AMISP: A Complete Content-Based MPEG-2 Error-Resilient Scheme

Pascal Frossard, *Member, IEEE,* and Olivier Verscheure, *Member, IEEE*

***Abstract*—We address a new error-resilient scheme for broadcast quality MPEG-2 video streams to be transmitted over lossy packet networks. A new scene-complexity adaptive mechanism, namely Adaptive MPEG-2 Information Structuring (AMIS) is introduced. AMIS modulates the number of resynchronization points (i.e., slice headers and intra-coded macroblocks) in order to maximize the perceived video quality, assuming that the encoder is aware of the underlying packetization scheme, the packet loss probability (PLR), and the error-concealment technique implemented at the decoding side. The end-to-end video quality depends both on the encoding quality and the degradation due to data loss. Therefore, AMIS constantly determines the best compromise between the rate allocated to encode pure video information and the rate aiming at reducing the sensitivity to packet loss. Experimental results show that AMIS dramatically outperforms existing structuring techniques, thanks to its efficient adaptivity. We then extend AMIS with a forward-error-correction (FEC)-based Protection algorithm to become AMISP. AMISP triggers the insertion of FEC packets in the MPEG-2 video packet stream. Finally, the performances of the AMISP scheme in an MPEG-2 over RTP/UDP/IP scenario are evaluated.**

***Index Terms*—Adaptive protection, error resilience, FEC, joint source/channel coding, MPEG-2, robust encoding, structuring.**

I. Introduction

BECAUSE of the increasing availability of Internet and ATM networks, packet video is becoming a common support. It is, therefore, important to fully understand the parameters that may affect the quality of the video delivered to the end user, and how to cope with the resulting impairments. Both the encoding and the transmission processes may affect the quality of service. The best quality at the lowest streaming bandwidth can thus only be obtained by optimizing the entire system end-to-end rather than its individual components in isolation [1], [2].

The choice of a compression standard depends mostly on the available transmission or storage capacity, as well as the application requirements. The MPEG-2 standard is an audio–visual standard developed by the International Standards Organization (ISO), together with the International Electrotechnical Commission (IEC) [3]. The video part of MPEG-2 permits data rates up to 100 Mbps and also supports interlaced video formats and a number of advanced features, including those supporting high-definition television (HDTV). MPEG-2 is capable of compressing NTSC or PAL TV-resolution video into an average bit rate of 3–7 Mbps with a quality comparable to analog broadcast TV [4].

Like any other compressed data, compressed video is highly sensitive to data loss (see Section II). Data loss propagates within the sequence and may thus become very annoying to the end user [5]. Error-resilience schemes have been introduced to limit these impairments [6]. These schemes could be roughly classified into three categories [7].

First, error-concealment techniques try to estimate missing video data using information available at the receiver. However, even for the most sophisticated concealment algorithms [8]–[10], important loss of data may lead to annoying degradation. It becomes, therefore, mandatory to minimize missing information. In the second category, the resynchronization or error-localization techniques aim at limiting spatial and/or temporal error propagation [11]–[13]. These techniques, however, do not take the local relevance of video data into account [14]. Finally, in the third category, unequal error-protection schemes try to efficiently recover the missing video information [15]–[17]. They try to minimize degradation due to losses by providing class-based degree of protection. Similar to the resynchronization techniques, the best results are however obtained only with a judicious packet prioritization process [18], [19]. In this category, layered coding [20], [21] and the Multiple Description Coding schemes [22] can be mentioned as the most promising algorithms.

Optimal error-resilient schemes should, however, not only combine techniques of the above three categories, but also exploit the local relevance of video data. Given bit-budget constraints, such a combination is indeed the only way to provide the best video quality. In this paper, we provide an adaptive MPEG-compliant structuring and protection of video data in order to maximize the end-to-end quality of service for given network constraints.

The paper is organized as follows. Section II proposes a brief overview of the MPEG-2 standard, with a focus on the MPEG-2 sensitivity to data loss. The structuring algorithm, namely AMIS, is described in Section III, where experimental results and comparisons are also presented. In Section IV, AMIS is extended with the protection scheme to become AMISP. The performances of AMISP are then evaluated. Finally, concluding remarks are given in Section V.

Manuscript received November 8, 1999; revised May 11, 2001. This paper was recommended by Associate Editor C. W. Chen.

P. Frossard was with the Signal Processing Laboratory, Swiss Federal Institute of Technology, Lausanne, Switzerland. He is now with IBM T. J. Watson Research Center, Hawthorne, NY 10532 USA (e-mail: frossard@us.ibm.com).

O. Verscheure is with the IBM T. J. Watson Research Center, Hawthorne, NY 10532 USA (e-mail: ov1@us.ibm.com).

Publisher Item Identifier S 1051-8215(01)08092-2.

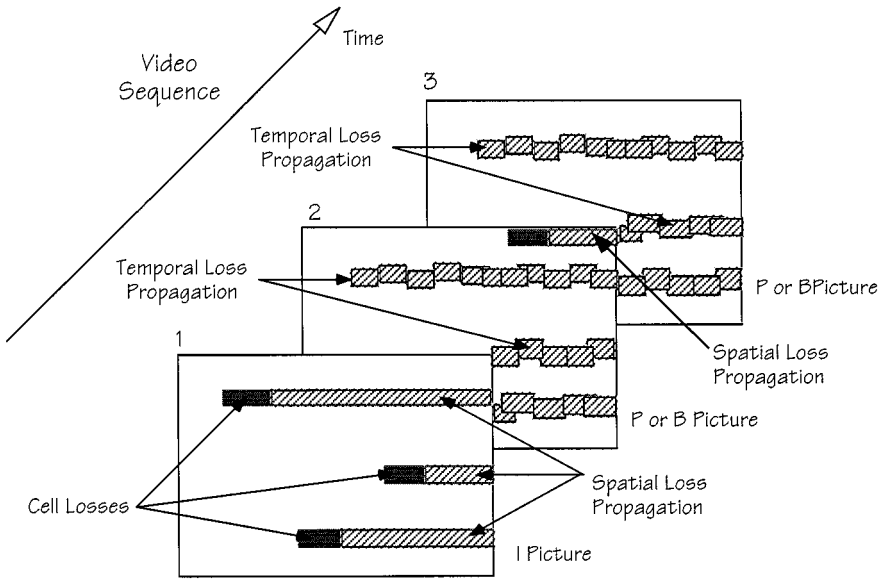

Fig. 1. Data loss propagation in MPEG-2 video streams.

II. MPEG-2 Sensitivity to Data Loss

An MPEG-2 video stream is highly hierarchically structured. The smallest entity defined by the standard is the *block*, which is an area of 8×8 pixels of luminance or chominance. A *macroblock* (16×16 pixels) contains four blocks of luminance samples and two, four, or eight blocks of chrominance samples, depending on the chrominance format. A variable number of macroblocks is encapsulated in an entity called *slice*. As required by the MPEG standard, each new line of macroblocks is the start of a new slice. However, there is no constraint on slice length. Thus, each *picture* is composed of a variable number of slices. To decrease the overhead and hence increase the compression, a slice very often continues all the way to the end of a macroblock line. Slices occur in the bitstream in the order in which they are produced.

