exactly on a constellation point ($R_i = R_q = 1$), then the point reliability is 1. If the received point is positioned on the border corners (positions P2, P4, P6, and P8 in Figure 4), then the point reliability will be 0.25, as these positions are in between four constellation points. If the received point is positioned in the middle between two constellation points (positions P1, P3, P5 and P7 in Figure 4), then the reliability is 0.5.

Substituting equation 1 in 2 gives:

$$\text{Reliability} = 1 - 0.5 \times (|X_I - P_I| + |X_Q - P_Q|) + 0.25 \times |X_I - P_I| \times |X_Q - P_Q|, \qquad (3)$$

For each subcarrier the receiver calculates an average reliability (AR) for the incoming symbol points within a packet. For a given QAM mode, AR can provide an estimation of the subchannel SNR as shown in Fig. 5, this shows AR as a function of SNR for Gaussian noise. Fig. 6 shows, by way of example, how this SNR estimation can be quite accurate in a fading environment. Therefore, based on the known BER/SNR relationships (as e.g. in [23]) a target maximum BER (BER_{max}) will correspond to a specific AR for each modulation mode. Note that in this work, as mentioned earlier, due to complex interactions of the several parameters these relationships are derived through simulation.

In the proposed adaptive system, if a subchannel's AR is less than a certain value (AR1 in Fig. 5) a more immune mode will be chosen for the next transmission. In this case the subchannel's contents will be marked as unreliable and may be retransmitted as explained later. Otherwise if AR is greater than AR1, the subchannel will be marked as reliable but if it exceeds a certain value AR2, it shows that the current subchannel has been overprotected and less immune modes (with more capacity) can be used. AR1 determines the maximum permitted BER ($BER_{max}=10^{-6}$ in Fig. 5 for illustration). Decreasing BER_{max} improves the BER range experienced but reduces the capacity of

the system. To be able to follow the envelope of Fig. 2 without losing protection, the simulation results indicate that BER_{max} should be less than about 10^{-5} (hence in this paper 10^{-6} is used).

B. Adapting the turbo code-rate

For each data packet, the turbo code-rate is fixed across all subchannels, but changes from one packet to another according to the channel conditions. This arrangement ensures an adequate turbo code interleaving length, which is equal to the length of the uncoded packet. Selection of packet length is also influenced by the need to obtain a representative measure of channel quality and yet the system must adequately track changes in channel conditions. In simulations, a packet length of 15 symbols was selected as a realistic compromise. After estimating the SNR in each subchannel according to the AR in Fig. 5, a mean is computed to determine the code-rate for the next packet. To do this, the receiver tries to maximize the source rate with regard to BER_{max} as follows.

First, based on the experimental BER/SNR tables for each code-rate R, it finds the subchannel modulation modes (M_i) that result in an estimated BER lower than BER_{max}. It then calculates the source rate for each R, according to Table 2:

$$SourceRate = 15 \times R \times \sum_{i=1}^{100} M_i \,, \quad R \in \{\frac{1}{3}, \frac{1}{2}, \frac{2}{3}, \frac{4}{5}\} \,. \tag{2}$$

Finally, the receiver chooses the code-rate that offers the highest source rate. This method is seen to perform well relative to alternatives [24]. Since the BER and SNR estimation tables are predeveloped in the design procedure based on the system parameters (Table 2), the above search will not impose any significant processing load.

C. Adapting the source coding

A real-time video transmission system cannot tolerate very large buffering delay and yet the channel rate, and hence the source rate, will vary with the channel conditions. Thus, the source coder must track these rate variations on a frame-by-frame basis in order to avoid any data being discarded. Also, since successive video frames are predicted from each other, as well as controlling the number of bits per frame, the source coder should have a means of preventing errors propagating through future frames [25]. Both these requirements are addressed in the applied Lagrangian optimized H.264 encoder [27].

The H.264 video codec consists of two main layers: the video coding layer (VCL) which provides the core compressed video contents, and the network abstraction layer (NAL) that performs packetization of the coded bitstream. Regarding the bitstream organization, each frame comprises a number of slices; the coded data of each are placed in the payload of one or more NAL units for

transmission over a network. To prevent error propagation an error detection provision with cyclic redundancy checks (CRC) for NAL units of each slice (in a manner described in Section IV) can be incorporated to report the errors from the receiver to the transmitter. Based on the error report, the encoder restricts the prediction modes to intra-modes and or to only those inter-modes that refer to successfully received slices [26]. The selection of the most efficient mode is carried out by a rate distortion optimization in which a fixed Lagrangian multiplier (λ) is selected. Controlling the source rate is also performed at this stage by adjusting the λ value. For further details of this type of rate control, the reader is referred to [27].

By restraining the prediction modes with the above scheme, error propagation can be avoided. However, despite all the efforts of the adaptive transmission, errors in a fading environment in particular could still destroy parts of the individual pictures. Although the destroyed slices undergo an error concealment procedure [28], they could still have a noticeable impact on the average quality. In the next section a data rearrangement and retransmission strategy is presented to limit the impact of such errors on individual frames.

