System and Cross-layer Design for H.264/AVC-based Wireless Video Applications

Thomas Stockhammer
Siemens AG, Mobile Devices
stockhammer@nomor.de

***Abstract* - H.264/AVC will be an essential component in emerging wireless video applications thanks to its excellent compression efficiency and network-friendly design. However, a video coding standard itself is only one component within the application and transmission environment. Its effectiveness strongly depends on the selection of appropriate modes and parameters at the encoder, at the decoder, as well as in the network. In this paper we discuss the integration of H.264/AVC in wireless real-time video applications. Modern wireless networks provide many different means to adapt quality of service, such as forward error correction methods on different layers and end-to-end or link layer retransmission protocols. The applicability of all these encoding and network features depends on application constraints, such as the maximum tolerable delay, the possibility of online encoding, and the availability of feedback and cross-layer information. We discuss the use of different coding and transport related features for different applications, namely video telephony and conferencing, video streaming, and video broadcasting. Guidelines for the selection of appropriate video coding tools, video encoder and decoder settings, as well as transport and network parameters are provided and justified. Selected simulation results are presented.**

***Index Terms* – H.264/AVC, error resilience, wireless video, video broadcast.**

1 INTRODUCTION

Most of the emerging and future mobile client devices will significantly differ from those being used for speech communications only: handheld devices will be equipped with a color display and a camera and they have sufficient processing power which allows presentation, recording, and encoding/decoding of video sequences. In addition, emerging and future wireless systems will provide sufficient bitrates to support video communication applications. Nevertheless, bitrates will always be a scarce resource in wireless transmission environments due to physical bandwidth and power limitations and thus efficient video compression is required. Nowadays H.263 and MPEG-4 Visual Simple Profile are used commonly in handheld products, but it is foreseen that H.264/AVC [1] will be the video codec of choice for many video applications. The compression efficiency of the new standard excels prior standards roughly by at least a factor of two. Although compression efficiency is the major feature for a video codec to be successful in wireless transmission environments, it is also necessary that a standard provides means to be integrated easily into existing and future networks as well as that it addresses the needs of different applications. This integration and appropriate system design is discussed in the following.

2 PRELIMINARIES

A. End-to-End Video Transmission

Figure 1 attempts to provide a suitable abstraction level of a video transmission system. In order to keep this paper focused, we have excluded capturing and display devices, user interfaces, and security issues; most computational complexity issues are ignored.

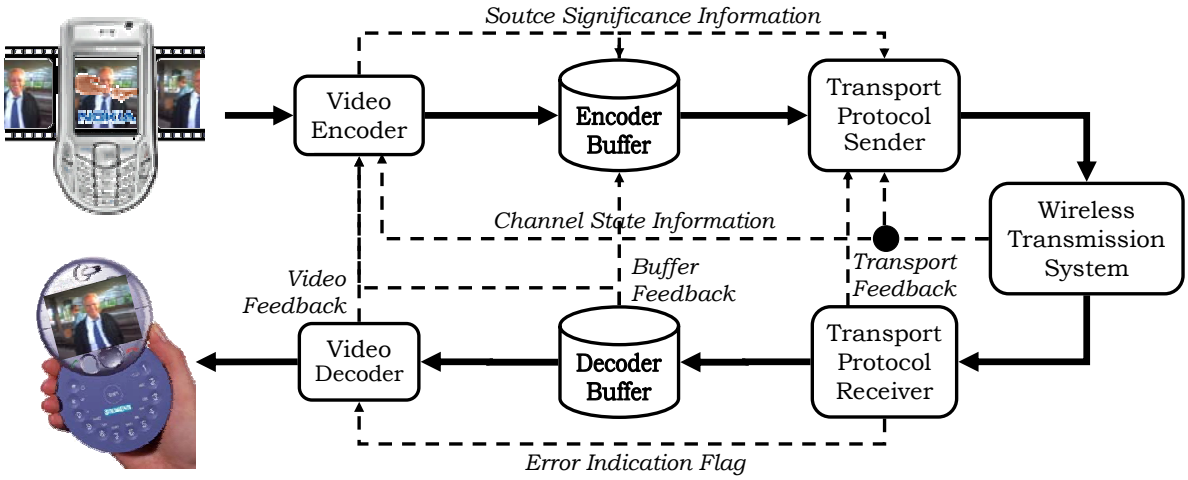

Figure 1 Abstraction of end-to-end video transmission systems.

In contrast to still image transmission video frames inherently include relative timing information, which has to be maintained to assure perfect reconstruction at the receiver's display. Furthermore, due to significant amount of spatial and temporal redundancy and irrelevancy in natural video sequences, video encoders are capable to reduce the actual amount of data significantly. However, too significant compression results in noticeable, annoying or even intolerable artifacts in the decoded video. A *trade-off* between *rate* and *distortion* is necessary. Real-time transmission of video adds additional challenges. According to Figure 1, the video encoder generates data units containing the compressed video stream possibly being stored in an encoder buffer before the transmission. A wireless transmission system might delay, loose, or corrupt individual data units. The unavailability of a single data unit usually has significant impact on the perceived quality due to spatio-temporal error propagation. In modern wireless system designs, data transmission is usually supplemented by additional information between the sender and the receivers and within the respective entities. Abstract versions of available messages are included in Figure 1; specific syntax and semantics as well the exploitation in video transmission systems will be discussed in more detail. Furthermore, each processing and transmission step adds some delay, which can be fixed, deterministic, or random. The encoder buffer and the decoder buffer allow compensating variable bit-rates produced by the encoder as well as channel delay variations to keep the end-to-end delay constant and maintain the timeline at the decoder. Nevertheless, if the *initial playout delay* Δ is or cannot be too extensive, late data units are commonly treated to be lost. Therefore, the system design also needs to find an appropriate tradeoff between *initial playout delay* and *data unit losses*.

B. *H.264-based Video Applications in 3GPP*

Digital coded video is used in different applications in wireless transmission environments. The integration of multimedia services in 3G wireless systems has been addressed in the recommendations of 3GPP depending on the application as well as the considered protocol stack: packet-switched one-to-one streaming (PSS) [2], multimedia multicast and broadcast service (MBMS) [3], circuit-switched video telephony (3G-324M) [4], packet-switched video telephony (PSC) [5], and multimedia messaging service (MMS) [6].

Applications can be distinguished by the maximum tolerable end-to-end delay, the availability and usefulness of different feedback messages, the availability and accurateness of channel state information at the transmitter, and the possibility of online encoding in contrast to pre-encoded content. Table 1 attempts to categorize and characterize wireless video applications with respect to these aspects. Especially the real-time services streaming and conversational services, but also broadcast services provide challenges in wireless transmission modes, as reliable delivery can in general not be guaranteed. The suitability of H.264/AVC for these services is discussed in the following.

Table 1 Characteristics of typical wireless video applications.

Video Application	3GPP	Max. delay	Video/Buffer Feedback		Transport Feedback		CSI	Encoding
			available?	useful?	available?	useful?	available?	
Download-and-Play	MMS	n.a.	No	-	yes	Yes	-	offline
On-demand streaming (pre-encoded content)	PSS	≥ 1 sec.	Yes	Yes	Yes	Yes	Partly	offline
Live streaming	PSS	≥ 200 ms	Yes	Yes	Partly	Yes	Partly	online
Multicast	MBMS	≥ 1 sec	Limited	Partly	Limited	Partly	Limited	both
Broadcast	MBMS	≥ 2 sec	No	-	No	-	No	both
Conferencing	PSC	≤ 250 ms	Limited	Yes	No	-	Limited	online
Telephony	PSC	≤ 200 ms	Yes	Yes	Limited	yes	Partly	online

In the following we will concentrate on packet-based real-time video services. Although in the first release of the 3G wireless systems, H.263 Profiles 0 and 3 and MPEG-4 Visual Simple Profile have been chosen, H.264/AVC was lately adopted as a recommended codec in these services, and it is expected that H.264/AVC will play a major role in emerging and future releases of wireless systems. It can also be foreseen that IP-based packet-switched communication will dominate multimedia transmission in wireless environments. 3GPP has chosen to use SIP and SDP for call control and RTP for media transport.

The elementary unit processed by an H.264/AVC codec is called Network Abstraction Layer (NAL) unit, which can be easily encapsulated into different transport protocols and file formats, such as MPEG-2 transport stream, RTP, and MP4 file format. There are two types of NAL units, Video Coding Layer (VCL) NAL units and non-VCL NAL units. VCL NAL units contain data that represents the values and samples of video pictures in form of a slice or slice data partitions. One VCL NAL unit type is dedicated for a slice in an Instantaneous Decoding Refresh (IDR) picture. A non-VCL NAL unit contains supplemental enhancement information (SEI), parameter sets, picture delimiter, or filler data.