Fig. 1 shows how network losses map into visual information losses in different types of MPEG frames (I, P, or B). Data loss spreads within a single picture up to the next resynchronization point (e.g., picture or slice headers) due to macroblock-to-macroblock differential and variable-length coding. This is referred to as spatial propagation. When loss occurs in a reference picture (I or P picture), the lost macroblocks will affect the predicted macroblocks in subsequent frame(s). This is known as temporal propagation.

Error concealment is generally used to reduce the impact of data loss on the visual information. The error-concealment algorithms include, for example, spatial interpolation, temporal interpolation and early resynchronization. The MPEG-2 standard [3] proposes an elementary error-concealment algorithm based on *motion compensation*. This simple technique is certainly not optimal, but it offers a satisfying decoding quality. The development of error-concealment technique is, however, outside the scope of this paper. The motion-compensation concealment estimates the motion vectors of the lost macroblock by using the motion vectors of neighboring macroblocks in the affected picture (provided that these have not also been lost). There is, however, an obvious problem with lost macroblocks whose neighbors are intra-coded, since usually they do not have associated motion vectors. To get around this problem, the encoding can also include motion vectors for intra macroblocks[1]. Even though error concealment may, in general, efficiently decrease the visibility of losses, severe data loss may, however, still lead to annoying degradation in the decoded video.

The robustness of compressed MPEG-2 video may be dramatically increased by judiciously inserting resynchronization points in the bit stream. These can be obtained by extra slice headers to limit spatial propagation and intra-coded macroblocks to stop temporal propagation. However, the addition of extra slice headers and/or intra-coded macroblocks is not costless. First the larger the number of slices, the bigger the overhead. Indeed, every new slice introduces a 5- to 6-bytes length header which compose the major part of the overhead. It also resets the differential coding of the DC values and motion vectors. Second, the larger the number of intra-coded macroblocks, the higher the overhead. The amount of overhead generated in this case is, however, not easy to predict. Indeed, it depends on the encoding complexity of each extra macroblock encoded in the intra mode.

Under the same video traffic constraints, extra resynchronization points therefore reduce the amount of bits available to code pure video information, and thus decrease the quality of the reconstructed video. In a lossy transmission, the end-to-end quality is no longer strictly decreasing with the amount of overhead. It results both from the encoding quality and the network degradation, as previously mentioned. Therefore, there is an optimal number of slice headers and intra-coded macroblocks that maximizes the end-to-end quality. This optimum is dependent on the encoding bit rate and the packet loss ratio. Fig. 2 illustrates this tradeoff under a uniform and independent packet loss process assumption, which represents the worst case in MPEG-2 delivery [5]. A two-state Markovian model-based [23] data loss

[1]Some MPEG-2 encoder chips automatically produce concealment motion vectors for all macroblocks.

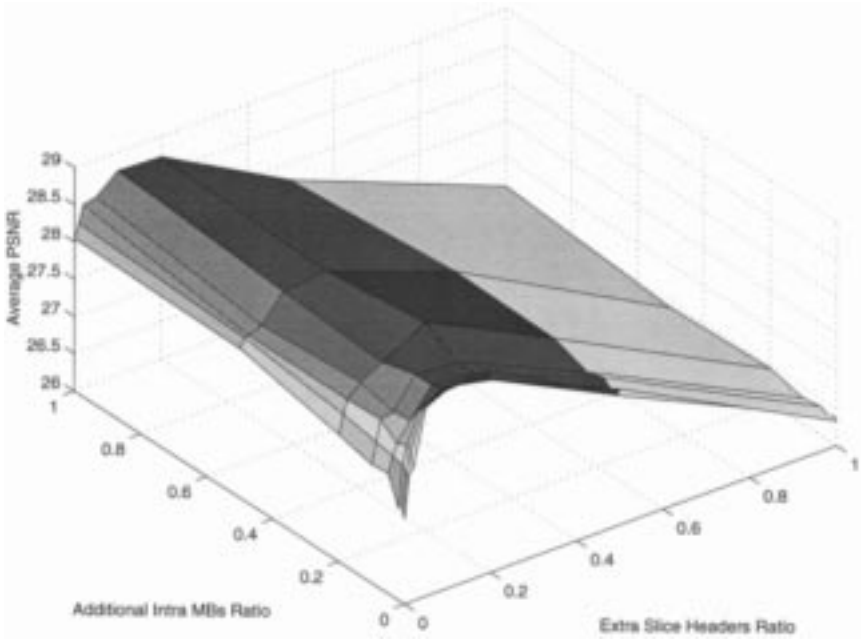

Fig. 2. End-to-end PSNR quality versus extra resynchronization points ratio for the football scene (CBR encoding at 5 Mbps and $\mathrm{PLR} = 10^{-2}$).

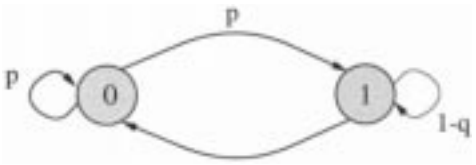

Fig. 3. Gilbert loss model.

generator (i.e., $p = \mathrm{PLR}$ and $q = 1 - \mathrm{PLR}$ in the loss model of Fig. 3) simulates losses on an MPEG-2/RTP video stream. The video sequence consists of 400 frames conforming to the ITU-R 601 format (TV-resolution, 720×576 at 25 fps). The sequence includes five video scenes that differ in terms of spatial and temporal complexities.

Problem Formulation: The problem addressed in this paper consists in finding the optimal tradeoff between video information and error protection. Clearly, a random insertion of extra resynchronization points in the bit stream or regular forward-error-correction (FEC) packets is not optimal. Indeed, the efficiency strongly depends on the content of the corresponding video area. There is no need to insert resynchronization points where the impact of data loss would not affect the video quality (under a given error-concealment technique). Moreover, the protection level has also to be adapted to the network performances or the expected loss probabilities.

Given the expected PLR [or dynamically measured through real-time control protocol (RTCP)] feedback messages), the error-concealment technique implemented at the decoder, and a distortion metric, two related problems are considered.

1) *MPEG-2 Structuring*: Determine the most appropriate MPEG-2 structure in terms of resynchronization points location.
2) *MPEG-2 Protection*: Derive a content-based FEC scheme to protect areas where structuring is not sufficient.

III. Adaptive MPEG-2 Structuring

A. Loss Probability Matrices

It has been noted that a macroblock may be damaged in any of the three following cases:

1) It belongs to an RTP packet that has been lost during transmission
2) It belongs to a slice that has been affected by a packet loss (spatial propagation)
3) It is temporally dependent on a damaged macroblock of a previous reference frame (temporal propagation).

The first factor that may affect a macroblock is the transmission error. If we assume a uniform and independent loss process, the probability θ for an RTP packet to be lost is given by the PLR experienced on the network. Therefore, without any other information about the packet loss process, every RTP packet has the same average probability to be lost: $\theta = \mathrm{PLR}$. Let us now call $B_n(i,j)$, the macroblock at the ith column and the jth row of a given frame n. Under the assumption that a macroblock is lost as soon as part of it is missing, the probability $\lambda_n(i,j)$ for the macroblock $B_n(i,j)$ to be lost is given by

$$\begin{aligned} &\lambda_n(i,j) = \theta \mathbf{N}_n(i,j), \\ &\forall (i,j) | 1 \leq i \leq B_{\mathrm{row}}, \quad 1 \leq j \leq B_{\mathrm{column}} \end{aligned} \tag{1}$$

where $\mathbf{N}_n(i,j)$ is the number of RTP packets containing the macroblock $B_n(i,j)$. B_{row} and B_{column} are, respectively, the number of macroblocks per frame row and column.

