

# CHANNEL RATE ALLOCATION FOR ERROR-ROBUST PACKET TRANSMISSION OF SNR-SCALABLE DPCM-CODED VIDEO

*Till Halbach*

Department of Telecommunications  
Norwegian University of Science and Technology (NTNU)  
Trondheim, Norway

*Thomas R. Fischer*

School of Electrical Engineering and Computer Science  
Washington State University (WSU) Pullman  
WA, USA

## ABSTRACT

An SNR-scalable hybrid video coding scheme with a quality constraint is considered. The encoded digital data are conveyed as packets over an error-prone channel. We derive a low-complexity algorithm which assigns the best code to each packet, based on a limited set of codes, and accounting for channel conditions, code properties, the data's importance, and inter-packet dependencies. A highly error-resilient transmission is hereby achieved, the image quality being equal to approximately high-quality layer  $SNR$  for a low error probability (0.002). The quality decrease with degrading channel conditions is only moderate.

## 1. INTRODUCTION

Digital video is in e.g. streaming applications often transmitted in packets over channels like RTP/UDP or ATM. Within and also across packet boundaries, single bit errors are likely to be disastrous for the decoding process due to spatial and temporal predictions during encoding, aiming at exploitation of redundancy inherent to natural image sequences. This paper describes a method to adapt the channel encoding scheme by means of unequal error protection to both the source's importance in terms of SNR and constraints given by the channel, i.e. a rate limitation. The method can be classified as a joint source channel coding technique with the objective of maximization of parameters like frame SNR and frequency under certain channel conditions.

The article is structured as follows. The problem formulation in Sec. 2 is based on the framework of the so-called Baseline profile of the upcoming standard H.264/H.26L [1]. Two solutions of the problem are then derived in Sec. 3, and Sec. 4 discusses simulation results in detail.

---

At the time of writing, Till Halbach was on sabbatical leave at WSU in Pullman. The authors can be contacted by [halbach@tele.ntnu.no](mailto:halbach@tele.ntnu.no) and [fischer@eecs.wsu.edu](mailto:fischer@eecs.wsu.edu). This work was supported by grants of the Norwegian Research Council and the Department of Telecommunications at NTNU.

## 2. PROBLEM FORMULATION

Consider a digital progressive-scan image sequence of  $F$  frames/pictures having some spatial resolution (e.g.  $352 \times 288$  for CIF-size material). The first frame in this group of pictures (GOP) is encoded in so-called INTRA (I) mode. The GOP is closed, i.e. there is no temporal prediction across I frames, enabling instantaneous access to the bit stream during decoding and also stopping temporal error propagation. The parameter  $F$  is chosen with regard to random-access and error resilience requirements. The remaining frames in the GOP are encoded in INTER (P) mode, that is employing forward temporal prediction based on a single reference frame, the previously encoded frame. One frame is further split into slices, non-overlapping regions of adjacent MBs, macroblocks, hereby filling the whole frame. A MB is a 3-tuple of one  $16 \times 16$  block of luminance samples and two  $8 \times 8$  blocks of chrominance samples. There are  $X$  slices in the horizontal direction and  $Y$  slices in the vertical direction. Slices are independently decodeable, which means that there neither be sample prediction in I mode nor motion vector prediction in P mode across slice boundaries. The encoder further generates  $L$  bit streams, each of which includes a certain quality layer, a BL and one or several ELs. A particular slice in frame  $f$  and level  $l$  with the spatial position  $(x, y)$  is then uniquely specified by the 4-tuple  $(f, l, x, y)$ , and is referred to in the following as segment or cell.