Figure 2 shows the basic processing of a H.264 VCL slice within RTP and 3GPP framework. The slice is packetized in a NAL unit wich itself is encapsulated in RTP/UDP/IP according to [7] and finally transported through the protocol stack of a wireless system such as GPRS, EGPRS, UMTS, or CDMA2000. The RTP payload specification supports different packetization modes: In the simplest mode a single NAL unit is transported in a single RTP packet, and the NAL unit header co-serves as an RTP payload header.

Each NAL unit consists of a one-byte header and the payload byte string. The header indicates the type of the NAL unit and whether a VCL NAL unit is a part of reference or non-reference picture. Furthermore, syntax violations in the NAL unit and the relative importance of the NAL unit for the decoding process can be signaled in the NAL unit header. More advanced packetization modes allow aggregation of several NAL units into one RTP packets as well the fragmentation of a single NAL unit into several RTP packets.

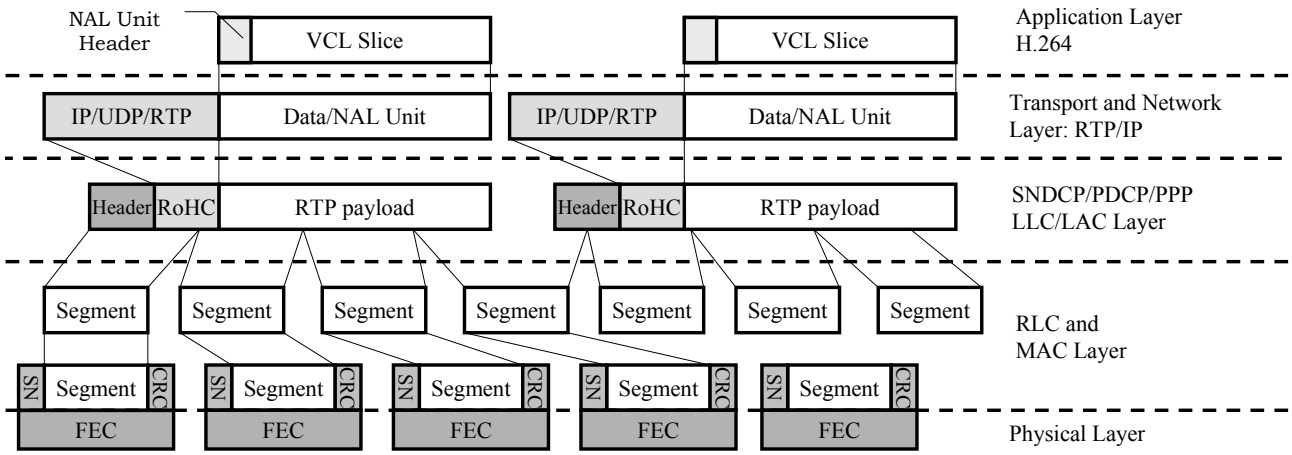

Figure 2 Integration example of an H.264 VCL slice in RTP payload and 3GPP framework.

For the wireless system we will in the following concentrate on UMTS terminology, the corresponding layers for other systems are shown in Figure 2. After Robust Header compression the generated IP/UDP/RTP packet is encapsulated into a single PDCP packet that becomes an RLC-SDU. As a typical RLC-SDU has a larger size than a RLC-PDU, it is then segmented into smaller units, so called link layer packets which serve as the basic units to be transmitted within the wireless system. The length of these segments depends on the selected bearer as well as the coding and modulation scheme in use. The physical layer generally adds FEC to RLC-PDUs depending on the coding scheme in use such that a constant length channel-coded and modulated block is obtained. This channel-coded block is further processed in the physical layer before it is sent to the far end receiver. The transmission time interval (TTI) between two consecutive RLC-PDUs determines the block length, but also the system delay. The receiver performs error correction and detection and possibly requests retransmissions. It is important to understand that in general the detection of a lost segment results in the loss of an entire PDCP packet, and therefore the encapsulated RTP packet as well as the NAL unit is lost. Wireless systems such as UMTS or GSM usually provide bearers with RLC-PDU error rates in the range of 1% to 10%, whereby 1% bearers are significantly more costly in terms of radio resources. About 10-25% more users can be supported with error rates 10% than with error rates of 1%.

C. System Design – Adding Reliability in the System

Due to the processing of IP packets in packet radio networks, the loss rate of IP packets strongly depends on the length of an IP packet. Common application with IP packet lengths in range of 500 to 1000 bytes would exceed loss rates in the wired Internet by even for low physical error rates. Therefore, to support video application of sufficient quality, additional means in the protocol stack for increased reliability are necessary. There exists an obvious tradeoff between compatibility and complexity aspects in wireless systems and the performance of means for reliability. Specifically, it has been considered to add means for reliability in four different layers of the wireless system, namely (i) on the physical layer, (ii) on RLC layer, (iii) on RTP layer, and finally (iv) in the application itself. Also, mixtures and combinations of reliability means could be considered. All included reliability features should be checked against the performance in terms of necessary overhead, residual overhead, and the added delay. Furthermore, the impact on legacy equipment especially on the network side has to be considered. This obviously results in multidimensional decisions which are to be taken in awareness of the considered application and the system constraints. However, for ultimate judgement of different features, the features need to be optimized by themselves. In the following we try to address these different aspects. Performance results, optimizations, as well as benefits and drawbacks of different system designs will be elaborated.

3 VIDEO ERROR-RESILIENCE FEATURES

A. H.264 Error Resilience Features

In some scenarios, the transmission link cannot provide sufficient QoS to guarantee a virtually error-free transmission link. The most common scenario are low-delay services such as video telephony and conferencing. For this purpose, H.264/AVC itself provides different features for increased error resilience. A suitable subset of those are presented and evaluated in the following, for exhaustive treatment we refer to, e.g. [8], [9], [10], [11]. Assume that the wireless system is treated as a simple IP link, whereby the packets to be transmitted are lost due to the RLC-PDU losses on the physical layer. The considered video transmission system is shown in Figure 3. In the simple mode of RTP payload specification each NAL unit is then carried in a single RTP packet. The encoding of a single video frame results in one or several NAL units each carried in single RTP packets. Each macroblock within the video frame is assigned to a certain RTP packet on the slice structuring and macroblock map. Further, assume that the RTP packets are transmitted over a channel which either delivers it correctly, or it is lost. Correctly delivered NAL units received after its decoding time has been expired are also treated to be lost.

At the encoder the application of *slice structured coding* and *flexible macroblock ordering* allows limiting the amount of lost data in case of transmission errors [9]. Although the mapping of macroblocks to NAL units basically provides any flexibility, there exist a few typical modes. With FMO typical macroblock maps with checkerboard patterns are suitable allocation patterns. Within a slice group, the encoder typically chooses a mode with the slice size bounded to some maximum $S_{\max}$ in bytes resulting in an arbitrary number of MBs per slice. This mode is especially useful to introduce some QoS as the slice size determines the loss probability in wireless systems due to the processing shown in Figure 2. The syntax in RTP and slice headers allows detecting missing slices. As soon as the erroneous mac-

roblocks are detected, *error concealment* should be invoked. In the following we exclusively use the error concealment introduced in the H.264 test model software [12].