Even at high encoding rates, loss entities (i.e., roughly multiples of 188 B) are much larger than the macroblock size. Thus, in general, macroblocks belong to at most two RTP packets (i.e., $\mathbf{N}_n(i,j) \in \{1,2\}$).

The second factor that may affect a macroblock is *spatial propagation*. In the case of a transmission error, an MPEG-2

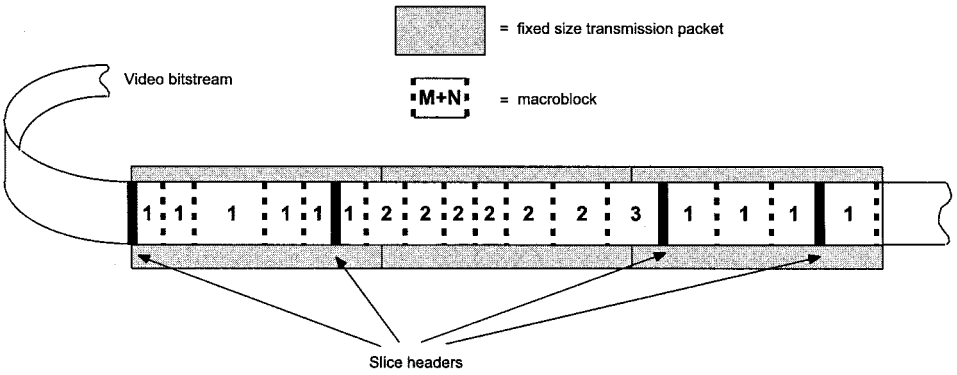

Fig. 4. Illustration of $\mathbf{P}_n(i,j)$. The numbers in each macroblock represent $M_n + N_n$ values.

decoder skips all video information up to the next slice header, which acts as a spatial resynchronization point. Consequently, when a macroblock is lost within a slice, all subsequent macroblocks of the same slice are considered as being damaged, even if they do not belong to the lost RTP packet.

Thus, for a given frame n, the probability $\mathbf{P}_n(i,j)$ for a macroblock $B_n(i,j)$ not to be correctly decoded (transmission error + spatial propagation) is given by

$$\begin{aligned}\mathbf{P}_n(i,j) &= \lambda_n(i,j) + \theta\mathbf{M}_n(i,j) \\ &= \theta[\mathbf{N}_n(i,j) + \mathbf{M}_n(i,j)] \\ \forall(i,j) | 1 \le i \le B_{\text{row}} &\text{ and } 1 \le j \le B_{\text{column}} \end{aligned} \quad (2)$$

where $\mathbf{M}_n(i,j)$ represents the number of RTP packets within the same slice before the first packet related to $B_n(i,j)$ (see Fig. 4).

There is an exception to this rule. Indeed, according to the MPEG-2 syntax, every frame is preceded by a header. If a packet containing a frame header is lost, the entire frame is skipped, making (2) useless. We assume this case to be rare enough to be neglected. This assumption is enforced when these headers are protected via a specific FEC scheme [24].

The third factor that may affect a macroblock is *temporal propagation* [16]. Our objective is to derive the loss probability matrix $\mathcal{E}_n^{n-k}$ for every pixel of frame n due to temporal propagation of damaged pixels in frame $(n-k)$, with $k \le n$.

Since motion estimation does not consider macroblock boundaries, but rather references areas of 16 by 16 pixels, the granularity of the loss probability matrix $\mathbf{P}_n$ must be refined to the pixel level. Indeed, in (1) and (2) the entries of the matrix $\mathbf{P}_n$ refer to macroblock,s whereas we need now reference to pixels. The *loss probability matrix* due to spatial propagation for every pixel of frame n is called $\mathcal{P}_n$. The mapping between $\mathbf{P}_n$ and $\mathcal{P}_n$ is straightforward. Indeed, every pixel of a given macroblock has the same probability to be lost. Hence, $\mathcal{P}_n$ is obtained by the Kronecker product of a 16×16 unity matrix $\mathbf{I}_{16}$ by $\mathbf{P}_n$

$$\mathcal{P}_n = \mathbf{I}_{16} \otimes \mathbf{P}_n. \quad (3)$$

The resulting matrix $\mathcal{P}_n$ has therefore the same size as the video frame (i.e., 720×576 in the ITU-R 601 format).

In the following development, the B frames are not considered for additional intra-coded macroblock insertion. Indeed, B frames do not propagate degradation, since they are never referenced. Therefore, the impact of data loss in B frames is barely visible (the temporal resolution of the human visual system is larger than a single frame duration [25]). These frames may also offer the highest compression ratio, and adding intra-coded macroblocks would result in the highest relative overhead. Finally, B frames generally have the smallest number of bits so that losses have a low chance to occur in these frames. Additional intra-coded macroblock in B frames would therefore result in a waste of bandwidth.

Temporal propagation means that a pixel in the current frame is damaged because it refers to a badly decoded pixel from a previous reference frame. This badly decoded pixel may result from transmission error, spatial propagation, or/and temporal propagation, as explained before. Thus, $\mathcal{E}_n^{n-k}$ obviously depends on $\mathcal{P}_m$, with $m = n-k, n-k+1, \ldots, n$. Moreover, $\mathcal{E}_n^{n-k}$ needs to be computed in a recursive manner. Indeed, video areas each pixel refers to have to be found by recursively following the successive motion vectors within the video sequence. Thus, within a macroblock, even though each pixel has the same probability to be lost, they do not have the same probability to be decoded into an erroneous value. They do not necessarily refer to the same macroblock in reference frames.

The loss probability matrix $\mathcal{E}_n^{n-1}$, derived from temporal propagation of errors occurring in reference frame $(n-1)$ and impacting frame n, is first computed. The probability for all the pixels in frame n to be damaged by losses occurring in frame $(n-1)$ can be easily derived. First, the motion vectors of frame n are used to reference matrix $\mathcal{P}_{n-1}$. Actually, the motion estimation performed by MPEG-2 is applied on the *"frame"* $\mathcal{P}_{n-1}$ on a macroblock basis. This mapping operation could be denoted by $\mathcal{M}_n(\mathcal{P}_{n-1})$ (see Fig. 5). Finally, each element of $\mathcal{M}_n(\mathcal{P}_{n-1})$ should be muliplied by the probability for the corresponding pixel not to be lost in frame n. Indeed, there is no need to compute the probability for a pixel to be damaged in a previous frame if it is lost in the current frame n (it would make the consideration of temporal propagation useless). Therefore, $\mathcal{E}_n^{n-1}$ can be written as follows:

$$\begin{aligned}\mathcal{E}_n^{n-1}(i,j) &= \mathcal{M}_n(\mathcal{P}_{n-1})(i,j)(1 - \mathcal{P}_n(i,j)) \\ \forall(i,j) \mid 1 \le i \le W, &\quad 1 \le j \le H \end{aligned} \quad (4)$$

where $\mathcal{E}_n^{n-1}(i,j)$ represents the probability for pixel at ith column and jth line in frame n to be damaged by losses in

