Hindawi Publishing Corporation
EURASIP Journal on Applied Signal Processing
Volume 2006, Article ID 89371, Pages 1–15
DOI 10.1155/ASP/2006/89371

Robust System and Cross-Layer Design for H.264/AVC-Based Wireless Video Applications

Thomas Stockhammer

BenQ Mobile, Haidenauplatz 1, 81667 Munich, Germany

Received 18 March 2005; Revised 30 September 2005; Accepted 4 October 2005

H.264/AVC is 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 a single component within a complex system. Its effectiveness strongly depends on the appropriate configuration of encoders, decoders, as well as transport and network features. The applicability of different features depends on application constraints, the availability and quality of feedback and cross-layer information, and the accessible quality-of-service (QoS) tools in modern wireless networks. We discuss robust integration of H.264/AVC in wireless real-time video applications. Specifically, the use of different coding and transport-related features for different application types is elaborated. Guidelines for the selection of appropriate coding tools, encoder and decoder settings, as well as transport and network parameters are provided and justified. Selected simulation results show the superiority of lower layer error control over application layer error control and video error resilience features.

Copyright © 2006 Hindawi Publishing Corporation. All rights reserved.

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 color displays and cameras and they will 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, bitrate 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 commonly used in handheld products, but it is foreseen that H.264/AVC [1] will be *the* video codec of choice for many video applications in the near future. The compression efficiency of the new standard excels prior standards roughly by at least a factor of two. These advantages also introduce additional processing requirements in both, the encoder and the decoder. However, dedicated hardware as well Moore's law will allow more complex algorithms on handheld devices in the future.

Although compression efficiency is the major attribute for a video codec to be successful in wireless transmission environments, it is also necessary that a standardized codec provides means to be integrated easily into existing and future networks as well as to be usable in different applications. A key property for easy and successful integration is robustness and adaptation capabilities to different transmission conditions. Thereby, rather than providing completely new and revolutionary ideas, H.264/AVC relies on well-known and proven successful concepts from previous standards such as MPEG-4 and H.263, but simplifies and generalizes those and attempts a natural integration of these technologies in the H.264/AVC syntax. Prior work on error resilience and network integration of preceding video coding standards has been presented in [2–5], as well as in references therein. Furthermore, H.264/AVC is designed such that it interfaces very well with packet-based networks such as RTP/IP [6].

In this work, the robustness and the suitability of the H.264/AVC design for wireless video applications are discussed. Specifically, we categorize and evaluate different features of the H.264/AVC standard for different applications. Therefore, Section 2 provides an overview of the considered application and transmission environments. Sections 3, 4, and 5 discuss robustness features within H.264/AVC as well as combinations with underlying transport protocol features based on forward error correction and retransmission protocols. For each case, we introduce the concepts, discuss system design issues, and provide experimental results within each section. Finally, Section 7 summarizes and compares these results and provides concluding remarks.

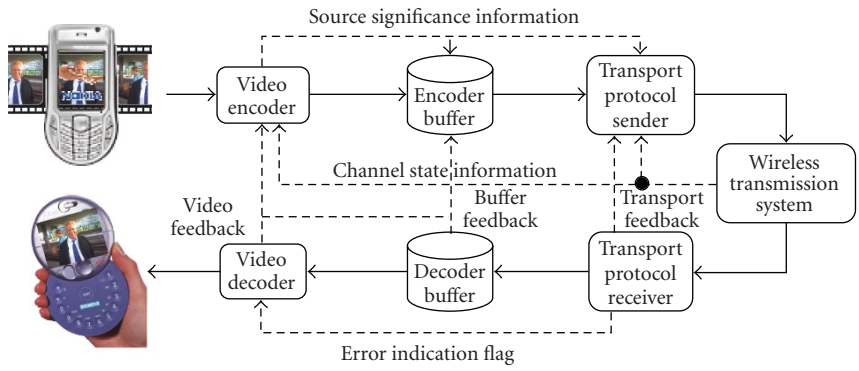


Figure 1: Abstraction of end-to-end video transmission systems.

2. PRELIMINARIES

2.1. End-to-end video transmission

Video applications are usually set up in an end-to-end connection either between a video encoding device or a media streaming server and a client. Figure 1 provides a suitable abstraction level of a video transmission system. In contrast to still image transmission, video frames inherently have assigned relative timing information, which has to be maintained to assure proper reconstruction at the receiver's display. Furthermore, due to significant amount of spatial and temporal redundancy in natural video sequences, video encoders are capable of reducing the actual amount of data significantly. However, too much compression results in noticeable, annoying, or even intolerable artifacts in the decoded video. A *tradeoff* 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. The generated video stream is encapsulated in appropriate transport packets, which are forwarded to a wireless transmission system. On the way to the receiver, the transport packets (and consequently the encapsulated data units) might be delayed, lost, or corrupted. At the receiver the transport packets are decapsulated, and in general the unavailability or late arrival of encapsulated data units is detected. Both effects usually have significant impact on the perceived quality due to frozen frames and spatio-temporal error propagation.

In modern wireless system designs, data transmission is usually supplemented by additional information between the sender and the receiver and within the respective entities. Some general messages are included in Figure 1, specific syntax and semantics as well as the exploitation in video transmission systems will be discussed in more detail. Specifically, the encoder can provide some information on the significance of certain data units, for example, whether a data unit is disposable or not without violating temporal prediction chains. The video encoder can exploit channel state information (CSI), for example, expected loss or bitrates, or information from the video decoder, for example, such as what reference signals are available. Buffer fullness at the receiver can be exploited at the transmitter, for example, for rate control purposes. The decoder can be informed about lost data units, which, for example, allow invoking appropriate error concealment methods. Finally, the transport layer itself can exchange messages, for example, to request retransmissions.

Each processing and transmission step adds some delay, which can be fixed or randomly varying. The encoder buffer and the decoder buffer allow compensating variable bitrates 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 not or cannot be too excessive, late data units are commonly treated as being lost. Therefore, the system design also needs to find an appropriate tradeoff between *initial playout delay* and *data unit losses*.

2.2. 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) [7], multimedia multicast and broadcast service (MBMS) [8], circuit-switched video telephony (3G-324M) [9], packet-switched video telephony (PSC) [10], and multimedia messaging service (MMS) [11].

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 CSI at the transmitter, and the possibility of online encoding in contrast to pre-encoded content. Table 1 categorizes and characterizes 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 in general, reliable delivery cannot be guaranteed. The suitability of H.264/AVC for these services is discussed.

In the remainder 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

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

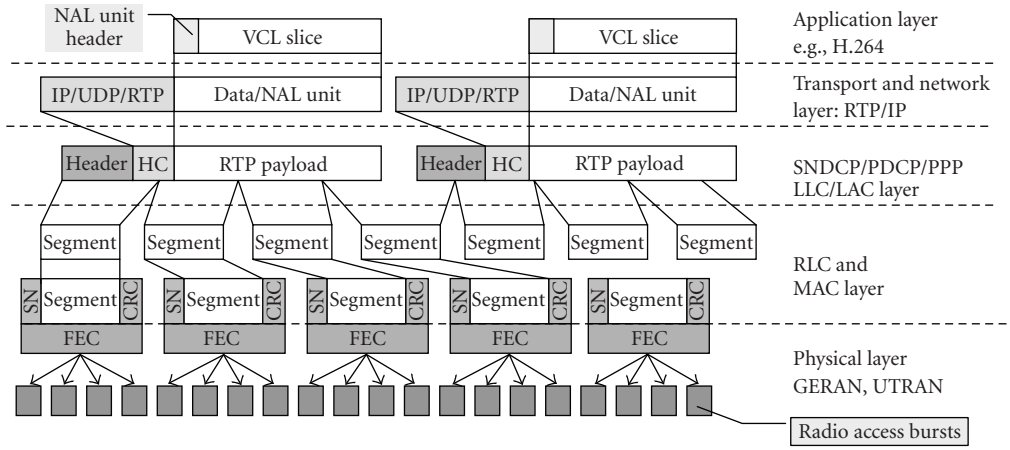


Figure 2: Protocol stack based on the exemplary encapsulation of an H.264 VCL slice in RTP payload and 3GPP packet-data mode.