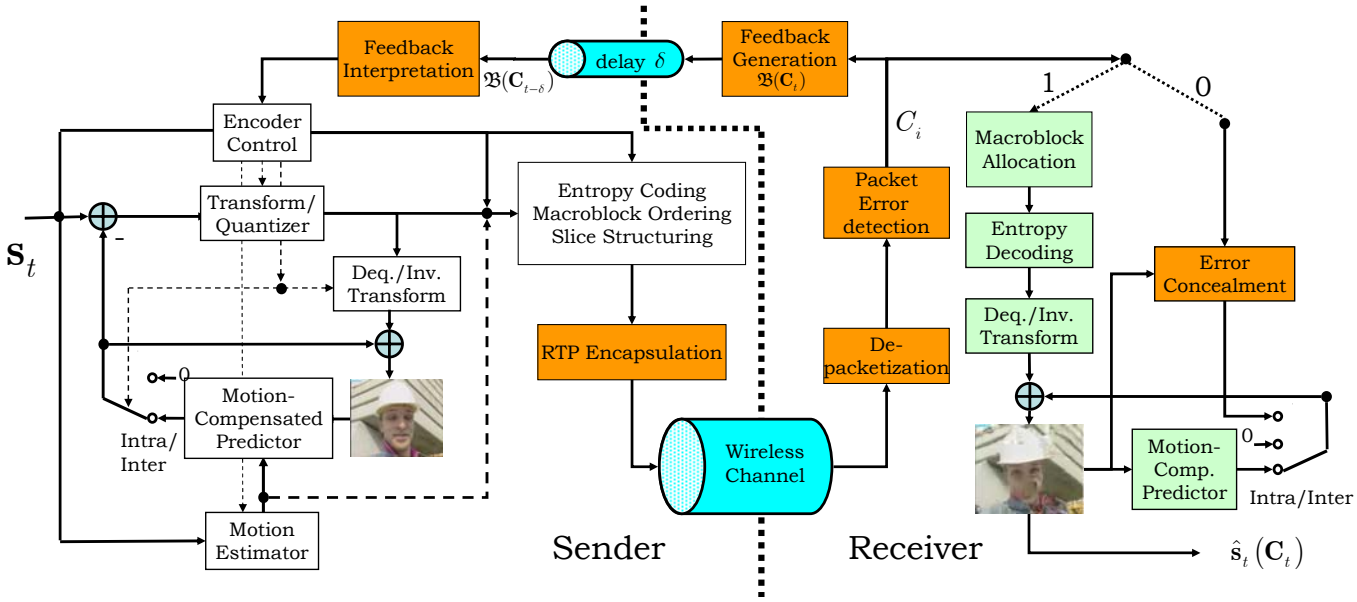

Figure 3 Hybrid video coding in RTP-based packet lossy environment.

Despite these advanced packetization modes and error concealment allow reducing the difference between the encoder and the decoder reference frames, a mismatch in the prediction signal in both entities is not avoidable. Then, the effects of spatio-temporal error propagation resulting from the motion compensated prediction can be severe. Although the mismatch decays over time to some extent, the leakage in standardized video decoders is not very strong. Therefore, the decoder has to be provided with other means which allow reducing or completely stopping error propagation. The straightforward way of inserting IDR frames is quite common for broadcast and streaming applications as these frames are also necessary to randomly access the video sequences. However, especially for low latency real-time applications such as conversational video, the insertion of complete intra frames increases the instantaneous bit-rate significantly. This can cause additional latency for the delivery over constant bit-rate channels and compression efficiency is significantly reduced when inserting intra frames too frequently. Therefore, more subtle methods are required to synchronize encoder and decoder reference frames. Two basic principles in H.264/AVC can be exploited to fight error propagation: applying *intra coded MBs* more frequently as well as the use *multiple reference frames*. A low-bitrate feedback channel from the video decoder to the encoder can support the selection of appropriate modes. For example, just recently, RTCP has been extended with *extended report* packet type allowing to transport information about each RTP packet. In addition new efforts within the AVT will allow timely and fast feedback especially for point-to-point scenarios giving senders new possibilities to react to packet losses. Nevertheless, feedback messages are usually delayed to at least some extent.

B. System Design Guidelines

It is important to understand that H.264/AVC provides features, such a flexible multiple reference frame concept, intra coding, switching pictures, slices and slice groups. However, rather than assigning these features to a single application, they are usually general enough to be used for different purposes. For example, intra coding of individual macroblocks

can be used for compression efficiency as well as error resilience. The encoder implementation is responsible to appropriately select the encoding parameters in the *operational coder control*. Thereby, the encoder must take into account constraints imposed by the application in terms of bit-rates, encoding and transmission delays, channel conditions, as well as buffer sizes. As the encoder is limited by the syntax of the standard, this problem is referred to as *syntax-constrained rate-distortion optimization* [13]. In case of a video coder such as H.264/AVC, the encoder must select parameters, such as motion vectors, macroblock modes, quantization parameters, reference frames, and spatial and temporal resolution as shown in , to provide good quality under given rate and delay constraints. Two simplify matters decisions on good selections of the coding parameters are usually divided in three levels:

Macroblock Level Decisions – Operational Encoder Control

Encoder control performs local decisions, e.g., the selection of macroblock modes, reference frames, or motion vectors on macroblock level. Most appropriately these decisions are based on rate-distortion optimizations applying Lagrangian techniques [14], [15]. The trade-off between rate and distortion is exclusively determined by the selection of the Lagrangian parameter λ. A coding option o^* from a set of coding options $\boldsymbol{O}$ is selected such that the linear combination of some distortion $D(o)$ and some rate, $R(o)$, - both resulting from the use of coding mode o - is minimized, i.e.

$$o^* = \arg\min_{o \in O} \left(D(o) + \lambda R(o) \right) \qquad (1)$$

In any case the rate $R(o)$ is selected as the number of bits necessary to encode the current MB with the selected mode o. Interestingly, also the Lagrangian parameter, which is connected with the quantization parameter, needs not be changed for different scenarios [19]. However, the distortion $D(o)$ as well as the set of coding options, $\boldsymbol{O}$, is selected depending on the expected channel conditions. If the encoder assumes an error-free channel, then for best compression efficiency it is wise to select $D(o)$ as the encoding distortion caused by mode o, e.g. the sum of squared distances, as well as $\boldsymbol{O}$ as all accessible coding options, e.g. all prediction modes and all reference frames.

In the anticipation or the knowledge of possible losses of NAL units additional intra information has to be introduced. It has been proposed [16], [17], [18] to modify the selection of the coding modes according to (1) to take into account the influence of the lossy channel. For example, when encoding a macroblock with a certain coding option o, it is suggested to replace the encoding distortion $D(o)$ by the decoder distortion $D(o,\mathbf{c})$ with $\mathbf{c}$ the observed channel sequence at the decoder. In general, the channel behavior is random and the realization $\mathbf{c}$, observed by the decoder is unknown to the encoder. However, with the knowledge of the statistic of the channel sequence $\mathbf{C}$ the encoder is able to compute some expected decoder distortion $\mathrm{E}\{D(o,\mathbf{C})\}$ which can be incorporated in the mode decision in (1) instead of the encoding distortion. The computation of the expected decoder distortion in the encoder is not trivial: In practical systems variants of the well-known so-called ROPE algorithm [16], [23] can be used providing an excellent estimate of $\mathrm{E}\{D(o,\mathbf{C})\}$ for most cases. Nevertheless, in the H.264/AVC test model encoder the expected decoder distortion is estimated based on a Monte-Carlo-like method [19], [10]. With this method as well as with model of the channel process assuming statistically independent NAL unit losses of some adapted loss rate p, one can generate bitstreams with good error resilience properties.

The availability of the fast feedback channel has led to different standardization and research activities on *interactive error control* (*IEC*) in recent years. Assume that a delayed version of the channel process experienced at the re-

ceiver, $\boldsymbol{c}$, is known at the encoder. This characteristic can be conveyed from the decoder to the encoder by acknowledging correctly received NAL units, sending a not-acknowledge messages for missing NAL units or both types of messages. Although retransmissions of lost data units are not possible, channel realizationexperienced by the receiver can still be useful to avoid or limit error propagation encoder though the erroneous frame has already been decoded and displayed at the decoder. In case of *online encoding*, this channel information is directly incorporated in the encoding process to reduce, eliminate, or even completely avoid error propagation. Whereas these initial Error Tracking and NEWPRED approaches must rely on existing simple syntax or has been supported by very specific syntax , the extended syntax allowing to select MB modes and reference frames on MB basis permits incorporating methods for reduced or limited error propagation in a straight-forward manner [17], [10]. Similarly to operational encoder control for error-prone channels, the delayed decoder state can also be integrated in a modified encoder control according to (1). Different operation modes are briefly discussed which can be distinguished only by the set of coding option $\boldsymbol{O}$ and the applied distortion metric $D(o)$; the principle of different modes is sketched in Figure 4.

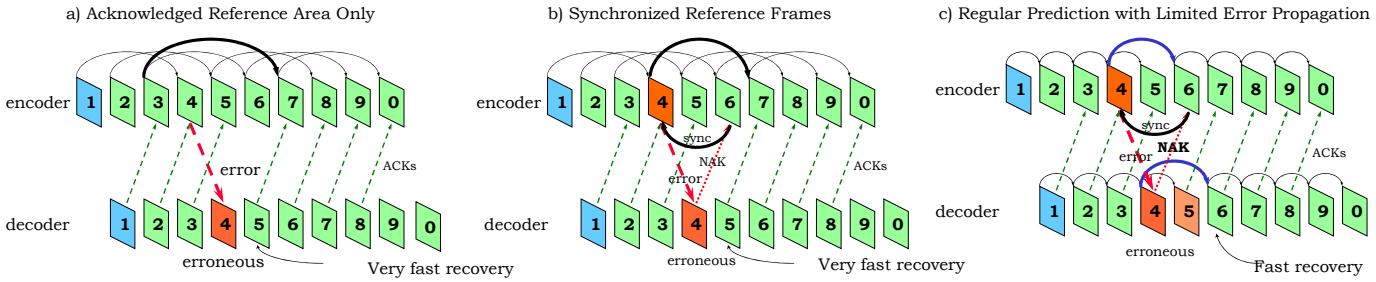

Figure 4 Operation of different interactive error control modes in the video encoder.