IV. NAL UNIT REARRANGEMENT AND TRANSMISSION STRATEGY

For layered transmission of coded video, H.264 supports data partitioning in which each slice is divided into three partitions in descending order of importance. The first Slice Data Partition is SDP-A which carries the most important data including addressing and motion data, the second partition SDP-B contains intra-residual data which prevent error propagation, and the third partition SDP-C accommodates inter-residual data. In the absence of SDP-A both SDP-B and SDP-C are useless and so more effort should be paid to the successful transmission of SDP-A than to the two other units [29]. In addition, in the system studied error propagation is inhibited so SDP-B and SDP-C units are likely to be of similar importance. So just two levels of importance: SDP-A and SDP-BC (SDP-B + SDP-C) unit types are considered, where each includes a 16-bit CRC, as shown in Fig. 7. The CRC decoding results are included in the feedback information guiding the encoder to prevent error-propagation. In this work, for simplicity, perfect CRC error detection is assumed. However, the incurred overhead of about 2.5% is taken into account and is maintained by fixing the slice sizes (160 bytes) for all frames in all circumstances.

In the experiments, each frame is divided into two packets. SDP-As are accommodated in the first packet on a symbol-by-symbol basis across all subchannels as mandated by the transmission layer. If this first packet has spare capacity, some of the SDP-BCs are also accommodated according to the order depicted in Fig. 7. Conversely, if the first packet cannot accommodate all the SDP-As (which rarely occurs, as they carry video slice headers, such as macroblock addressing, motion

vectors etc, which compared to the DCT coefficients carried by SDP-BC is very small) the overflow is placed at the beginning of the second packet. To ensure that the high priority units (SDP-As) of the first packet have been transmitted successfully some of its contents may be repeated in the second packet as explained in the following.

The basic assumption of the proposed scheme is that in a fading environment different subchannels each conveying a small number of information bits may have different SNR values. In addition, a NAL unit can contain several bytes of data and so may be affected by several subchannels. Intuitively, one can conclude that it is better to only retransmit the erroneous subchannels rather than all the NAL units (SDPs) that have their CRCs violated. However, incorporating such an error recovery policy is not practical for individual subchannels since it would introduce a significant overhead. Instead, for an erroneous NAL unit, the data of those subchannels is retransmitted according to the feedback information that has been received unreliably and hence, has been switched to a higher level of protection. Note that to minimize delay, the retransmitted data is repeated only in the second packet of each frame. Thus, the low priority data do not have such an opportunity for retransmission. It should be mentioned that as some subchannels of the first packet could be included in the second packet, to preserve the turbo-coded structure of the first packet, the code-rate for the repeated data would not be changed.

The contents of the second packet are arranged as follows. For simplicity of discussion, assume the first packet contains only SDP-As. The capacity of the second packet is unequally allocated to three prioritized contents as follows. The first priority is for retransmission of subchannels described above. The second priority is allocated for those SDP-BCs that have the corresponding successfully received SDP-As. This is because in some situations there is not sufficient capacity to transmit all the available data and therefore, the SDPs that are certainly useful will take the earlier priorities. Finally, the rest of SDP-BCs are accommodated in the packet provided there is a spare capacity. Fig. 8 shows an example of the two packets of a coded frame, where there is a possibility that some of the contents of the second packet are discarded. This follows because; the more repeated data that is inserted at the beginning of the second packet, the more low-priority data (normally SDP-BCs) will be discarded. Furthermore, since some of the modulation modes may be readjusted, the capacity of the packet could change. In a real-time application, frame delay should be minimised, so, all information appertaining to a frame is contained within the two packets, and any excess is discarded.

V. SIMULATION RESULTS FOR THE COMPLETE SYSTEM

The adaptive methods described above have been incorporated into the JM 7.1 H.264 reference software so that coded video transmission can be simulated. As this work is especially focused on multimedia in a suburban or urban mobile environment, the multipath fading channel is of particular interest but the Gaussian channel is also investigated to define performance bounds. For the former, the COST207 fading model was adopted with an exponential decaying profile, four paths and an rms delay spread of 1.2546 μs [9]. The channel is assumed to remain unchanged throughout a packet period. Assumed parameters for the H.264 video codec are as follows: Foreman QCIF size video clip at 10 Hz, slice size 160 bytes, and 3 reference frames. Each video sequence has one initial error-free intraframe followed by 33 interframes. In computing the average PSNR, to ensure the requisite confidence, each experiment was repeated until the average value became stable.

Fig. 9(a) shows a frame-by-frame PSNR of a randomly selected experiment of the proposed adaptive system for a mean channel SNR of 30 dB when both error-propagation-prevention and subchannel-retransmission are disabled. Fig. 9(b) shows the same experiment with only error-propagation-prevention and in Fig. 9(c) both error-propagation-prevention and subchannel-retransmission are enabled. It can be seen that in fading environments, where errors can often occur even at relatively high SNRs, error-propagation-prevention on top of the adaptive channel coding and modulation yields a significant improvement in performance. Comparing Fig. 9(b) with Fig. 9(c), indicates that the retransmission scheme has significantly reduced the impact of SDP-A loss at the expense of an increase in the number of SDP-BC losses. Although the total number of affected frames is almost the same for both schemes, the average decoded PSNR will be shown to exhibit a reasonable improvement.

In order to assess the effectiveness of the subchannel retransmission scheme (SCH-RTX), it is compared with two other schemes: (i) a non-retransmission scheme (NON-RTX) without any retransmission, and (ii) a NAL (or SDP) retransmission scheme (NAL-RTX) where the retransmission is performed but in a conventional manner, i.e. the whole of an erroneous NAL unit is repeated. In all three schemes considered, the receiver-to-transmitter feedback is assumed to be available and used for adaptation of the modulation modes and code-rates, as well as for the error-propagation prevention system. Fig. 10 depicts the average SDP-A and SDP-BC loss rates for the three transmission schemes under consideration over a fading channel. It can be seen that both retransmission schemes provide an excellent protection for SDP-A units, and in this regard NAL-RTX is even more successful than SCH-RTX. However, the SDP-BC error rate of SCH-RTX is

better than that of NAL-RTX. The reason is that in NAL-RTX a significant proportion of SDP-BCs have been discarded. It should be noted that as well as affecting the individual frames, nontransmission of NAL units influences the coding efficiency, because the prediction choices in the source coder will be more restricted.