Visual Simple Profile have been chosen, H.264/AVC was lately adopted as a recommended codec in all services, and it is expected that H.264/AVC will play a major role in emerging and future releases of wireless systems.

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. 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, parameter sets, picture delimiter, or filler data. Figure 2 shows the basic processing of an H.264 VCL data within real-time protocol (RTP) and third generation partnership project (3GPP) framework. The VCL data is packetized in NAL units which themselves are encapsulated in RTP according to [12] and finally transported through the protocol stack of any wireless system such as enhanced general packet radio services (GPRS) or universal mobile telecommunication system (UMTS). The RTP payload specification [12] supports different packetization modes: in the simplest mode a single NAL unit is transported in a single RTP packet, and the NAL unit header coserves 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 a reference or nonreference 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 packet as well the fragmentation of a single NAL unit into several RTP packets.

Furthermore, Figure 2 shows the protocol stack for the integration of RTP packets encapsulated in UDP and IP packets in a typical wireless packet-switched mode. For the wireless system we will concentrate on UMTS terminology, the corresponding layers for other systems are shown in Figure 2. Robust header compression (RoHC) is applied to the generated RTP/UDP/IP packet resulting in a single packet data convergence protocol (PDCP)-protocol data unit (PDU) that becomes a radio link control (RLC)-service data unit (SDU). As typically an RLC-SDU has a larger size than an RLC-PDU, the SDU is then segmented into smaller RLC-PDUs 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. Typically, RLC-PDUs have sizes between 20 bytes and 100 bytes. The physical layer generally adds forward error correction (FEC) to RLC-PDUs depending on the

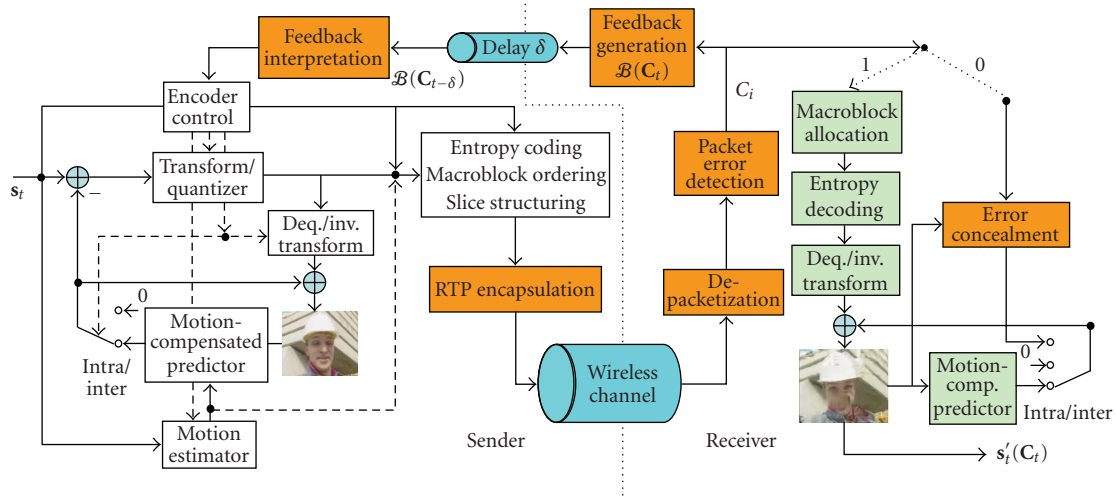


Figure 3: Hybrid video coding in RTP-based packet-lossy environment.

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 system delay and the bearer bitrate. 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 EGPRS 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%.

2.3. System design-adding reliability in the system

Due to the discussed processing of IP packets in packet-radio networks, the loss rate of IP packets strongly depends on their length. Common applications with IP packet lengths in range of 500 to 1000 bytes would exceed loss rates in the wired Internet 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 reliability methods. Specifically, we have considered to add means for reliability to four different layers of the wireless system, namely, (i) on the physical layer, (ii) on RLC layer, (iii) on the transport layer, and finally (iv) in the application itself. Also, mixtures and combinations of reliability means have been 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. These obviously result 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 themselves need to be optimized. In what follows we address these different aspects.

3. DESIGN WITH VIDEO ERROR RESILIENCE FEATURES

3.1. 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 scenarios are low-delay services such as video telephony and conferencing. For this purpose, H.264/AVC itself provides different features such as a flexible multiple reference frame concept, intra-coding, switching pictures, slices, and slice groups for increased error resilience [13–15]. A suitable subset of those is presented and evaluated, for exhaustive treatment we refer to references. 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 (MB) within the video frame is assigned to a certain RTP packet based on the applied slice structuring and macroblock map. Further, assume that the RTP packets are either delivered correctly (indicated with $C_i = 1$), or they are lost ($C_i = 0$). However, correctly delivered NAL units received after their decoding time has been expired are usually also considered to be lost.

At the encoder the application of *flexible macroblock ordering* (FMO) and *slice-structured coding* allows limiting the amount of lost data in case of transmission errors. FMO enables the specification of MB allocation maps which specify

the mapping of MBs to slice groups, where a slice group itself may contain several slices. Employing FMO, MBs might be transmitted out of raster scan order in a flexible and efficient manner. Out of several ways to map MBs to NAL units, the following are typical modes. With FMO typical MB maps with checkerboard patterns are suitable allocation patterns. Within a slice group, the encoder typically chooses a mode with the slice sizes bounded to some maximum $S_{\max}$ in bytes resulting in an arbitrary number of MBs per slice. This mode is especially useful since it introduces 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 the detection of missing slices. As soon as the erroneous MBs are detected, *error concealment* should be applied.

Despite the fact that 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 as the error concealment cannot reconstruct the encoder's reference frame. Then, the effects of spatio-temporal error propagation resulting from the motion-compensated prediction can be severe and the decoded video frame $\mathbf{s}'_t(\mathbf{C}_t)$ at time instant t strongly depends on observed channel behavior $\mathbf{C}_t$ up to time t. Although the mismatch decays over time to some extent, the recovery in standardized video decoders is not sufficient and fast enough. Therefore, the decoder has to reduce or completely stop 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 bitrate significantly. This increase can cause additional latency for the delivery over constant bitrate channels and compression efficiency is significantly reduced when intra-frames are inserted 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 of *multiple reference frames*. A low-bitrate feedback channel, denoted as $\mathfrak{B}(\mathbf{C}_t)$, might allow reporting either statistics or loss patterns on the observed channel behavior $\mathbf{C}_t$ from the video decoder to the encoder and can support the selection of appropriate modes. Despite recent efforts within the Internet Engineering Task Force to provide timely and fast feedback, feedback messages are still usually delayed, at least to some extent, such that the information $\mathfrak{B}(\mathbf{C}_t)$ is available at the video encoder with some delay δ; the delayed information is denoted by $\mathfrak{B}(\mathbf{C}_{t-\delta})$.