In the mode shown in Figure 4a) only the decoded representation of NAL units which have been positively acknowledged at the encoder, are allowed to be referenced in the encoding process which can be accomplished by restricting the option set $\boldsymbol{O}$ in (1) to acknowledged area only. Note that the restricted option set depends on the frame to be encoded and is basically applied to both, the motion estimation as well as in the reference frame selection. If no reference area is available, the option set is restricted to intra modes only. In the mode presented in Figure 4b) the encoder synchronizes its reference frames to the reference frames of the decoder by using exactly the same decoding process for the generation of the reference frames in the encoder. The important difference is that not only positively acknowledged NAL units are allowed to be referenced, but also a concealed version of not-acknowledged NAL units. Therefore, the encoder must be aware of the error concealment applied in the decoder. Although error propagation is completely eliminated, in case of longer feedback delays as well as low error rates, a significant amount of good prediction signals are excluded from the accessible reference area in the encoder control resulting in significantly reduced coding efficiency. Therefore, in mode 3 as shown in Figure 4c) the encoder only alters its operation in case that it receives a NACK. This mode obviously performs well in case of lower error rates. However, for higher error rates and longer feedback delays error propagation still occurs quite frequently. Finally, in [16] and [17] techniques have been proposed which combine this mode with the encoder control for error-prone transmission, but unfortunately add significant complexity. It is worth to mention with the concept of switching-pictures, similar techniques can also be applied for pre-encoded content [24].

Frame-Level Decisions - Rate Control

Rate control targets to maintain the constraints imposed by the application and the hypothetical reference decoder (HRD) by dynamically adjusting quantization parameters, or more elegantly, the Lagrangian parameter in the operational encoder control for each frame [13],[25], [26]. The rate control mainly controls the timing and bit-rate constraints of the application and is usually applied to achieve a CBR encoded video suitable for transmission over CBR channels. The aggressiveness of the change of the quantization/Lagrangian parameter allows a trade-off between quality and instantaneous bit-rate characteristic of the video stream. If the quantization /Lagrangian parameter is kept constant over the entire sequence, the quality is almost equal over the entire sequence, but the rate usually varies over time resulting in a VBR-encoded video.

Sequence and GOP-level Decisions - Global parameter selection

In addition to the decision made during the encoding process, usually a significant amount of parameters are predetermined taking into account application, profile and level constraints. For example Group-of-Picture (GOP) structures, temporal and spatial resolution of the video, as well as the number of reference frames are fixed. In addition, commonly packetization modes such slice sizes, error resilience tools such as FMO, are not determined on the fly but are reasonably selected. Nevertheless, these issues still provide rooms for improvements as the selection of the packetization modes is hardly done on the fly.

C. Experimental Results

The validation and comparison of the presented concepts needs extensive simulations which have partly been presented in the references provided. Nevertheless, it is basically completely infeasible to test and investigate different system designs due to the huge amount of possible parameters. Therefore, the video coding expert group (VCEG) of the ITU has defined and adopted appropriate common test conditions for 3G mobile transmission of PSC and PSS [27]. The common test conditions include simplified offline 3GPP/3GPP2 simulation software basically implementing the stack as presented in Figure 2. Radio channel conditions are simulated with bit-error patterns, which were generated from mobile radio channel simulations. The bit-error patterns are captured above the physical layer and below the RLC layer, and, therefore, they are used as the physical layer simulation in practice. The bit-error patterns for a walking user can basically be mapped to statistically independent RLC-PDU loss rates with about 1% and with about 10%. Note that the latter mode allows about 10-25% more users supported in a system due to the less restrictive power control. The RTP/UDP/IP overhead after RoHC, and the link layer overhead is taken into account in the bit-rate constraints. Furthermore, the H.264/AVC test model software has been extended to allow channel adaptive rate-distortion optimized mode selection with a certain assumed NAL unit loss rate p, slice structured coding, FMO with checkerboard patterns, interactive error control (IEC) with synchronized reference frames, as well as variable bit-rate encoding with a fixed quantization parameter for the entire sequence and CBR encoding with the quantization parameter selected such that number of bits for each frame are almost constant.

In the following we report simulation results applying the average PSNR which is computed as the arithmetic mean over the decoded luminance PSNR over all frames of the encoded sequence and over 100 transmission and decoding

runs. In any case we use the QCIF test sequence "Foreman" (30 Hz, 300 frames) coded at a constant frame rate of 7.5 fps for a walking user with 64 kbit/s with regular IPPP… structure.

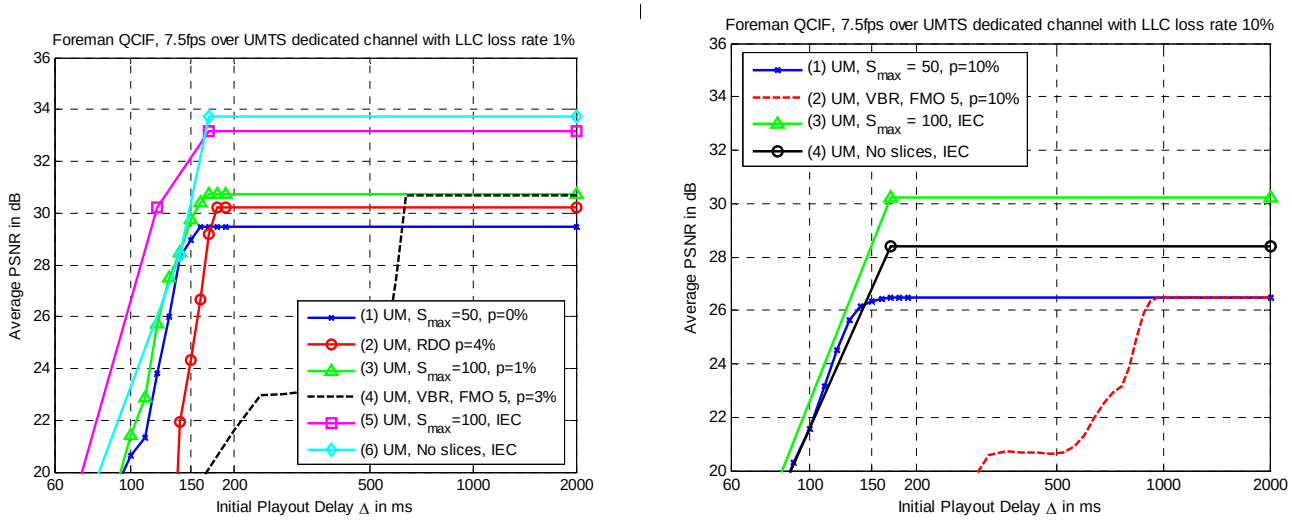

Figure 5 Performance in average PSNR for different video systems over initial playout delay for UMTS dedicated channel with link layer error rates of 1% and 10%.

We have chosen to present the results in terms of average PSNR over the initial playout delay at the decoder, Δ, whereby for the delay components in the system only the encoder buffer delay and the transmission delay on the physical link is considered. Additional processing delay as well as transmission delays on the backbone networks might add in practical systems. Figure 5a) shows the performance for link layer loss rates of about 1%. Curve (1)-(4) can be applied without any feedback channel, but the video encoder assumes a link layer loss rate of about 1%. In curve (1), (2), and (3) CBR encoding is applied to match the bitrate of the channel taking into account the overhead with bit rates 50, 60, and 52 kbits/s, respectively. Curve (1) relies on slices of maximum size S_{max}=50 bytes only, no additional intra updates to remove error propagation is introduced. Curve (2) in contrast neglects slices, but uses optimized intra updates with $p=4\%$, curve (3) uses a combination of the two features with S_{max}=100 bytes and p=1%. The transmission adds a delay of about 170ms for the entire frame, for lower initial delays NAL units are lost due to late losses. For initial play-out delays above this, only losses due to link errors occur. If the initial play-out delay is not that critical, a similar performance can be achieved by VBR encoding combined with FMO with 5 slice groups in checkerboard pattern as well as optimized intra with $p=3\%$ as shown in curve (4). However, the VBR encoding causes problems for low-delay applications in wireless bottleneck links, and therefore, a CBR-like rate control is essential. Curves (5) and (6) assume the availability of a feedback channel from the receiver to the transmitter, which is capable to report the loss or acknowledgement of NAL units. They use IEC, only results for synchronized reference frames for a feedback delay of about 250ms are shown. Other feedback modes show similar performance for this typical feedback delay. For the slice mode with S_{max}=100 bytes shown in curve (5) significant gains can be observed for delays suitable for video telephony applications, but due to the avoided error propagation it is even preferable to abandon slices and only rely on IEC as shown in curve (6). The average PSNR is about 3 dB better than the best mode not exploiting any feedback.