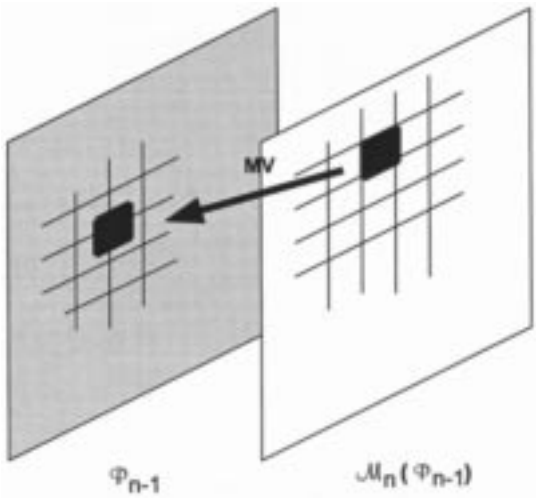

Fig. 5. Mapping function $\mathcal{M}_n$ of loss probability matrices.

reference frame $n-1$. The variables W and H correspond respectively to the number of pixels per row and column. Obviously, if the pixel given by (i',j') in frame n does not reference any video area of frame $(n-1)$, or if it belongs to an intra-coded macroblock, $\mathcal{E}_n^{n-1}(i',j')=0$.

The probability matrix for pixels not to be spatially lost could be written as

$$\overline{\mathcal{P}}_n = \mathbf{I_{W,H}} - \mathcal{P}_n \tag{5}$$

where $\mathbf{I_{W,H}}$ is a $W \times H$ unity matrix. Equation (4) could now be generalized, taking into account losses in any of the k previous reference frames, with $k \leq n$. The generic loss probability matrix due to temporal propagation, $\mathcal{E}_n^{n-k}$, can be obtained via recursion. Indeed, similar to (4), $\mathcal{E}_{n-k+1}^{n-k}$ is given by

$$\begin{aligned}\mathcal{E}_{n-k+1}^{n-k}(i,j) =& \mathcal{M}_{n-k+1}\left(\mathcal{E}_{n-k}^{n-k}\right)(i,j)\overline{\mathcal{P}}_{n-k+1}(i,j)\\ &\forall (i,j) | 1 \leq i \leq W, \quad 1 \leq j \leq H\end{aligned} \tag{6}$$

with

$$\mathcal{E}_{n-k}^{n-k}(i,j) = \mathcal{P}_{n-k}(i,j). \tag{7}$$

The process can then be generalized starting with (7) above. It becomes

$$\begin{aligned}\mathcal{E}_{n-k+m}^{n-k}(i,j) =& \mathcal{M}_{n-k+m}\left(\mathcal{E}_{n-k+m-1}^{n-k}\right)(i,j)\overline{\mathcal{P}_{n-k+m}}(i,j),\\ & m = 1,2,3,\ldots,k.\end{aligned} \tag{8}$$

Following the same notation, $\mathcal{M}_{n-k+m}$ uses the motion vectors of frame $(n-k+m)$. Moreover, when a pixel given by (i',j') in one of the reference frames $(n-k+m)$ belongs to an intra-coded macroblock, or has no correspondence in its direct reference frame (according to $\mathcal{M}_{n-k+m}$), then

$$\mathcal{E}_{n-k+m}^{n-k}(i',j') = 0. \tag{9}$$

Finally, $\mathcal{E}_n^{n-k}$ is obtained when $m=k$ in (8).

$\mathcal{E}_n^{n-k}$ represents the generic loss probability matrix for frame n due to temporal propagation of data loss in frame $(n-k)$. For the sake of simplicity, the impact of data loss in each reference frames $(n-k')$ on $\mathcal{E}_n^{n-k}$ is considered independently.

B. Proposed Algorithm: Adaptive MPEG-2 Information Structuring (AMIS)

The proposed algorithm for adaptively inserting resynchronization points in an MPEG-2 bit stream, namely the AMIS algorithm, is now presented. AMIS strongly relies on the study presented in the previous subsection. Intuitively, it works as follows: an extra resynchronization point is inserted in the bit stream whenever hypothetical data loss, following a uniform loss process, would lead to video degradation above a desired threshold, after error concealment.

The *mean luminance difference* (MLD) has first been chosen as distortion measure. It corresponds to the simplest metric correlated with human perception [9] (under the assumption that the viewer stands far enough from the monitor). The distortion is computed between the current macroblock after encoding (generally given by the encoding scheme) and the same macroblock impaired by loss. This one is obtained by simulating losses and concealment in the encoder. The real effects of hypothetical losses are thus captured by the encoding scheme. The MLD for $B(i,j)$ is defined as follows:

$$\delta(i,j) = \left| \frac{1}{256}\sum_{p=1}^{256} B^p(i,j) - \frac{1}{256}\sum_{p=1}^{256} \widetilde{B}^p(i,j) \right| \tag{10}$$

where

- (i,j) macroblock position in the frame;
- p pixel position in the corresponding macroblock.
- $B^p(i,j)$ pixels of the correctly (error-free) decoded macroblock;
- $\widetilde{B}^p(i,j)$ pixels of the corresponding damaged macroblock at position (i,j).

The error-concealment technique implemented at the decoder should also be specified to build the optimal structuring. However, if the error-concealment technique is not known *a priori*, the structuring algorithm would still produce good results, since major error-concealment schemes have similar features. To be specific, the motion-compensated concealment technique has been chosen for its simplicity.

It has to be noted that the AMIS algorithm would not need any major modification if a different distortion measure and/or error-concealment technique was imposed.

AMIS is divided in two distinct parts: 1) the *spatial* part, which deals with slice headers insertion, and 2) the *temporal* part, which is in charge of deciding when a macroblock should be intra-coded. Indeed, inserting extra slice headers has no effect on temporal error propagation. Also, adding intra-coded macroblock does not help in limiting the spatial error propagation. Therefore, these two parts are considered independently. However, it is clear that the slice structure of reference frames may influence the insertion decision of intra-coded macroblocks.

1) AMIS-Spatial: The *Spatial* part of AMIS [26] aims at limiting the spatial error propagation, or at least its visible degradation. It introduces an extra slice header as soon as the distortion due to hypothetical loss reaches a given threshold, Δ_s. Clearly a new slice is inserted as soon as

$$\sum_{B_n(i,j)\in S} \delta_n^s(i,j)\mathbf{P}_n(i,j) \geq \Delta_s \tag{11}$$

where, $B_n(i,j)$ is the current macroblock belonging to slice S and $\delta_n^s(i,j)$ corresponds to the expected MLD in case $B_n(i,j)$ was damaged. $\mathbf{P_n}(i,j)$ defined in (2), represents the probability for $B(i,j)$ to be spatially damaged, by packet loss or spatial propagation.

Actually, the expected distortion is weighted by its likelihood to occur. There is indeed no need to protect an area not likely to be lost, even if the corresponding distortion would be high.