Next, each encoded segment is treated as an entity and passed to the channel encoder as a packet of variable size  $R_i$ , resulting in  $P_v = F \cdot L \cdot X \cdot Y$  packets for one GOP. The packet size is source-dependent and can be controlled by putting a quality constraint on the encoder; a quality parameter  $QP$  is passed to the quantization process to select the appropriate uniform quantizer with desired step size. At the channel encoder, an 8-bit code specifier  $CS_{i+1}$  is added to every packet  $i$ , which specifies the channel code of the next transmitted packet  $i + 1$ . This scheme assumes a certain channel code for the first conveyed packet and requires that packets are received in order. A 16-bit cyclic redundancy

check (CRC) [2] is computed over both encoded data and code specifier for detection of (residual) bit errors and appended at the end of the packet. Finally, all data is channel-encoded, adding  $C_i$  redundancy bits to each packet, and passed to the channel which can be modeled as a binary symmetric channel (BSC) with bit error rate (BER),  $\epsilon \geq 0$ . See Fig. 1 for an illustration of the architecture of a packet.

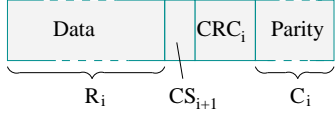


Figure 1: Packet structure

The encoder exploits spatial and temporal redundancies inherent to a natural-image sequence and creates a data dependency by applying predictive coding to the sequence. This means, however, that in case of transmission errors, even a single residual bit error in the data of one segment is likely to propagate through all segments which depend by prediction on this particular segment. Therefore, all dependent segments after error detection are discarded, including the contaminated cell. This applies to all levels with higher quality than the affected level and all subsequent frames until the end of the GOP is reached; for illustration see Fig. 4(b). In order to describe the temporal dependency among segments introduced by motion estimation/compensation (ME/MC), we have to put some constraints on the source encoder.

Even though specification of the search area for ME limits the prediction possibilities somewhat, the source dependency does not allow a simple description of segment dependencies. There are two ways to navigate around this problem; both require that the outer slice boundaries be treated as frame boundaries, i.e. reference samples outside the respective slice(s) are extrapolated for ME. Scenario A makes use of as few (large) slices as possible. The reduction of error propagation comes at the cost of reduced coding efficiency. Fig. 2 (left) illustrates the data dependency. In scenario B, slice boundaries overlap across frames, see Fig. 2 (right). There should be as many (small) slices as possible. This approach should not decrease coding efficiency as much as scenario A (and mainly due to an increase in side information caused by e.g. the slice header), however, the temporal error may spread spatially over many frames; in the example, the slice dependency region increases by one MB in each direction for each picture back in time. In the following, scenario A is assumed.

Let  $P_e(R_{f,l,x,y}, \Gamma_{f,l,x,y}, \epsilon)$  be the probability of at least one residual error in the  $i^{\text{th}}$  decoded packet which is protected by the channel code  $\Gamma_i$ . The expected distortion of

this particular segment can then be computed by

$$\tilde{D}_{f,l,x,y} = \sum_{g=1}^f \sum_{h=1}^l D_{g,h,x,y} P_e(R_{g,h,x,y}, \Gamma_{g,h,x,y}, \epsilon) \prod_{i=0}^{g-1} \prod_{j=0}^{h-1} (1 - P_e(R_{j,i,x,y}, \Gamma_{j,i,x,y}, \epsilon)), \quad (1)$$

where  $P_e(R_{j,i,x,y}, \Gamma_{j,i,x,y}, \epsilon) = 0$  with  $j = 0$  or  $i = 0$ . The true distortion  $D_{g,h,x,y}$  of a cell is here set equal to the accumulated error made by discarding the current and all subsequent packets. However, also other distortion measures are possible. The average distortion of frame  $f$  at level  $l$  is  $\bar{D}_{f,l} = \sum_{x=1}^X \sum_{y=1}^Y \tilde{D}_{f,l,x,y}$ , and the expected distortion of the GOP  $\tilde{D}_{\text{GOP}} = \tilde{D}_{F,L}$ .