Figure 5b) shows similar curves for a UMTS bearer with 10% link layer error rate. The resulting high NAL unit error rates need a significant amount of video error resilience if applied over unacknowledged mode. Curve (1) applying

slice structured mode with S_{max}=50 bytes and p=10% is necessary for good quality under these circumstances. For VBR with FMO similar quality can be achieved, but only if the initial playout delay is higher. However, in both cases the quality is not satisfying. Only IEC with slice structured coding with S_{max}=100 according to curve (3) can provide average PSNR over 30 dB for initial playout delay below 200ms, whereas in this case dispensing with slices is not beneficial in combination with IEC according to curve (4).

In summary it is observed that for low delay wireless applications, it is necessary that the underlying layer provides bearers with sufficient QoS. Adaptation to the transmission conditions by the use of slice-structured coding and especially the use of macroblock intra updates is essential. Best performance is achieved by the use IEC as long as the feedback delay is reasonably low. In this case it is even suitable to use not any other error resilience tools if the measure of interest in the average PSNR.

4 FORWARD-ERROR CORRECTION

A. Forward Error Correction Mechanisms on Different Layers

A powerful mean for additional reliability in error-prone systems is forward error correction (FEC), especially for applications where no feedback is available and/or end-to-end delay is relaxed. A typical scenario are video broadcast services, e.g. within 3GPP MBMS. With recent advances in the area of channel coding practical codes such as Turbo codes and LDPC codes as well as variants of those allow transmission very close to the channel capacity. Examining the protocol stack in the most obvious point of attack would be to enhance the *FEC in the physical layer*. For increased coding and diversity gains, it is beneficial to increase the block length of the code, but at the expense of additional latency. Such an approach has been undertaken for UTRAN-based MBMS bearers where the physical layer channel coding provides sufficient freedom to introduce such modifications. Longer RLC-PDUs are in general also beneficial for the residual IP-packet loss rate due to the processing as shown in Figure 2. However, this approach usually requires significant changes in legacy hardware and existing network infrastructure. Thus, solutions on higher level of the protocol stack are often preferred. GERAN-based MBMS systems allow blind repetitions of RLC-PDUs, which can be combined with Chase combining at the receiver. Furthermore, erasure correction schemes based on Reed-Solomon codes within the *RLC/MAC layer* have been considered for MBMS scenarios (see [28] and references therein).

Despite their good performance as well as the manageable complexity, the required changes have still been considered too complex; it was decided to keep existing packet-radio systems below the IP-layer basically unchanged and introduce reliability above the IP/UDP layer. Means as presented in Section 3 could be used, but initial results in [29] as well some following results show that no sufficient QoS for real-time video can be provided with video-resilience tools only especially lacking a feedback channel. Therefore, FEC *above* the UDP layer is considered. For RTP-based transmission, simple existing schemes such as RFC2733 [29] might have been used. However, for non-real-time services the powerful FLUTE/ALC/LCT framework [31] - all protocols defined with the RMT working group of the IETF - is used in 3GPP providing significantly better performance than RFC 2733. The FLUTE framework has been modified to be used also for RTP-based FEC [3]; the RTP-FEC framework is also considered to be adopted within the IETF AVT. The MBMS video streaming delivery system is shown in Figure 6. In this case the source RTP packets are transmitted almost unmodified to the receiver. However, in addition a copy of the source RTP packet is forwarded to

the FEC encoder and placed in a so-called *source block*, a virtual two-dimensional array of width T bytes, referred to as encoding symbol length. Further RTP packets are filled into the source block until the second dimension of the source block, the height K determining the information length of the FEC code to be used, is reached. Each RTP packet starts at the beginning of a new row in the source block. The flexible signaling specified in [3] allows adapting T for each session as well as the height K for each source block to be encoded. After processing all original RTP packets to be protected within one source block, the FEC encoder generates N-K repair symbols by applying a code over each byte column-wise. These repair symbols can be transmitted individually or as blocks of P symbols within a single RTP packet. Sufficient side information is transmitted in payload headers of both, source and repair RTP packet, such that the receiver can insert correctly received source and repair RTP packets in its encoding block. If sufficient data for this specific source block is received, the decoder can recover all packets inserted in the encoding block, in particular the original source RTP packets. These RTP packets are forwarded to the RTP decapsulation process which itself hands the recovered application layer packets to the media decoder. Codes considered in the MBMS framework are Reed-Solomon codes [32], possible extended by 2D-RS codes to allow K greater than 255 as well as Raptor codes [33] which have some unique properties in terms of performance, encoding and decoding complexity, as well as flexibility.

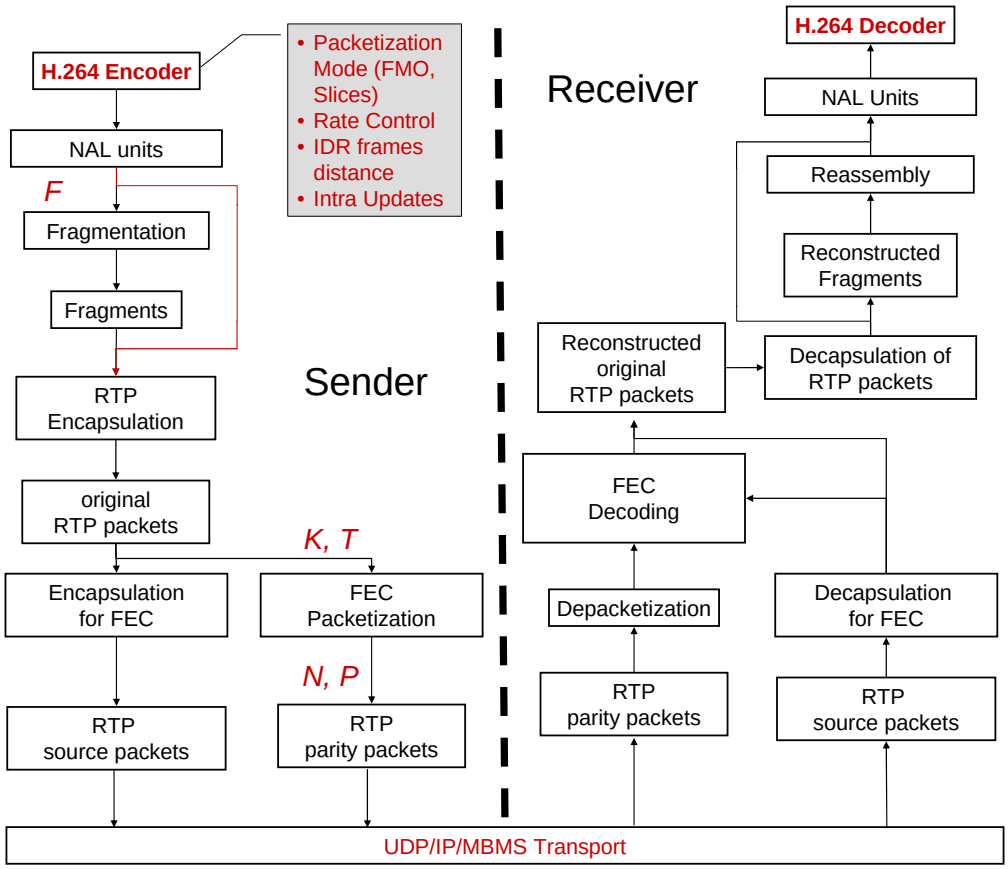

Figure 6 MBMS FEC framework for H.264-based streaming video delivery.

B. System Design Guidelines

Figure 6 shows the video streaming system and also highlights a several optimization parameters. They should be adequately selected taking into account the application and transmission constraints as well as advanced receiver algo-

rithms. Among others, H.264 encoding parameters, fragmentation of NAL units, the dimension and the rate of the error protection, as well as the transport and physical layer options are to be selected. Some rationales will be discussed in the following, an implemented optimization will be presented and simulations as shown in following subsections will provide further indication on good system design.