The spatial threshold Δ_s regulates the acceptable level of distortion. The smaller the threshold, the higher the number of slices.

AMIS-Spatial also takes the packetization process into account: no more than one slice header is encapsulated in the same network loss entity [13].

2) AMIS-Temporal: The *temporal* part of AMIS is more complex [27]. First, let us assume that losses in different reference frames can be considered independently in regard to their impact on the current frame. Even though not completely correct, this assumption places the encoding process in the worst case from the distortion point of view. It will tend to generate more protection than effectively needed, but greatly simplifies the AMIS mechanism.

AMIS-Temporal analyzes every single macroblock and decides whether or not to intra-code it. Again, this decision depends on the macroblock distortion due to temporal propagation of data loss.

The decision may be expressed as follows. The distortion due to temporal error propagation is weighted by the corresponding loss probability matrix and compared to a threshold Δ_t. This weighted distortion is obtained by summing the effects of uniformly distributed packet losses in every single previous reference frame, up to the last intra-coded picture $(n-I)$. Finally, the condition for a macroblock $B_n(i,j)$ to be intra-coded in frame n is given by

$$\sum_{k=1}^{I}\left(\frac{1}{256}\sum_{p\in B_n(i,j)}\mathcal{E}_n^{n-k}(p)\delta_{n,k}^t(i,j)\right)\geq\Delta_t \qquad (12)$$

where $\mathcal{E}_n^{n-k}$ is given by (8). The expected MLD between the current MB correctly decoded and its substitute in case of loss in the reference frame k is given by $\delta_{n,k}^t(i,j)$. Again, the temporal threshold Δ_t regulates the acceptable level of distortion. The smaller the threshold, the higher the number of intra-coded macroblocks.

Finally, a maximum refresh period, $T_{\max}$, is also imposed. This period corresponds to the maximum number of frames a pixel may subsist without any intra-reference. When a pixel has no intra-reference for a period longer than $T_{\max}$, the macroblock shall be intra-coded. This consideration is particularly useful in large GOP encoding schemes, or in case of large intervals between consecutive I-frames.

C. Experimental Results and Comparisons

The AMIS algorithm is now evaluated and compared to other encoding schemes in terms of final video quality. Figs. 6 and 7 compare the behavior of: 1) AMIS; 2) a random resynchronization points insertion scheme; 3) the common TM-5 model [28]; and 4) the algorithm proposed by Richardson and Riley [11].

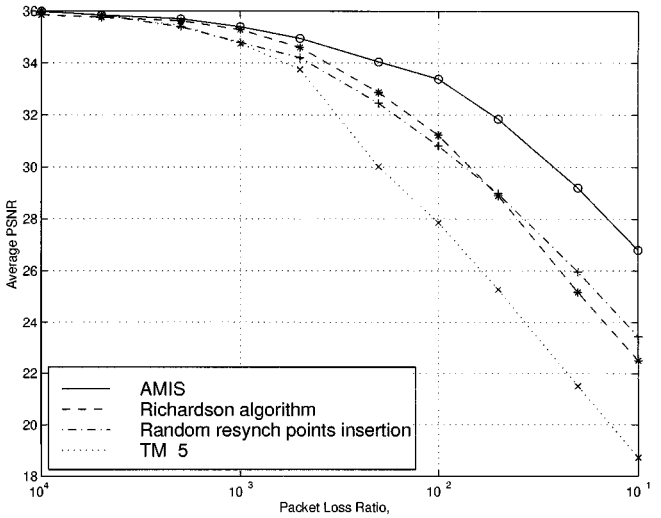

Fig. 6. AMIS end-to-end PSNR quality versus PLR in comparison to random resynchronization points insertion, TM-5 encoding scheme and encoding algorithm proposed by Richardson (CBR encoding at 6 Mbps).

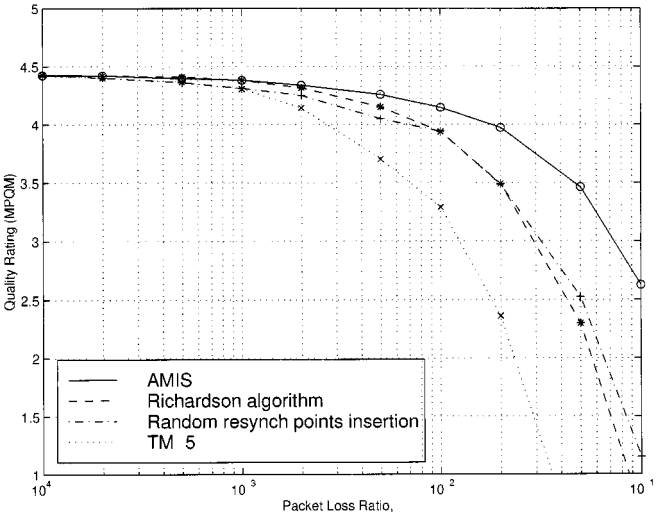

Fig. 7. AMIS perceptual end-to-end quality versus PLR in comparison to random resynchronization points insertion, TM-5 encoding scheme and encoding algorithm proposed by Richardson (CBR encoding at 6 Mbps).

The comparison is performed in terms of PSNR and MPQM [29], [30] final video quality, respectively, under several packet loss ratios and a given bit budget. The PLR has been allowed to vary between 10^{-1} and 10^{-4}. These values could seem quite larger than commonly accepted networking performances. However, the latter are generally average values computed over entire sessions. They do not reflect the short-term characteristics of the losses. Such RTP packet loss ratio values could likely be met during small periods of time (e.g., network congestion, atmospherical conditions). Finally, the video streams have been encoded at a constant bit rate of 6 Mbps. CBR encoding mode imposes the most stringent constraint on bit-rate allocation. However, similar (and, certainly better) results could easily be obtained in OL-VBR encoding mode.

AMIS is obviously dramatically better than the TM-5 algorithm under medium to high PLRs. Also, under low PLRs, it is comparable to the TM5. Indeed, AMIS judiciously shares the

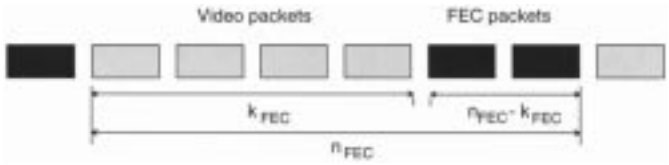

Fig. 8. Media-independent FEC scheme.

total bit budget between pure video information and additional resynchronization points, according to the expected PLR. Moreover, the comparison of AMIS versus the random insertion of extra resynchronization points scheme, for the same overhead, clearly shows the relevance of the content-based structuring. Finally, AMIS offers better results than the algorithm proposed by Richardson *et al.*, especially in bad transmission conditions. The latter algorithm indeed basically uses a static slice length for each frame type (i.e., I , P, or B frame). This length stays valid for the whole sequence, without PLR values considerations. All these results highlight the need for adaptivity to both video content and transmission quality.

IV. Adaptive MPEG-2 Structuring and Protection

A. FEC-Based Protection

In Section V, AMIS has been presented. It smartly structures the MPEG-2 bit stream to make it more robust against data loss. However, it is clear that data loss may still induce unacceptable degradation in the reconstructed video. Indeed, some video areas may be highly sensitive to loss (e.g., fast-moving areas). In this section, AMIS is extended with a FEC-based protection scheme, to become AMISP.