3.2. System design guidelines

In general, the encoder is not specified in a video coding standard, leaving significant freedom to the designer. It is not only important that a video standard provides error resilience features, but also that the encoder appropriately chooses the provided options. Therefore, we will discuss operational encoder control, rate control, and sequence level control from an error resilience perspective. The encoder implementation is responsible for appropriately selecting the encoding parameters in the *operational coder control*. Thereby, the encoder must take into account constraints imposed by the application in terms of bitrates, 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* [16]. In case of a video coder such as H.264/AVC, the encoder must select parameters, such as motion vectors, MB modes, quantization parameters, reference frames, and spatial and temporal resolution as shown in [17], to provide good quality under given rate and delay constraints. To simplify matters decisions on good selections of the coding parameters are usually divided in three levels.

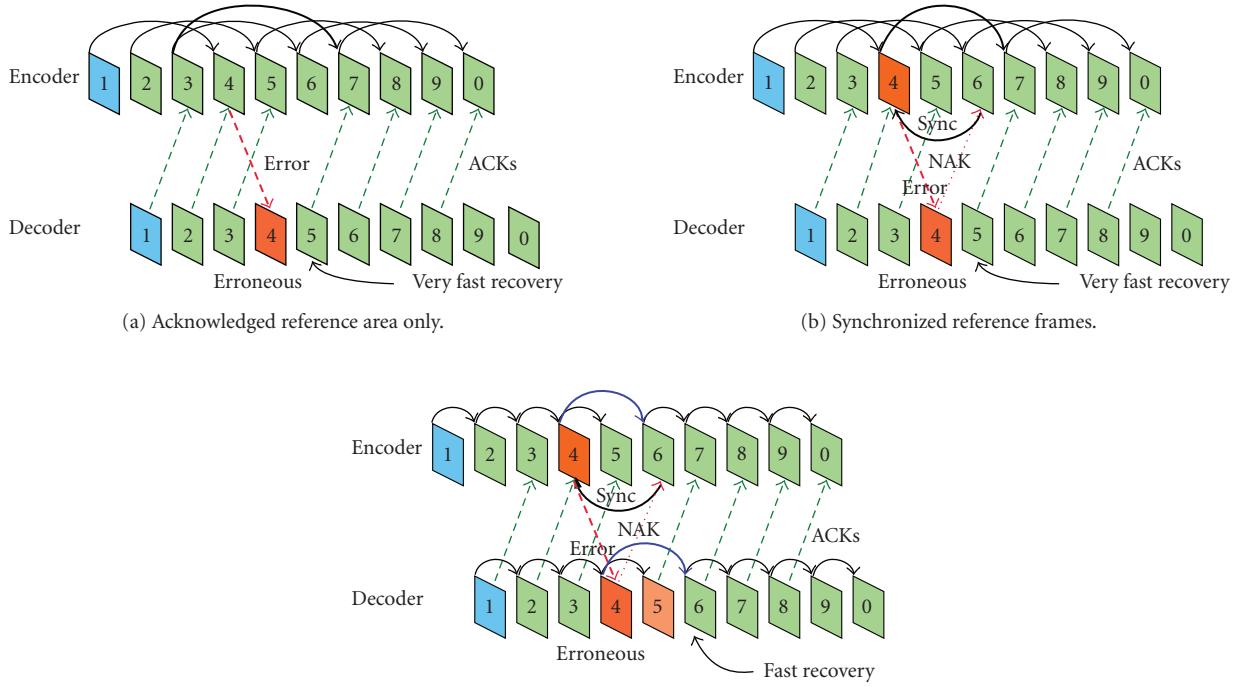
Macroblock level decisions: operational encoder control

Encoder control performs local decisions, for example, the selection of MB modes, reference frames, or motion vectors at MB level. More often than not these decisions are based on rate-distortion optimizations applying Lagrangian techniques [17, 18]. The tradeoff 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, that is,

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

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. 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 we propose to select $D(o)$ as the encoding distortion caused by mode o, for example, the sum of squared errors between the original and the encoded signal, as well as $\boldsymbol{O}$ as the set of all accessible coding options, for example, all prediction modes and all reference frames. Interestingly, the Lagrangian parameter, which is connected with the quantization parameter, needs not be changed in packet-lossy environments [19].

In the anticipation or the knowledge of possible losses of NAL units additional intra-information might be introduced. In [20–22], modifying the selection of the coding modes according to (1) to take into account the influence of the lossy channel has been proposed. For example, when encoding an MB with a certain coding option o, the encoding distortion $D(o)$ may be replaced by the decoder distortion $D(o, \mathbf{C}_t)$ with $\mathbf{C}_t$ the observed channel sequence at the decoder. In general, the channel behavior is random and the realization $\mathbf{C}_t$, observed by the decoder is unknown to the encoder. However, with the knowledge of the statistics of the channel sequence $\mathbf{C}_t$ the encoder is able to compute some expected decoder distortion $\mathrm{E}\{D(o, \mathbf{C}_t)\}$ which can be



(a) Acknowledged reference area only.

(b) Synchronized reference frames.

(c) Regular prediction with limited error propagation.

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

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 recursive optimal per-pixel estimate (ROPE) algorithm [20, 23] can be used providing an excellent estimate of $E\{D(o, \mathbf{C})\}$ 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 [14, 19]. With this method as well as with a model of the channel process that assumes statistically independent NAL unit losses of some adapted loss rate, p, one can generate streams with excellent error resilience and robustness properties.

The availability of expected channel conditions at the encoder can help reduce the error propagation. However, such propagation is usually not completely avoided, and, in addition, a non-negligible amount of redundancy is necessary as the advanced prediction methods are significantly restricted by the robust mode selection. However, if a feedback channel is available from the decoder to the encoder, the channel loss pattern as observed by the receiver can be conveyed to the encoder. Assume that a delayed version of the channel process experienced at the receiver, $\mathbf{C}_{t-\delta}$, is known at the encoder. This characteristic can be conveyed from the decoder to the encoder by acknowledging correctly received NAL units (ACK), sending a not-acknowledge messages (NAK) for missing NAL units or both types of messages. Even if retransmissions of lost data units are not possible due to delay constraints, channel realizations experienced by the receiver can still be useful to avoid or limit error propagation at the decoder 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. These *interactive error control* (*IEC*) techniques have been investigated in different standardization and research activities in recent years. Initial approaches such as error tracking [24] and new prediction (NEWPRED) [25–27] rely on existing simple syntax or have been incorporated by the definition of very specific syntax [28]. However, the extended syntax in H.264/AVC, which allows selecting MB modes and reference frames on MB basis, permits incorporating IEC methods for reduced or limited error propagation in a straightforward manner [14, 21]. Similarly to operational encoder control for error-prone channels, the delayed decoder state $\mathbf{C}_{t-\delta}$ can be integrated in a modified encoder control according to (1). Different operation modes, which can be distinguished only by the set of coding options $\boldsymbol{O}$ and the applied distortion metric $D(o)$, are illustrated in Figure 4.

In the mode shown in Figure 4(a) only the decoded representations of NAL units, which have been positively acknowledged at the encoder, are allowed to be referenced in the encoding process. This 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 4(b) 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. The important difference is that not only positively acknowledged NAL units, but also a concealed version of not-acknowledged NAL units, are allowed to be referenced. 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 is excluded from the accessible reference area in the encoder control resulting in significantly reduced coding efficiency. Therefore, in mode 3 shown in Figure 4(c) the encoder only alters its operation when it receives NAK. 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 [20, 21] techniques have been proposed which combine this mode with the robust encoder control for error-prone transmission, but unfortunately add significant complexity. It is worth to mention that with the concept of switching pictures, similar techniques can also be applied for pre-encoded content [29].

Frame-level decisions: rate control

Rate control aims to meet 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 [16, 30, 31]. The rate control mainly controls the delay and bitrate constraints of the application and is usually applied to achieve a constant bitrate (CBR)-encoded video suitable for transmission over CBR channels. The aggressiveness of the change of the quantization/Lagrangian parameter allows a tradeoff between quality and instantaneous bitrate 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 variable bitrate (VBR)-encoded video.