Assume that a maximum end-to-end delay constraint δ has to be maintained for the application. Furthermore, assume that the MBMS transport parameters in terms of RLC-PDU N_p sizes, header overheads H_{IP}, and bit-rates R are given. Furthermore, assume that we target for a specific target code rate r_t which results in a specific supported video bitrate R_v. Finally, assume that the symbol size T is suitably pre-determined [3] and P is selected as the largest integer such that $P \leq (N\text{p}-H_{IP}-H_{RTP})/T$ with H_{RTP} the overhead in the repair FEC packets. This avoids that at most two repair RTP packets are lost by the loss of single RLC-PDU. Then, our transmitter optimizes the actual code parameters N and K for each source block under delay and code constraints such that K is as large as possible under the delay constraints and N is as large as possible under the constraint that the actual code rate is below the target code rate, i.e. $K/N \leq r_t$. It is obvious that lower target code rate r_t results in lower video bit-rate R_v, but also lower NAL unit loss rate p_{NALU}, and vice versa. For conciseness reasons, we skip the details of this optimization.

This leaves the appropriate dimensioning of the video and the transmission parameters. For the video parameters, we assume a loose rate control which maintains the target bit-rate R_v for a GOP. The GOP itself is bounded by an IDR frame and consists of regular P-frames only. It is further assumed that the stream is encoded such that in the operational rate control the macroblock modes are chosen assuming an NAL unit loss rate, p, in the encoding process which matches the NAL unit loss rate of some worst case user for the selected transmission parameters, i.e. $p=p_{NALU,wc}$. For the packetization modes we assume different parameters, namely that

- no slices are used and each NAL unit is transported in a single RTP packet,
- slices are used in the encoding such that the size of the resulting RTP/IP packet does not exceed the length of an RLC-PDU or at least does not exceed some reasonable multiple of the RLC-PDU,
- FMO with checkerboard pattern is used, whereby the number of slice groups is varied and no specific optimization on the packet sizes is performed.
- no slices are used, but the NAL unit is fragmented into multiple fragmentation units according to RFC3984, each fragmentation unit is transported in a separate RTP packet and reassembly of NAL units at the receiver is only possible if all fragments are received correctly. The fragmentation size is chosen as the slice sizes. The benefits of this approach are discussed in detail in [34].

To obtain insight in the performance of FEC in 3GPP applications, especially in the case of MBMS, we have implemented the different options and tried to find suitable parameter settings and overall performance figures for these kind of applications.

C. Experimental Results

To obtain reasonable results for the MBMS environment, we have extended simulation software for 3G mobile transmission by the RTP-FEC framework. This software allows setting the different parameters as presented in the previous subsection. Any pre-coded H.264 NAL unit sequence can be transmitted taking into account timing information. In the

following we will restrict ourselves to ideal erasure codes as the performance of Raptor codes is only marginal worse when compared to ideal codes and we safe the extra burden of Raptor implementation and simulation. For comparison reason we again use the same video sequence, namely the QCIF test sequence "Foreman" (30 Hz, 300 frames) coded at a constant frame rate of 7.5 fps with regular IPPP… structure. This results in an IDR frequency of 10 seconds which has assumed to be reasonable. Flexibility in the video encoding is provided by allowing to adapt the bit rate R_v including packetization overhead for NAL headers as well as the macroblock intra update ratio specified by p_{NALU}. Specifically, we have selected operation points which result in application layer error rates p_{AL}={0, 0.1, …, 2, 3,…, 20}% for each of the systems presented in Figure 7. The video is encoded with a variable bit-rate rate control to match the application layer throughput η_{AL}. Note that in any case the maximum delay constraint of δ=5s is not exceeded. In addition, we might apply fragmentation of NAL units to obtain RTP packets of size 300 bytes and 600 bytes. Also, FMO can be applied, we restrict ourselves to two slice groups ordered in checkerboard pattern. The channel is again assumed to support 64 kbit/s and different RLC-PDU loss rates are considered. Figure 7 shows the average PSNR over the application layer throughput η_{AL} for different system designs for RLC-PDU loss rate of 1% (left-hand side) and 10% (right-hand side). For both cases, we assume that the considered user is also the worst case user for which the system is optimized. For each point shown in the figures we assume a certain target code rate r_t. The channel is assumed to have transmission time interval of 80ms, for comparison also one result with TTI=10ms is shown for the RLC-PDU loss rate 1%. We use T=20 and in case of TTI=80 ms P=30 and for TTI=10ms P=6. In addition, header compression is assumed such that PDCP/IP/UDP header is reduced to 10 bytes.

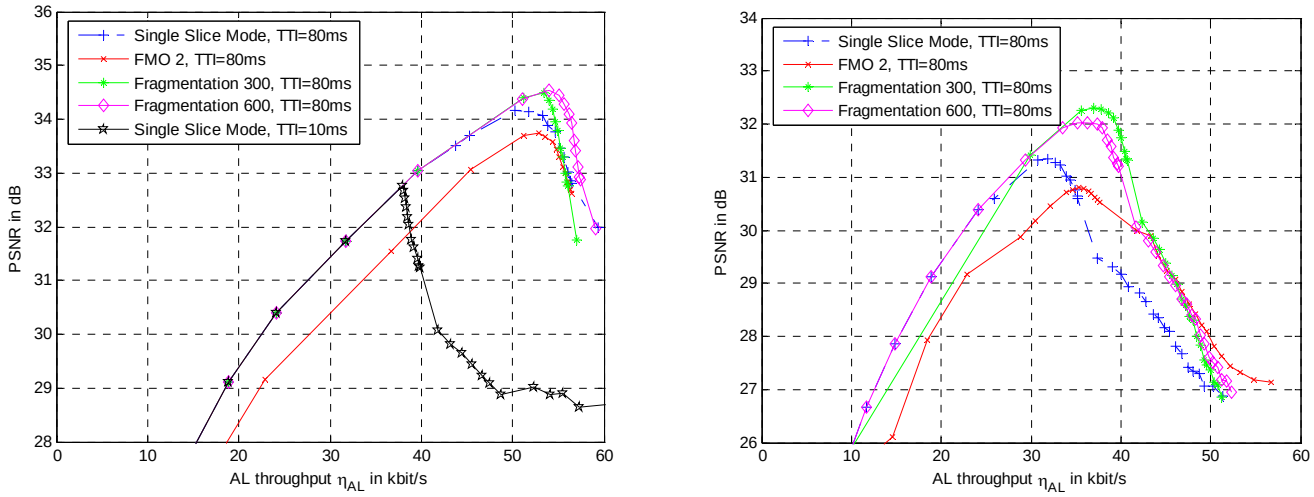

Figure 7 Average Peak Signal-to-Noise Ratio (PSNR) over the application layer throughput η_{AL} for different system designs.

Many interesting insights are observed from the results. Let us first investigate the 1% case. In any case for low throughput the FEC is sufficient to obtain an error-free received video resulting in only encoding distortion. The reduced compression efficiency due to FMO is observed, the single slice mode as well as the fragmentation operate in all cases with the same encoded bit-stream. The TTI 10ms results in significantly higher IP-packet loss, as the likelihood that a long IP-packet is hit by an error is significantly larger. Hence, longer TTIs are beneficial, if the RLC-PDU loss rate is the same or even lower for the longer TTI. Note that for Turbo Codes as used in UMTS this is the case. With increasing throughput the quality increases as long a the FEC is sufficient to correct the errors. If the FEC is correctly

designed the best performance is achieved by fragmentation, as the RTP packets are most suitably aligned with RLC-PDUs [34]. Shorter fragments are worse due to higher packet overheads. If the FEC is not appropriately designed the quality degrades again although the video is coded with optimized macroblock updates. One can observe that without any FEC – represented by the end-points in the graphs – that FMO performs best. Therefore, it is obvious that the redundancy is better spent for FEC than for error resilience in the video. Similar results are obtained for the RLC-PDU loss rate of 10%, but the PSNR is obviously lower. Again, FMO only exceeds the other schemes in case of high error rates, but overall the performance of optimized FEC and fragmentation performs best. However, we note that in this case an end-to-end delay of at least 5 seconds has to be accepted. Dimensioning for other delays is subject of ongoing work.

5 ADVANCED TRANSPORT-LAYER FEATURES

A. Retransmission Protocols in Wireless Systems

In point-to-point connections, usually the communication setup is bi-directional. In less time-critical applications, protocols can be employed allowing the retransmission of lost entities. Wireless systems usually support a so-called *acknowlegded mode* on the RLC layer which allows retransmitting lost radio blocks.at the expense of usually unpredictable and variable delay. The retransmission delay usually depends on different factors such as the transmission time interval, as well as the syntax and semantics retransmission request message. If designed appropriately retransmission requests can be at the receiver with only very few TTIs. Usually, the acknowledged mode can be further distinguished in a persistent mode applying retransmissions until the radio block is correctly received and a non-persistent mode applying only a limited number of retransmissions, but resulting residual error rates. An appropriate mode should be selected suitable for the delay constraints of the application.