FEC means that redundancy is added to the data so that the receiver can recover from losses or errors without any further intervention from the sender. Considering the delay requirements for interactive video, FEC is more appropriate than retransmission (i.e., ARQ scheme) because it fulfills the timing constraints. Usually, FEC schemes build n_{FEC} packets blocks where k_{FEC} video packets are protected by means of $n_{\mathrm{FEC}} - k_{\mathrm{FEC}}$ redundancy packets (see Fig. 8). The FEC blocks use Reed–Solomon codes or simply XOR-based functions. The description of the FEC blocks is however outside the scope of this paper [31]. Recall that such schemes are able to recover up to $n_{\mathrm{FEC}} - k_{\mathrm{FEC}}$ lost packets in a block of n_{FEC} packets [32]. The video packet loss process is then modified, and the resulting packet loss ratio for FEC-protected packets becomes θ_{FEC}. Under the assumption of independent losses, the PLR θ_{FEC} becomes [33]

$$\theta_{\mathrm{FEC}} = \theta(1 - P_{\mathrm{rec}}) \tag{13}$$

where

$$P_{\mathrm{rec}} = \sum_{i=0}^{n_{\mathrm{FEC}}-k_{\mathrm{FEC}}-1} \binom{n_{\mathrm{FEC}}-1}{i} \theta^i (1-\theta)^{n_{\mathrm{FEC}}-i-1} \tag{14}$$

represents the probability of FEC recovery in an (n_{FEC}, k_{FEC}) FEC scheme. A packet lost during transmission can be recovered by the receiver is less than $n_{\mathrm{FEC}} - k_{\mathrm{FEC}}$ packets are lost among the other $n_{\mathrm{FEC}} - 1$ packets from the FEC block.

Several criteria have then to be considered in the choice of the FEC parameters n_{FEC} and k_{FEC}. First, the overhead $(n_{\mathrm{FEC}} - k_{\mathrm{FEC}})/n_{\mathrm{FEC}}$ has to be kept as small as possible and to be adapted to the expected loss ratio. However, this ratio does not need to be very large to ensure a good recovery probability. It has indeed been shown that, even for a small $(n_{\mathrm{FEC}} - k_{\mathrm{FEC}})/n_{\mathrm{FEC}}$ ratio, FEC can be very effective and reduces the loss probability by several orders of magnitude [34]. The ratio efficiency/overhead is moreover larger for large k_{FEC} values, assuming that losses occur independently [35].

Second, the FEC scheme has to satisfy strict delay constraints in interactive applications. The delay introduced by FEC reconstruction[2] should not be much larger than one frame, since other delays are also introduced along the transmission path. Since one TS packet represents already a delay of about 5.6 ms in a 6-Mbps connection, n_{FEC} should not be larger than 10–15 packets. This value is, however, directly dependent on the bit rate.

Third, it has been shown that for a given overhead, large values of k_{FEC} lead to the best reconstruction probabilities [15]. On the other hand, small k_{FEC} values ensure a more efficient protection of elected packets in an adaptive algorithm.

All the previous statements suggests that the value of n_{FEC} should be chosen as large as possible, given some delay constraints. Then, the k_{FEC} value should be computed accordingly to offer a sufficient protection, and also to minimize the overhead.

Finally, FEC parameters could vary dynamically according to loss patterns (i.e., PLR and ABL). On-going work is currently trying to optimize these parameters according to network conditions and the degree of protection accuracy. For the sake of simplicity, $n_{\mathrm{FEC}} - k_{\mathrm{FEC}} = 1$ in the following experiments. This allows moreover a very simple and rapid XOR-based FEC scheme.

B. The Adaptive Protection Algorithm: AMISP

The proposed protection algorithm is the following. During the encoding process, a packet p is marked to be protected whenever its hypothetical loss would introduce an unacceptable degradation. Similarly to (11), the loss probability weighted distortion is compared to a third threshold Δ_{FEC}

$$\sum_{B_n(i,j)\in p} \delta_s(i,j)\theta \geq \Delta_{\mathrm{FEC}}. \tag{15}$$

Whenever AMISP decides to protect a packet, it triggers the underlying network adaptation layer (NAL). The NAL starts counting k_{FEC} video packets and then inserts $n_{\mathrm{FEC}} - k_{\mathrm{FEC}}$ FEC packets in the MPEG-2 bit stream. Of course, if the elected packet already belongs to a FEC block, no additional overhead is inserted. As in the structuring scheme, the amount of redundancy is driven by the threshold Δ_{FEC}, which represents the quality of service desired at the receiver. Finally, the adaptive FEC algorithm is easily implemented on RTP protocols, thanks to the support for FEC protection [36].

[2]The FEC block construction at the sender does not introduce any queueing delay.

The structuring part of AMISP still works in the same manner. However, the macroblock loss probability $\mathbf{P}_n$ [see (2)] becomes $\tilde{\mathbf{P}}_n$ and is now given by

$$\tilde{\mathbf{P}}_n(i,j) = \sum_{p=1}^{N_n(i,j)} \theta_p + \sum_{p=1}^{M_n(i,j)} \theta_p, \text{ with } \theta_p \in \{\theta, \theta_{\mathrm{FEC}}\} \quad (16)$$

where $M_n(i,j)$ still represents the number of RTP packets within the same slice before and excluding $B_n(i,j)$. $N_n(i,j)$ represents the number of packets containing part of the macroblock $B_n(i,j)$. The PLR θ_p is either equal to θ_{FEC} or θ, depending on whether the packets are FEC protected or not.

It has to be noticed that packets are FEC-protected in regard to their influence onto spatial distortion. These packets very likely contain a slice header due to the similarity between relations (11) and (15). However, the temporal propagation phenomenon is not taken into account by the protection decision process. The reasons of this choice are twofold. First, the temporal propagation of an error in the current frame cannot be predicted in one-pass encoding. Second, it can be assumed that the most relevant packets (i.e., FEC-protected packets) are the packets that would also cause the highest temporal distortion.

Moreover, the major MPEG-2 headers (i.e., sequence and picture headers) are also FEC protected [24]. Their loss would indeed cause a really important degradation. Each packet containing such crucial information is therefore protected by an FEC packet. Finally, the rate control part of the encoding algorithm has been slightly modified. The video encoding rate has to be adapted to the protection overhead to ensure a constant total bit rate. Basically, the modifications simply includes the number of bits used for protection in the loop of the TM-5 rate control algorithm [28].

C. Experimental Results and Comparisons

As stated before, the algorithm inserts only a single packet per FEC block (i.e, $n_{\mathrm{FEC}} - k_{\mathrm{FEC}} = 1$). The length of the FEC blocks (i.e., k_{FEC}) could be determined through simulations of different AMISP schemes [35]. Therefore, $k_{\mathrm{FEC}} = 10$ seems to fit both delay and robustness requirements, at least in the most common θ range (i.e., between 10^{-4} and 10^{-1}). The total bit rate (i.e., video information and FEC overhead) is fixed to 6 Mbps for each transmission. Similar results are presented in terms of both PSNR and perceptual quality.