Sequence and GOP-level decisions: global parameter selection

In addition to the decisions made during the encoding process, usually a significant amount of parameters is 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 typically fixed. In addition, commonly packetization modes such slice sizes, error resilience tools such as FMO, are not determined on the fly but are selected a priori. Nevertheless, these issues still provide rooms for improvements as the selection of the packetization modes is hardly done on the fly.

3.3. Experimental results

The validation and comparison of the presented concepts need extensive simulations which have partly been presented in the references provided. Nevertheless, it is infeasible to exhaustively test and investigate different system designs due to the huge amount of possible parameters. Therefore, the video coding expert group (VCEG) has defined and adopted appropriate common test conditions for 3G mobile transmission of PSC and PSS [32]. The common test conditions include simplified offline 3GPP/3GPP2 simulation software that implements the stack presented in Figure 2. The bearers can be configured in unacknowledged mode (UM) to support low-delay applications. 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 provided bit-error patterns for a walking user can basically be mapped to statistically independent RLC-PDU loss rates of about 1% and about 10%. Note that the latter mode allows about 10–25% more users to be supported in a system due to the less restrictive power control. The RTP/UDP/IP overhead after RoHC, and the link layer overhead are taken into account in the bitrate 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, IEC with synchronized reference frames, as well as variable bitrate 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 is almost constant. We exclusively use the error concealment introduced in the H.264 test model software [33].

We report simulation results using the average PSNR (computed as the arithmetic mean over the decoded luminance PSNR over all frames of the encoded sequence and over 100 transmission and decoding runs). We exclusively use the QCIF test sequence "Foreman" (30 fps, 300 frames) coded at a constant frame rate of 7.5 fps for a walking user with 64 kbp/s with regular IPPP… structure.

We have chosen to present the results in terms of average PSNR over the initial playout delay at the decoder, Δ, for the delay components in the system only the encoder buffer delay and the transmission delay on the physical link are considered. Additional processing delay as well as transmission delays on the backbone networks might cumulate in practical systems. Figure 5(a) shows the performance for link layer loss rates of about 1%. Graphs (1)–(4) can be applied without any feedback channel, but the video encoder assumes a link layer loss rate of about 1%. In graphs (1), (2), and (3) CBR encoding is applied to match the bitrate of the channel taking into account the overhead with bitrates 50, 60, and 52 kbp/s, respectively. Graph (1) relies on slices of maximum size $S_{\max} = 50$ bytes only, no additional intra-updates to remove error propagation are introduced. Graph (2) in contrast neglects slices, but uses optimized intra-updates with $p = 4\%$, graph (3) uses a combination of the two features with $S_{\max} = 100$ bytes and $p = 1\%$. The transmission adds a delay of about 170 ms for the entire frame, for lower initial delays NAL units are

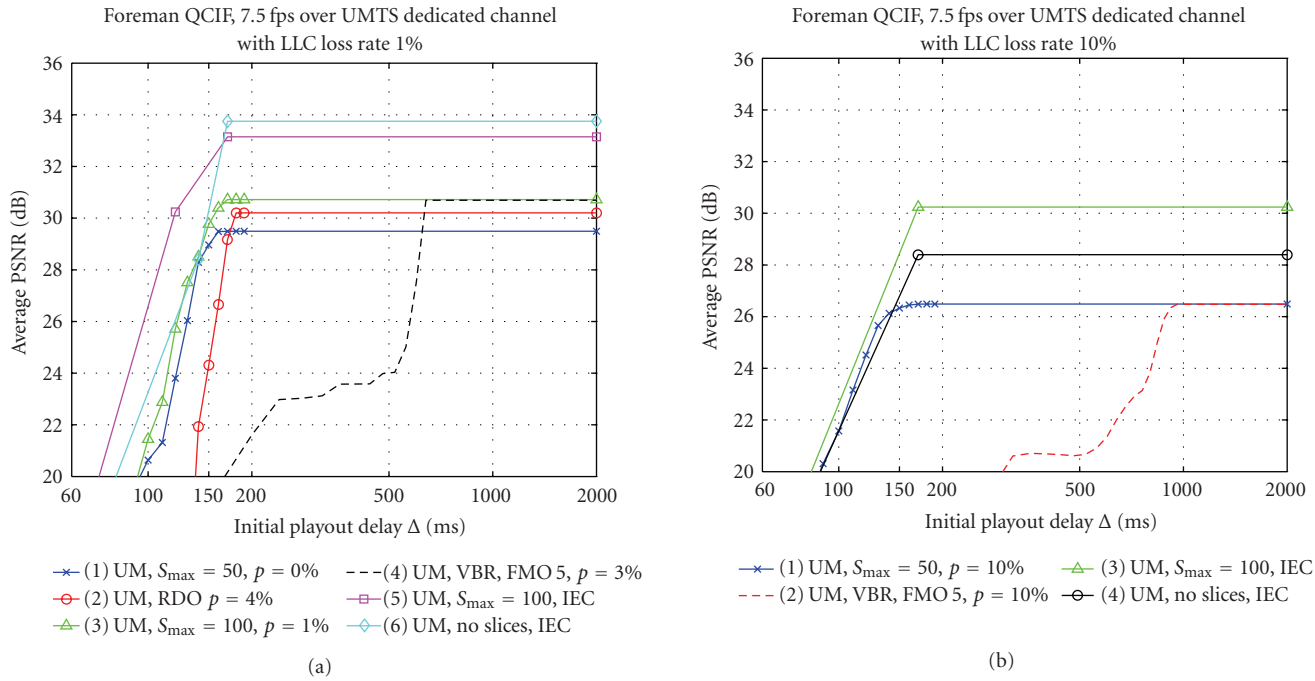


Figure 5: Performance in average PSNR for different video systems over initial playout delay for UMTS dedicated channel with link layer error rates of 1% and 10%.

lost due to late losses. For initial playout delays above this value, only losses due to link errors occur. If the initial playout 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 graph (4). However, the VBR encoding causes problems for low-delay applications in wireless bottleneck links, and therefore, a CBR-like rate control is essential. Graphs (5) and (6) assume the availability of a feedback channel from the receiver to the transmitter, which is capable of reporting the loss or reception of NAL units. They use IEC, only results for synchronized reference frames for a feedback delay of about 250 ms are shown. Other feedback modes show similar performance for this typical feedback delay. For the slice mode with $S_{max} = 100$ bytes shown in graph (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 graph (6). The average PSNR is about 3 dB better than the best mode not exploiting any feedback.

Figure 5(b) shows similar graphs 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. Graph (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 graph (3) can provide average PSNR over 30 dB for initial playout delay below 200 ms, whereas in this case dispensing with slices is not beneficial in combination with IEC according to graph (4).

In summary, 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 MB intra-updates is essential. Best performance is achieved using IEC as long as the feedback delay is reasonably low. Interestingly, with the use of IEC the PSNR is highest if no other error resilience tools are used.

4. DESIGN WITH FORWARD ERROR CORRECTION

4.1. *Forward error correction mechanisms on different layers*

A powerful method to add 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 is that of video broadcast services, for example, within 3GPP MBMS. With recent advances in the area of channel coding practical codes such as Turbo codes and LDPC codes as well as their variants allow transmission very close to the channel capacity. From the protocol stack in Figure 2, 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 MBMS bearers in UMTS where the physical layer channel coding provides sufficient freedom to introduce such modifications [34]. Instead of common TTIs of 10 ms, for MBMS the TTI

can be up to 80 ms. 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 levels of the protocol stack are often preferred. EGPRS-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 [35] and references therein).