Obviously, retransmissions can also be carried out above the IP level. Within IETF AVT a selective retransmission scheme has been proposed [35] which allows retransmitting RTP packets. To speed up decoding, Arbitrary Slice Ordering (ASO) as supported by H.264 is beneficial in this case as the NAL units might be delivered out-of-order. The applicability for applications with less stringent delay constraints is obvious, but its inferiority to link layer retransmission protocols will also be obvious. However, RTP retransmission is found useful to combat packet losses happening in other elements of the transmission system than the radio access link. To provide at least some amount of information to the decoder in case of losses, it might also be possible to re-send redundant coded slices and pictures instead of high rate primary frame. Finally, it is worth to mention that there exists a huge amount of work where streaming media over TCP is investigated, mainly due to the high distribution of this reliable protocol. However, it is also well know that TCP is not capable to deal with wireless losses as it is optimized for congestion awareness. If TCP is applied to transmit video data reliably, it is necessary that the link layer provides sufficient QoS.

An obvious problem when transmitting real-time data over error-prone channels results from the fact that the use of retransmissions or powerful forward error correction is not permitted due to the delay constraints. However, this timing only relates to data which is live generated as for example in case of conversational video, live streaming, or live broadcasting where the sending time of the data is usually closely coupled to the display time referred to as time-stamp based streaming (TBS). In case that pre-encoded data is transmitted and the decoder buffer is sufficiently large, one can

transmit data earlier than its nominal sending time, so called ahead-of-time streaming (ATS), which allows better exploitation of the channel. This strategy can be even extended by transmitting more important data earlier allowing for example more retransmissions for this important data [36]. Other advanced transport issues which take into account multiple users in a wireless system are not further discussed. For some specific video related issues and system design of schedulers and network buffers we refer for example to [37].

B. Experimental Results

The experimental results have been obtained using an extended version of the common test conditions for 3G mobile transmission. The simulation software has been extended to allow running different modes in addition to the unacknowledged (UM), namely acknowledged mode (AM) on the RLC layer with persistent and non-persistent mode, application layer retransmissions (ALR), timestamp-based streaming (TBS) and ahead-of-time streaming (ATS). In Figure 8 the results are compared to those presented in Figure 5.

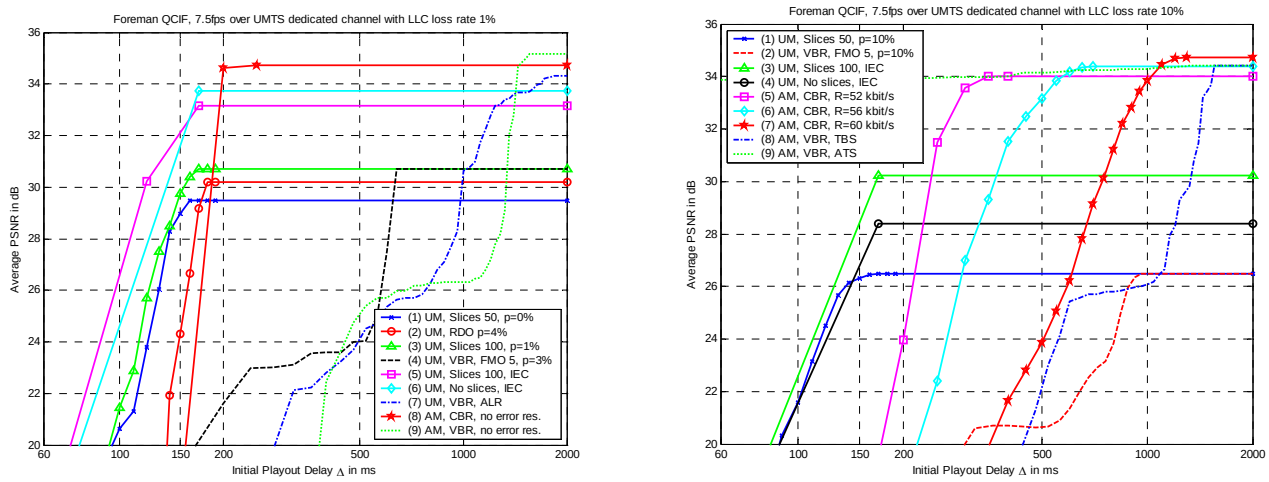

Figure 8 Performance in average PSNR for different advanced video transport systems over initial playout delay for UMTS dedicated channel with link layer error rates of 1% and 10%.

For the 1% RLC-PDU loss rate, curve (7) shows the case where the feedback is exploited for application layer retransmission of RTP packets, VBR encoding is used. It is obvious that this mode is not suitable for low-delay applications, but as delay does not matter, it provides better performance than any other scheme relying on methods in the video layer. Curve (8) and (9) show the performance of CBR encoded video and VBR encoded video, respectively, with matching bit-rates for the acknowledged mode. The performance of the CBR mode is excellent even for lower delays, but at least 200ms of initial play-out delay must be accepted which makes the applicability for conversational modes critical, but not infeasible, if the system supports fast retransmissions. It is also observed that for VBR encoding low-delay applications cannot be well supported, but if initial playout delays of a few seconds can be accepted, VBR encoding with acknowledged mode on the link layer provides the best overall performance.

For the 10% RLC-PDU loss rate, the advanced transport system enhances the overall system. Significantly better performance can be achieved by the use of the acknowledged mode, but only for initial play-out delays well over 300ms according to curve (5) with CBR and bitrate 52 kbit/s. Interestingly, if the initial play-out delay is increased, one can also support higher bitrates resulting in higher quality. This behaviour has been exploited in the HRD specification of

H.264 where it was recognized that an encoded stream is contained not just by one, but many leaky buckets [38]. Finally, curve (8) and (9) show the performance for VBR encoded video in case of timestamp-based streaming and ahead-of-time streaming over the AM mode. It is interesting that with ATS low playout delays can be achieved, but obviously this requires that the data can not be generated online. In addition, in practical systems some kind of start-up delay might occur due to TCP-like congestion control. It is also worth to note that the performance of video over the 10% link layer loss bearer does not differ significantly from the 1% one, if the initial play-out delay constraints are not really stringent.

When compared to the FEC, it is observed that in case of the 1% RLC-PDU rate, the advanced transport system designs cannot gain significantly, but for the 10% RLC-PDU rate the AM mode with RLC layer retransmission outperforms MBMS by about 2dB.

6 CONCLUSIONS

In this work the benefits of H.264/AVC in wireless transmission environments have been shown. In addition to excellent compression efficiency, H.264/AVC provides features which can be used in one or several application scenarios and also allows easy integration in most networks. The selection and combination of different features strongly depends on the system and application constraints, namely bit-rates, maximum tolerable playout delays, error characteristics, online encoding possibility, as well as availability of feedback and crosslayer information. Although the standardization process is finalized, the freedom at the encoder as well as the combination with transport modes such as FEC and retransmission strategies promises optimization potentials. Table 2 provides an overview of proposed video and transport features for different applications with performance in terms of delay and average PSNR for different RLC-PDU loss rates and for a typical video sequence. In general, it is observed that error resilience on lower layers provides better performance than doing it in the video codec or on the RTP layer. However, in any case the system options as well as the application constraints have to be taken into account. Therefore, further research in the area of optimization, crosslayer design, feedback exploitation, and error concealment are necessary to fully understand the potential of H.264/AVC in wireless environments. However, researchers are encouraged to integrate transport protocol as well as wireless system options into their considerations rather than assuming QoS-unaware link and transport layers.

Table 2 Proposed Video and transport features for different applications with performance in terms of delay and average PSNR for different RLC-PDU loss rates and QCIF video sequence Foreman coded at 7.5 fps.