The average PSNR for the three transmission schemes are illustrated in Fig. 11, shown on two scales of low and high channel SNRs for clarity. The figures confirm that both retransmission schemes (SCH-RTX and NAL-RTX) achieve a better video quality (by up to 2 dB) than the non-retransmission method (NON-RTX). Furthermore, it can be seen that subchannel retransmission outperforms NAL retransmission especially at low SNRs by more than 0.5 dB. This improvement is due to its better retransmission strategy, where only the unreliable subchannels are repeated, as detailed earlier. It should also be mentioned that SCH-RTX may be more practical to implement since it involves the data-link-layer rather than NAL-RTX which can be associated with the application layer and so may experience delays in its feedback.

Finally, the overall performance of the proposed adaptive system for a Gaussian channel is shown in Fig. 12 where the BER_{max} on the decision boundary is set to 10^{-6}. Note that the performance of the adaptive system over a Gaussian channel closely follows the fixed envelope of Fig. 2. This means that the adaptive source/modulation/channel coding system outperforms all the individual graphs of Fig. 2 (three of them are depicted for reference) and provides a graceful degradation of video quality over a wide range of channel SNRs. These simulations indicate that for a Gaussian channel, the adaptive system can provide a lower NAL loss rate (less that 0.2%) for all channel SNRs better than about 6 dB. This means that for a Gaussian channel there is no need for retransmission. It can be seen from Fig. 13 that the system performance over a fading channel follows a similar trend to that of the Gaussian, but requires higher values of channel SNR. Also as noted earlier, the NAL loss rates are slightly higher, thus the retransmission scheme provides improved performance and shifts the curve towards that of the Gaussian channel (which is included for comparison). The figure also shows the envelope of fixed (modulation, channel code-rate, and source rate) systems in a fading channel. Two envelopes are depicted: 1) when error propagation is prevented in the source coder, and 2) when there is no feedback at all. It can be seen that even when the feedback is used for error propagation prevention, the fixed systems in fading environments are considerably worse than the adaptive system. The reason is that in frequency-selective fading, some subchannels experience poor quality, even at high Gaussian SNR and hence, some video slices are always likely to suffer from error.

VI. CONCLUSION

This paper has shown that in the presence of a feedback channel, good quality video transmission can be achieved over a wide range of channel SNRs. To achieve this, the channel coder and the modulators need to adapt themselves according to channel conditions. A relatively simple estimate of subchannel reliabilities at the receiver is shown to be sufficient to enable selection of appropriate modulation modes and channel code-rates. The source coder should also adapt to the available channel bit rate, while feedback provides a means of limiting error propagation. A low-delay retransmission scheme is included that repeats only the unreliable subchannels of an erroneous unit; this is shown to outperform the non-retransmission schemes by up to 2 dB in fading environments. If all these adaptations are implemented, results show that they can combine to provide a graceful trade-off between video quality and channel SNR that can offer a near ideal performance over a wide range of possible channel states.

VII. APPENDIX A – BIT RELIABILITY

Bit reliability depends on the amplitude and phase of the received point with respect to the nearest constellation point. Using a Gray coded constellation, at most two bits in the received point could exhibit low reliability. Fig. 14 shows the upper-right quadrant for a 64-QAM constellation with Gray coded distribution. A binary 1 will have positive certain reliability, +1, and binary 0 will have negative certain reliability, -1. Bits marked as "?" have uncertain reliabilities (zero reliability) as they have equal probability of being binary 1 or binary 0. If the received signal is positioned in the middle between four constellation points then two of its bits will have zero reliability (marked as "?"). If it is positioned in the middle between two constellation points then one of its bits will have zero reliability. In other positions, these unreliable bits will have real values between -1 and +1.

The region around the constellation point plays an important role in determining the two bits that have less than certain reliabilities. If we divide the region around the constellation point into four subregions, as in Fig. 4, then in subregion q1, bits number 5 and 6 have their reliabilities less than certain, while in q2, bits number 1 and 6 have uncertain reliability, and so on. This also depends on the position of the point within the signal constellation.

To find a general form, we note from Fig. 14 that all the intermediate-border points (B_i, B_q) (the smaller grey points in the figure), which are in the same row or in the same column, have one of the two uncertain bits in the same position inside the point. For example, if we number the borders on I and Q axes from 1 to 4 as shown in Fig. 14, all the border points in column 2 will have the fifth bit with uncertain reliability, while all the border points in row 2 will have their sixth bit with uncertain

reliability. Table 3 gives the position number of the two uncertain bits associated with the intermediate-border points.

After identifying the nearest constellation point (x_i, x_q) and the subregion of the received point (this means that now we know the positions of the two bits that have uncertain reliability), the phase and the amplitude of the received point are calculated as follows:

$$\theta = \arctan \frac{|x_q - p_q|}{|x_i - p_i|}$$

$$\Delta = \sqrt{(x_i - p_i)^2 + (x_q - x_q)^2} \qquad \text{------(4)}$$

The reliability of the uncertain bit in the received point is given by:

$$b_{p,\,i} = b_{x,\,i} - b_{x,\,i} \times \Delta \times F(\theta) \qquad \text{------(5)}$$

where $b_{p,\,i}$ is the *ith* bit in the received point *p*. $b_{x,\,i}$ is the *ith* bit in the constellation point *x* (has value +1 for binary 1 and value -1 for binary 0). *F(θ)* equals *cos(θ)* (if the uncertain bit is the same for all the border points in the same column), and equals *sin(θ)* (if the uncertain bit is the same for all the border points in the same row). Note that equation (5) is only applied to bits having uncertain reliability. The other bits will have either +1 or -1 values.