Despite their good performance as well as the manageable complexity, the required changes have still been considered too complex; existing packet-radio systems below the IP layer have stayed unchanged and reliability was introduced above the IP/UDP layer. Methods as presented in Section 3 could be used, but initial results in [36] as well some following results show that sufficient QoS for real-time video can be provided with video resilience tools only for the case when a feedback channel is present. Therefore, FEC *above* the IP layer is considered. For RTP-based transmission, simple existing schemes such as RFC2733 [36] might have been used. However, for non-real time services the powerful file delivery over unidirectional transport (FLUTE) framework [37] has been introduced in 3GPP providing significantly better performance than RFC 2733. The FLUTE framework has been modified to be used also for RTP-based FEC [8].

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 [8] allows the adaptation of T for each session, as well as that of 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 added 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 having been considered in the MBMS framework are Reed-Solomon codes [38], possibly extended to multiple dimensions as well as Raptor codes [39] which have some unique properties in terms of performance, encoding and decoding complexity, as well as flexibility.

4.2. System design guidelines

With the optional integration of FEC, the amount of adjustable parameters for robustness increases even more. Figure 6 shows an MBMS video streaming system and also highlights several optimization parameters. They should be adequately selected taking into account the application constraints and transmission conditions. Among others, H.264/AVC 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 reasons will be discussed, an implemented optimization will be presented and simulations as shown in following subsections will provide further good indication for 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 RLC-PDU size N_{PDU}, header overhead H_{IP}, and bitrate R are given and that we aim for a specific target code rate r_{t} which results in a specific supported application throughput η_{AL} matching the available video bitrate R_{v}. The symbol size T is appropriately predetermined according to [8]. 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, that is, $K/N \leq r_{\text{t}}$. It is obvious that lower target code rate r_{t} results in lower video bitrate R_{v}, but also lower NAL unit loss rate p_{NALU}, and vice versa.

This leaves the appropriate selection of the video and the transmission parameters. For the video parameters, a relaxed rate control which maintains the target bitrate R_{v} for each GOP is sufficient. The GOP itself is bounded by an IDR frame and consists of regular P-frames only. For increased robustness the video stream is encoded such that in the operational rate control the MB modes are chosen assuming an NAL unit loss rate, p. Thereby, the NAL unit loss rate matches the loss rate of some worst-case users for the selected transmission parameters. Different packetization modes are considered, namely,

(i) no slices are used and each NAL unit is transported in a single RTP packet;
(ii) 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;
(iii) FMO with checkerboard pattern is used, whereby the number of slice groups is varied and no specific optimization on the packet sizes is performed;
(iv) 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 appropriately [40].

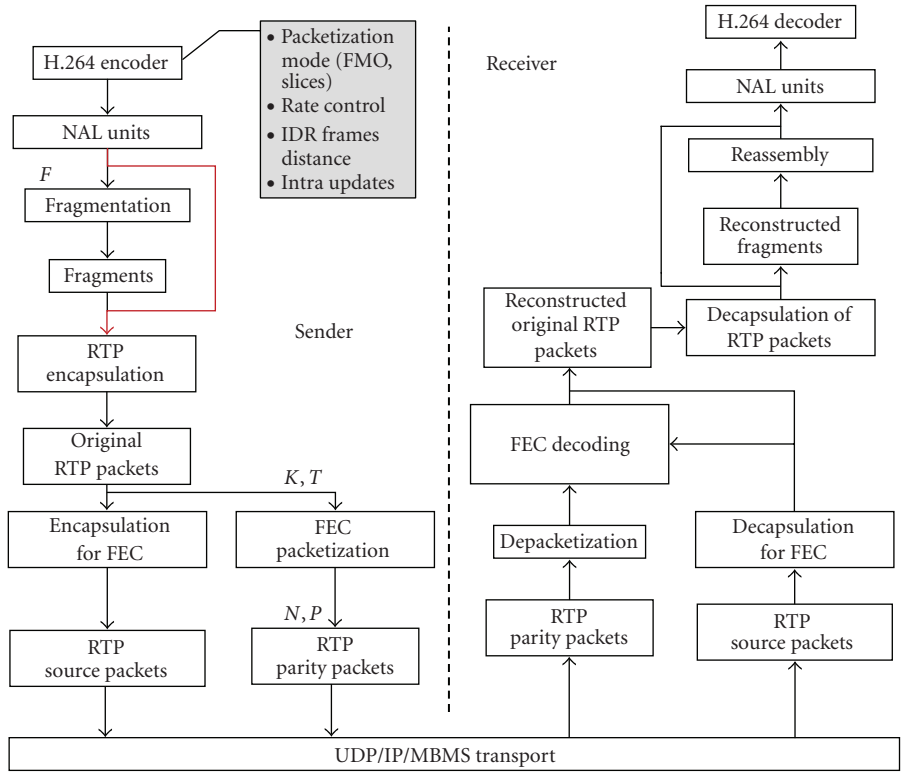


Figure 6: MBMS FEC framework for H.264-based streaming video delivery with F the fragmentation size, K the number of virtual source symbols, N-K the number of repair symbols, T the symbol length, and P the number of symbols per packet.

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

4.3. 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 precoded H.264 NAL unit sequence can be transmitted taking into account timing information. We will restrict ourselves to ideal erasure codes as the performance of all considered codes is equal to or only marginally worse than that of ideal codes and we save the extra burden of code 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. The video encoding parameter selection results in an IDR frequency of 10 seconds which seems reasonable. Flexibility in the video encoding is provided by allowing to adapt the bitrate R_v including packetization overhead for NAL headers as well as the MB intra-update ratio specified by p_{NALU}. Specifically, we have selected operation points which result in application layer error rates $p_{\text{AL}} = \{0, 0.1, \ldots, 2, 3, \ldots, 20\}\%$ for each of the systems presented in Figure 7. The video is encoded with a VBR rate control to match the application layer throughput η_{AL}. Note that the maximum delay constraint of $\Delta = 5$ second is never exceeded. In addition, we might apply fragmentation of NAL units to obtain RTP packets of size 300 bytes and 600 bytes. Also, FMO is included and we restrict ourselves to two slice groups ordered in checkerboard pattern. The channel is again assumed to support 64 kbp/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 a certain target code rate r_t is applied. The RLC-PDUs are transmitted with a TTI of 80 ms, for comparison also one result with TTI = 10 ms 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 = 10 ms, $P = 6$. In addition, header compression is assumed such that PDCP/IP/UDP header is reduced to 10 bytes.

Let us first investigate the case when the loss rate is equal to 1%. For all investigated parameter settings we observe that for low throughput the FEC is sufficient to receive error-free video such that only the distortion caused by the encoding process matters. The reduced compression efficiency due to

FMO is observed, the single slice mode as well as the fragmentation operates in all cases with the same encoded bitstream. A TTI equal to 10 ms 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. With increasing throughput the quality increases as long as 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 [40]. 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 MB updates. One can observe that without any FEC—represented by the end points in the graphs— FMO performs best. Therefore, we conclude 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.

5. DESIGN WITH ADVANCED TRANSPORT LAYER FEATURES

5.1. Retransmission protocols in wireless systems

In point-to-point connections, usually the communication setup is bidirectional. 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 depends on different factors such as the TTI as well as the syntax and semantics of retransmission request messages. If designed appropriately, retransmission requests can be conveyed to the receiver within only very few TTIs. The acknowledged mode can be further distinguished in a persistent mode applying retransmissions until the radio block is correctly received and a nonpersistent mode applying only a limited number of retransmissions, but resulting in residual error rates. Obviously, retransmissions can also be carried out above the IP level. A selective retransmission scheme has been proposed [41] which allows retransmitting RTP packets. In combination with arbitrary slice ordering (ASO) as supported by H.264 even out-of-order delivered NAL units might be decoded in time. On application level, the transmitter might also decide to resend a lower quality, but also a lower rate, representation of the requested RTP packet if it contains a VCL NAL unit. This feature is supported in H.264/AVC by the application of redundant coded slices and pictures which can be sent instead of high rate primary frame. Finally, it is worth to mention that the majority of commercial IP-based video streaming employs TCP for transport layer services, mainly due to the high penetration of this reliable protocol. However, it is also well know that TCP is not capable of dealing with wireless losses as it is optimized for congestion awareness [42]. If TCP is applied to transmit video data reliably, it is necessary that the link layer provides sufficient QoS.