Next, given a set of channel codes  $\mathcal{C}$ , we seek to assign each packet a channel code  $\Gamma_{f,l,x,y} \in \mathcal{C}$  such that the overall expected distortion of a particular GOP is as small as possible. With other words, we aim at limiting the impact of channel errors to a minimum by our code selection. Fig. 3 shows the whole framework. Channel coding adds  $C_i$  parity bits to the packet, resulting in the packet length  $L_i = R_i + 24 + C_i$ ; thus, we have a channel rate allocation problem and seek to  $\min_{\Gamma_i \in \mathcal{C}} \tilde{D}_{\text{GOP}}$  subject to the rate constraint

$$\sum_{f=1}^F \sum_{l=1}^L \sum_{x=1}^X \sum_{y=1}^Y (R_{f,l,x,y} + 24 + C_{f,l,x,y}) \leq r^{-1}(R + 24P_v), \quad (2)$$

where  $C_{f,l,x,y}$  are the redundancy bits generated by code  $\Gamma_{f,l,x,y}$ ,  $r$  is the total channel code rate (e.g. 1/3), and  $R$  is the overall bit budget consumed by source encoding. The channel rate is determined by taking into account the data's importance which is reflected in Eq. 1 by the distortion measure  $D_i$ . Also, the source code rate is represented by the dependency of  $P_e$  on  $R_i$ . The GOP's segments have to be transmitted sequentially with index  $i$  over the channel in order of increasing level index  $l$  and increasing frame number  $f$ , such that first the low-level data of a complete encoded picture is available at the decoder, then all levels of that respective frame, and finally all remaining frames.

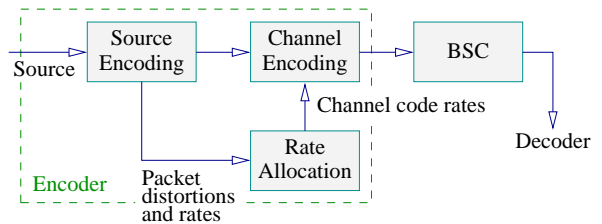


Figure 3: System overview



Figure 2: The dark-gray shaded segments are recursively predicted from the light-gray shaded segments. Left: Scenario A with  $F = 2$ ,  $L = 3$ ,  $X = 2$ ,  $Y = 3$ . Right: Scenario B with  $F = 3$ ,  $L = 3$ ,  $X = 6$ ,  $Y = 5$ . Both scenarios are examples for segment structuring of the QCIF image format.

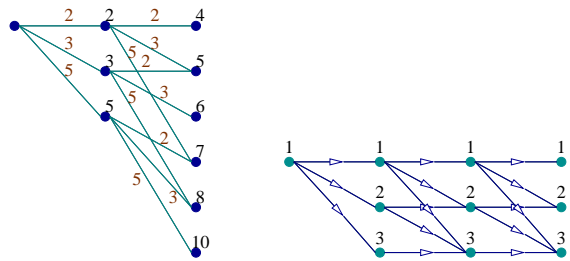
### 3. CHANNEL CODE RATE DETERMINATION

The problem of assigning the optimum channel codes to each of the packets is a discrete optimization problem and can be solved using a brute-force search algorithm. However, considering the typical values  $F = 30$ ,  $L = 4$ ,  $X = 4$ , and  $Y = 6$ , the number of packets and hereby the computational burden becomes very large – the number of combinations  $|\mathcal{C}|^P$  is impossible to trace. In order to allow a lower-complexity approach (which generally provides a sub-optimum solution), the Viterbi algorithm, we convey the variable-length bit stream generated by the source encoder by means of fixed-length packets. With other words,  $P_f$  packets are sent  $P_v$  data cells. A data cell may hereby be split and transported by more than one channel packet. The length of a channel packet is in this work chosen to be  $L_i = S = 4136$  bits (517 bytes). The number of payload/data bits  $R_i$  of a packet depends then on the chosen channel code  $\Gamma_i$ . Also, we can define the encoder trellis states as  $\sum_i R_i$  along the path. The rate constraint expressed by Eq. 2 is hereby turned into  $\sum_{i=1}^{P_f} (R_i + 24 + C_i) = S P_f$  and is inherent to the trellis, see Fig. 4(a). That is, the rate constraint is given by the  $P_f$ , number of channel packets. It is required for successful decoding that only complete cells be accepted as decoder input. Padding is used as necessary such that all packets have the same length. By means of the mapping  $(f, l, x, y) \mapsto i$  of cell specifiers to packet index, the cost metric of a node in stage  $s$  then becomes