The rest of the constellation quadrants can be extracted by mirroring the upper-right quadrant around the appropriate Q and I axis.

REFERENCES

[1] R. Van Nee, R. Prasad, "OFDM for wireless multimedia communications", Artech House, 2000.

[2] ETSI, "Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for digital terrestrial television", EN 300 744, V1.4.1 (2001-01).

[3] R. Van Nee, G. Awater, M. Morikura, H. Takanashi, M. Webster, K. W. Halford, "New High-Rate Wireless LAN Standards", *IEEE Comms. Magazine*, Dec. 1999, pp. 82-88.

[4] T. S. Rappaport, A. Annamalai, R. M. Buehrer, W.H. Tranter, "Wireless communications: past events and a future perspective", *IEEE Communications Magazine*, vol. 40, no. 5, May 2002, pp. 148-61.

[5] ITU-T, "Advanced video coding for generic audiovisual services," *ITU-T Recommendation* H.264, May 2003.

[6] M. Ghanbari, Standard Codecs: Image Compression to Advanced Video Coding. *IEE London, UK*, 2003.

[7] L. Piazzo, "Fast algorithm for power and bit allocation in OFDM systems", *Electronics Letters*, Vol. 53, No. 25, Dec. 1999, pp. 2173-2174.

[8] A. J. Goldsmith, and S. G. Chua, "Adaptive coded modulation for fading channels", *IEEE Transactions on Communications*, Vol. 46, No. 5, May 1998, pp. 590-594.

[9] T. Keller, M. Munster, and L. Hanzo, "A turbo-coded burst-by-burst adaptive wide-band speech transceiver", *IEEE Journal on Selected Area in Communications*, Vol. 18, No. 11, Nov. 2000, pp. 2362-2372.

[10] H. Matsuoka, S. Sampei, N. Morinaga, and Y. Kamio, "Adaptive modulation system with variable coding rate concatenated code for high quality multimedia communication systems", *IEEE Proc. 46th Vehicular Technology conf.* 1996, pp 478-491.

[11] T. Stockhammer, H. Jenkac, and C. Weis, "Feedback and error protection strategies for wireless progressive video transmission," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 6, June 2002, pp. 465-482.

[12] N. Fukui, A. Shibuya, K. Murakami, "Performance of combined ARQ with SR and GBN for broadband wireless systems on a 40GHz band radio channel", *IEEE Communications Magazine*, Sept. 2001, pp. 122-126.

[13] C. Berrou, A. Glavieux, "Near optimum error correcting coding and decoding: Turbo Codes", *IEEE Trans. Commun*. Vol. 44, NO 10, Oct 96, pp. 1261-1271.

[14] C. Berrou, R. Pyndiah, P. Adde, C. Douillard, R. Le Bidan, "An overview of turbo codes and their applications", The European Conference on Wireless Technology, 2005, 3-4 Oct. 2005, pp. 1-9.

[15] E. Guizo, "Closing In On the Perfect Code", IEEE Spectrum, , March 2004, pp.36-42.

[16] Y. Ould-Cheikh-Mouhamedou, S. Crozier, P. Kabal, "Distance Measurement Method For Double Binary Turbo Codes and A New Interleaver Design For DVB-RCS", Proc. IEEE Global Telecommun. Conf. (Dallas, TX), Nov. 2004, pp 172-178.

[17] B. Vucetic, J. Yuan, "Turbo codes: principles and applications", Kluwer Academic Publishers, 2002.

[18] S. Benedetto, G. Montorsi, D. Divsalar, "Concatenated convolutional codes with interleavers", IEEE Communications Magazine, Aug. 2003.

[19] C. Douillard, C. Berrou, “Turbo codes with rate-m/(m+1) constituent convolutional codes”, Communications, IEEE Transactions on Volume 53, Issue 10, Oct. 2005, pp.1630–1638.

[20] Y. Ould-Cheikh-Mouhamedou, S. Crozier, K. Gracie, P. Guinand, P. Kabal, “A Method for Lowering Turbo Code Error Flare using Correction Impulses and Repeated Decoding”, Turbo-Coding-2006 · April 3–7, 2006, Munich.

[21] C. Berrou, “The ten-year-old turbo codes are entering into service,” *IEEE Communications Magazine,* August 2003, pp. 110-116.

[22] A. Ortega, and K. Ramchandran, “Rate-distortion methods for image and video compression,” *IEEE Signal Processing Magazine*, vol. 15, Nov. 1998, pp. 23-50.

[23] K. Cho, and D. Yoon, "On the general BER expression of one- and two-dimensional amplitude modulations," *IEEE T. Commun*., Vol. 50, no. 7, pp. 1074-1080, July 2002.

[24] B. Barmada, and E. V. Jones, “Adaptive modulation and coding for multimedia services”, *3G 2004 Mobile Communication Technologies,* IEE, London, UK, Oct. 2004.

[25] S. Wenger, “H.264/AVC over IP,” *IEEE Trans. Circuits Syst. Video Technol*., vol. 13, July 2003, pp. 645-656.

[26] B. Girod, and N. Farber, “Feedback-based error control for mobile video transmission,” *Proceedings of the IEEE*, vol. 87, No. 10, October 1999, pp. 1707-1723.