5.2. System design guidelines

In general, automatic repeat request protocols can provide low error rates or even completely reliable services with high efficiency. However, the application of retransmissions is obviously restricted to the case where a feedback channel is available. In addition, the retransmissions generally result in delay jitter which can be undesirable or even unacceptable for some applications. If the retransmissions are applied to the RLC layer, then with appropriate setting of the initial delay and receiver buffer size a QOS comparable to an error-free constant bitrate channel can be guaranteed with only slightly increased initial delay [43]. If this technique is not sufficient, *adaptive media playout* might be applied which allows a streaming media client, without the involvement of the server, to control the rate at which data is consumed by the playout process [44]. The applicability of retransmission protocols above the IP layer to services with less stringent delay constraints has been proven [45], but its inferiority when compared to link layer retransmission protocols will be shown in some simulation results. However, RTP retransmission is still found useful to combat packet losses happening in elements of the transmission system other than the radio access link. The work in [45] also provides a flexible framework to allow *rate-distortion optimized packet scheduling*. This can be supported if media streams are pre-encoded with appropriate packet dependencies such that selective retransmissions for higher priority data units can be applied.

For applications where the data is generated online, for example, in case of conversational video, live streaming, or live broadcasting, the sending time of the data is usually closely coupled to the display time. We refer to this transmission mode as timestamp-based streaming (TBS). However, in case when pre-encoded data is transmitted and the decoder buffer is sufficiently large, one can transmit data earlier than its nominal sending time. This so-called ahead-of-time streaming (ATS) or progressive transmission allows better exploitation of the channel, but usually needs to be combined with some TCP-like congestion control. ATS can be even extended by transmitting more important data earlier which, for example, allows more retransmissions for this important data or providing more robustness against delay jitter [46]. 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 [47].

5.3. 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 UM, namely, acknowledged mode (AM) on the RLC layer with

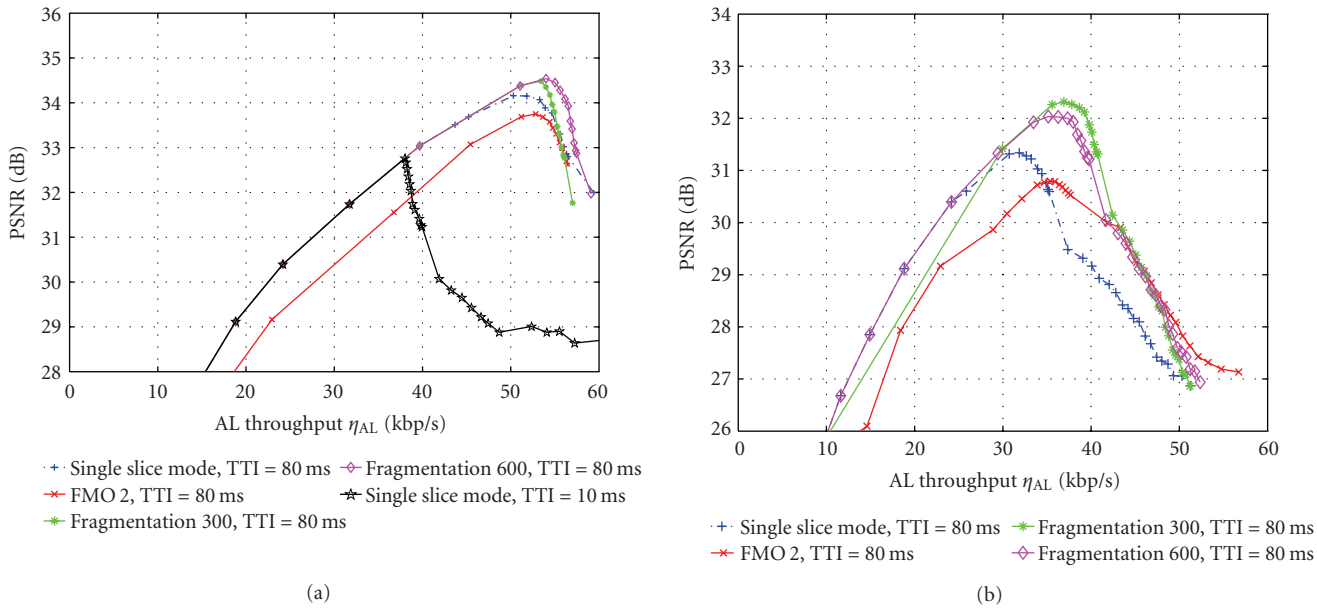


FIGURE 7: Average peak signal-to-noise ratio (PSNR) over the application layer throughput η_{AL} for different system designs; transmission time intervals (TTI); and flexible macroblock ordering (FMO).

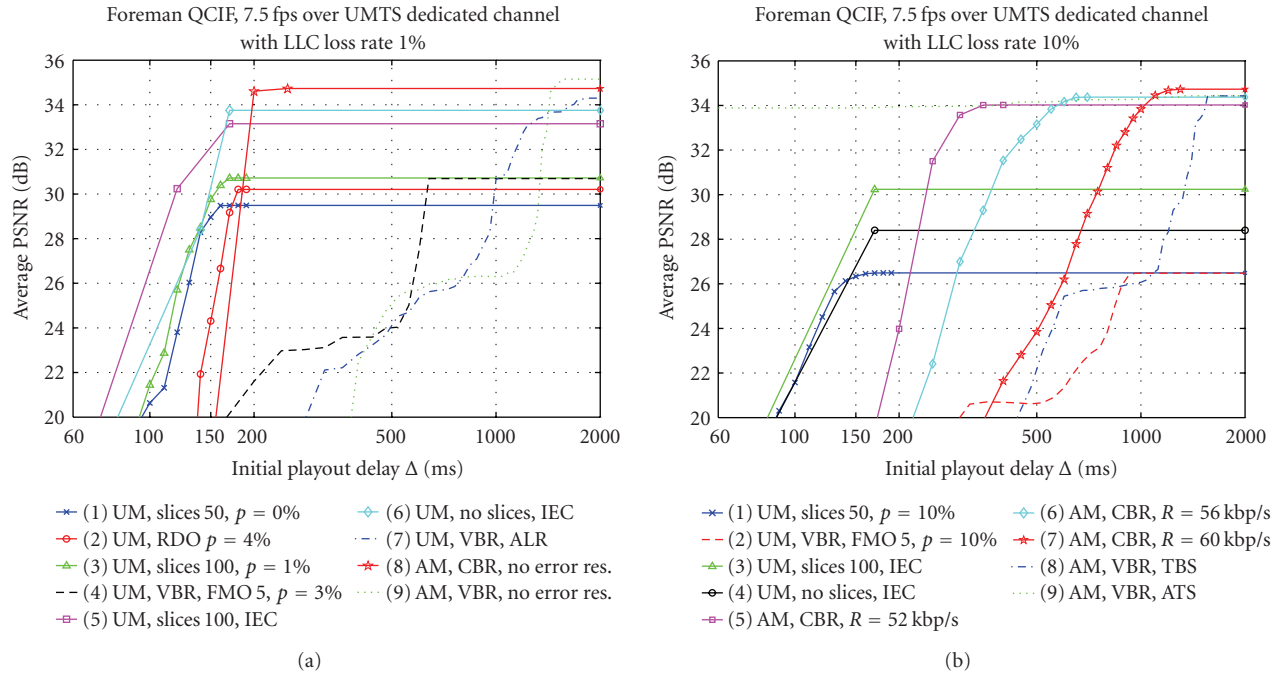


FIGURE 8: Average PSNR for different advanced video transport systems over initial playout delay for UMTS dedicated channel with link layer error rates of 1% and 10%.

persistent and nonpersistent modes, 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.