Figs. 9 and 10 compare AMISP with several protection schemes. First a basic TM-5 video encoding protected by a regular (by opposition to adaptive) FEC scheme is proposed. It generates one redundancy packet every ten video packets. It is clearly visible that AMISP provides a better end-to-end quality over the complete packet loss ratio range. At low θ values, the improvement in quality is mainly due to the adaptivity feature of AMISP: it generates less redundancy, and thus provides more accurate video information. At high loss rates, both schemes perform similarly in terms of protection. However, the quality offered by AMISP is much higher thanks to the underlying structuring scheme (i.e., AMIS). This scheme indeed greatly limits the residual error propagation within the decoded sequence.

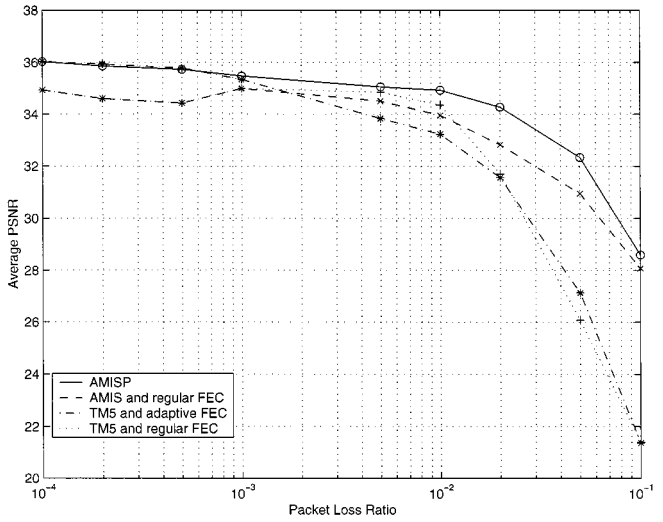

Fig. 9. End-to-end PSNR quality versus the packet loss ratio. Comparison of the AMISP algorithm ($k_{\mathrm{FEC}} = 10$ and $n_{\mathrm{FEC}} = 11$) with a TM-5 encoding with an adaptive and regular FEC scheme ($k = 10$ and $n = 11$). The total bit rate is about 6 Mbps.

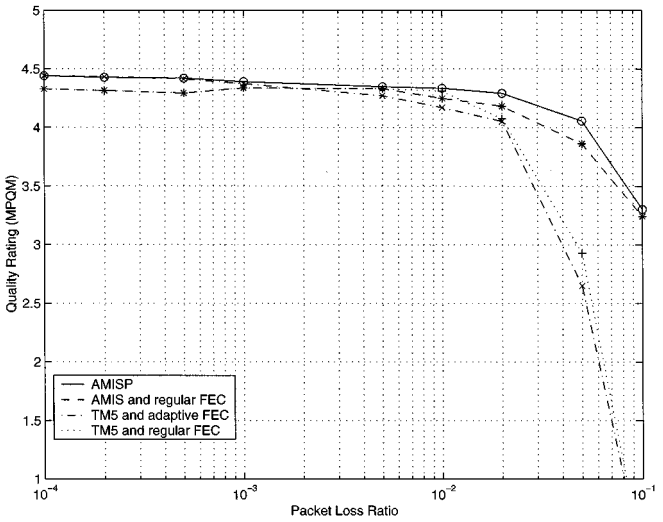

Fig. 10. MPQM end-to-end perceptual quality versus the packet loss ratio. Comparison of the AMISP algorithm ($k_{\mathrm{FEC}} = 10$ and $n_{\mathrm{FEC}} = 11$) with a TM-5 encoding with an adaptive and regular FEC scheme ($k = 10$ and $n = 11$). The total bit rate is about 6 Mbps.

Figs. 9 and 10 also emphasize the useful adaptivity feature to network conditions. The end-to-end quality of AMISP is compared to the same AMIS video bit rate but protected by a regular FEC scheme. At a low loss ratio, AMISP provides a better quality since it does not generate useless overhead. However, the difference is not very large. The relatively small FEC overhead only decreases slightly the encoding quality at medium encoding rate. Meanwhile, both algorithms are equivalent at high loss ratios. The number of packets AMISP has to protect becomes very large in these conditions. Hence, the adaptive protection becomes very close to a regular protection scheme.

Figs. 9 and 10 finally demonstrate the advantages of the underlying structuring scheme. AMISP is compared to a TM-5 encoding protected by the same adaptive FEC scheme used in AMISP. It is clear that both algorithms lead to the same quality

at low loss rates. Losses that would cause important degradation are recovered by the FEC algorithm. However, the gap between both schemes grows rapidly with the loss ratio, as the protection algorithm looses some of its efficiency indeed. The errors propagate within the TM-5 sequence, while they are kept to an acceptable level in AMIS.

V. Conclusion

We have presented a new adaptive error-resilient scheme for TV-resolution MPEG-2 video streams interactive delivery, namely AMISP. It includes a media-dependent FEC algorithm relying on an MPEG-2 syntactic structuring technique. A judicious combination of protection redundancy, MPEG syntactic data, and pure video information were shown to greatly improve the final quality under a given bit budget. Experimental results have shown that AMISP dramatically outperforms existing techniques, thanks to its efficient adaptivity. Major improvements are due to adaptivity of both FEC protection and structuring at respectively low and high loss ratios. Moreover, it must be noted that AMISP does not significantly increase the MPEG-2 encoding complexity. However, the protection part of AMISP requires the underlying layer (NAL) to provide FEC capabilities. If this is not the case, only the structuring part can be used.

In this work, retransmission of missing packets was assumed not to be feasible due to stringent timing constraints. However, we believe that retransmission might lead to an improved scenario in one-way real-time MPEG-2 delivery. Finally, AMISP could also be applied to other video standards at the cost of a few modifications.

Acknowledgment

The authors are particularly grateful to Prof. M. Kunt from the Signal Processing Laboratory and to the anonymous reviewers for their relevant comments and remarks on the paper.

References

[1] G. Karlsson, "Asynchronous transfer of video," *IEEE Commun. Mag.*, vol. 34, pp. 118–126, Aug. 1996.

[2] O. Verscheure, X. G. Adanez, G. Karlsson, and J.-P. Hubaux, "User-oriented QoS in packet video delivery," *IEEE Network Mag.*, vol. 12, pp. 12–21, Nov./Dec. 1998.

[3] *Information Technology—Generic Coding of Moving Pictures and Associated Audio Information—Part 1, 2 and 3*, ISO/IEC 13 818, 1996.

[4] B. G. Haskell, A. Puri, and A. N. Netravali, *Digital Video: An Introduction to MPEG-2*, ser. Digital Multimedia Standards. London, U.K.: Chapman & Hall, 1997.

[5] Y.-Q. Zhang and X. Lee, "Performance of MPEG codes in the presence of errors," *J. Vis. Commun. Image Repres.*, vol. 5, no. 4, pp. 378–387, Dec. 1994.

[6] Y. Wang and Q.-F. Zhu, "Error control and concealment for video communication: a review," *Proc. IEEE*, vol. 86, pp. 974–997, May 1998.