[27] M. M. Ghandi, and M. Ghanbari, “A Lagrangian optimized rate control algorithm for the H.264/AVC encoder,” *Proceedings of IEEE International Conference on Image Processing (ICIP)*, October 2004, vol 1, pp. 123-126.

[28] T. Stockhammer, M. M. Hannuksela, and T. Wiegand, “H.264/AVC in wireless environment,” *IEEE Trans. Circuits Syst. Video Technol*., vol. 13, July 2003, pp. 657-673.

[29] M. M. Ghandi, and M. Ghanbari, "Error concealment for SNR scalable video coding," *Elsevier J. Signal Processing: Image Communications*, Vol 21, No 2, February 2006, pp. 91-99.

TABLES

Table **1.** Puncturing patterns for coding rate 1/2, 2/3 and 4/5

1/2	1 1 0 1 0 1
2/3	1 1 0 1 0 0 1 0 1 1 0 0 1 0 0 1 1 0 1 0 0 1 0 1
4/5	1 1 0 1 0 0 1 0 0 1 0 1 1 0 0 1 0 0 1 0 0 1 1 0 1 0 0 1 0 0 1 0 0 1 0 1 1 0 0 1 0 0 1 0 0 1 1 0 1 0 0 1 0 0 1 0 1

Table 2. OFDM and turbo coding parameters used in the study system.

OFDM		Turbo coding	
FFT period	80 μs	Code-rate (R)	1/3, 1/2, 2/3, 4/5
Cyclic prefix	20 μs	Polynomial generator	[5, 7]
Cyclic postfix	6.4 μs	Interleaver type	Non-uniform
Windowing roll-off factor	0.05	Interleaver size	Packet-size×R bits
Subchannels	100	Decoding algorithm	Log-MAP
Bandwidth	1.25 MHz	Iterations	3
Symbols per packet (S)	15		
Synchronization	Assumed perfect		
Modulation modes	4-/16-/64-QAM	(Gray mapping)	
Bits per symbol (M_i, i=1-100)	2 / 4 / 6		
Packet-size	$S\times\sum M_i$ bits		

Table 3. Position number of the uncertain bit in the received signal with respect to the nearest intermediate-border point (B_i , B_q).

B_i (Column No.)	bit number	B_q (Row No.)	bit number
1	1	1	2
2	5	2	6
3	3	3	4
4	5	4	6

FIGURES

Fig. 1. Basic system diagram for combined OFDM and turbo coding.

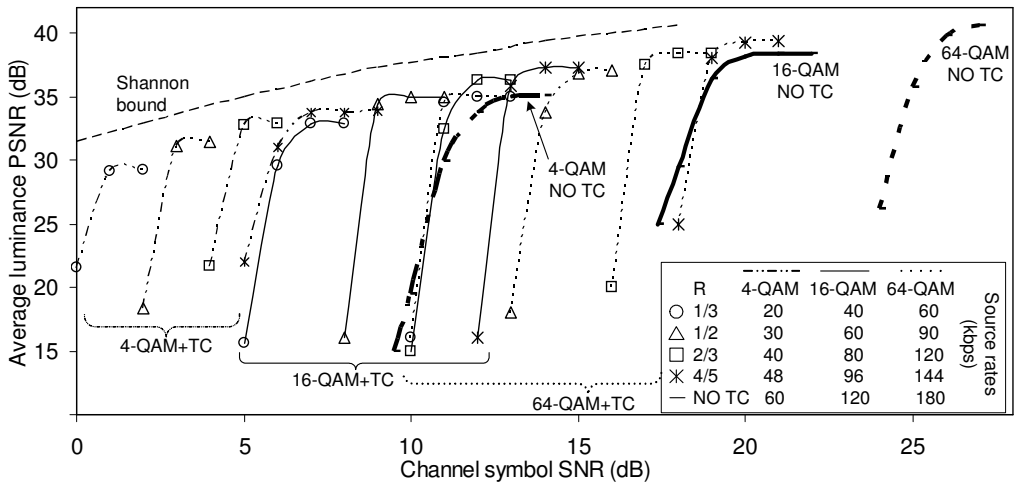


Fig. 2. Luminance PSNR vs. channel symbol SNR for turbo coding (TC) with code-rates: R= 1/3, 1/2, 2/3 and 4/5, combined with 4, 16 and 64-QAM modulations (Foreman QCIF@10Hz H.264 video over a Gaussian channel).

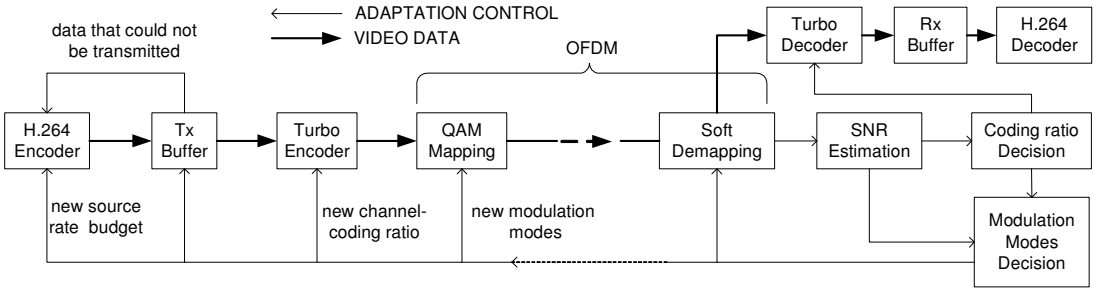


Fig. 3. System diagram for adaptive modulation, adaptive code-rate and adaptive source rate for H.264 video.