For the 1% RLC-PDU loss rate, graph (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 if delay does not matter, it provides better performance than any other scheme relying on methods in the video layer. Graphs (8) and (9) show the performance of CBR encoded video and VBR encoded video, respectively, with matching bitrates for the acknowledged mode. The performance of the CBR mode is excellent even for lower delays, but at least

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 kbp/s UMTS transmission scenario			Delay	PSNR	Delay	PSNR
Download-and-play	VBR, no error res.	ATS,	> 1.5 s	35.2 dB	> 1 s	34.4 dB
On-demand streaming	Playout buffering	AM on RLC			> 10 s	35.1 dB
Live streaming	CBR/VBR, no error res.	TBS,	> 250 ms	34.7 dB	> 400 ms	34.0 dB
	Playout buffering	AM on RLC			> 1.5 s	34.7 dB
Broadcast	VBR, regular IDR, no other error resilience	FEC, long TTI, Fragmentation	> 5 s	34.5 dB	> 5 s	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

200 ms of initial playout delay must be accepted which makes the applicability for conversational modes critical, but not infeasible, if the system supports fast retransmissions. We also observe 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 playout delays well over 300 ms according to graph (5) with CBR and bitrate 52 kbp/s. Interestingly, if the initial playout delay is increased, one can also support higher bitrates resulting in higher quality. This behavior 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 [48]. Finally, graphs (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 cannot be generated online. In addition, in practical systems some kind of startup delay might occur due to TCP-like congestion control. It is also worth noting that the performance of video over the 10% link layer loss bearer does not differ significantly from the 1% one if the initial playout delay constraints are not really stringent.

6. SUMMARY: SYSTEM DESIGN GUIDELINES

The obtained results allow comparing different options for different applications. A summary of proposed video and transport features for the video test sequence Foreman over a 64 kbp/s UMTS link with RLC-PDU loss rates of 1% and 10% is provided in Table 2. For download-and-play as well as on-demand streaming applications with initial playout delays beyond one to two seconds, the video should mainly be encoded for compression efficiency, that is, relatively relaxed variable bitrate (VBR) rate control and no explicit error resilience features. The reliability should be added in the link layer by running acknowledged mode (AM) on the RLC link layer. The resulting delay jitter can be compensated with playout buffering. Ahead-of-time streaming (ATS) can be applied if the receiver buffer has sufficient size. For higher error rates, the quality even scales with the initial playout delay, as the jitter can be better compensated for larger delays. For live applications, timestamp-based streaming (TBS) must be applied. For lower requested delays in range of 250–500 ms, CBR-like rate control is also preferable. Error resilience is still not very essential as the acknowledged mode in general provides sufficient QoS. For broadcast applications without any feedback, we propose to apply additional FEC. This can be accomplished by longer TTIs in the physical layer and/or application layer FEC. In addition, we suggest using regular IDR frames for random access and error resilience, but a relatively relaxed rate control. For low RLC-PDU loss rates, the FEC-based scheme is almost as good as the one relying on feedback mechanisms. However, for higher loss rates the acknowledged mode with RLC layer retransmission outperforms MBMS FEC by about 2–3 dB. In any case, broadcast systems with application layer FEC add significant delay.

For low-delay applications, the video will apply CBR-like rate control and the transport and link layer basically must operate in a transparent or unacknowledged mode (UM) without any retransmission or long FEC schemes. The video application itself must take care to provide sufficient robustness. In case that feedback is not available or only limited to reporting statistics, for example, in conferencing applications, more frequent intra-MB updates based on robust mode decision as well as slice-structured coding are proposed. However, in this case compared to the acknowledged mode significant degradations in the video quality must be accepted, especially if the RLC-PDU loss rates are high. Therefore, the physical layer must provide sufficient QoS to support these applications. For video telephony, the fast feedback channel can be exploited for interactive error control (IEC). No additional means of error resilience are necessary. In this case and for low loss rates, the achieved video quality is significantly better, about 3 dB, when compared to video error resilience without feedback. The degradation compared to reliable download-and-play applications is only about 1.5 dB.

7. CONCLUSIONS

In this work we have shown how the robust coding features of H.264/AVC in wireless transmission environments can be successfully and appropriately employed. 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 any networks. The selection and combination of different features strongly depend on the system and application constraints, namely, bitrates, maximum tolerable playout delays, error characteristics, online encoding possibility, as well as availability of feedback and cross-layer information. Although the standardization process for H.264/AVC is finalized, the freedom at the encoder as well as the combination with transport modes such as FEC and retransmission strategies promises have optimization potential. In general, 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, cross-layer design, feedback exploitation, and error concealment is necessary to fully understand the potential of H.264/AVC in wireless environments. However, integration of transport protocol and wireless options into the design is needed, rather than assuming QoS-unaware link and transport layers.

ACKNOWLEDGMENTS

The author would like to thank Thomas Wiegand, Stephan Wenger, Günther Liebl, Miska M. Hannuksela, Waqar Zia, Imre Varga, and Ingo Viering for useful discussions on the subject of this work. Especially appreciated are the comments of the anonymous reviewers and editors, in particular Adriana Dumitras.

REFERENCES

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

[2] R. Talluri, "Error-resilient video coding in the ISO MPEG-4 standard," *IEEE Communications Magazine*, vol. 36, no. 6, pp. 112–119, 1998.

[3] Y. Wang and Q.-F. Zhu, "Error control and concealment for video communication: a review," *Proceedings of the IEEE*, vol. 86, no. 5, pp. 974–997, 1998.

[4] Y. Wang, G. Wen, S. Wenger, and A. K. Katsaggelos, "Review of error resilient techniques for video communications," *IEEE Signal Processing Magazine*, vol. 17, no. 4, pp. 61–82, 2000.

[5] Q.-F. Zhu and L. Kerofsky, "Joint source coding, transport processing, and error concealment for H.323-based packet video," in *Visual Communications and Image Processing (VCIP '99)*, vol. 3653 of *Proceedings of SPIE*, pp. 52–62, San Jose, Calif, USA, January 1999.

[6] S. Wenger, "H.264/AVC over IP," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 645–656, 2003.

[7] 3GPP Technical Specification 3GPP TS 26.234, Rel-6, "Transparent end-to-end packet-switched streaming service (PSS); protocols and codecs."

[8] 3GPP Technical Specification 3GPP TS 26.346, Rel-6, "Multimedia multicast and broadcast service (MBMS); protocols and codecs."

[9] 3GPP Technical Specification 3GPP TS 26.110, "Circuit-switched video telephony (3G-324M)."

[10] 3GPP Technical Specification 3GPP TS 26.235 and 26.236, "Packet-switched conversational multimedia applications."

[11] 3GPP Technical Specification 3GPP TS 26.140 "Multimedia messaging service (MMS); media formats and codecs ".

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

[13] G. J. Sullivan and T. Wiegand, "Video compression - from concepts to the H.264/AVC standard," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 18–31, 2005.

[14] T. Stockhammer, M. M. Hannuksela, and T. Wiegand, "H.264/AVC in wireless environments," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 657–673, 2003.