$$\tilde{D}_i = \sum_{i=1}^s D_i P_e(\Gamma_i, \epsilon) \prod_{j=0}^{i-1} (1 - P_e(\Gamma_j, \epsilon)), \quad (3)$$

with  $P_e(\Gamma_0, \epsilon) = 0$ . All out-going branches of a node correspond to the available channel codes. All in-coming branches of a node correspond to the same number of already channel-encoded data bits. An example of the channel encoding trellis is given in Fig. 4(a) for three channel codes

with input rates  $\{2, 3, 5\}$ . At the last stage/packet, the state/node of overall minimum error has to be traced back along the in-coming branches with the smallest local error metric. The complexity of the Viterbi algorithm grows moderately according to  $O(|\mathcal{C}| \cdot P_f^2)$  because the number of states in each stage increases in a linear manner.



(a) Encoder trellis with initializing stage. A stage corresponds to a transmitted packet.

(b) Decoder trellis for two layers. A stage corresponds to a received frame. In state 3 all layers have been lost. In state 2 the high-quality layer has been lost. Both layers have been received error-free in state 1.

Figure 4: Examples of encoder and decoder trellises.

A set of eight turbo codes is employed for channel encoding and error correction in this work, the code rates being  $r_i = k_i/n$  with  $n = 12$  and  $k_i = \{4, 5, 6, 8, 9, 10, 11, 12\}$ . For a 517-byte packet size, this results in source byte lengths of 169, 212, 255, 341, 384, 427, 470, and 513. The codes consist of punctured parallel concatenated recursive convolutional codes as recommended in [3] and [4]. Based on these rates, the probabilities  $P_e$  of a cell having at least one bit error after 20 turbo decoder iterations have been computed in extensive Monte-Carlo simulations of 10,000 blocks in [5] and can hence be tabulated for use in Eq. 3.

#### 4. TESTING, RESULTS, AND DISCUSSION

To test the aforementioned concept, a preliminary codec version of the upcoming international video compression standard H.264/MPEG-4 AVC is considered [1]. The system design has been extended by limited ME/MC as specified in Sec. 2, scenario A, and SNR scalability [6].

| Video               | $\epsilon$ |       |       |       |       |
|---------------------|------------|-------|-------|-------|-------|
|                     | 0.002      | 0.008 | 0.02  | 0.08  | 0.12  |
| <i>Foreman</i>      | 41.13      | 40.96 | 40.55 | 39.80 | 38.97 |
| <i>Container</i>    | 42.41      | 41.89 | 41.21 | 40.62 | 40.01 |
| <i>Silent (CIF)</i> | 41.86      | 40.75 | 39.90 | 38.45 | 37.94 |

Table 1: Expected distortion ( $PSNR$  in dB) at the decoder as a function of the channel BER,  $\epsilon$ , after transmission over a BSC at a total rate 0.99 bpp.

The code assignment results are presented in Tab. 1. A GOP compound of 30 frames is taken from an image sequence and coded at the three  $QPs$  40, 30, 20, resulting – with *Foreman* – in the average luminance  $SNRs$  28.75, 35.45, and 41.76 dB. The corresponding source bit rates are 45.91, 109.93, and 373.17 Kbit/s. The videos have a YCbCr color space with 4:2:0 chrominance subsampling and are of size QCIF if not noted otherwise. We set one frame slice equal to one frame.