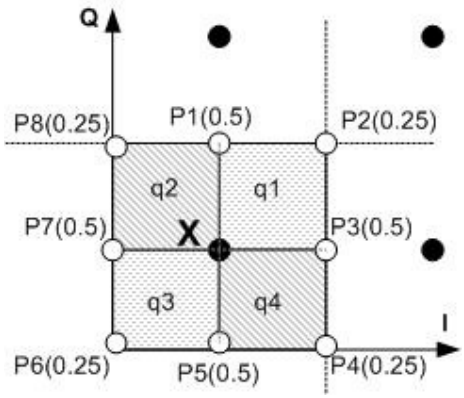


Fig. 4. Possible positions of a received signal around a constellation point and the related point reliabilities. An example is given to the nearest constellation point to (0, 0) coordinates in the up-right quadrant for squared M-QAM.

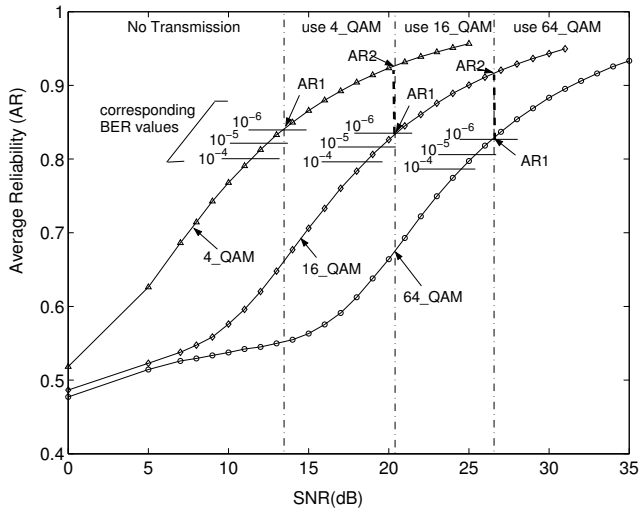


Fig. 5. Determining the modulation mode - average point reliability as a function of SNR (assuming $BER_{max} = 10^{-6}$, no turbo coding).

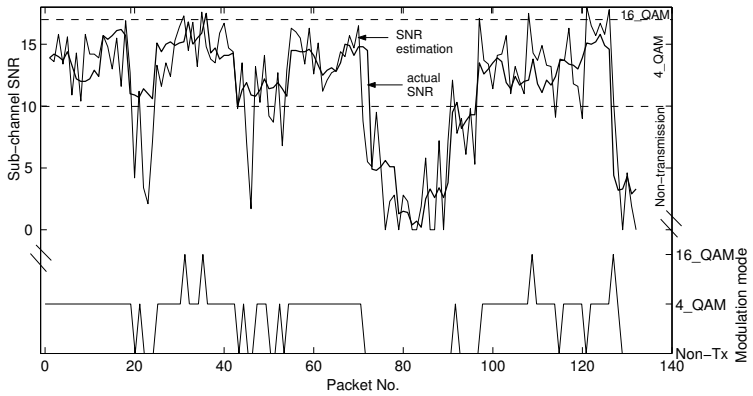


Fig. 6. Subchannel SNR estimation and the corresponding modulation mode. The Gaussian SNR = 16dB, and the fading channel model is COST 207 with four paths. The dash lines are the switch decision borders between 16-QAM, 4-QAM, and non-transmission.

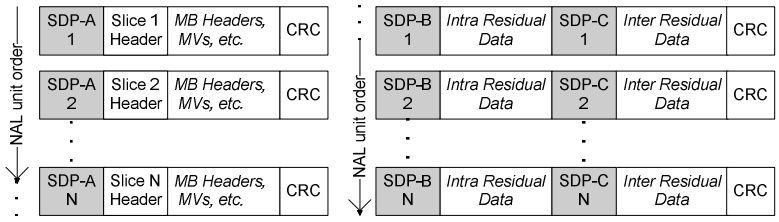


Fig. 7. Placement of NAL units within the transmission packets (showing a complete frame).

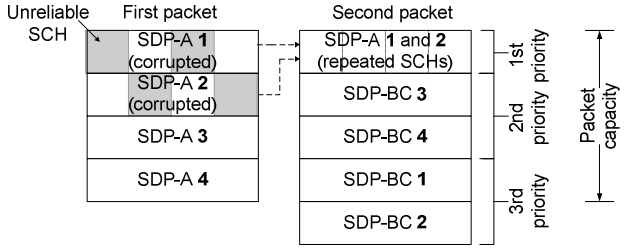
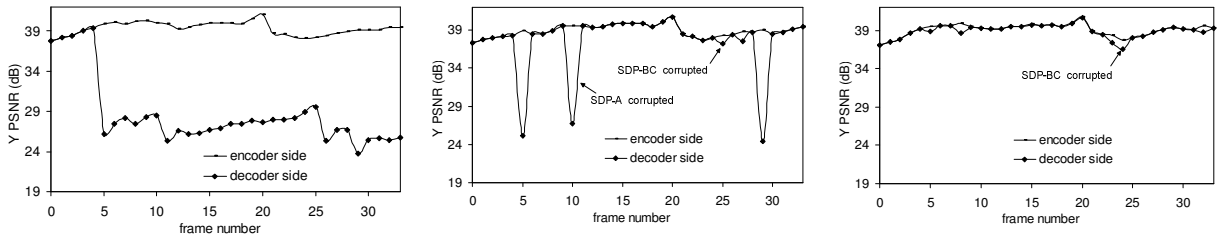
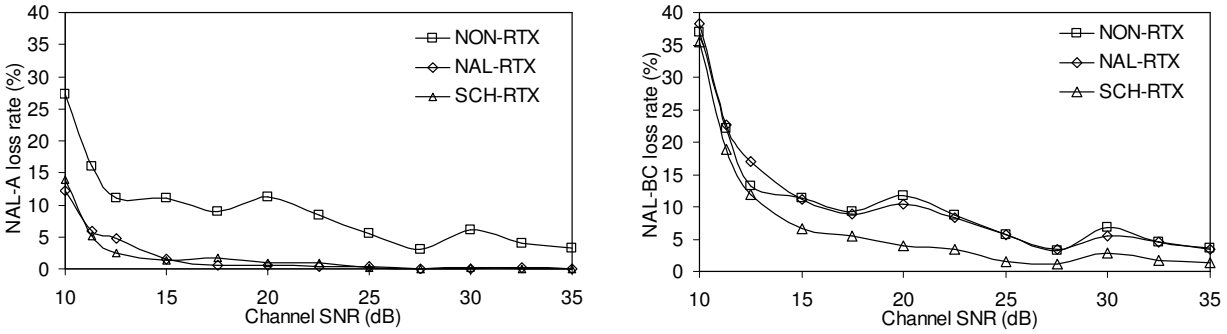