[7] W. S. Lee, M. R. Pickering, M. R. Frater, and J. F. Arnold, "Error resilience in video and multiplexing layers for very low bit-rate video coding systems," *IEEE J. Select. Areas Commun.*, vol. 15, pp. 1764–1774, Dec. 1997.

[8] C.-T. Chen, "Error detection and concealment with an unsupervised MPEG2 video decoder," *J. Vis. Commun. Image Repres.*, vol. 6, no. 3, pp. 265–279, Sept. 1995.

[9] M.-J. Chen, L.-J. Chen, and R.-M. Weng, "Error concealment of lost motion vectors with overlapped motion compensation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 560–563, June 1997.

[10] H.-C. Shyu and J.-J. Leou, "Detection and concealment of transmission errors in MPEG-2 images—a genetic algorithm approach," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 937–948, Sept. 1999.

[11] I. E. G. Richardson and M. J. Riley, "Varying slice size to improve error tolerance of MPEG video," *Proc. SPIE*, vol. 2668, pp. 365–371, 1996.

[12] S. McCanne and V. Jacobson, "Vic: A flexible framework for packet video," in *Proc. ACM Multimedia Conf.*, San Francisco, CA, Nov. 1995, pp. 511–522.

[13] J. Zhang, M. R. Frater, J. F. Arnold, and T. M. Percival, "MPEG 2 video services for wireless ATM networks," *J. Select. Areas Commun.*, vol. 15, pp. 119–128, Jan. 1997.

[14] P. Batra and S.-F. Chang, "Effective algorithms for video transmission over wireless channels," *Signal Processing: Image Commun.*, vol. 12, no. 2, pp. 147–166, Apr. 1998.

[15] G. Carle and E. W. Biersack, "Survey of error recovery techniques for IP-based audio–visual multicast applications," *IEEE Network*, vol. 11, pp. 24–36, Nov./Dec. 1997.

[16] M. Wada, "Selective recovery of video packet loss using error concealment," *IEEE J. Select. Areas Commun.*, vol. 7, pp. 807–814, June 1989.

[17] V. Parthasarathy, J. W. Modestino, and K. S. Vastola, "Design of a transport coding scheme for high-quality video over ATM networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 358–376, Apr. 1997.

[18] A. Albanese, J. Blomer, J. Edmonds, M. Luby, and M. Sudan, "Priority encoding transmission," *IEEE Trans. Inform. Theory*, vol. 42, pp. 1737–1744, Nov. 1996.

[19] M. Bystrom, V. Parthasarathy, and J. W. Modestino, "Hybrid error concealment schemes for broadcast video transmission over ATM networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 868–881, Sept. 1999.

[20] S. McCanne, M. Vetterli, and V. Jacobson, "Low-complexity video coding for receiver-driven layered multicast," *IEEE J. Select. Areas Commun.*, vol. 6, pp. 983–1001, 1997.

[21] U. Horn, K. Stuhlmuller, M. Link, and B. Girod, "Robust internet video transmission based on scalable coding and unequal error protection," *Signal Processing: Image Commun.*, vol. 15, no. 1–2, pp. 77–94, 1999.

[22] W. Jiang and A. Ortega, "Multiple description coding via polyphase transform and selective quantization," in *Proc. Visual Communications and Image Processing, VCIP'99*, San Jose, CA, Jan. 1999.

[23] E. N. Gilbert, "Capacity of a burst-noise channel," *Bell Syst. Tech. J.*, pp. 1253–1265, Sept. 1960.

[24] X. G. Adanez, "Designing new network ndaptation and ATM adaptation layers for interactive multimedia applications," Ph.D. dissertation, Swiss Federal Institute of Technology, Lausanne, Switerland, 1998.

[25] C. J. van den Branden Lambrecht, "Perceptual models and architectures for video coding applications," Ph.D. dissertation, Swiss Federal Institute of Technology, Lausanne, Switzerland, 1996.

[26] O. Verscheure and P. Frossard, "Perceptual MPEG-2 syntactic information coding: a performance study based on psychophysics," in *Proc. Picture Coding Symp.*, Berlin, Germany, Sept. 1997, pp. 297–302.

[27] P. Frossard and O. Verscheure, "MPEG-2 video over lossy packet networks: perceptual syntactic information coding," in *Proc. SPIE Int. Symp. Voice, Video, and Data Communications*, vol. 3528, Boston, MA, Nov. 1998, pp. 113–123.

[28] C. Fogg, "mpeg2ncode/mpeg2decode," MPEG Software Simulation Group, 1996.

[29] C. J. van den Branden Lambrecht and O. Verscheure, "Perceptual qualtiy measure using a spatio-temporal model of the human visual system," *Proc. SPIE*, vol. 2668, pp. 450–461, Jan. 1996.

[30] S. Winkler, "A perceptual distortion metric for digital color images," in *Proc. Int. Conf. Image Processing*, vol. 3, Chicago, IL, Oct. 1998, pp. 399–403.

[31] R. E. Blahut, *Theory and Practice of Error Control Codes*. Reading, MA: Addison-Wesley, 1983.

[32] A. J. McAuley, "Reliable broadband communications using a burst erasure correcting code," in *Proc. ACM SIGCOMM'90*, Philadelphia, PA, Sept. 1990, pp. 297–306.

[33] J. Nonnenmacher, E. W. Biersack, and D. Towsley, "Parity-based loss recovery for reliable multicast transmission," *IEEE/ACM Trans. Networking*, vol. 6, pp. 349–361, Aug. 1998.

[34] E. Biersack, "Performance evaluation of forward error correction in an ATM environment," *IEEE J. Select. Areas Commun.*, vol. 11, pp. 631–640, May 1993.

[35] P. Frossard and O. Verscheure, "Content-based MPEG-2 structuring and protection," in *Proc. SPIE Int. Symp. Voice, Video, and Data Communications*, vol. 3845, Boston, MA, Sept. 1999.

[36] J. Rosenberg and H. Schulzrinne, "An RTP payload format for generic forward error correction," IETF, Internet Draft, 1999.

Pascal Frossard (M'96) received the M.S. and Ph.D. degrees in electrical engineering from the Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland, in 1997 and 2000, respectively. He graduated from the Communication Systems Doctoral School, EPFL, in 1998.

From 1997 to 1998, he was a Research and Teaching Assistant in the Institute for Computer Communications and their Applications (ICA) at EPFL. From 1998 to 2000, he worked with the Signal Processing Laboratory of EPFL as Reasearch and Teaching Assistant under a grant from Hewlett–Packard. He is currently a Research Staff Member at the IBM TJ Watson Research Center, Hawthorne, NY. His main research interests include signal and video compression, video streaming, error control, and video distribution.

Olivier Verscheure (M'00) received the B.S. degree in electrical engineering from Ecole Polytechnique de Mons, Belgium, in 1995, and the Ph.D. degree from the Swiss Federal Institute of Technology, Lausanne, Switzerland, in 1999.

He was a Visiting Researcher at the Hewlett–Packard Laboratories, Palo Alto, CA, during the summer of 1997. He joined IBM Research in July 1999, and is presently a Research Staff Member at the IBM T. J. Watson Research Center, Hawthorne, NY. His research lies within the areas of scalable multimedia servers, packet video, and vision science.