We observe that the approach favours the early frames within a GOP: On the average nine complete frames, i.e. all layers, out from 30 are transmitted with quite strong channel codes (4/12, . . . , 10/12), the remaining frames are ignored. This is because the number of channel packets is determined by termination of the algorithm when all bits of the compressed sequence are used up by weak codes which allow a maximum of source bits in one packet. Since the number of channel packets is constant, less source packets are conveyed with strong codes which are found to be optimal in a Viterbi algorithm sense for the early frames of a GOP.

The expected decoder distortion of the received frames is very high due to the assignment of the 4/12 code to up to 74% of all packets. This means further that only few packets can become more protected by stronger codes. As a result, the performance drops only about 2.5–3.1 dB with increasing  $\epsilon$ . The CIF sequence shows a larger difference of about 3.8 dB, which is because of the fact that more packets are needed for transmission of a single frame. The developed system, combining SNR scalability and a nearly optimum code assignment scheme, achieves a quite good objective performance, i.e.  $PSNR$ , also for high channel BERs larger than 0.1. By including a 12/12-rate code in the set of channel codes, it is assured that maximum quality can be yielded towards the error-free case. Even though the degradation in quality is not very large, the code set has to be

chosen carefully. Removal of e.g. the 5/12 code gave more than 1 dB worse performance. This means that only very few codes are of major interest at a certain  $BER$ . The result is consistent with [5] where the codes are sequentially distributed over the range of channel  $BERs$ .

#### 5. CONCLUSIONS AND OUTLOOK

By applying unequal error protection (UEP) to the scalable H.264 bit stream, sensitive data is encapsulated by much redundancy, and less sensitive data is assigned few protection. The criterion of sensitivity takes into account the distortion introduced by decoding termination after error detection, and hereby the lost data’s spatial, temporal, and layer position. Furthermore, the channel codes are assigned to the packets with regard to the channel statistics. The performance of the system is quite good also in highly error-prone environments, and hence our scheme is a promising candidate in e.g. streaming applications.

It is obvious that the  $D_i$  term can also include decoding strategies like error concealment. E.g. the scheme of simply replacing the current erroneous frame by the last correctly transmitted and decoded frame will typically lead to a strongly reduced distortion. How concealment influences the performance of the code assignment algorithm may thus be an interesting task for future research.

#### 6. ACKNOWLEDGMENTS

The authors wish to thank Brian A. Banister and Tor A. Ramstad for valuable discussions.

#### 7. REFERENCES

- [1] Thomas Wiegand, “Joint Final Committee Draft (JFCD) of joint video specification ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC,” Tech. Rep. D157, ITU-T VCEG | ISO/IEC MPEG (JVT), Aug. 2002.
- [2] G. Castagnoli, J. Ganz, and P. Graber, “Optimum cyclic redundancy-check codes with 16-bit redundancy,” *IEEE Trans. Commun.*, vol. 38, no. 1, pp. 111–114, Jan. 1990.
- [3] Ö. Açikel and W. Ryan, “Punctured turbo-codes for BPSK/QPSK channels,” *IEEE Trans. Commun.*, vol. 47, no. 9, Sept. 1999.
- [4] D. Rowitch and L. Milstein, “On the performance of hybrid fec/arq systems using rate compatible punctured turbo (rcpt) codes,” *IEEE Trans. Commun.*, vol. 48, no. 6, June 2000.
- [5] B.A. Banister, B. Belzer, and T.R. Fischer, “Robust image transmission using JPEG2000 and turbo-codes,” *IEEE Signal Processing Letters*, vol. 9, no. 4, pp. 117–119, Apr. 2002.
- [6] Till Halbach and Thomas R. Fischer, “SNR scalability by transform coefficient refinement for block-based video coding,” Submitted to Visual Communications and Image Processing Conference (Jul. 2003).