Fig. 8. The two packets of a coded frame with 4 slices; the data from unreliable subchannels (SCH) of the first packet are repeated in the second packet.



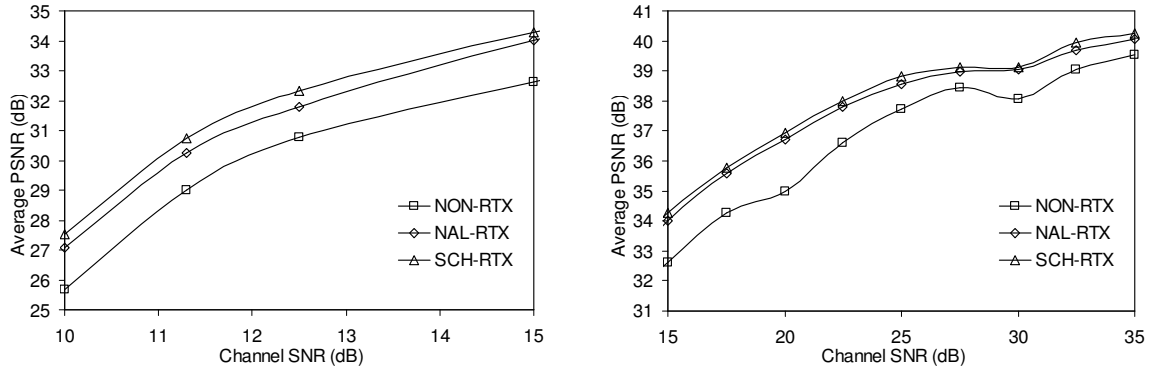
(a) no error-propagation-prevention (EPP) (b) with EPP, no retransmission (NON-RTX) (c) with EPP, with subchannel retransmission (SCH-RTX)

Fig. 9. Frame-by-frame luminance (Y) PSNR, Gaussian SNR = 30 dB in a fading environment.



(a) SDP-A loss rate (b) SDP-BC loss rate

Fig. 10. SDP-A and SDP-BC loss rate percentages for the three transmission schemes.



(a) SNR 10-15 dB (b) SNR 15-35 dB

Fig. 11. Average PSNR vs. channel Gaussian symbol SNR in a fading environment for the three transmission schemes.

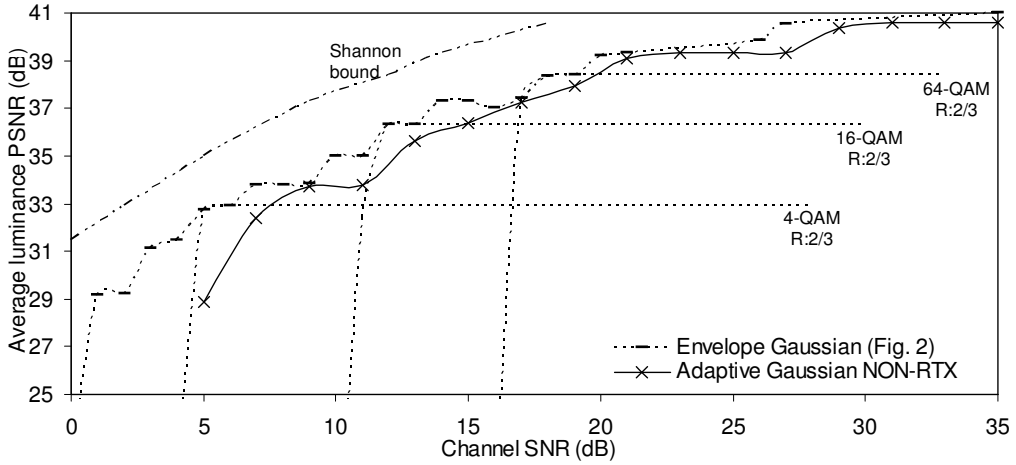


Fig. 12. PSNR performance of the adaptive system for Gaussian channels, the envelope and selected curves from Fig. 2 are shown (dotted) for reference.