Video Application	Video Features	Transport Features	1% RLC-PDU loss rate		10 % RLC-PDU loss rate	
64 kbit/s UMTS transmission scenario			Delay	PSNR	Delay	PSNR
Download-and-Play On-demand streaming	VBR, no error res., playout buffering	ATS, AM on RLC	> 1.5 sec	35.2 dB	> 1 sec > 10 sec	34.4 dB 35.1 dB
Live streaming	CBR/VBR, no error res. playout buffering	TBS, AM on RLC	> 250 ms	34.7 dB	> 400 ms > 1.5s	34.0 dB 34.7 dB
Broadcast	VBR, regular IDR, no other error resilience	FEC, long TTI, fragmentation	> 5 sec	34.5 dB	> 5 sec	32.0 dB
Conferencing	CBR, Intra Updates, Slices	UM	> 150 ms	30.7 dB	> 150 ms	26.5 dB
Telephony	CBR, IEC, no slices	UM	> 150 ms	33.7 dB	> 150 ms	30.2 dB

ACKNOWLEDGEMENTS

The author would like to thank Thomas Wiegand, Stephan Wenger, Ye-Kui Wang, Günther Liebl, Miska Hannuksela, and Ingo Viering for useful discussions on the subject of this work.

REFERENCES

[1] ITU-T and ISO/IEC JTC 1, "Advanced Video Coding for Generic Audiovisual Services," ITU-Recommendation H.264 and ISO IEC 14996-10 AVC, 2003.

[2] 3GPP Technical Specification 3GPP TS 26.234, Rel-6, "Transparent end-to-end packet-switched streaming service (PSS); protocols and codecs."

[3] 3GPP Technical Specification 3GPP TS 26.346, Rel-6, "Multimedia multicast and broadcast service (MBMS); protocols and codecs."

[4] 3GPP Technical Specification 3GPP TS 26.110, "Circuit-switched video telephony (3G-324M)."

[5] 3GPP Technical Specification 3GPP TS 26.235 and 26.236, "Packet-switched conversational multimedia applications."

[6] 3GPP Technical Specification 3GPP TS 26.140 "Multimedia messaging service (MMS); media formats and codecs."

[7] S. Wenger, T. Stockhammer, M.M. Hannuksela, M. Westerlund, and D. Singer, "RTP payload Format for H.264 Video", IETF RFC3984, February 2005.

[8] G. Sullivan and T. Wiegand, "Video Compression – From Concepts to H.264/AVC Standard," *Proc. of the IEEE*, *Special Issue on Advances in Video Coding and Delivery*, Vol. 93, No. 1, pp. 18-31, January 2005.

[9] S. Wenger, "H.264/AVC over IP," IEEE Transactions on Circuits and Systems, vol. 13, no. 7, pp. 645-656, July 2003.

[10] T. Stockhammer, M.M. Hannuksela, and Thomas Wiegand, "H.264/AVC in Wireless Environments," IEEE Transactions on Circuits and Systems, vol. 13, no. 7, pp. 657-673, July 2003.

[11] S. Kumar, L. Xu, M. K. Mandal and S. Panchanathan, *Overview of Error Resiliency Schemes in H.264/AVC Video Coding Standard*, To appear in Elsevier Journal of Visual Communication and Image Representation (Special issue on H.264/AVC Video Coding Standard), 2005

[12] Y.-K. Wang, M.M. Hannuksela, V. Varsa, A. Hourunranta, and M. Gabbouj, "The error concealment feature in the H.26L test model," in Proc. ICIP, vol. 2, pp. 729-732, Sept. 2002.

[13] A. Ortega and K. Ramchandran, "Rate-Distortion Techniques in Image and Video Compression," *IEEE Signal Processing Magazine* , pp. 23-50, Vol. 15, No. 6, Nov. 1998.

[14] T. Wiegand, M. Lightstone, D. Mukherjee, T. G. Campbell, and S. K. Mitra. "Rate-Distortion Optimized Mode Selection for Very Low Bit Rate Video Coding and the Emerging H.263 Standard," IEEE Circuits and Systems on Video Technology, vol. 6, no. 2, pp. 182–190, April 1996.

[15] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, "Rate-Constrained Coder Control and Comparison of Video Coding Standards," IEEE Transactions on Circuits and Systems for Video Technology, vol. 13, pp. 688-703, July 2003.

[16] R. Zhang, S. L. Regunathan, and K. Rose, "Video Coding with Optimal Inter/Intra-Mode Switching for Packet Loss Resilience," in IEEE JSAC, vol. 18, no. 6, pp. 966-976.
[17] T. Wiegand, N. Färber, K. Stuhlmüller, and B. Girod, "Error-resilient video transmission using long-term memory motion-compensated prediction", IEEE JSAC, vol. 18, no. 6, pp. 1050-1062.
[18] G. Cote, S. Shirani, F. Kossentini, "Optimal mode selection and synchronization for robust video communications over error-prone networks," IEEE JSAC, vol. 18, no. 6, pp. 952 -965.
[19] T. Stockhammer, D. Kontopodis, and T. Wiegand, "Rate-Distortion Optimization for H.26L Video Coding in Packet Loss Environment," in Proc. Packet Video Workshop 2002, Pittsburgh, PY, April 2002.
[20] B. Girod, and N. Färber, "Feedback-Based Error Control for Mobile Video Transmission," Proceedings of IEEE, vol. 97, no. 10, pp. 1707-1723, Oct. 1999.
[21] S. Fukunaga, T. Nakai, and H. Inoue, "Error resilient video coding by dynamic replacing of reference pictures," in Proc. IEEE Globecom, vol.3, Nov. 1996.
[22] D. Tian, M.M. Hannuksela, and M. Gabbouj, "Sub-Sequence Video Coding for Improved Temporal Scalability," Annex E of 3GPP SA4 document S4-040048, February 2004.
[23] H. Yang and K. Rose, "Recursive end-to-end distortion estimation with model-based cross-correlation approximation," IEEE ICIP 2003, Barcelona, Spain, September 2003.
[24] M. Karczewicz, and R. Kurçeren, "The SP and SI Frames Design for H.264/AVC", IEEE Transactions on Circuits and Systems, vol. 13, no. 7, July 2003.Circuits and Systems, vol. 13, no. 7, July 2003.
[25] Chi-Yuan Hsu, Antonio Ortega and Masoud Khansari, "Rate Control for Robust Video Transmission over Burst-Error Wireless Channels," *IEEE Journal on Selected Areas in Communications*, Special Issue on Multimedia Network Radios, Vol. 17, No. 5, pp. 756-773, May 1999.
[26] T. V. Lakshman, A. Ortega, and A. R. Reibman, "VBR video: Trade-offs and potentials," Proceedings of the IEEE, vol. 86, pp. 952-973, May 1998.
[27] G. Roth, R. Sjöberg, G. Liebl, T. Stockhammer, V. Varsa, and M. Karczewicz, "Common Test Conditions for RTP/IP over 3GPP/3GPP2," ITU-T SG16 Doc. VCEG-N80, Santa Barbara, CA, USA, Sep. 2001.
[28] H. Jenkac, G. Liebl, T. Stockhammer and W. Xu, "Flexible Outer Reed-Solomon Coding on RLC Layer for MBMS over GERAN, " in Proc. VTC Spring 2004, Milan, Italy, May 2004.
[29] Stockhammer, T.; Liebl, G.; Jenkac, H.: H.264/AVC Video Transmission over MBMS. - In: *Proc. IEEE International Workshop on Multimedia Signal Processing*, September 2004.
[30] J. Rosenberg and H. Schulzrinne, "An RTP Payload Format for Generic Forward Error Correction", IETF RFC2733, December 1999.
[31] T. Paila, M. Luby, R.Lehtonen, V. Roca and R. Walsh, "FLUTE - File Delivery over Unidirectional Transport", IETF RFC3926, October 2004.
[32] 3GPP SP-050088, "Report of FEC selection for MBMS," SA-WG4, March 2005.
[33] A. Shokrollahi, "Raptor Codes", Digital Fountain, DR2003-06-001, June 2003.
[34] 3GPP S4-050090 , "Alignment of H.264/AVC NAL Units for MBMS ", Siemens, February 2005.
[35] J. Rey, D. Leon, A. Miyazaki, V. Varsa, and R. Hakenberg, "RTP Retransmission Payload Format," Internet Draft, draft-ietf-avt-rtp-retransmission-06.txt, February 2003.
[36] T. Stockhammer and M. Walter, "Optimized H.264-Based Bitstream Switching for Wireless Video Streaming," in Proc. ICME, July 2005, Amsterdam, NL.
[37] G. Liebl, T. Stockhammer, H. Jenkac, C. Buchner, and A. Klein, "Radio Link Buffer Management and Scheduling for Video Streaming over Wireless Shared Channels," in Proc. International Packet Video Workshop 2004, Irvine, CA, December 2004.
[38] J. Ribas-Corbera, P.A. Chou, and S. Regunathan, "A Generalized Hypothetical Reference Decoder for H.264/AVC", IEEE Transactions on Circuits and Systems, vol. 13, no. 7, pp. 674-687, July 2003.