[15] S. Kumar, L. Xu, M. K. Mandal, and S. Panchanathan, "Error resiliency schemes in H.264/AVC video coding standard," *Journal of Visual Communication and Image Representation*, vol. 17, no. 2, pp. 425–450, 2006, (Special issue on H.264/AVC video coding standard).

[16] A. Ortega and K. Ramchandran, "Rate-distortion techniques in image and video compression," *IEEE Signal Processing Magazine*, vol. 15, no. 6, pp. 23–50, 1998.

[17] T. Wiegand, M. Lightstone, T. G. Campbell, D. Mukherjee, and S. K. Mitra, "Rate-distortion optimized mode selection for very low bit rate video coding and the emerging H.263 standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, no. 2, pp. 182–190, 1996.

[18] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. J. Sullivan, "Rate-constrained coder control and comparison of video coding standards," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 688–703, 2003.

[19] T. Stockhammer, D. Kontopodis, and T. Wiegand, "Rate-distortion optimization for H.26L video coding in packet loss environment," in *Proceedings of the 12th International Packet Video Workshop (PVW '02)*, Pittsburgh, Pa, USA, April 2002.

[20] R. Zhang, S. L. Regunathan, and K. Rose, "Video coding with optimal inter/intra-mode switching for packet loss resilience," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 6, pp. 966–976, 2000.

[21] T. Wiegand, N. Färber, K. Stuhlmüller, and B. F. Girod, "Error-resilient video transmission using long-term memory motion-compensated prediction," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 6, pp. 1050–1062, 2000.

[22] G. Cote, S. Shirani, and F. Kossentini, "Optimal mode selection and synchronization for robust video communications over error-prone networks," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 6, pp. 952–965, 2000.

[23] H. Yang and K. Rose, "Recursive end-to-end distortion estimation with model-based cross-correlation approximation," in *Proceedings of International Conference on Image Processing (ICIP '03)*, vol. 2, pp. 469–472, Barcelona, Spain, September 2003.

[24] B. F. Girod and N. Färber, "Feedback-based error control for mobile video transmission," *Proceedings of the IEEE*, vol. 87, no. 10, pp. 1707–1723, 1999.

[25] S. Fukunaga, T. Nakai, and H. Inoue, "Error resilient video coding by dynamic replacing of reference pictures," in *Proceedings of Global Telecommunications Conference (GLOBECOM '96)*, vol. 3, pp. 1503–1508, London, UK, November 1996.

[26] M. Wada, "Selective recovery of video packet loss using error concealment," *IEEE Journal on Selected Areas in Communications*, vol. 7, no. 5, pp. 807–814, 1989.

[27] Y. Tomita, T. Kimura, and T. Ichikawa, "Error resilient modified inter-frame coding system for limited reference picture memories," in *Proceedings of Picture Coding Symposium (PCS '97)*, pp. 743–748, Berlin, Germany, September 1997.

[28] T. Nakai and Y. Tomita, "Core experiments on feedback channel operation for H.263+," ITU-T SG15 LBC 96-308, November 1996.

[29] M. Karczewicz and R. Kurçeren, "The SP- and SI-frames design for H.264/AVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 637–644, 2003.

[30] C.-Y. Hsu, A. Ortega, and M. Khansari, "Rate control for robust video transmission over burst-error wireless channels," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 5, pp. 756–773, 1999, Special issue on multimedia network radios.

[31] T. V. Lakshman, A. Ortega, and A. R. Reibman, "VBR video: tradeoffs and potentials," *Proceedings of the IEEE*, vol. 86, no. 5, pp. 952–973, 1998.

[32] 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, Calif, USA, September 2001.

[33] Y-K. Wang, M. M. Hannuksela, V. Varsa, A. Hourunranta, and M. Gabbouj, "The error concealment feature in the H.26L test model," in *Proceedings of IEEE Internantional Conference on Image Processing (ICIP '02)*, vol. 2, pp. 729–732, Rochester, NY, USA, September 2002.

[34] 3GPP Technical Specification 3GPP TS 25.346, "Introduction of the Multimedia Broadcast/Multicast Service (MBMS) in the Radio Access Network (RAN); Stage 2".

[35] T. Stockhammer, G. Liebl, H. Jenkac, and W. Xu, "Flexible Outer Reed-Solomon Coding on RLC Layer for MBMS over GERAN," in *Proceedings of IEEE Semiannual Vehicular Technology Conference (VTC '04)*, Milano, Italy, May 2004.

[36] T. Stockhammer, G. Liebl, and H. Jenkac, "H.264/AVC video transmission over MBMS," in *Proceedings of IEEE International Workshop on Multimedia Signal Processing*, Siena, Italy, September 2004.

[37] T. Paila, M. Luby, R. Lehtonen, V. Roca, and R. Walsh, "FLUTE - File Delivery over Unidirectional Transport," IETF RFC3926, October 2004.

[38] 3GPP S4-050090, "Report of FEC selection for MBMS," SA-WG4, March 2005.

[39] A. Shokrollahi, "Raptor codes," Digital Fountain, DR2003-06-001, June 2003.

[40] 3GPP S4-050090, "Alignment of H.264/AVC NAL Units for MBMS," Siemens, February 2005.

[41] J. Rey, D. Leon, A. Miyazaki, V. Varsa, and R. Hakenberg, "RTP Retransmission Payload Format," Internet Draft, draft-ietf-avt-rtp-retransmission-11.txt, March 2005.

[42] P. Mehra and A. Zakhor, "TCP-based video streaming using receiver-driven bandwidth sharing," in *Proceedings of the 13th International Packet Video Workshop (PVW '03)*, Nantes, France, April 2003.

[43] T. Stockhammer, H. Jenkac, and G. Kuhn, "Streaming video over variable bit-rate wireless channels," *IEEE Transactions on Multimedia*, vol. 6, no. 2, pp. 268–277, 2004.

[44] M. Kalman, E. G. Steinbach, and B. F. Girod, "Adaptive media playout for low-delay video streaming over error-prone channels," *IEEE Transactions on Circuits and Systems on Video Technology*, vol. 14, no. 6, pp. 841–851, 2004.

[45] P. A. Chou and Z. Miao, "Rate-distortion optimized streaming of packetized media," to appear in *IEEE Transactions on Multimedia*.

[46] T. Stockhammer, M. Walter, and G. Liebl, "Optimized H.264-based bitstream swiching for wireless video streaming," in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME '05)*, Amsterdam, The Netherlands, July 2005.

[47] G. Liebl, H. Jenkac, T. Stockhammer, C. Buchner, and A. Klein, "Radio link buffer management and scheduling for video streaming over wireless shared channels," in *Proceedings of International Packet Video Workshop (PVW '04)*, Irvine, Calif, USA, July 2004.

[48] J. Ribas-Corbera, P. A. Chou, and S. Regunathan, "A generalized hypothetical reference decoder for H.264/AVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 674–687, 2003.

Thomas Stockhammer has been working at the Munich University of Technology, Germany, and was a Visiting Researcher at Rensselear Polytechnic Institute (RPI), Troy, NY and at the University of San Diego, California (UCSD). He has published more than 70 conference and journal papers, is a Member of different program committees, and holds several patents. He regularly participates in and contributes to different standardization activities, for example, JVT, IETF, 3GPP, and DVB, and has coauthored more than 100 technical contributions. He is the Acting Chairman of the video ad hoc group of 3GPP SA4. He is also the cofounder and CTO of Novel Mobile Radio (NoMoR) Research, a company working on the simulation and emulation of future mobile networks. Since 2004, he has been working as a research and development Consultant for Siemens Mobile Devices, now BenQ mobile in Munich, Germany. His research interests include video transmission, cross-layer and system design, forward error correction, content delivery protocols, rate-distortion optimization, information theory, and mobile communications.