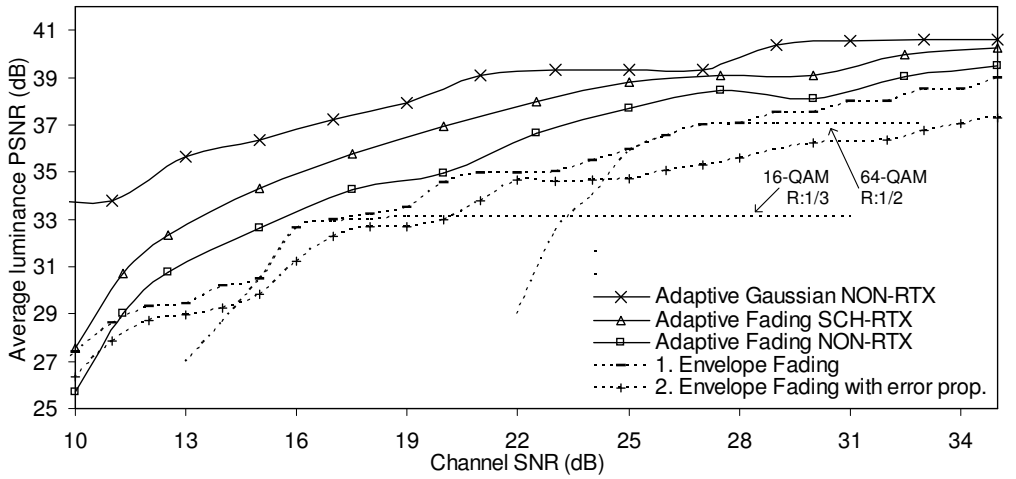


Fig. 13. PSNR performance of the adaptive system for fading channels, fixed system envelopes shown for reference.

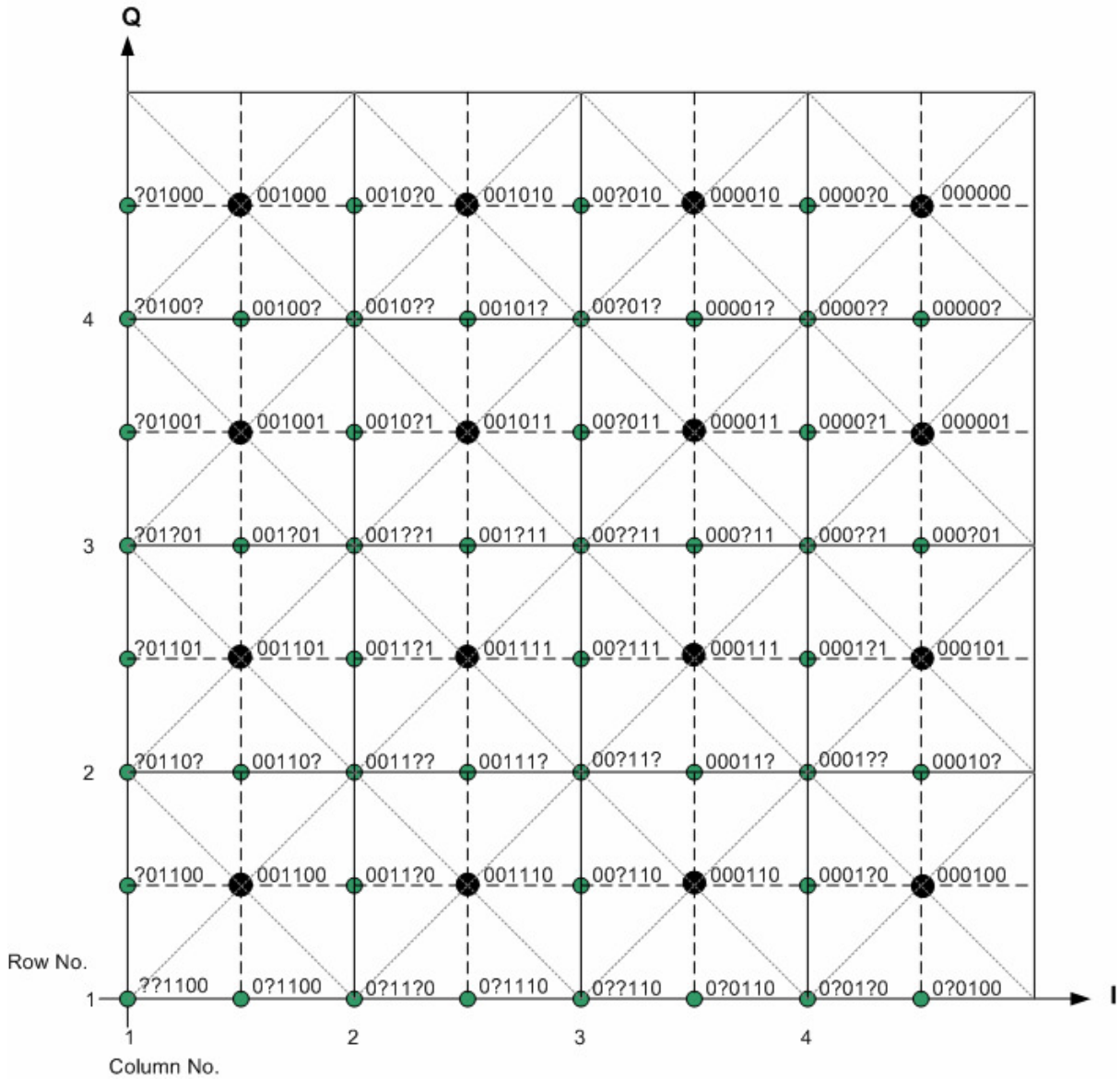


Fig. 14. The upper-right quadrant for a Gray-bit distribution 64-QAM constellation, showing the bits values of the constellation and border points. "?" denotes uncertain bit reliability.